Réseau de PLLs distribuées pour synthèse automatique d'horloge de MPSOCs synchrones

Anton Korniienko

▶ **To cite this version:**

Anton Korniienko. Réseau de PLLs distribuées pour synthèse automatique d'horloge de MPSOCs synchrones. Autre. Ecole Centrale de Lyon, 2011. Français. NNT : 2011ECDL0040 . tel-00676933

HAL Id: tel-00676933

https://theses.hal.science/tel-00676933

Submitted on 6 Mar 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° d'ordre 2011-40 Année 2011

ÉCOLE CENTRALE DE LYON

THÈSE

pour obtenir le grade de

DOCTEUR

Spécialité : "Informatique Automatique"

préparée dans le laboratoire Ampère et dans le Laboratoire d'Électronique des Technologies de l'Information (CEA, LETI, MINATEC)

dans le cadre de **l'École Doctorale "Électonique, Électrotechnique et Automatique"**

par

Anton Korniienko

Réseau de PLLs distribuées pour synthèse automatique d'horloge de MPSOCs synchrones

---

soutenue publiquement le 6 décembre 2011 devant la commission d'examen

JURY

Président du jury	M. Ian O'CONNOR	Professeur des Universités, École Centrale de Lyon
Rapporteur	M. Jamal DAAFOUZ	Professeur des Universités, ENSEM Nancy
Rapporteur	M. Yann LE GORREC	Professeur des Universités, ENSMM Besançon
Examinateur	M. Stéphane FONT	Chef de Département SSE, SUPÉLEC
Examinateur	M. Dimitri GALAYKO	Professeur des Universités, UPMC, Paris VI
Invité	M. Jérôme JUILLARD	Professeur adjoint, HDR, SUPÉLEC
Encadrant	M. Eric COLINET	Chercheur, HDR, CEA, LETI, MINATEC
Encadrant	M. Eric BLANCO	Maître de Conférences, École Centrale de Lyon
Directeur de thèse	M. Gérard SCORLETTI	Professeur des Universités, École Centrale de Lyon

Every action, every change in nature, is set within
a network of feedback loops.

Jay W. Forrester

ÉCOLE CENTRALE LYON — **Direction de la Recherche**

Liste des personnes Habilitées à Diriger des Recherches en poste à l'Ecole Centrale de Lyon

Nom-Prénom	*Corps grade*	Laboratoire ou à défaut département ECL	Etablissement
BEROUAL Abderrahmane	*professeur*	AMPERE	ECL
BURET François	*professeur*	AMPERE	ECL
JAFFREZIC-RENAULT Nicole	*directeur de recherche*	AMPERE	CNRS/ECL
KRÄHENBÜHL Laurent	*directeur de recherche*	AMPERE	CNRS/ECL
NICOLAS Alain	*professeur*	AMPERE	ECL
NICOLAS Laurent	*directeur de recherche*	AMPERE	CNRS/ECL
SCORLETTI Gérard	*professeur*	AMPERE	ECL
SIMONET Pascal	*chargé de recherche*	AMPERE	CNRS/ECL
VOLLAIRE Christian	*professeur*	AMPERE	ECL
		Nbre Ampère 9	
HELLOUIN Yves	*maître de conférences*	DER EEA	ECL
		Nbre DER EEA 1	
GUIRALDENQ Pierre	*professeur émérite*	DER STMS	ECL
VINCENT Léo	*professeur*	DER STMS	ECL
		Nbre DER STMS 2	
LOHEAC Jean-Pierre	*maître de conférences*	ICJ	ECL
MAITRE Jean-François	*professeur émérite*	ICJ	ECL
MARION Martine	*professeur*	ICJ	ECL
MIRONESCU Elisabeth	*professeur*	ICJ	ECL
MOUSSAOUI Mohand	*professeur*	ICJ	ECL
MUSY François	*maître de conférences*	ICJ	ECL
ZINE Abdel-Malek	*maître de conférences*	ICJ	ECL
		Nbre ICJ 7	
DAVID Bertrand	*professeur*	ICTT	ECL
		Nbre ICTT 1	
CALLARD Anne-Ségolène	*maître de conférences*	INL	ECL
CLOAREC Jean-Pierre	*maître de conférences*	INL	ECL
GAFFIOT Frédéric	*professeur*	INL	ECL
GAGNAIRE Alain	*maître de conférences*	INL	ECL
GARRIGUES Michel	*directeur de recherche*	INL	CNRS/ECL
GENDRY Michel	*directeur de recherche*	INL	CNRS/ECL
GRENET Geneviève	*directeur de recherche*	INL	CNRS/ECL
HOLLINGER Guy	*directeur de recherche*	INL	CNRS/ECL
KRAWCZYK Stanislas	*directeur de recherche*	INL	CNRS/ECL
LETARTRE Xavier	*chargé de recherche*	INL	CNRS/ECL
O'CONNOR Ian	*professeur*	INL	ECL
PHANER-GOUTORBE Magali	*professeur*	INL	ECL
ROBACH Yves	*professeur*	INL	ECL
SAINT-GIRONS Guillaume	*chargé de recherche*	INL	CNRS/ECL
SEASSAL Christian	*chargé de recherche*	INL	CNRS/ECL
SOUTEYRAND Eliane	*directeur de recherche*	INL	CNRS/ECL
TARDY Jacques	*directeur de recherche*	INL	CNRS/ECL
VIKTOROVITCH Pierre	*directeur de recherche*	INL	CNRS/ECL
		Nbre INL 18	
CHEN Liming	*professeur*	LIRIS	ECL
		Nbre LIRIS 1	

BAILLY Christophe	*professeur*	LMFA	ECL
BERTOGLIO Jean-Pierre	*directeur de recherche*	LMFA	CNRS/ECL
BLANC-BENON Philippe	*directeur de recherche*	LMFA	CNRS/ECL
BOGEY Christophe	*chargé de recherche*	LMFA	CNRS/ECL
CAMBON Claude	*directeur de recherche*	LMFA	CNRS/ECL
CARRIERE Philippe	*chargé de recherche*	LMFA	CNRS/ECL
CHAMPOUSSIN J-Claude	*professeur émérite*	LMFA	ECL
COMTE-BELLOT geneviève	*professeur émérite*	LMFA	ECL
FERRAND Pascal	*directeur de recherche*	LMFA	CNRS/ECL
GALLAND Marie-Annick	*professeur*	LMFA	ECL
GODEFERD Fabien	*chargé de recherche*	LMFA	CNRS/ECL
GOROKHOVSKI Mikhail	*professeur*	LMFA	ECL
HENRY Daniel	*directeur de recherche*	LMFA	CNRS/ECL
JEANDEL Denis	*professeur*	LMFA	ECL
JUVE Daniel	*professeur*	LMFA	ECL
LE RIBAULT Catherine	*chargée de recherche*	LMFA	CNRS/ECL
LEBOEUF Francis	*professeur*	LMFA	ECL
PERKINS Richard	*professeur*	LMFA	ECL
ROGER Michel	*professeur*	LMFA	ECL
SCOTT Julian	*professeur*	LMFA	ECL
SHAO Liang	*chargé de recherche*	LMFA	CNRS/ECL
SIMOENS Serge	*chargé de recherche*	LMFA	CNRS/ECL
TREBINJAC Isabelle	*maître de conférences*	LMFA	ECL
		Nbre LMFA 23	

BENAYOUN Stéphane	*professeur*	LTDS	ECL
CAMBOU Bernard	*professeur*	LTDS	ECL
COQUILLET Bernard	*maître de conférences*	LTDS	ECL
DANESCU Alexandre	*maître de conférences*	LTDS	ECL
FOUVRY Siegfrid	*chargé de recherche*	LTDS	CNRS/ECL
GEORGES Jean-Marie	*professeur émérite*	LTDS	ECL
GUERRET Chrystelle	*chargé de recherche*	LTDS	CNRS/ECL
HERTZ Dominique	*past*	LTDS	ECL
ICHCHOU Mohamed	*professeur*	LTDS	ECL
JEZEQUEL Louis	*professeur*	LTDS	ECL
JUVE Denyse	*ingénieur de recherche*	LTDS	ECL
KAPSA Philippe	*directeur de recherche*	LTDS	CNRS/ECL
LE BOT Alain	*chargé de recherche*	LTDS	CNRS/ECL
LOUBET Jean-Luc	*directeur de recherche*	LTDS	CNRS/ECL
MARTIN Jean-Michel	*professeur*	LTDS	ECL
MATHIA Thomas	*directeur de recherche*	LTDS	CNRS/ECL
MAZUYER Denis	*professeur*	LTDS	ECL
PERRET-LIAUDET Joël	*maître de conférences*	LTDS	ECL
SALVIA Michelle	*maître de conférences*	LTDS	ECL
SIDOROFF François	*professeur*	LTDS	ECL
SINOU Jean-Jacques	*maître de conférences*	LTDS	ECL
STREMSDOERFER Guy	*professeur*	LTDS	ECL
THOUVEREZ Fabrice	*professeur*	LTDS	ECL
TREHEUX Daniel	*professeur*	LTDS	ECL
VINCENS Eric	*maître de conférences*	LTDS	ECL
		Nbre LTDS 25	
	Total HdR ECL	87	

Remerciements

Les travaux de thèse présentés dans ce mémoire ont été effectués dans le cadre du projet ANR « HODISS » et d'une collaboration entre le LETI du CEA Grenoble et le laboratoire Ampère de l'École Centrale de Lyon. A ce propos, je remercie Messieurs Eric Colinet, Jérôme Juillard et Dimitri Galayko d'avoir monté et accompli ce projet ainsi que Messieurs Eric Colinet, Eric Blanco et Gérard Scorletti pour la réalisation de la collaboration et de l'encadrement réussi de la thèse. Sincèrement, grâce à vous, j'ai appris énormément de choses et j'ai beaucoup évolué en différents aspects, comme la personnalité au niveau scientifique, au niveau de maitrise de la langue française, *etc.*. Je tiens à remercier en particulier Monsieur Eric Colinet non seulement pour son encadrement et les discussions scientifiques que nous avons pu avoir durant ces années de thèse et qui ont souvent donné des résultats intéressants, mais aussi pour son soutien humain et amical. Merci Eric d'avoir appelé la mutuelle pour résoudre des questions de santé ou bien le fameux service de RH du CEA pour débloquer certains verrous administratifs afin de faciliter la vie d'un « pauvre » doctorant ukrainien. Je voudrais également remercier en particulier le directeur de thèse Monsieur Gérard Scorletti pour avoir dirigé avec succès les travaux. Merci Gérard pour tes grandes pédagogie et patience durant des heures passées ensemble discutant de la thèse ou de la vie en général, et pour toutes les corrections et des 20-èmes versions des chapitres, des articles, des présentations, *etc.*. On peut vraiment remarquer la différence par rapport aux versions numéro 10. Merci de m'avoir hébergé quand c'était nécessaire surtout pour rester en règle avec l'administration française. Je voudrais remercier en outre Gérard Scorletti et Eric Blanco, ainsi que toute l'équipe d'enseignement ATS de l'Ecole Centrale de Lyon pour m'avoir fait découvrir le monde de l'enseignement non seulement en théorie mais aussi en pratique.

Je remercie Messieurs Jamal Daafouz et Yann Le Gorrec pour m'avoir fait l'honneur de rapporter la thèse, d'y avoir consacré du temps et donné les remarques constructives. N'étant pas écrit dans ma langue maternelle, je crois que le manuscrit comporte des passages un petit peu plus dur à comprendre que les autres. Je suis très honoré que Monsieur Ian O'Connor ait accepté de présider le jury de thèse. Enfin, je remercie Messieurs Stéphane Font, Dimitri Galayko, Jérôme Juillard et les autres membres du jury d'avoir y participer, d'avoir posé pleins de questions intéressantes en m'incitant à analyser les résultats obtenus et à réfléchir sur les perspectives futures possibles de travail.

Je voudrais remercier Messieurs Cyril Condemine, Eric Colinet et Eric Blanco pour l'encadrement mon stage de fin d'étude, qui a été, après tout, mes premiers pas dans le domaine scientifique, avant que je décide de faire les études en thèse. Merci en particulier à Jonathan Soen, qui à l'époque, a été un exemple de doctorant à suivre, me motivant à prendre une voie de recherche et dont les travaux en thèse m'ont

beaucoup aidé durant mon stage.

Je remercie l'ensemble des personnes ayant contribué à ce travail de thèse : merci à Monsieur François Anceau, à Mohammad, Eldar, Jean-Michel et Chuan ainsi que à tous les hodissiens pour leur participation dans le projet, pour les discussions et les réunions animées et le travail intéressant qu'on a pu avoir ensemble. Je tiens à remercier Bénédicte Lafay pour son aide dans la description de l'exemple de synchronisation en nature, Micha, Benoit, Matthieu, Michael Pelissier, Émeric De Foucauld, Cédric, Chady pour les discussions qu'on a eu autour d'un café ou autre, qui m'ont donné assez souvent des idées intéressantes à explorer.

Pour tous leurs efforts et leur disponibilité, je remercie chaleureusement Armelle, Caroline et Marie-Christine : le côté administratif n'a pas été toujours facile avec moi, mais tout a été bien résolu. A ce propos, je tiens à remercier Monsieur Jean-René Lequepeys non seulement pour m'avoir accueilli au sein de son département au CEA et soutenu durant la thèse dans toutes les démarches entreprises, mais aussi un grand merci pour avoir appelé l'ambassade irlandaise afin de résoudre le problème de VISA, ce qui m'a permis, en dépit des difficultés, d'aller quand même en conférence à Cork et de présenter nos travaux. Je remercie Messieurs Cyril Condemine et Pierre Vincent, les chefs de laboratoires au CEA dans lesquels j'ai eu l'occasion de travailler durant les années en thèse. Mes remerciements vont aussi à Monsieur Laurent Nicolas, directeur du laboratoire Ampère, pour m'avoir accueilli, lui aussi, au sein de son unité de recherche. J'adresse également mes remerciements à tous les personnels de mes trois laboratoires d'accueil que ce soit les chercheurs, les enseignants, les post-docs, les ATERs, les doctorant, les stagiaires les élèves ou les personnels techniques pour ces jours passés ensemble.

Je ne saurais terminer sans adresser mes plus vifs remerciements à toutes les personnes, qui par leur existence et soutien (moral, physique ou autres, l'accueil toujours chaleureux, et les repas animés auxquels j'ai toujours bien participé) directement ou non, ont contribué à la réalisation de ce travail et plus particulièrement à ma mère, à ma famille, à toute la famille Martin. Merci à tous mes amis de Grenoble, Lyon, Kiev, Moukatchevo, de toute la Zakarpatie et le monde entier, qui m'ont beaucoup entendu parler de mon sujet, de mes recherches et dire que « l'Automatique c'est très fort », durant des conversations intéressantes... ou pas. Merci pour une super collocation Micha et Christophe, et pour l'atmosphère si chaleureuse que j'ai subie et dans laquelle ma thèse a dû se dérouler.

Merci à toi ma petite Marion, et pas seulement pour tes corrections d'orthographe ou de grammaire françaises...

Résumé

Sujet : Réseau de PLLs distribuées pour synthèse automatique d'horloge de MPSOCs synchrones

Les arbres classiques de distribution du signal d'horloge au sein des microprocesseurs synchrones présentent un certain nombre de limitations : skew, jitter, limitation de la fréquence, influence de perturbations et de dispersions quelles que soient leurs natures. Ces facteurs, critiques pour les microprocesseurs modernes complexes, sont devenus la raison principale qui a poussé à la recherche d'autres types d'architecture de génération et de distribution du signal d'horloge. Un exemple d'un tel système alternatif est le réseau de PLLs couplées, où les PLLs sont géographiquement distribuées sur la puce, et génèrent des signaux d'horloge locaux qui sont ensuite synchronisés, en temps réel, par un échange d'information entre les PLLs voisines et une rétroaction locale réalisé par leur correcteurs.

La nature active du réseau de PLLs de génération et de distribution du signal d'horloge, qui peut permettre de surpasser les limitations mentionnées plus tôt, oblige à sortir du cadre classique des outils et des méthodes de la Microélectronique habituellement appliqués à l'étude et à la conception de ce type de systèmes. En effet, les aspects dynamiques de bouclage et de transformation de signaux au sein de tels systèmes complexes rendent leur conception extrêmement difficile voire parfois impossible. La difficulté principale consiste en un changement des propriétés d'un sous-système local indépendant par rapport aux propriétés du même sous-système faisant partie du réseau. Effectivement, il existe beaucoup de méthodes et d'outils de conception d'une PLL isolée garantissant un comportement et des propriétés locales désirés. Néanmoins, ces propriétés désirées locales, selon la topologie d'interconnexion considérée, ne sont pas forcément conservées quand il s'agit d'un réseau de PLLs interconnectées et de son comportement global. Le but principal de cette thèse est ainsi de développer une méthode de synthèse de la loi de commande décentralisée réalisée au sein de chaque sous-système (tel qu'une PLL) assurant le comportement désiré pour le réseau global.

Une méthode de transformation du problème de synthèse globale en un problème équivalent de synthèse d'une loi de commande locale est proposée en se basant sur l'hypothèse des sous-systèmes identiques interconnectés en réseau. Le lien entre les propriétés locales et globales est établi grâce aux approches d'Automatique avancée telles que les approches entrée-sortie et la dissipativité. Ce choix de méthode permet non seulement de réduire considérablement la complexité du problème initial mais aussi de ramener le problème de synthèse à une forme proche des méthodes de conception locale utilisées en Microélectronique, ce qui garantit une continuité

logique de leur évolution. Ensuite la méthode proposée est combinée avec la commande H_∞ et l'optimisation sous contraintes LMIs conduisant au développement d'algorithmes efficaces de résolution du problème posé. Elles sont à la fois particulièrement bien adaptées à l'application considérée, c'est-à-dire à la synchronisation d'un réseau de PLLs, et sont facilement généralisables aux autres types de problèmes de commande de systèmes de grande dimension. Le premier aspect permet une intégration naturelle et aisée de la méthode dans le flux de conception existant en Microélectronique, très riche et mature à ce jour, alors que le deuxième offre une solution à d'autres problèmes de commande de systèmes interconnectés en réseau, un champ d'application aujourd'hui en plein essor.

Mots clefs : Systèmes en réseau, système Multi-Agents, Commande Décentralisée, Commande H_∞, Dissipativité, Optimisation LMI, Conception du réseau de PLLs, Synchronisation, Distribution Active du Signal d'horloge.

Abstract

Subject : Distributed PLL network for automatic clock synthesis of synchronous MPSOCs

The classical clock distribution trees used in the synchronous microprocessor systems in nowadays have several drawbacks such as skew, jitter, frequency limitation, perturbation and disturbance behavioral impact independently of their origin, *etc.*. These factors, critical for the modern microprocessors, motivate the research of an alternative architecture of the clock generation and distribution system. An example of such alternative architectures is the network of coupled PLLs where the PLLs are geographically distributed on the chip and produce the local clock signals. These local clock signals are then synchronized, in real time, by an exchange of information between the PLLs and by local feedback corrections realized by its controllers.

Distributed PLLs network allows overcoming the mentioned limitation encountered for the classical clock distribution system. However, the active nature of this network requires going beyond the scope of usual stand-alone PLL design methods. Indeed, the dynamical aspects of the feedback loops and the transformations of the signal inside this complex system make the design problem extremely difficult to solve. The main issue consists in ensuring certain properties of the global network as well as local properties of each subsystem PLL because those properties may change drastically from independent stand-alone PLL designed with standard tools and methods. Indeed, depending on the network topology, the local properties and global dynamical behavior are not necessarily ensured for the overall network. The main contribution of this PhD thesis is the development of a control law design method for each subsystem (such as PLL) ensuring the desired behavior of the global network.

A method for transforming the global design problem to an equivalent local control law design problem is proposed. It is based on the assumption that all subsystems are identical. The relation between the local and global properties is established using advanced Control System Theory tools such as input-output and dissipativity principle. This principle decreases significantly the problem complexity by transforming the design problem into a form that is closed to the design of a stand-alone closed loop system. The proposed method is combined with robust H_∞ control and LMI optimization that can be solved efficiently with appropriate algorithms that are well suited for the considered application *i.e.* the PLLs network synchronization. The proposed approach can be easily generalized to other types of networked system to be controlled.

Keywords : Networked systems, Multi-Agent systems, Decentralized Control, H_∞ Control, Dissipativity, LMI Optimization, PLL network Design, Synchronization, Active Clock Distribution.

Резюме

Тема : Сеть распределенных систем ФАПЧ для автоматического синтеза сигнала тактовой частоты для синхронных микропроцессорных систем на чипе

Классические системы генерирования и распределения сигнала тактовой частоты в современных синхронных микропроцессорных системах обладают рядом недостатков: сдвиг и дрожание фаз, ограничение частоты, влияние возмущающих воздействий и дисперсии технологических параметров и т.п. Эти факторы, критические для комплексных микропроцессорных систем, стали основной причиной побуждающей к исследованиям новых типов архитектуры систем генерирования и распределения сигнала тактовой частоты. Один из примеров такой альтернативной системы является сеть географически распределенных на чипе и взаимосвязанных между собой систем Фазовой Автоподстройки Частоты (ФАПЧ). Эти системы генерируют локальные сигналы тактовой частоты которые вследствие синхронизируются между собой в реальном времени за счет обмена информации между соседними ФАПЧ.

Сеть распределенных ФАПЧ позволяющая превысить упомянутые ранее ограничения, вынуждает выйти из рамок классических методов и инструментов в Микроэлектронике, обычно применяемых в изучении и проектировании такого типа систем. В действительности, динамические аспекты обратных связей и преобразования сигналов существующих в таких комплексных системах производить к тому, что их разработка становится крайне затруднительной и даже иногда невозможной. Основная трудность заключается в изменении характеристик независимой локальной подсистемы (ФАПЧ) по отношению к характеристик той же подсистемы, входящей в состав глобальной сети. Действительно, существует много методов и инструментов проектирования одной независимой системы ФАПЧ, гарантирующие ее нужные локальные качественные характеристики. Однако, эти характеристики, в зависимости от данной топологии взаимосвязей, не обязательно сохраняются когда речь идет об сети связанных между собой ФАПЧ и ее глобальных качественных характеристик. Основная задача этой работы является разработка метода синтеза децентрализованного закона управления для каждой подсистемы (ФАПЧ) обеспечивающий нужные характеристики глобальной сети.

В результате работы, базируясь на предположение идентичности подсистем, был предложен метод преобразования проблемы глобального синтеза в эквивалентную проблему локального синтеза закона управления. Благодаря методам передовой Теории Автоматического Управления, а именно подходам входа-выхода и диссипативности, была установлена связь между локальными и глобальными качественными характеристиками системы позволяющая это преобразование. Разработанный метод позволяет не только значительно снизить уровень сложности исходной проблемы, но и преобразовать проблему синтеза в форму близкую к существующим локальным методам разработки используемых в Микроэлектронике, гарантирующее их логическую и непрерывную эволюцию. Затем, предложенный метод в комбинации с методами робастного контроля и оптимизации Линейных Матричных Неравенств (ЛМН) привел к разработке эффективных алгоритмов решения поставленной проблемы. Таким образом, разработанный метод с одной стороны в особенности хорошо адаптирован к рассматриваемой инженерной проблеме, то-есть синхронизации сети ФАПЧ, и, с другой стороны, легко обобщаем, что позволяет решать другие типы проблем автоматического контроля систем связанных в сети. Первый аспект позволяет легкую и естественную интеграцию метода в существующую в Микроэлектронике и очень развитую САПР, между тем как второй аспект предлагает решение проблем контроля систем в сети, сферы инженерии, являющейся на этапе бурного развития на сегодняшний день.

Ключевые слова : Сеть взаимосвязанных систем, Системы Мульти-Агентов, Децентрализованное Управление, H_∞-управление, Диссипативность, ЛМН Оптимизация, Проектирование сети ФАПЧ, Синхронизация, Активные системы генерирования и распределения сигнала тактовой частоты.

Table des matières

Introduction générale **1**

1 Distribution du signal d'horloge **7**
Introduction . . . 7
1.1 Architectures synchrones et asynchrones de circuits intégrés . . . 7
1.2 Arbres classiques de distribution du signal d'horloge . . . 9
1.2.1 Génération du signal d'horloge . . . 10
1.2.2 Distribution du signal d'horloge . . . 11
1.2.3 Incertitudes du signal d'horloge : Skew et Jitter . . . 12
1.2.4 Dissipation d'énergie : réchauffement . . . 16
1.2.5 Du problème de distribution au problème de synchronisation . 17
1.3 Synchronisation dans la nature . . . 18
1.4 D'une architecture classique à une architecture distribuée active . . . 20
1.4.1 Grilles . . . 20
1.4.2 Compensation centralisée du skew . . . 21
1.4.3 Compensation décentralisée du skew . . . 22
1.4.4 Filtrage centralisé du jitter . . . 23
1.4.5 Réseau actif de distribution du signal d'horloge . . . 24
Conclusion : Problématique de la thèse . . . 26

2 Conception automatique d'une ADPLL locale **33**
Introduction . . . 33
2.1 Généralités sur les PLLs . . . 34
2.1.1 Comparateur de phases . . . 36
2.1.2 Filtre . . . 40
2.1.3 Oscillateur contrôlé par un mot numérique . . . 42
2.1.4 Diviseur de fréquence . . . 42
2.2 Modélisation d'une ADPLL . . . 43
2.2.1 Comparateur de phase . . . 44
2.2.2 Filtre . . . 46
2.2.3 Oscillateur . . . 47
2.2.4 Diviseur de fréquence . . . 53
2.2.5 Modèle complet de l'ADPLL . . . 54
2.3 Les aspects d'auto-échantillonnage . . . 56
2.4 Méthodes de synthèse continues de correcteurs numériques . . . 57
2.5 Synthèse de la loi de commande d'une ADPLL . . . 60
2.5.1 Cahier des charges pour la synthèse d'une ADPLL . . . 61
2.5.2 Cahier des charges sous forme d'un critère mathématique . . . 63
2.5.3 Mise en forme standard d'une synthèse H_∞ . . . 74

2.5.4 Exemple numérique de la synthèse d'une loi de commande d'une ADPLL locale . . . 76
2.6 Robustesse vis-à-vis d'incertitudes paramétriques . . . 79
2.6.1 Valeur singulière structurée et son calcul . . . 82
2.6.2 μ-analyse . . . 83
2.6.3 Application de la μ-analyse au cas d'une ADPLL . . . 88
2.6.4 Exemple numérique de la μ-analyse . . . 90
2.7 Simulations . . . 91
2.8 D'une ADPLL locale au réseau d'ADPLLs global . . . 96
2.8.1 Interconnexion en réseau . . . 96
Conclusion . . . 101
2.9 Annexe A : Optimisation LMI . . . 102
2.10 Annexe B : Résolution du problème H_∞ standard par la méthode d'optimisation sous contrainte LMI . . . 106

3 Systèmes en réseau de grande dimension 115
Introduction . . . 115
3.1 Systèmes Multi-Agents . . . 117
3.1.1 Définition . . . 117
3.1.2 Stabilité du système Multi-Agents . . . 117
3.1.3 Résultats d'analyse de MASs . . . 122
3.1.4 Résultats de synthèse de MASs . . . 123
3.2 Commande Décentralisée . . . 125
3.2.1 Définitions . . . 125
3.2.2 Théorème de séparation des graphes . . . 129
3.2.3 Commande décentralisée des systèmes de grande dimension . . . 131
Conclusion . . . 132
3.3 Annexe C : Théorie des Graphes et Théorie algébrique de matrices . . . 134
3.3.1 Définitions générales . . . 134
3.3.2 Produit de Kronecker . . . 138
3.3.3 Propriétés algébriques de matrices . . . 140
3.4 Annexe D : Opérations mathématiques sur les signaux . . . 143

4 Méthode de la synthèse de la loi de commande décentralisée 145
Introduction . . . 145
4.1 Formulation du problème . . . 146
4.2 Les conditions locales pour la stabilité et la performance globales . . . 148
4.3 Synthèse de la loi de commande . . . 156
4.3.1 Synthèse H_∞ de la loi de commande . . . 156
4.3.2 Performance Locale . . . 157
4.3.3 Performance Globale . . . 159
4.4 Choix de caractérisation de dissipativité . . . 164
4.4.1 Conditions du Théorème 4.6 . . . 164
4.4.2 Conditions du Théorème 4.10 . . . 166
4.4.3 L'algorithme de la synthèse de la loi de commande décentralisée 167
4.5 Exemple numérique . . . 170
4.5.1 Stabilité globale performance locale . . . 170

4.5.2 Performance globale . . . 173
Conclusion . . . 180

5 Conception automatique du réseau d'ADPLLs 183
Introduction . . . 183
5.1 Topologies d'interconnexion considérées . . . 184
5.2 Modèle de synthèse et d'analyse du réseau . . . 184
5.3 Modèle de simulation du réseau et Mode-Locking . . . 190
5.3.1 Description . . . 190
5.3.2 Élimination de Mode-Locking . . . 192
5.4 Synthèse de la loi de commande décentralisée pour le réseau d'ADPLL 197
5.4.1 Cahier des charges pour le réseau d'ADPLL . . . 197
5.4.2 Cahier des charges sous forme d'un critère mathématique . . . 197
5.4.3 Exemple numérique de la synthèse d'une loi de commande pour le réseau d'ADPLL . . . 201
5.5 Choix d'emplacement de la référence en réseau . . . 215
5.6 Robustesse du réseau vis-à-vis d'incertitudes paramétriques . . . 224
5.6.1 Stabilité robuste du réseau . . . 226
5.6.2 Performance globale robuste du réseau . . . 226
5.7 Simulation du réseau complet de modèles d'ADPLLs . . . 233
Conclusion . . . 238

Conclusion générale et perspectives 239

Liste des symboles 243

Bibliographie 245

Introduction générale

Avec l'évolution des technologies de circuits intégrés de grande densité (VLSI, Very Large Scale Integration), la technique classique de distribution centralisée du signal d'horloge est de moins en moins bien adaptée aux systèmes MPSOCs (Multi Processor System On Chip) modernes. En effet, dans les circuits intégrés modernes à la fois denses et complexes, le nombre de transistors atteint des dizaines de millions, leur taille quelques dizaines de nanomètres et la fréquence du signal d'horloge des gigahertzs. Dans ces conditions, les délais de propagation augmentent et deviennent difficilement maîtrisables, l'environnement bruyant n'est plus négligeable à cause de la petite taille de composants, et la fréquence élevée en combinaison avec une grande densité de circuits fait également augmenter la consommation. Toutes ces raisons rendent inefficaces et onéreuses les techniques traditionnelles de génération et de distribution du signal d'horloge telles que l'arbre de distribution du signal d'horloge, grille du signal d'horloge, *etc.*. C'est pour cela que des architectures alternatives aux arbres de distribution du signal d'horloge sont aujourd'hui recherchées.

Une génération distribuée de l'horloge, à travers un réseau actif, représente une alternative aux solutions classiques. Un tel système complètement décentralisé est composé de sous-systèmes identiques interconnectés entre eux et géographiquement répartis sur la puce. Chaque sous-système est une boucle de rétroaction. Le réseau actif de distribution du signal d'horloge permet potentiellement *(i)* une génération des signaux d'horloges synchrones et disponibles en différents endroits du circuit sans leur distribution préalable en haute fréquence (réduction de la consommation) ; *(ii)* un filtrage des bruits et une réjection des perturbations extérieures en temps réel, d'où l'adjectif d'architecture active. Ces deux tâches sont robustes aux incertitudes et moins sensibles aux influences des dynamiques non-linéaires vue la nature intrinsèque des boucles fermées de rétraction.

Les premiers prototypes du réseau actif confirment ces résultats en montrant une opportunité intéressante de générer un signal d'horloge rapide (500-1000 MHz ou plus) dans les circuits numériques très complexes, à faible coût énergétique. Néanmoins, la conception de ces premiers prototypes a été effectuée dans le cas particulier d'une topologie donnée du réseau générant les signaux d'horloges à fréquence donnée. Il n'existe pas de méthode de conception d'un tel réseau pour le cas général d'une topologie et d'une fréquence quelconques. Étant données les perspectives offertes par les réseaux actifs de génération du signal d'horloge pour aller au-delà des limitations technologiques des systèmes traditionnels, il est important de développer une méthode générale pour leur analyse et leur synthèse.

Cependant, malgré ces avantages, la nouvelle architecture est associée à une complexité de conception plus importante que son homologue à distribution centralisée. En effet, dans la mesure où le réseau actif de distribution du signal d'horloge est

composé de sous-systèmes formant des boucles d'interconnexion globales et en représentant eux-mêmes des boucles de rétraction locales, le comportement dynamique du système complet est d'une telle complexité qu'il est difficile de l'analyser, sans parler de le concevoir. Le projet ANR HODISS (HOrlogerie Distribuée pour les SOCs Synchrones) dans le cadre duquel s'inscrit cette thèse, a pour but principal le développement de la méthode générale d'analyse et de synthèse d'un tel système complexe de génération et de distribution du signal d'horloge pour les SOCs synchrones.

En raison de la pluridisciplinarité du problème, les contributeurs principaux du projet : LIP6 (PARIS VI), Supélec, CEA-LETI et laboratoire Ampère (École Centrale de Lyon - INSA - UCB) apportent leurs compétences respectives en Microélectronique, la Théorie et Modélisation de systèmes électroniques non-linéaires, analogiques et mixtes ainsi qu'en Automatique. La contribution de ma thèse dans le projet HODISS consiste en l'utilisation et le développement de méthodes et d'approches d'Automatique dans le but de résoudre la problématique de synthèse et d'analyse du comportement du réseau actif de génération et de distribution du signal d'horloge. D'un point de vue Automatique, ces deux problèmes consistent en la synthèse de lois de commande assurant les propriétés désirées du système complet et en l'analyse de son comportement.

Dans ces conditions, il est important de préciser deux aspects principaux auxquels la contribution de la thèse doit correspondre : il est nécessaire que les méthodes d'analyse et de synthèse de l'Automatique à développer soient à la fois *(i)* adaptées aux méthodes et aux outils existants en Microélectronique vue la nature de la problématique considérée et *(ii)* les plus générales possibles afin d'imposer un large spectre d'exigence de performance et de traiter d'autres types de problèmes semblables à la problématique considérée ici. Ces deux aspects, à première vue contradictoires, ne le sont pas en réalité et inspirent tous les choix faits tout au long de cette thèse. Ils permettront de développer des méthodes et des approches d'Automatique qui soient facilement intégrables au flux de conception existant de la Microélectronique et en même temps aisément applicables à des types d'applications similaires intéressantes et/ou plus larges que l'application considérée dans cette thèse. Il s'agit de la question primordiale dans les sciences de l'équilibre entre la pratique et la problématique utilitaire (une conception et d'analyse du comportement du réseau actif de distribution du signal d'horloge) d'un côté et la théorie plus générale, poussée (la synthèse et l'analyse du comportement du système décentralisé de grande dimension) d'un autre côté. Comme le montre ce mémoire, il est possible de trouver un équilibre, l'objectif étant d'améliorer le processus de conception.

Organisation du mémoire

Ce mémoire de thèse se décompose en cinq chapitres.

Chapitre 1 : Distribution du signal d'horloge

Ce chapitre présente la problématique de distribution du signal d'horloge en MP-SOCs synchrones. Les architectures synchrones et asynchrones ainsi que leurs différences sont présentées au début du chapitre. Ensuite, les arbres classiques de génération et de distribution du signal d'horloge, un système vital pour les architectures de microprocesseurs synchrones, sont brièvement présentés en soulignant les limitations technologiques les plus importantes. Un exemple de la synchronisation des lucioles est présenté par la suite afin de bien comprendre les mécanismes principaux de synchronisation existant dans la nature et de donner une explication logique d'évolution technologique, présentée dans la section suivante, des architectures de génération et de distribution des signaux d'horloge synchrones d'une architecture classique à une architecture active. En conclusion du chapitre, une formalisation de la problématique considérée dans cette thèse du point de vue d'Automatique est présentée en détails.

Chapitre 2 : Conception automatique d'une PLL locale

Étant donné que la conception du réseau complet actif de distribution du signal d'horloge ayant certaines propriétés désirées est un problème complexe n'ayant pas à ce jour de solutions en Microélectronique, une autre approche « locale » est proposée par les concepteurs en Microélectronique. Elle est beaucoup plus simple et consiste en deux étapes *(i)* une conception locale d'un seul sous-système, correspondant à une boucle à verrouillage de phase locale (PLL), garantissant ses propriétés de performance désirées séparément du réseau et *(ii)* une analyse du comportement de ce dernier après une telle conception et une interconnexion de sous-systèmes en réseau. Le prix à payer pour cette simplicité est le fait que l'approche « locale » ne permet pas de garantir lors de la conception que les propriétés assurées pour un seul sous-système sont conservées ou faiblement dégradées dans le cas du réseau complet. Une réalisation de cette approche « locale » est illustrée dans le chapitre 2. A ce sujet, différentes architectures de boucles à verrouillage de phase, l'architecture retenue d'une boucle à verrouillage de phase (ADPLL) ainsi que sa modélisation, ses aspects d'auto-échantillonnage et du contrôle numérique sont étudiés dans les quatre premières sections du chapitre. Ensuite, des méthodes de synthèse locale de la loi de commande et d'analyse de la robustesse vis-à-vis des incertitudes paramétriques d'une seule ADPLL sont présentées. Ces méthodes issues de l'Automatique peuvent être facilement combinées avec des méthodes et des approches existantes de conception et d'analyse du comportement d'une seule ADPLL en Microélectronique et donc intégrées dans le flux des outils CAO, fortement développés dans ce domaine ces dernières années. Les simulations de la section suivante confirment l'efficacité des méthodes proposées de synthèse et d'analyse du comportement d'une seule ADPLL. A la fin du chapitre, une analyse du comportement d'un exemple d'interconnexion en réseau des ADPLLs identiques obtenus, c'est-à-dire la dernière étape de l'approche « locale » de la conception utilisée en Microélectronique, illustre ses limitations principales et la nécessité du développement d'une autre méthode de conception qui puisse garantir a priori les propriétés désirées du système complet.

Chapitre 3 : Systèmes en réseau de grande dimension

La question qui se pose est de savoir s'il existe une méthode de synthèse réalisant la même approche « locale » mais avec une ou plusieurs conditions supplémentaires assurant que les propriétés désirées garanties en local soient également garanties en global lors de la conception ? Cette sorte de lien « global-local » permettra non seulement de conserver la simplicité de la synthèse locale tout en garantissant les propriétés du système global mais aussi d'avoir une méthode qui soit dans la continuité logique de la méthode de l'approche « locale » dont les outils sont bien maîtrisés, connus et compatibles tant en Automatique qu'en Microélectronique. Afin de préserver la même démarche de la conception « locale » et de généraliser en même temps le problème de synthèse aux autres problèmes d'application, une étude bibliographique exhaustive de l'état de l'art des systèmes en réseau de grande dimension a été réalisée dans le chapitre 3. Ainsi, les deux grandes approches dans ce domaine, les systèmes Multi-Agents et la Commande Décentralisée, ont été considérées. Bien que chaque approche apporte une contribution intéressante à notre problématique, aucune ne peut être appliquée directement du fait de leurs limitations.

Chapitre 4 : Méthode de la synthèse de la loi de commande décentralisée

Inspirés par les idées importantes retenues lors de l'étude de l'état de l'art de systèmes en réseau de grande dimension du chapitre précédent, nous proposons et développons dans ce chapitre une nouvelle méthode de synthèse de la loi de commande décentralisée utilisant le lien « global - local » mentionné précédemment, ce que constitue la contribution principale de cette thèse. En nous appuyant sur l'approche entrée-sortie et la caractérisation des systèmes par la dissipativité, en utilisant l'information importante sur l'identité de sous-systèmes, nous avons pu obtenir des conditions suffisantes locales sur la dynamique de l'interconnexion et la dynamique de sous-systèmes identiques garantissant la stabilité et les propriétés de performance globales d'un tel système de grande dimension. L'information sur l'identité de sous-systèmes a été exploitée avec succès afin de réduire le conservatisme potentiel du résultat. Par la suite, une synthèse de la loi de commande ainsi qu'un choix de la caractérisation de dissipativité nécessaire à l'application de la méthode sont présentés. A la fin du chapitre, la méthode de synthèse développée a été appliquée à l'exemple numérique de la fin du chapitre 2 pour lequel l'approche « locale » initialement proposée en Microélectronique n'a pas donné des résultats satisfaisants.

Chapitre 5 : Conception automatique du réseau de ADPLLs

Ce chapitre illustre l'application de la méthode de synthèse présentée dans le chapitre 4 à différentes topologies de réseau actif de génération et de distribution du signal d'horloge. Les deux premières sections présentent des questions pratiques

d'implémentation : la linéarisation du réseau complet, les aspects de modes parasites Mode-Locking ainsi que les solutions possibles pour leur élimination. Par la suite, les résultats développés dans le chapitre 4 sont appliqués afin de synthétiser une loi de commande décentralisée pour chaque topologie garantissant la stabilité et les propriétés de performance désirées. Les mêmes résultats sont utilisés dans la section qui suit afin de choisir un emplacement optimal du signal de référence au sein du réseau actif de distribution du signal d'horloge de façon à améliorer la rapidité de la synchronisation. Par la suite, une application de la méthode d'analyse de la robustesse de la stabilité et de la performance du réseau complet semblable à celle présentée dans le chapitre 2 dans le cas de la robustesse d'une seul ADPLL est présentée. Le chapitre se termine avec des simulations confirmant les résultats obtenus.

Liste de publications et séminaires

Durant les 3 années de thèse, les travaux réalisés ont fait l'objet des publications, brevets et séminaires suivants.

Articles de conférences internationales, avec comité de lecture, publiés

C.1. A. Korniienko, E. Colinet, C. Condemine, « Performance enhancement of a closed-loop accelerometer tuned close to its instability regime », *IEEE Conf. Newcas-Taisa*, Toulouse, France, juin – juillet 2009, pp. 1–4.

C.2. A. Korniienko, E. Colinet, G. Scorletti, E. Blanco, « H_∞ loop shaping control for distributed PLL network », *IEEE Conf. PRIME*, Cork, Ireland, juillet 2009, pp. 336–339.

C.3. A. Korniienko, E. Colinet, G. Scorletti, E. Blanco, D. Galayko, J. Juillard, « A clock network of distributed ADPLLs using an asymmetric comparison strategy », *IEEE Conf. ISCAS*, Paris, France, mai – juin 2010, pp. 3212–3215.

C.4. M. Javidan, E. Zianbetov, A. Korniienko, E. Colinet, G. Scorletti, J.–M. Akré, J. Juillard, « All-digital PLL array provides reliable distributed clock for SOCs », *IEEE Conf. ISCAS*, Rio de Janeiro, Brazil, mai 2011, pp. 2589–2592.

C.5. A. Korniienko, G. Scorletti, E. Colinet, E. Blanco, J. Juillard, D. Galayko, « Control Law Synthesis for Distributed Multi-Agent Systems : Application to Active Clock Distribution Networks », *IEEE American Control Conference*, San Francisco, USA, juin – juillet 2011, pp. 4691–4696.

C.6. J.M. Akre, J. Juillard, M. Javidan, D. Galayko, E. Zianbetov, D. Galayko, A. Korniienko, E. Colinet, « A Design Approach for Networks of Self-Sampled All-Digital Phase-Locked Loops », *European Conference on Circuit Theory and Design*, Linköping, Sweden, août 2011, pp. 725–728.

Articles de conférence, avec comité de lecture, soumis

C.7. A. Korniienko, G. Scorletti, E. Colinet, E. Blanco, « Approche de la performance relative pour la commande de systèmes de grande dimension », soumis au *Conférence Internationale Francophone d'Automatique*, Grenoble, France, juillet 2012.

Articles de journal, soumis

A.1. A. Korniienko, G. Scorletti, E. Colinet, "Control Law Design for Distributed Multi-Agent Systems : Application to Active Clock Distribution Networks", soumis au *International Journal of Robust and Nonlinear Control*, soumis en octobre 2011.

Articles de journal, en préparation

A.2. M. Javidan, E. Zianbetov, A. Korniienko, G. Scorletti, E. Colinet, J.–M. Akré, J. Juillard, "All-digital PLL network for distributed clock generation in GSLS integrated systems", en préparation 2012.

Séminaires, présentés

S.1. Réseaux actifs de distribution d'horloges pour MPSOCs synchrones : les méthodes de contrôle associées. Séminaire Département Méthodes pour l'Ingénierie des Systèmes, Laboratoire Ampère, Lyon, France. Février 2011.

S.2. Decentralized control of the Multi-Agent systems. California Institute of Technology (Caltech), États–Unis. Juillet 2011.

S.3. Réseau actif de distribution d'horloges pour MPSOCs synchrones : les méthodes de contrôle associées. CEA, LETI, MINATEC, Grenoble, France. Octobre 2011.

Brevets, déposés

B.1. E. Colinet, D. Galayko, and A. Korniienko. Device for generating clock signals for asymmetric comparison of phase, PCT/EP2010/066405, Octobre 2010.

B.2. E. Colinet, A. Korniienko. Dispositif et procédé de compensation de délai de propagation d'un signal, DD 12189 ST, Décembre 2010.

Rapport technique, disponible en ligne

R.1. A. Korniienko, G. Scorletti, E. Colinet, E. Blanco, « Control Law Design for Distributed Multi-Agent Systems ». *Rapport technique* Laboratoire Ampère, École Centrale de Lyon. France. Octobre 2011, http ://hal.archives-ouvertes.fr/hal-00630543/fr/.

1 Distribution du signal d'horloge

Introduction

Dans ce chapitre, nous allons étudier en détails la problématique considérée dans la thèse du point de vue Microélectronique. Dans ce but, la première section présente une brève description des architectures synchrones et asynchrones des circuits intégrés en justifiant le but principal du système de génération et de distribution du signal d'horloge d'un circuit intégré synchrone.

Ensuite, la section suivante donne un aperçu du système de génération et de distribution du signal d'horloge traditionnel à savoir l'arbre classique de distribution du signal d'horloge. Nous allons ainsi étudier les incertitudes principales des signaux d'horloge générés par un tel système, et discuter sur ses limites actuelles technologiques liés principalement à la distribution de signaux d'horloges. Nous allons voir par la suite que pour dépasser ces limites technologiques, il est intéressant de remplacer le problème de distribution par un problème de synchronisation.

Les questions « comment la synchronisation se passe dans la nature et quels sont ses mécanismes principaux ? » sont abordées dans la section 3. En complément, la section suivante décrit une évolution des architecture de génération et de distribution de signal d'horloge d'une architecture classiques aux architectures actives possédant tous ces mécanismes de synchronisation dans la nature.

Le chapitre se conclut par une formalisation du problème de point de vue Automatique considéré dans cette thèse.

1.1 Architectures synchrones et asynchrones de circuits intégrés

De manière simplifiée, un circuit intégré, souvent appelé *puce* (ou *chip* en littérature anglo-saxonne), peut être représenté par un ensemble de blocs opérationnels ayant pour objectif d'envoyer, transformer, recevoir et stocker des données. Les données au sein d'un circuit intégré sont codées par des signaux électriques dont la tension prend deux valeurs, la valeur maximale correspondant au « 1 » logique, la valeur minimale au « 0 » logique. L'ensemble des règles d'échange et de transformation de données détermine un *protocole de communication*. La plupart des circuits intégrés modernes sont réalisés en ***architecture synchrone*** c'est-à-dire que l'ensemble des opérations mentionnées ci-dessus ainsi que le protocole de communication sont synchrones. L'adjectif « synchrone » vient du mot grec *« synchronisation »* où *« syn »*

signifie « avec » ou « ensemble avec » et *« chronos »* signifie « le temps ». Par conséquent, la synchronisation ou simplement « être synchrone » se traduit par le fait que les différentes parties du circuit partagent la même notion du temps ou bien atteignent une *coïncidence temporelle* d'évènements de transformation, d'échange et de stockage de données.

Ainsi, dans une architecture synchrone, les blocs effectuent l'ensemble des opérations élémentaires sur des signaux de données à un temps bien précis et synchrone. Par exemple, imaginons deux blocs adjacents dans une chaîne de traitement de données comme représenté sur la Figure 1.1. Le bloc 1 envoie à l'aide de son émetteur les signaux de données à un instant déterminé t_0. Grâce à un récepteur, le bloc 2 récupère les données envoyées à un instant $t_1 = t_0 + T$ où le temps T est un temps suffisamment grand pour garantir que les données arrivent au récepteur avant qu'il ne les récupère. A l'instant suivant $t_2 = t_1 + T$, les données sont déjà transformées dans le bloc 2 et prêtes à être envoyées aux blocs suivants. Simultanément avec l'opération de récupération de données effectuée dans le bloc 2, à l'instant t_1 l'émetteur du bloc 1 peut envoyer une autre portion de données à traiter qui apparaîtront à l'instant t_2 à l'entrée du récepteur du bloc 2 qui est prêt à les récupérer. Toutes ces opérations d'envoi, récupération, transformation et stockage de données dans l'architecture synchrone s'effectuent ainsi à des instants t_i bien déterminés et sont espacés l'un par rapport à l'autre par un intervalle de temps identique T. On dit que les opérations sont cadencées par un signal périodique de période T. Le signal donnant le rythme aux blocs est appelé le signal d'***horloge*** (ou ***clock*** dans la littérature anglo-saxonne). C'est un signal de tension représentant la valeur numérique d'un bit « 0 » et « 1 ». Les fronts montants et/ou descendants du signal d'horloge indiquent la référence temporelle pour les événements ou les opérations. Par abus de langage dans la suite de ce document, les fronts montants et/ou descendants sont appelés « les évènements ». Le temps T est appelé la période d'échantillonnage et sa valeur inverse $f = 1/T$ est appelée la fréquence d'échantillonnage.

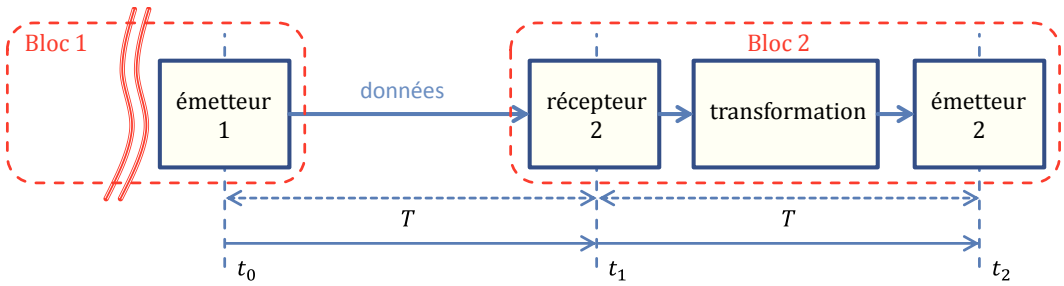


Figure 1.1: Principe de fonctionnement d'un circuit intégré en architecture synchrone

Il existe un autre type d'architecture de circuits intégrés s'appelant ***architecture asynchrone***. Dans ce type d'architecture, les opérations s'effectuent à des instants de temps déterminés par un protocole de communication et un mode de fonctionnement asynchrone. Dans ce cas-là, les opérations sont cadencées par un signal qui n'est pas forcement périodique et même parfois de nature aléatoire dans le cas de circuits complexes de grande dimension. Ce signal, appelé *signal d'acquittement*

(*acknowledgement signal* dans la littérature anglo-saxonne), est envoyé par les blocs eux-mêmes afin de signaler qu'ils sont prêts à effectuer telle ou telle opération. Le signal d'acquittement dépendant de la réalisation du protocole de communication peut être codé par plusieurs bits contenant l'information sur le temps auquel l'opération doit être réalisée et la référence des blocs concernés. Il s'agit souvent d'une succession d'échanges de signaux d'acquittement entre les blocs.

Les blocs d'un circuit intégré effectuant une ou plusieurs opérations simultanées, sont souvent regroupés en sous-systèmes fonctionnels. Les sous-systèmes d'un même circuit intégré peuvent être réalisés selon des architectures différentes : synchrone ou asynchrone. Il s'agit donc d'une architecture mixte du circuit. Un exemple de cette architecture est l'architecture *Globalement Asynchrone et Localement Synchrone* (GALS). Pour ce type d'architecture, contrairement aux opérations synchrones au sein de chaque sous-système, les opérations effectuées entre les sous-systèmes sont asynchrones.

Bien que les architectures asynchrones et mixtes aient un certain nombre d'avantages comme par exemple une consommation potentiellement plus basse par rapport aux architectures synchrones étant donné que le signal d'acquittement est envoyé seulement si cela est nécessaire, elles possèdent un certain nombre d'inconvénients. En plus de la complexité de réalisation liée à la complexité du protocole de communication, il existe des problèmes de robustesse et de sûreté. En effet, théoriquement, dans une séquence irrégulière de signaux, il existe toujours un risque de **métastabilité** c'est-à-dire d'une situation où deux ou plusieurs d'évènements apparaissent en même temps ou quasiment en même temps. Si le temps écoulé entre les deux événements n'est pas suffisamment grand pour qu'une opération s'achève proprement, son résultat peut être erroné et devenir critique pour le fonctionnement du circuit complet. C'est pour ces raisons que les architectures synchrones restent une approche incontournable dans le cas d'applications où un grand niveau de sûreté est demandé : médecine, aviation, nucléaire, *etc.*.

Dans la suite de ce manuscrit, on considère les architectures synchrones de circuits intégrés et la problématique de génération et de distribution du signal d'horloge, un signal vital pour leur fonctionnement.

1.2 Arbres classiques de distribution du signal d'horloge

Comme mentionné précédemment, les blocs élémentaires d'un circuit intégré synchrone sont cadencés par un signal de référence du temps ou tout simplement par un signal d'horloge. Cela veut dire que le signal d'horloge doit être mis à disposition des blocs à tout moment et aux différents endroits du circuit, c'est-à-dire qu'il doit être généré, multiplié et distribué. Le système de génération et de distribution du signal d'horloge dans chaque circuit intégré synchrone a pour objectif principal la réalisation de cette tâche. Il peut être séparé en deux parties ou sous-systèmes fonctionnels : une première partie réalisant la génération du signal d'horloge-source, une deuxième partie réalisant sa multiplication et sa distribution à travers la puce.

1.2.1 Génération du signal d'horloge

De manière générale, le signal d'horloge est généré par un quartz dont la sortie est une tension électrique oscillante. Les quartz sont réputés pour avoir une bonne *stabilité fréquentielle*, c'est-à-dire que les variations de la fréquence et donc de la période du signal horloge au cours du temps de ce dernier sont faibles [Rub10]. Nous allons voir par la suite que les variations de la période d'un signal périodique généré peuvent être modélisées par un bruit de phase de l'oscillateur. Le rapport entre la densité spectrale du signal en sortie d'oscillateur à la fréquence idéale, appelée aussi la porteuse, et la densité spectrale moyenne du bruit de phase détermine la qualité de l'oscillateur.

En dépit de bonnes caractéristiques en bruit des quartz, leur inconvénient majeur est la difficulté d'intégration dans un circuit intégré numérique à cause de leurs dimensions. Il est également compliqué dans le cas d'utilisation des quartz d'atteindre des valeurs de fréquence et d'avoir une plage de variation importante de cette dernière. C'est pour cette raison qu'on utilise d'autres types d'oscillateurs tels que les oscillateurs de relaxations, en anneau, *etc.*. Les avancées techniques ainsi que la réduction de la taille des transistors en Microélectronique moderne ont permis, pour les oscillateurs en anneau, d'atteindre des fréquences et des plages de variation importantes. L'inconvénient de ces oscillateurs est d'avoir, contrairement aux quartz, un niveau du bruit important. Pour diminuer les effets du bruit, on les utilise dans des ***Boucles à Verrouillage de Phase*** (ou ***Phase Locked Loop***, ***PLL*** en littérature anglo-saxonne) avec un signal issu d'un quartz à son entrée. Une PLL est un dispositif permettant de générer un signal d'horloge qui est synchrone avec un signal périodique à son entrée et qui a une fréquence multiple de ce dernier, tout en profitant de la bonne stabilité fréquentielle du signal d'entrée [Roh97, Raz98]. Il s'agit donc d'une multiplication de la fréquence par un facteur d quand pour un front montant du signal périodique à l'entrée de la PLL, sa sortie génère d fronts montants.

Un schéma fonctionnel d'une PLL est représenté sur la Figure 1.2. Il est en générale composé de quatre blocs réalisant chacun une fonction précise. *L'Oscillateur Contrôlé en Tension* (ou *Voltage Controlled Oscillator, VCO* en littérature anglo-saxonne) est un dispositif qui génère un signal périodique local *Loc* dont la fréquence dépend du signal de commande à son entrée. Ensuite, la fréquence de ce signal est divisée par un facteur d grâce au *diviseur de fréquence* qui fournit à sa sortie un signal de fréquence divisée Div. C'est notamment ce bloc de division de fréquence qui permet de multiplier la fréquence du signal à la sortie de la PLL *Loc* par rapport à la fréquence du signal à son entrée Ref. Le signal Div avec le signal de référence extérieure Ref sont envoyés à l'entrée du *Comparateur de Phases* (ou *Phase Detector PD*) qui fournit la différence de phase entre les deux signaux à son entrée. Cette différence de phase considérée comme une erreur à annuler est injectée dans un *Filtre* (F) pour générer une commande nécessaire à la minimisation de l'erreur de la différence de phase et ainsi assurer la synchronisation entre Div et Ref. Le filtre doit par ailleurs rejeter au maximum les fluctuations aléatoires de la phase du signal d'horloge issu de l'oscillateur local.

Les méthodes de conception d'une PLL sont bien connues et maîtrisées en Micro-

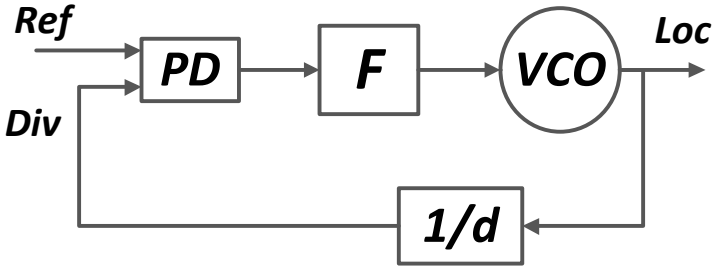


Figure 1.2: Schéma fonctionnel d'une PLL

électronique. De manière simplifiée, elles peuvent être divisées en deux étapes : *(i)* la conception de chaque bloc de la PLL indépendamment les uns des autres assurant leurs comportements locaux désirés de ces blocs ; *(ii)* la conception de la PLL complète au niveau du système global assurant un certain comportement désiré de la PLL globale. La dernière étape consiste généralement en le choix des paramètres du filtre, le bloc le plus souple de la PLL. Ce choix se repose sur les principes de l'Automatique fréquentielle classique. D'un point de vue d'Automatique, une PLL est une boucle de rétroaction où le filtre délivre un signal de commande pour sa correction : c'est pour cela qu'il est aussi appelé le ***correcteur***. Le signal de commande est réalisé par le correcteur selon un algorithme ou ***loi de commande*** dont la conception est appelée la ***synthèse***. De plus amples détails sur le fonctionnement d'une PLL, sur sa modélisation ainsi qu'un exemple de mise en œuvre d'une des méthodes de conception selon une approche automaticienne sont présentés dans le Chapitre 2.

1.2.2 Distribution du signal d'horloge

Une fois généré par le quartz et/ou la PLL, le signal d'horloge est ensuite multiplié et distribué à travers la puce à l'aide des fils conducteurs selon la Figure 1.3. Il s'agit d'un arbre de distribution du signal d'horloge dont la racine est sa source, les branches sont les fils le multipliant et le distribuant et les feuilles sont des cibles vers lequelles le signal d'horloge doit être fourni (les blocs à cadencer).

La Figure 1.3 représente un schéma simplifié d'un arbre de distribution du signal d'horloge. A titre d'illustration, on peut estimer aujourd'hui que les arbres de distribution du signal d'horloge des microprocesseurs occupent typiquement une surface de $20 \times 20\, mm^2$ et peuvent atteindre une longueur totale de $6\, m$! Chaque fil conducteur du signal d'horloge n'est pas idéal et provoque nécessairement une perte d'énergie de ce dernier. Du fait de leur longueur, considérable par rapport à la taille du microprocesseur, les signaux doivent être amplifiés. Cette fonction est assurée par les répéteurs (ou buffers en anglais), représentés par les triangles sur la Figure 1.3. Les répéteurs se composent en général d'une ligne d'inverseurs [1] [Hof04], ce qui explique la représentation graphique.

1. Un inverseur est un dispositif permettant d'inverser une valeur du signal logique à son entrée, par exemple de la valeur « 1 » à « 0 » ou inversement.

L'objectif principal du système de distribution du signal d'horloge est de fournir à sa sortie les signaux d'horloge synchrones nécessaires aux blocs en dépit de sa grande taille et de sa complexité.

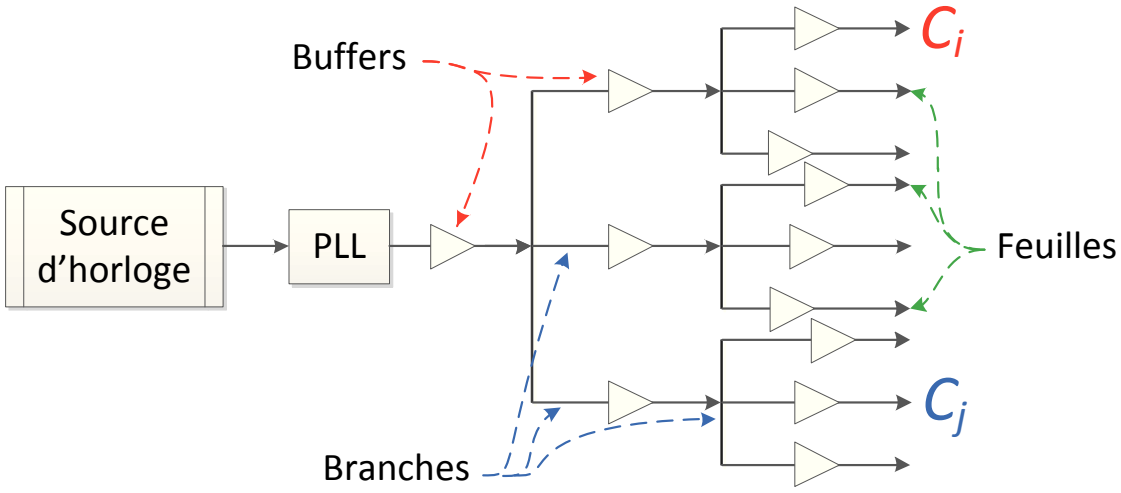


Figure 1.3: L'arbre de distribution classique

1.2.3 Incertitudes du signal d'horloge : Skew et Jitter

Définitions et influence des incertitudes du signal d'horloge

Avant de présenter les différentes sources d'incertitude sur le signal d'horloge, on se propose de comprendre quelles sont les incertitudes principales et quelles sont leurs influences sur le fonctionnement du système entier. Pour des raisons de clarté, on parlera de phénomènes liés aux incertitudes du signal d'horloge de manière simplifiée. Le lecteur intéressé pourra cependant trouver de plus amples précisions sur ce sujet dans [Fri01, RMW+01].

La Figure 1.4 donne une représentation graphique dans le domaine de phase de deux incertitudes du signal d'horloge : skew (décalage en français) et jitter (gigue en français).

Définition 1.1. Le ***skew*** T_{skew} est la différence moyenne entre le temps d'arrivée de deux signaux d'horloge donnant un rythme aux blocs de circuit, c'est-à-dire le retard entre leurs fronts montants C_i et C_j .

Il peut être défini comme [Fri01] :

$$T_{skew} = T_{Cj} - T_{Ci}. \tag{1.1}$$

Regardons maintenant quelle est l'influence du skew sur le fonctionnement du circuit. Imaginons la situation simple de deux bascules adjacentes interconnectées par

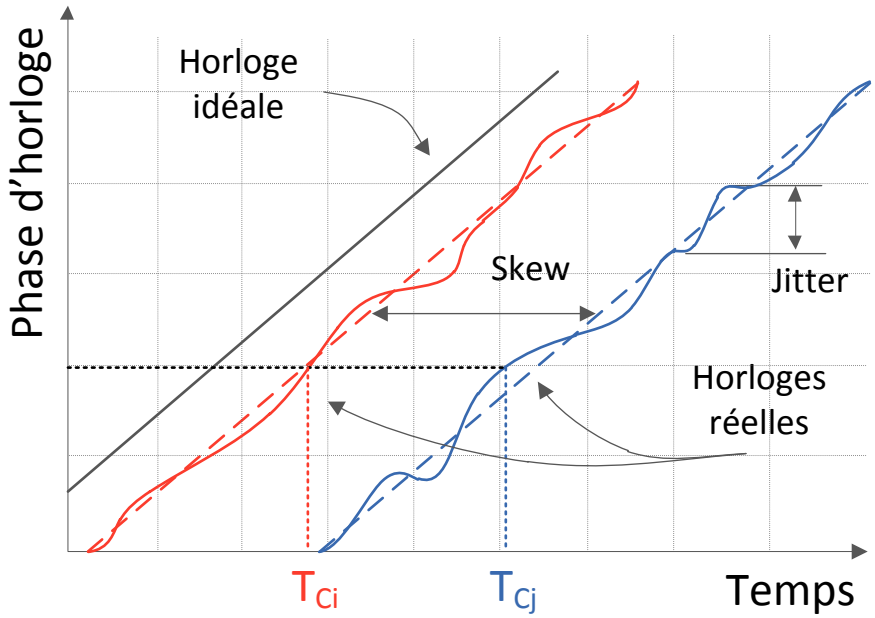


Figure 1.4: Représentation graphique des incertitudes du signal d'horloge

un chemin où la première bascule B_i envoie des données et la deuxième B_j les enregistre (voir la Figure 1.5). Les opérations d'envoi et d'enregistrement des bascules i et j s'effectuent aux instants t_i et $t_j = t_i + T + T_{skew}$ fixés respectivement par les fronts montants de signaux d'horloge C_i et C_j. Cette situation simple est assez représentative de la situation réelle du fonctionnement de circuits modernes complexes, et, en réalité, modélise un échange local de données entre n'importe quelles bascules d'une puce réelle [Fri01].

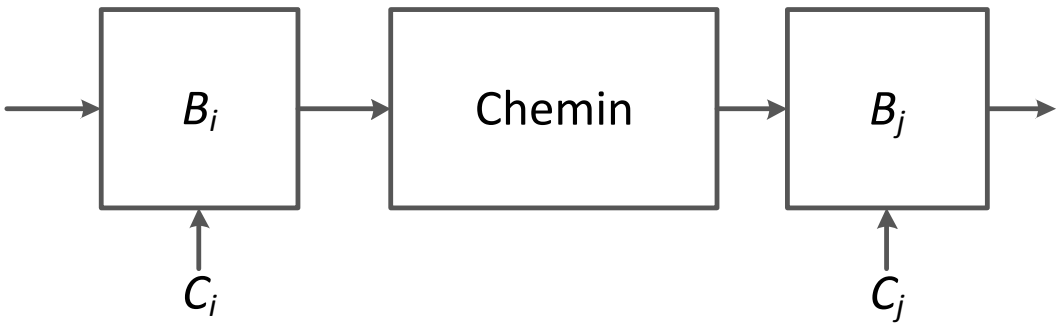


Figure 1.5: Chemin local d'échange de données d'un circuit intégré

En absence de décalage ($T_{skew} = 0$), les données envoyées par la première bascule B_i sont correctement transmises et enregistrées dans la deuxième bascule B_j.

Si les données de la bascule B_i sont maintenant envoyées par le signal d'horloge C_i qui est en retard par rapport à C_j ($T_{skew} < 0$) (ou de la même manière enregistrées dans la bascule B_j par un signal d'horloge C_j en avance par rapport à C_i), le temps

nécessaire à la propagation de données à travers le chemin reliant les deux bascules peut être insuffisant. On observe dans ce cas une sorte « d'allongement » du chemin entre les deux bascules. Cette situation est souvent appelée une *erreur de chemin long* (ou équivalent en anglais : *long-path error*). Pour ne pas enregistrer les données erronées issues du cycle précèdent, on doit augmenter la période d'échantillonnage T et donc diminuer la fréquence d'échantillonnage en dégradant ainsi les performances du circuit.

A l'inverse, quand le signal d'horloge d'envoi de données C_i est en avance par rapport à C_j (ou de manière équivalente le signal d'horloge C_j est en retard par rapport à C_i), les données ont suffisamment de temps pour traverser le chemin de propagation. Cependant, des erreurs fonctionnelles peuvent également se produire dans ce cas-là. Si le signal d'horloge d'envoi est trop en avance, les anciennes données de la bascule B_j, issues du cycle précèdent, peuvent être écrasées par les nouvelles données arrivant en avance. On observe donc une sorte de « réduction » de la longueur du chemin entre les deux bascules. Cette situation est appelée une *erreur de chemin court* (*short-path error* en anglais) et est particulièrement dangereuse car on ne peut pas l'éviter par un simple ralentissement de la fréquence du circuit.

Une autre incertitude du signal d'horloge représentée sur la Figure 1.4 est appelée jitter.

Définition 1.2. Le ***jitter*** est le phénomène de fluctuation de la phase du signal d'horloge qui provoque des variations aléatoires de la période d'échantillonnage.

Un niveau de jitter important peut aussi déclencher des événements en avance ou en retard et donc provoquer les mêmes erreurs fonctionnelles décrites précédemment.

Nous allons par la suite observer les principales sources d'incertitudes évoquées auparavant.

Sources d'incertitudes du signal d'horloge

Les sources d'incertitudes sur le signal d'horloge peuvent être partagées en deux parties : *(i)* causées par le sous-système générant le signal d'horloge et *(ii)* causées par le sous-système le multipliant et le distribuant. La contribution du premier sous-système est négligeable par rapport à la contribution du deuxième. En effet, étant donné que les signaux d'horloge générés par les arbres classiques de distribution d'horloges proviennent de la même source, la seule raison possible de leur déphasage est l'arbre de distribution lui-même. Il n'y a donc pas de skew provoqué par le sous-système de génération du signal d'horloge. D'un autre côté, le niveau de jitter est faible pour le signal généré par un quartz et une PLL bien dimensionnée réduit le jitter causé par son oscillateur [Roh97]. Nous allons donc nous concentrer sur les sources d'incertitudes provoquées par le sous-système de distribution du signal d'horloge.

Étudions tout d'abord le skew. Toutes les sources principales du skew d'un arbre classique de distribution du signal d'horloge peuvent être classées de la manière suivante [Fri01] :

1. Différence de longueurs des branches de l'arbre de distribution du signal d'horloge ;
2. Dispersions technologiques ;
3. Complexité d'interconnexion ;
4. Perturbations extérieures.

La première source de désynchronisation des signaux d'horloge distribués, c'est-à-dire la différence de distances dans leurs parcours à travers la puce, semble être naturelle. En effet, les branches des arbres de distribution avec leurs répéteurs induisent un retard inévitable de propagation du signal d'horloge. Si les longueurs des branches ne sont pas identiques, les retards ne le seront pas non plus, ce qui provoque au final l'apparition du skew. Une des techniques souvent employées est l'utilisation d'arbres de type-H ou de type-X comme représenté sur la Figure 1.6. Dans ce cas-là, la distance parcourue par le signal d'horloge dans chaque branche est la même et par conséquent le retard induit est identique aussi. Un autre problème peut donc se poser ici quand des blocs fonctionnels du circuit, placés géographiquement à des endroits bien précis, interdisent le passage des branches à ces endroits. Il existe cependant des algorithmes de placement et de routage permettant de placer les branches, les blocs fonctionnels du circuit ainsi que des retards supplémentaires afin de garantir certaines contraintes sur le skew [2] [Cad09]. Les retards supplémentaires sont en général réalisés à l'aide d'inverseurs (même principe que les répéteurs) et servent à égaliser le retard de propagation du signal d'horloge à travers les branches qui n'ont pas forcément la même longueur.

Malheureusement, dans un circuit réel, la valeur des retards induits par les blocs (les répéteurs ainsi que des inverseurs supplémentaires de chaque branche) fluctuent. En effet, les dispersions technologiques de fabrication provoquent des différences sur les paramètres des composants passifs et actifs de l'arbre de distribution du signal d'horloge, ce qui entraîne des différences de retard entre les différentes branches par rapport au cas nominal. De nature aléatoire, ces différences sont imprédictibles et leur amplitude augmente avec la complexification du circuit.

La troisième source de skew, la complexité d'interconnexion, est intrinsèquement liée aux dispersions technologiques mais se manifeste plutôt en régime dynamique. La diaphonie [3] entre un grand nombre de branches provoque une impédance parasite ajoutant un certain retard aux signaux d'horloge. A différents endroits de l'arbre de distribution du signal d'horloge, la complexité et la densité d'interconnexion sont différentes et par conséquent les retards induits aussi.

La dernière source de skew correspond aux perturbations extérieures. Elles sont en général provoquées par le fonctionnement du circuit (changement de la température, de l'alimentation, du champ électromagnétique, *etc.*) et influent en temps réel sur le retard de propagation du signal d'horloge à travers les branches.

La première et la deuxième sources de skew sont relativement bien maîtrisées et peuvent être minimisées par les techniques mentionnées précédemment. Néanmoins,

2. Un exemple de contraintes à garantir peut être une valeur maximale de skew pour un endroit critique du circuit.

3. La diaphonie est l'interférence d'un premier signal avec un second causée généralement par de phénomènes d'induction électromagnétique.

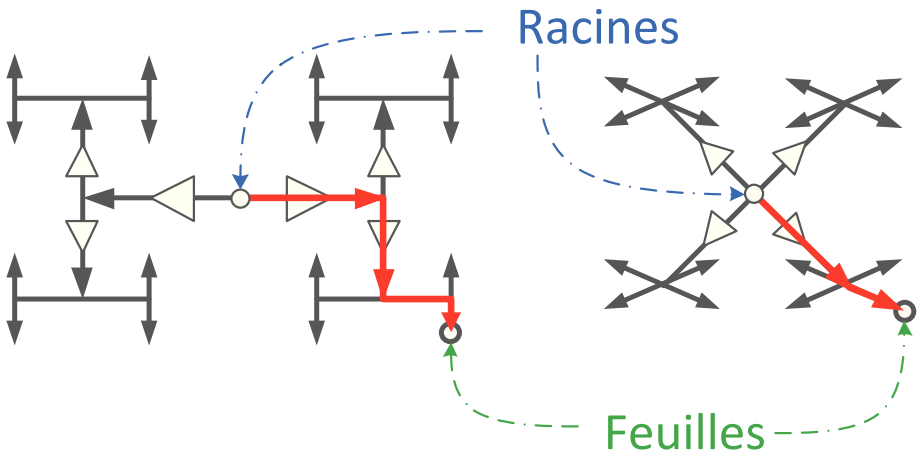


Figure 1.6: Arbres de distribution du signal d'horloge de type-H et type-X

les arbres classiques de distribution du signal d'horloge ne peuvent pas de manière générale compenser en temps réel les deux dernières sources du skew. Elles deviennent encore plus critiques pour des fréquences du signal d'horloge élevées et incitent les concepteurs en Microélectronique à chercher d'autres moyens de génération et de distribution du signal d'horloge incluant des éléments actifs de compensation du skew.

En ce qui concerne le jitter provoqué par le sous-système de multiplication et de distribution du signal d'horloge, ses deux sources principales sont la diaphonie entre les branches voisines et les composants actifs (bruit de flicker [4] de transistors) de l'arbre de distribution du signal d'horloge. Les deux sont fortement liées à la complexité et à la taille de l'arbre de distribution du signal d'horloge et peuvent être minimisées en réduisant la quantité de branches, de répéteurs ainsi que le nombre d'inverseurs supplémentaires dans l'arborescence du signal d'horloge. Pour une architecture de puce donnée, la minimisation du jitter n'est pas toujours possible, ce qui pousse à chercher des moyens actifs de sa compensation ou simplement de son filtrage.

1.2.4 Dissipation d'énergie : réchauffement

Un autre grand problème des arbres classiques de distribution du signal d'horloge est la consommation d'énergie. Une partie de cette consommation est liée à l'alimentation de tous les composants actifs : les répéteurs de branches nécessaires pour amplifier les signaux d'horloge ainsi que les inverseurs introduits artificiellement pour égaliser les retards de branches. La diaphonie dûe à la proximité des lignes entraîne une consommation énergétique supplémentaire de plus en plus importante dans les circuits modernes. Elle devient encore plus critique si l'on augmente la fréquence du signal d'horloge. La courbe sur la Figure 1.7 représente la consommation totale de

4. Le bruit de transistors principalement dû aux impuretés dans le matériau libérant de manière aléatoire des porteurs de charge.

processeurs Intel en fonction de l'évolution des familles technologiques [Anc03] et donc de l'augmentation de la fréquence. Pour les trois derniers microprocesseurs, la consommation dédiée au réseau de distribution du signal d'horloge représente plus de 40% de la consommation totale. Cette consommation d'énergie est dissipée en partie sous forme de chaleur, et si l'évolution de fréquence continuait de progresser au même rythme, elle serait comparable à la dissipation thermique par millimètre carré équivalent à la dissipation d'un réacteur nucléaire classique. C'est l'une des raisons qui a conduit à travailler de nos jours sur la multiplication des cœurs plutôt que sur l'augmentation de la fréquence du signal d'horloge.

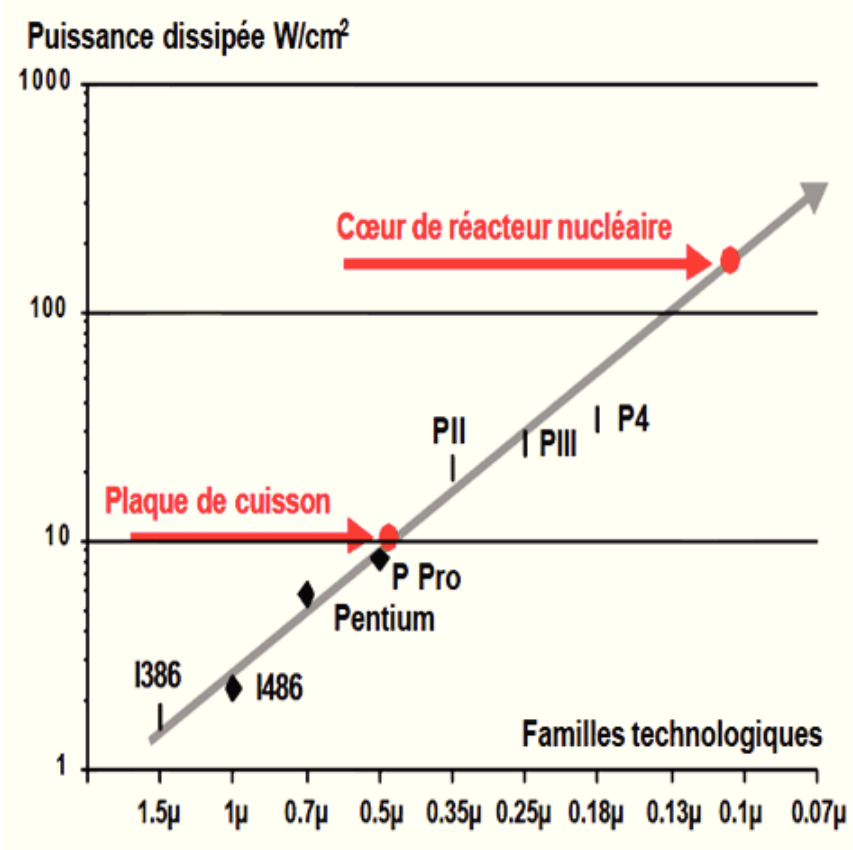


Figure 1.7: Évolution de la dissipation d'énergie de microprocesseurs Intel [Anc03] en fonction de la longueur des transistors de familles technologiques associées

1.2.5 Du problème de distribution au problème de synchronisation

En regardant de plus près, la majorité des problèmes des réseaux d'horlogerie classiques sont liés à la multiplication et à la distribution d'un signal d'horloge source. Une autre alternative à cette architecture pourrait être une architecture distribuée avec des moyens de génération de plusieurs signaux d'horloge déjà distribués en différents points d'un circuit. Le problème n'est alors plus la multiplication et la distribution d'un signal d'horloge mais la synchronisation des signaux d'horloge générés, ce qui est potentiellement plus compliqué. En effet, l'architecture classique a

été initialement choisie pour sa simplicité car elle ne nécessite pas de dispositifs supplémentaires de génération et de synchronisation des signaux d'horloge. Dans le cas idéal, provenant de la même source, les signaux sont déjà synchrones. Néanmoins, avec l'évolution des circuits intégrés ainsi qu'avec l'augmentation de leur complexité et de leur fréquence, l'architecture classique du réseau de distribution du signal d'horloge est de plus en plus complexe. Elle inclut maintenant un nombre de plus en plus grand de dispositifs supplémentaires nécessaires pour respecter les contraintes de plus en plus serrées sur les incertitudes du signal d'horloge. En prenant en compte le fait que certaines de ces contraintes ne peuvent toujours pas être satisfaites en temps réel, la question se pose alors : « pourquoi ne pas utiliser des architectures alternatives dont la complexité n'est pas forcement beaucoup plus importante ? ». Il s'agit de remplacer le problème de multiplication et de distribution d'une seule source du signal d'horloge par le problème de synchronisation de plusieurs sources. L'avantage de ce dernier est qu'il peut être assuré et maintenu en temps réel. On obtient donc une architecture active qui permet de générer et de synchroniser des signaux d'horloge en temps réel pour les circuits intégrés. Mais avant de parler de différentes architectures actives, on se propose de voir comment la synchronisation est réalisée dans la nature.

1.3 Synchronisation dans la nature

L'un des exemples de synchronisation les plus spectaculaires que l'on puisse trouver dans la nature est la synchronisation des lucioles en Asie de Sud-Ouest, en Afrique et en Amérique du Nord. Il existe une race de lucioles dont les mâles émettent de la lumière en clignotant avec une période d'environ $0.9\,sec$. Il s'agit de la *Lampyris noctiluca L.* (Animalia : Arthropoda : Insecta : Coleoptera : Lampyridae) dont la photographie est présentée sur la Figure 1.8.

Lorsque les lucioles mâles se rassemblent sur un ou plusieurs arbres voisins, un phénomène intéressant de synchronisation se produit. Chaque luciole observe les lucioles voisines et adapte sa propre fréquence de clignotement à celle des autres. Dans le cas où elle est trop en avance ou en retard par rapport aux lucioles voisines, elle ralentit ou accélère afin de s'aligner avec les autres. Au final, toutes les lucioles clignotent en même temps, à l'unisson c'est-à-dire de manière synchrone. Le phénomène a attiré une grande attention chez les biologistes. Le premier travail fut réalisé par [Buc38], qui rédigea 50 ans plus tard l'article [Buc88] expliquant et modélisant le phénomène en détail. Il a été étudié également dans [Han83] la possibilité d'imposer une fréquence de clignotement par une source de lumière extérieure si cette fréquence n'est pas trop éloignée de la fréquence naturelle des lucioles. Un programme modélisant le comportement des lucioles basé sur le modèle décrit dans [Buc88] peut être retrouvé dans [Wil97].

Dans la présente étude, on ne rentre pas dans les détails du phénomène de la synchronisation des lucioles d'un point de vue biologique, mais on va plutôt essayer de retenir les idées principales permettant sa réalisation. Pour atteindre un objectif global, c'est-à-dire la synchronisation globale de certains évènements (clignotement à l'unisson), les lucioles doivent être capables de réaliser deux tâches :

Figure 1.8: Photo de *Lampyris noctiluca L.*, luciole émettant de la lumière

1. au niveau local : adapter une grandeur locale après sa comparaison avec des grandeurs d'entrée extérieures. La grandeur doit être liée à la synchronisation, qui, dans le cas des lucioles, peut être par exemple la fréquence et la phase de clignotement ;
2. au niveau global : interagir et échanger une certaine information avec d'autres lucioles. Par exemple, afin de se synchroniser, chaque luciole doit envoyer aux autres sa propre phase et sa propre fréquence de clignotement.

Il est important de mentionner que l'échange d'information ne doit pas être obligatoirement réalisé entre toutes les lucioles, mais peut être fait seulement entre une partie de l'ensemble des lucioles, les lucioles voisines par exemple. En effet, il n'est pas nécessaire que les lucioles placées au sommet des arbres puissent voir les lucioles en bas des arbres pour se synchroniser globalement. Il est également possible de maintenir la synchronisation en effectuant l'échange et l'adaptation de la fréquence tous les d cycles au lieu de chaque cycle de clignotement.

On trouve d'autres exemples de synchronisation dans la nature comme le mouvement synchrone de bancs de poissons ou de volées d'oiseaux qui se basent sur le principe décrit ci-dessus. Dans ce cas-là, chaque espèce est capable de remplir les deux tâches mentionnées précédemment nécessaires pour atteindre un comportement global désiré. Un être humaine n'est pas une exception et peut aussi servir comme un exemple de la synchronisation dans la nature : applaudissements rythmés dans une salle de concert, défilé d'une troupe de soldats lors d'une revue militaire *etc.*.

Ces exemples de synchronisation ainsi que beaucoup d'autres peuvent être trouvés dans [Str01]. Sans entrer dans plus des détails sur l'analogie de la synchronisation dans la nature, nous allons parcourir dans la section suivante l'évolution de l'architecture des arbres classiques de génération et de distribution du signal d'horloge

aboutissant à une architecture dont le principe de fonctionnement est semblable au phénomène de la synchronisation des lucioles. En effet, au fil de l'évolution des technologies de Microélectronique, on remarque que les architectures classiques centralisées se transforment peu à peu en une architecture distribuée active de génération de signaux d'horloge. Les sous-systèmes de cette architecture distribuée réalisent en temps réel, comme les lucioles, les deux tâches élémentaires évoquées précédemment nécessaires à la synchronisation globale.

1.4 D'une architecture classique à une architecture distribuée active

1.4.1 Grilles

Un premier pas dans l'évolution de l'architecture classique vers une architecture distribuée est l'utilisation de grilles. Des grilles introduites localement en différents endroits d'un circuit sont souvent utilisées dans les architectures classiques de distribution du signal d'horloge [BB98, APK+01]. Un exemple de grille est représenté sur la Figure 1.9. Les grilles permettent de diminuer la différence de potentiel de différents points à leurs entrées et donc de minimiser le skew. Les grilles très denses présentent une meilleure caractéristique de ce point de vue. Même si quelques lignes de la grille ne sont pas parfaitement positionnées, l'impact sur l'incertitude du signal d'horloge sera négligeable. De plus, le jitter et les perturbations locales sont moyennés sur la structure entière. Cependant, les grilles peuvent dissiper beaucoup d'énergie car, par conception, elles contiennent un grand nombre de lignes redondantes dont l'impédance est grande surtout pour les hautes fréquences. En outre, les grilles restent toujours un moyen de distribution du signal d'horloge passif sans aucune compensation en temps réel.

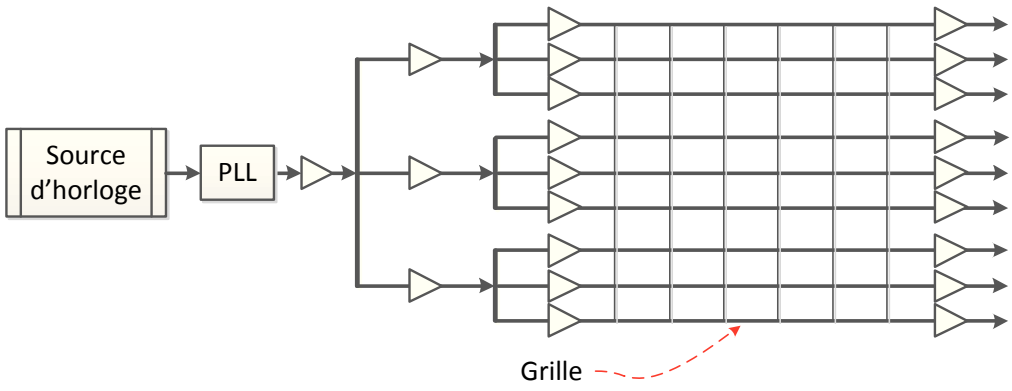


Figure 1.9: Exemple d'utilisation d'une grille dans les arbres classiques de distribution du signal d'horloge

Beaucoup d'architectures avec une stratégie de compensation active d'incertitudes

ont été proposées dans la littérature mais peu d'entre elles ont été utilisées en pratique. Les différentes stratégies de compensation des incertitudes du signal d'horloge peuvent être classées en deux catégories : centralisées ou décentralisées.

1.4.2 Compensation centralisée du skew

Les réseaux de distribution du signal d'horloge avec la compensation centralisée du skew se basent habituellement sur l'asservissement global de la phase du signal d'horloge. Ils ont un correcteur centralisé, appelé encore « le compensateur » du skew. Son rôle est d'assurer que les signaux envoyés vers les différentes régions du circuit sont synchrones. Ces régions sont aussi appelées les régions isochrones du fait qu'elles se synchronisent après une certaine période transitoire.

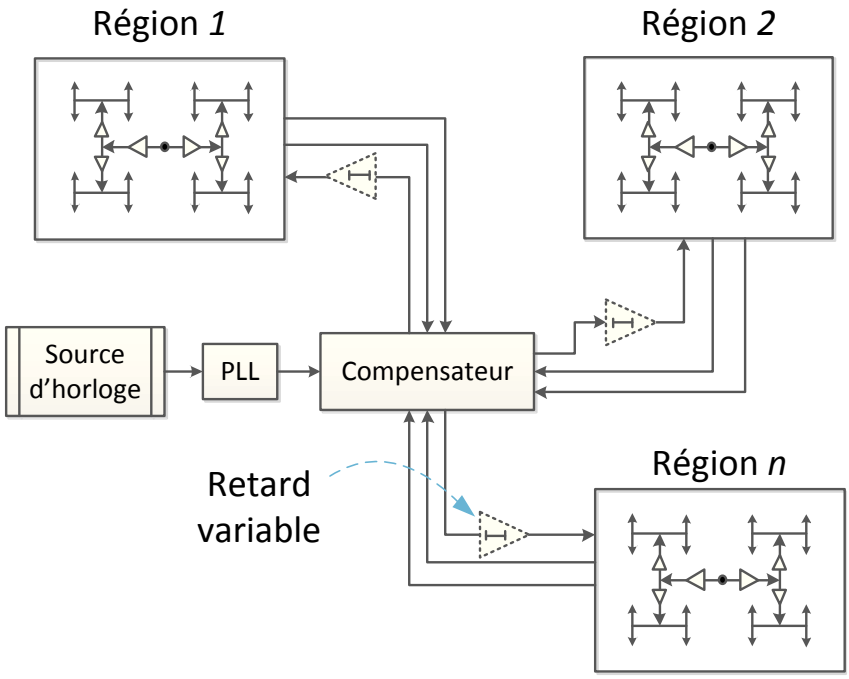


Figure 1.10: Compensation centralisée du skew proposée dans [HLCF98] et [SY98]

L'implémentation proposée dans [HLCF98] est montrée sur la Figure 1.10. Chaque région isochrone possède un arbre local de distribution du signal d'horloge. Le réseau de distribution exige la réalisation propre des segments d'interconnexion de chaque région : un lien pour le chemin direct (forward path) et deux liens pour le retour (feedback path). Les signaux directs fournissent le signal d'horloge aux racines de chaque arbre de la région locale. Un des signaux de retour est également connecté à la racine, l'autre est connecté à une des feuilles de l'arbre local. Le premier chemin a pour but de mesurer le retard d'aller-retour du signal d'horloge, le deuxième est utilisé pour mesurer un retard de propagation du signal d'horloge au sein de l'arbre local. Le compensateur est capable de corriger l'erreur de phase par une accélération ou un ralentissement du chemin direct. L'approche proposée a deux inconvénients principaux : la résolution du circuit de mesure est relativement faible et la précision de la correction de phases appliquée aux signaux directs est limitée. De plus, il est

difficile d'implémenter cette stratégie à cause des exigences d'égalité imposées sur les branches aller-retour d'interconnexions des différentes régions.

Un autre réseau de distribution du signal d'horloge avec la compensation active semblable à la précédente est proposé dans [SY98]. Cependant, les auteurs de [SY98] utilisent pour chaque région isochrone un seul signal de retour, et non deux. Ils y ajoutent aussi un comparateur de phase dans le compensateur et une ligne à retard avec un retard variable (en pointillé sur la Figure 1.10). Ce dernier est implémenté dans le chemin direct. La compensation du skew s'effectue en deux étapes. Durant la première étape, le comparateur de phase compare le temps d'arrivée du signal d'horloge de référence au niveau du compensateur de skew au temps d'arrivée du signal d'horloge issu de la région locale. Le compensateur ajuste la ligne à retard jusqu'à ce qu'il ne subsiste plus de retard entre les deux signaux d'horloge. Dans ce cas-là, le signal d'horloge issu de la région locale est synchronisé avec le signal d'horloge de référence et le retard du chemin aller-retour est mémorisé. La deuxième étape consiste en une division par deux du retard de la ligne directe, ce qui garantit la synchronisation des signaux d'horloge délivrés à chaque région (en opposition par rapport à l'approche précédente) avec le signal d'horloge de référence. Le principal problème de cette stratégie de compensation est qu'elle ne peut être effectuée que durant la phase du « reset » de la puce. La raison est que le processus d'alignement perturbe significativement la phase des signaux d'horloge distribués. Cet algorithme exige aussi un comportement très linéaire des lignes à retard avec une grande résolution. Par ailleurs, il ne résout pas, comme dans l'approche précédente, le problème de précision des lignes d'interconnexion de chaque région.

1.4.3 Compensation décentralisée du skew

Pour pouvoir compenser le skew du signal d'horloge, il n'est pas nécessaire de le renvoyer vers la source. Cela permet de réduire la dissipation d'énergie et diviser par deux le niveau du jitter dans le détecteur de phase. Une alternative plus simple et plus pratique consiste à utiliser la compensation distribuée de skew. Le microprocesseur de 650 MHz décrit dans [SFRY99] utilise un réseau de distribution du signal d'horloge composé de deux troncs. Chaque tronc est gouverné par son propre arbre d'horloge (voir la Figure 1.11). Il est possible d'ajuster finement le signal l'horloge distribué dans chaque tronc au moyen d'un retard réglable. Un détecteur de phase (PD) localisé entre les deux troncs compare les temps d'arrivée des fronts montants du signal d'horloge. En cas de déséquilibre entre les deux troncs, les retards de la chaîne de distribution sont ajustés. Cette approche est particulièrement efficace pour compenser le déphasage lié aux interconnexions, aux variations de température de processus et de tension.

Une autre étape dans le contournement du problème de distribution du signal d'horloge par le problème de synchronisation est l'utilisation de multiples oscillateurs. Les réseaux de distribution du signal d'horloge avec de multiples oscillateurs évitent, en règle générale, le routage de lignes d'interconnexion de l'horloge à travers de longues distances. Ils peuvent potentiellement résoudre le problème d'accumulation de jitter issu des stratégies de distribution classiques en utilisant des filtres pour son atté-

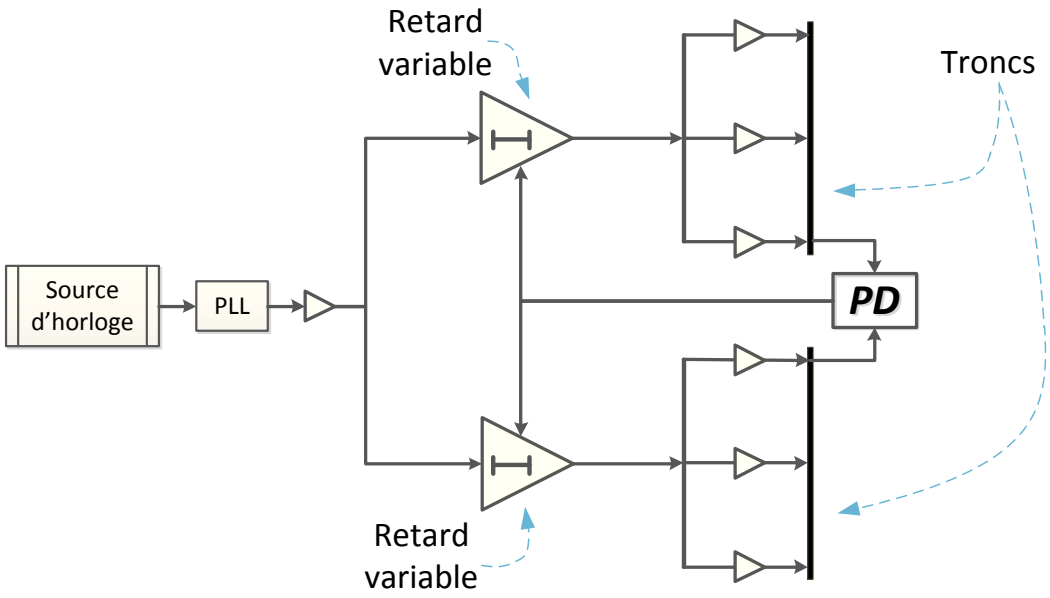


Figure 1.11: Compensation décentralisée du skew [SFRY99]

nuation. Comme précédemment pour le cas de compensation du skew, le filtrage de jitter peut être centralisé ou non. L'approche centralisée correspond à un réseau de distribution du signal d'horloge avec un seul filtre [MI98] ce qui est présenté dans la sous-section 1.4.4. Les réseaux de distribution du signal d'horloge à plusieurs filtres correspondent à une stratégie décentralisée [PN95] et [GC00] ce qui est présenté dans la sous-section 1.4.5.

1.4.4 Filtrage centralisé du jitter

Les réseaux de distribution du signal d'horloge utilisant un filtrage centralisé du jitter ressemblent à une PLL conventionnelle, avec comme différence principale l'utilisation d'oscillateurs multiples distribués sur la puce. Un tel système reçoit comme un signal de référence le signal issus de quartz (source d'horloge) et/ou une autre PLL classique représentée en pointillé sur la Figure 1.12. Comme nous le montre la Figure 1.12, le filtre commande les différents oscillateurs individuels (*VCO*s). Les signaux issus des différents oscillateurs sont moyennés afin de délivrer des signaux d'horloge synchrones aux récepteurs.

Un réseau de distribution du signal d'horloge complet utilisant les oscillateurs contrôlés en tension est décrit dans [MI98]. L'architecture proposée dans ce travail est montrée sur la Figure 1.13. La PLL considérée ici est construite autour d'un comparateur de phase *PD*, d'un filtre passe-bas *F*, et de *VCO*s. Le chiffre à côté de chaque *VCO* indique le nombre d'oscillateurs voisins les plus proches. Les sorties des différents oscillateurs sont connectées par des lignes de longueurs égales l. Cette interconnexion oblige tous les oscillateurs à osciller de manière synchrone et à fournir leur signal d'horloge synchrone à chaque réseau de distribution local. Les auteurs

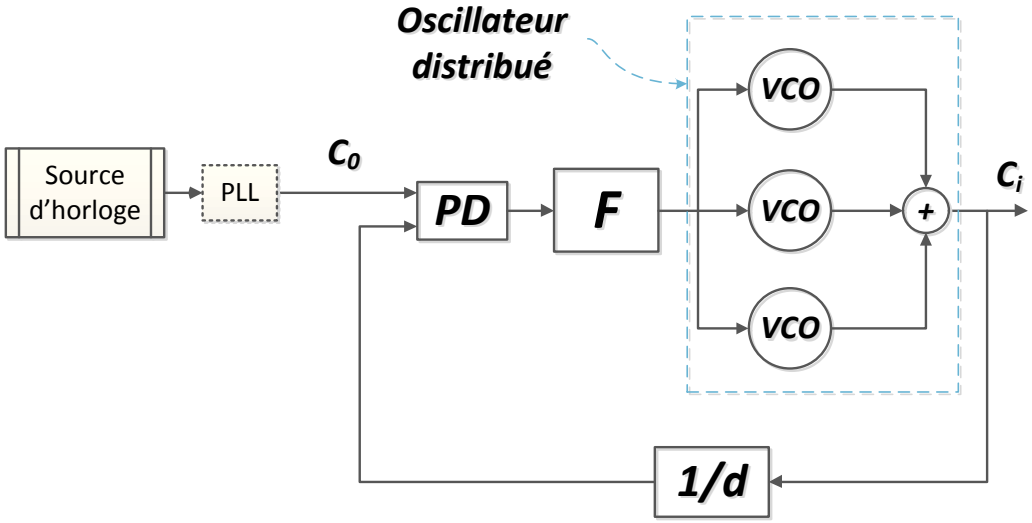


Figure 1.12: Filtrage centralisé de la jitter

de [MI98] proposent aussi, comme une alternative à l'architecture représentée sur la Figure 1.13, de relier les sorties d'oscillateurs par une grille 2D avec des longueurs égales.

Le problème de distribution du signal d'horloge depuis une seule source est remplacé par le même problème de distribution mais cette fois du signal de tension analogique issu du filtre passe-bas F. Le signal à distribuer maintenant est un signal basse-fréquence de commande qui n'oscille pas. Malheureusement il dissipe beaucoup d'énergie statique et une protection adéquate contre le bruit est également nécessaire. Par ailleurs, il faut prendre en compte l'effet des bords et les défaillances possibles qui peuvent entraîner un déphasage supplémentaire.

1.4.5 Réseau actif de distribution du signal d'horloge

La prochaine étape est l'utilisation d'un réseau de distribution du signal d'horloge complètement décentralisé y compris pour le filtrage. Un réseau de distribution du signal d'horloge avec de multiples oscillateurs, filtres et détecteurs de phase présente d'une certaine façon les mêmes caractéristiques qu'une pluralité de PLLs totalement distribuées sur la puce, ce qui signifie que la génération, la comparaison et le filtrage du signal d'horloge local sont décentralisés. Cette architecture a été initialement proposée par [PN95] et réalisée par [GC00]. Schématiquement, elle peut être représentée par une interconnexion orthogonale en deux dimensions (2D) de nœuds Figure 1.14. Chaque nœud est une PLL à entrées multiples (la partie gauche de la Figure 1.14) qui est composée comme une PLL classiques d'un VCO, d'un filtre F, d'un diviseur de fréquence $1/d$, des comparateurs de phase PDs ainsi qu'un bloc de sommation pondérée Σ. En fonction de la localisation de la PLL au sein du réseau, elle peut avoir deux, trois ou quatre voisines. Grâce aux comparateurs de phase, chaque PLL mesure le déphasage de son signal d'horloge locale par rapport aux signaux d'horloge

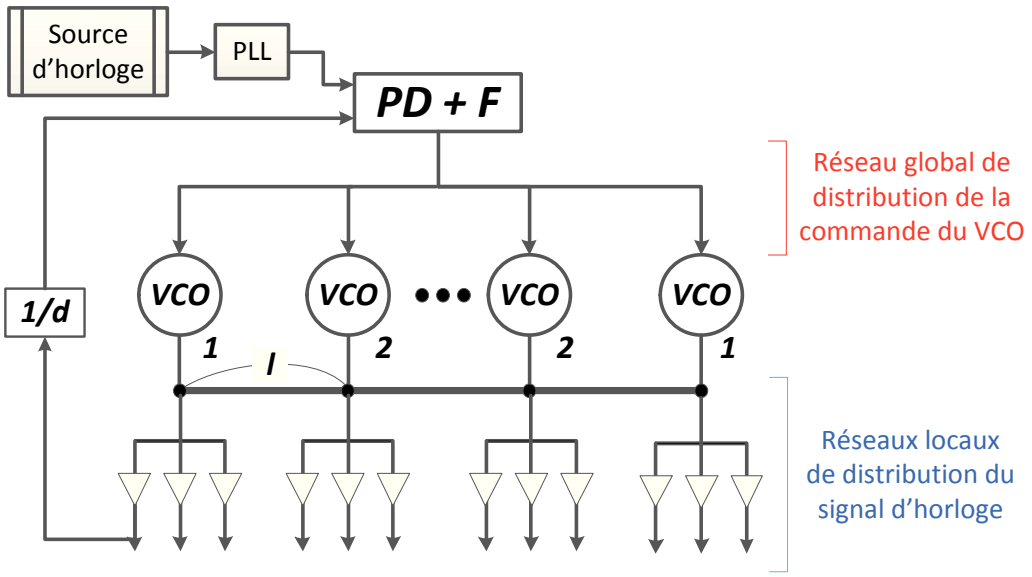


Figure 1.13: Réseau de distribution du signal d'horloge avec *VCO*s distribués et un seul filtre proposé dans [MI98]

des PLLs voisines. Elle a par conséquence la même quantité d'entrées et donc de comparateurs de phase que la quantité de ses PLLs voisines. Les auteurs de [GC00] proposent de partager chaque comparateur de phase entre deux PLLs adjacentes en les plaçant à leur mi-chemin. Cela va permettre de compenser l'influence des retards éventuels en propagation des signaux d'horloge et d'utiliser moins de comparateurs de phase et donc de composants électriques. Les erreurs de phase provenant des comparateurs de phases sont additionnées, filtrées à l'intérieur de chaque nœud et ensuite utilisées pour ajuster la phase et la fréquence de l'oscillateur local. Cela permet de les rendre égales à la phase et à la fréquence moyennes des oscillateurs voisins. Lorsque chaque nœud est synchronisé en moyenne avec ses voisins, alors l'ensemble du réseau est censé être synchronisé en fréquence et en phase. Les signaux d'horloge locales synchrones peuvent être ensuite facilement distribués à l'aide de méthodes classiques simples au vue de la petite taille des zones locales. Pour imposer une fréquence fixe désirée, il suffit de remplacer un ou plusieurs nœuds par une source d'oscillation extérieure (horloge maître). Au final, on obtient un réseau de génération et de distribution du signal d'horloge actif. « Actif » signifie ici que la mesure et la compensation du déphasage se font en temps réel et donc l'influence de toutes les perturbations possibles (température, impédance parasite, dispersion, vieillissement, défaillance, bruit) sur la synchronisation peut être palliée à tout instant par une loi de commande appropriée. C'est ce qui constitue la différence majeure entre des réseaux actifs et des réseaux classiques de distribution du signal d'horloge.

On peut établir une certaine analogie entre la synchronisation du réseau actif de PLLs et le phénomène de synchronisation de lucioles décrit dans la section précédente (page 18). En effet, comme les lucioles, les PLLs ont la capacité de remplir les deux tâches principales nécessaires à leur synchronisation.

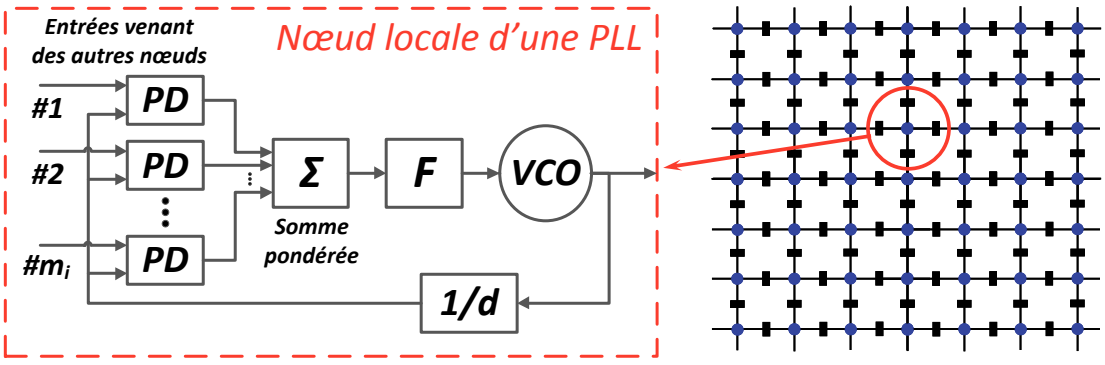


Figure 1.14: Architecture active de distribution du signal d'horloge

1. Au niveau local : adapter une grandeur locale via les filtres et *VCO*s, c'est-à-dire adapter la phase et la fréquence du signal d'horloge après leur comparaison avec les phases et les fréquences de signaux extérieurs, réalisée par les PDs.
2. Au niveau global : interagir et échanger une certaine information via les fils d'interconnexion, c'est-à-dire l'information de la phase et de la fréquence du signal d'horloge avec d'autres PLLs.

Comme dans le cas des lucioles, l'échange d'information ne s'effectue pas avec toutes les PLLs du réseau mais seulement avec une partie d'entre elles, notamment avec les PLLs voisines. Le bloc de division de fréquence $1/d$ donne la possibilité de réaliser cet échange chaque d cycle, c'est-à-dire en basses fréquences, tout en gardant les fréquences locales élevées.

En plus des avantages dynamiques possibles mentionnés précédemment, un réseau de distribution du signal d'horloge actif possède un certain nombre d'avantages intrinsèques. En effet, dans cette architecture, plutôt que d'utiliser de larges branches de distribution, on utilise de simples lignes d'interconnexion des nœuds voisins. Cela permet de diminuer la diaphonie parasite entre les lignes d'interconnexion existant dans les arbres d'horloge classiques et donc de réduire la consommation et l'échauffement de la puce. De plus, la possibilité d'utiliser des signaux en basses fréquences pour l'interconnexion diminue significativement la consommation et l'échauffement liés à la diaphonie parasite. En outre, le problème d'accumulation de jitter associée à la stratégie de distribution conventionnelle est évité car le signal d'horloge est régénéré à chaque fois, juste avant son interaction avec la logique séquentielle.

Conclusion : Problématique de la thèse

Dans la section précédente, on a évoqué les avantages potentiels de réseaux actifs de distribution du signal d'horloge. Mais comment concevoir un tel système complexe ?

Une première approche que nous allons appeler par la suite de ce document *l'approche locale*, utilisée actuellement du fait de sa simplicité en Microélectronique est l'approche suivante composée de trois étapes.

(i) Synthèse locale : on suppose que l'influence du réseau ou de l'interconnexion de PLLs sur la dynamique globale du système est négligeable et on réalise la synthèse d'une PLL seule afin de garantir son comportement désiré local.

(ii) Analyse locale de robustesse : une fois que la synthèse locale est effectué, on réalise l'analyse locale du comportement de la PLL en présence des incertitudes paramétriques inévitables lors de sa fabrication.

(iii) Analyse du comportement globale nominale : une fois que la synthèse et l'analyse de robustesse locales sont effectuées, on interconnecte les modèles LTIs des PLLs identiques ainsi obtenues en réseau selon une topologie donnée en vérifiant si le comportement désiré est conservé en global. Si ce n'est pas le cas, on évalue les différences.

Les méthodes de synthèse (point *(i)* de l'approche locale de la conception du réseau de PLLs) d'une PLL seule ainsi que les méthodes d'analyse de son comportement dynamique sont relativement bien connues et maîtrisées par les concepteurs électroniques [Roh97, Kro03]. Elles se basent en générale sur les approches en domaine fréquentiel étant donné que ces dernières permettent une formulation mathématique puissante et plus simple par rapport aux approches en domaine temporel et donc facilement exploitable sur l'ordinateur. Ce point est important dans l'optique de faciliter l'insertion de la procédure de synthèse d'une PLL dans le flux de conception assisté par ordinateur (CAO) existant en Microélectronique.

D'un point de vue automatique, une PLL peut être représentée, sous certaines hypothèses, par un modèle *Linéaire Invariant dans le Temps* (LTI - *Linear Time Invariant* en anglais) dans le domaine de phase [5] et on peut de façon maîtrisée synthétiser une loi de commande dans le cas général assurant les bonnes marges de stabilité et certaines exigences en terme de performance utilisant les méthodes de synthèse classiques [Roh97, Kro03] et plus récents [Hou07, KCV08] de l'Automatique fréquentielle.

Les deux approches de commande récentes citées se basent sur la synthèse H_∞ standard « 2 blocs » [Hou07] et la synthèse « H_∞ loop-shaping » [KCV08] à partir du modèle LTI d'une PLL. Les deux résultats permettent de calculer des lois de commande assurant la stabilité vis-à-vis d'une incertitude dynamique dans le modèle de la PLL ainsi que certaines spécifications de performance dans le cas nominal : la synchronisation de la PLL et le filtrage de bruit de phase de l'oscillateur (ou jitter) à sa sortie.

Cependant, selon l'application, ces deux critères de performance peuvent être insuffisants pour garantir un bon fonctionnement de la PLL. En effet, comme nous allons le voir dans la sous-section 2.5.1 du chapitre 2, il est également important d'assurer que le correcteur filtre le bruit et rejette la perturbation non seulement à la sortie mais aussi à l'entrée de l'oscillateur tout en garantissant un niveau de la commande modéré. Comment peut-on garantir les spécifications de performance plus générales ?

D'un autre côté, une seule incertitude dynamique peut être insuffisante pour modéliser les dispersions réelles dans le circuit réalisant la PLL. En réalité il existe

5. Nous développerons plus en détails cet aspect dans le chapitre 2.

plusieurs incertitudes au sein d'une PLL qui sont en général modélisées par des incertitudes paramétriques. Afin d'assurer le bon comportement de la PLL vis-à-vis de ces incertitudes des méthodes d'analyse basées sur les simulations Monte-Carlo [RK08, Kra98, Kra06] sont généralement utilisé dans le domaine de Microélectronique. Malheureusement, si dans ce domaine on dispose de méthodes d'analyse efficaces pour des circuits simples, l'analyse d'une PLL et a fortiori d'un réseau de PLLs est beaucoup plus complexe, voire pratiquement impossible. Est-il possible d'obtenir une méthode efficace pour analyser une PLL ou un réseau de PLLs interconnectées en prenant en compte les dispersions technologiques.

L'analyse du comportement du modèle LTI globale nominal (l'étape *(iii)* de l'approche de conception locale) peut être résolu de manière efficace même dans le cas d'un tel système complexe comme le réseau de PLLs. En effet, la stabilité globale peut être testée en faisant un calcul de valeurs propres du système global [6]. Concernant la performance, ce qui est souvent utilisé dans le domaine de Microélectronique, c'est l'évaluation de la « *différence* » entre la performance d'une PLL unique et la performance correspondante du réseau de PLLs. Si la différence est « faible », la performance du réseau de PLLs global obtenu est jugée acceptable.

Mais comment doit-on procéder dans le cas où le réseau de PLLs a une influence non négligeable sur la performance globale ? Est-ce qu'on peut garantir un bon comportement global d'un réseau de PLLs tout en gardant la démarche de synthèse locale simple évoquée précédemment ? En général, il n'est pas toujours possible d'obtenir un résultat satisfaisant en procédant selon l'approche de conception locale décrite ci-dessus, soit parce que la stabilité n'est pas garantie, soit parce que le niveau de performance du système global est considérablement dégradé par rapport au cas local. Ceci est vrai même dans le cas où toutes les PLLs sont identiques et proprement localement synthétisées.

Le chapitre suivant présente l'application de l'approche classique de la synthèse locale et de l'analyse de robustesse locale (point *(i)* et *(ii)*) avec quelques améliorations.

1. On propose d'améliorer la synthèse de la loi de commande par l'utilisation du « critère H_∞ 4-blocs » [Fon95]. Cela permet par rapport au critère utilisé dans [Hou07] de prendre en compte plus de spécifications lors de la synthèse.
2. On propose de combiner la méthode d'analyse Monte-Carlo en Microélectronique avec la méthode de μ-analyse en Automatique fréquentielle avancée, ce qui permet de réduire considérablement la difficulté du problème d'analyse.

La dernière section du chapitre 2 illustre les limites de l'approche locale de conception et l'importance de l'interconnexion sur le comportement du réseau entier à travers d'un exemple numérique. Il est donc primordial de prendre en compte la contribution de l'interconnexion et de réaliser le lien « local-global » afin de garantir les propriétés globales *a priori* en phase de conception, ce qui constitue la problématique principale de cette thèse.

Cependant, on retient les idées important de l'approche locale qui vont être utilisées tout au long de ce travail de thèse afin de proposer une approche originale tenant en

6. Il s'agit de valeurs propres de la matrice d'évolution d'une représentation d'état du système globale.

compte ce lien, à savoir : *(i)* la facilité de la synthèse locale de la loi de commande, *(ii)* l'évaluation de la performance du système global à travers de sa comparaison à la performance local d'un sous-système.

Malheureusement, dans le domaine de la Microélectronique il n'existe ni méthode permettant une synthèse du réseau de PLLs garantissant *a priori* certaines propriétés globales, ni méthode d'analyse de la robustesse du réseau complet efficace du fait de la complexité de ce dernier. En effet, par nature, étant un système dynamique actif complexe nécessitant la synthèse appropriée d'une loi de commande et une étude approfondie de son comportement dynamique, il sort du cadre d'étude du microélectronicien et oblige à prendre en compte les outils et les méthodes de l'Automatique. Mon travail de thèse s'inscrit dans ce cadre-là. L'Automatique souvent négligée en Microélectronique peut apporter les solutions à ses problèmes et notamment sur celui de la distribution du signal d'horloge active.

Pour ces raisons, la fin de ce chapitre présente une formalisation des problèmes de synthèse et d'analyse considérés dans la thèse du point de vue Automatique. Nous supposons que le réseau de génération et de distribution du signal d'horloge est modélisé dans domaine des phases par un système LTI de grande dimension. Un exemple d'une telle modélisation est présenté dans la section 5.2 du chapitre 5. Le système de grande dimension considéré est un système modélisé par une interconnexion de N sous-systèmes identiques T_s (voir la Figure 1.15). Pour des raisons de clarté et sans perte de généralité, le cas du sous-système carrés c'est-à-dire le sous-système T_s avec le même nombre p d'entrées et de sorties, est considéré par la suite. Chaque sous-système est LTI et peut être divisé en deux parties : *(i)* une partie de sous-système H, appelé « procédé à commander » et *(ii)* une partie réalisant la commande K, appelé « correcteur ». L'ensemble de sous-systèmes est décrit par un opérateur LTI global $\widetilde{T}$ et leur interconnexion par un opérateur LTI M :

$$\begin{array}{rcl} \varphi & = & \overbrace{(I_N \otimes T_s)}^{\widetilde{T}} \quad r \\ \begin{bmatrix} r \\ z \end{bmatrix} & = & \underbrace{\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}}_{M} \begin{bmatrix} \varphi \\ w \end{bmatrix} \end{array} \tag{1.2}$$

avec $T_s = \underbrace{\begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}}_{H} \star K$, $r(t), \varphi(t) \in \mathbb{R}^{pN}$, $w(t) \in \mathbb{R}^{n_w}$, $z(t) \in \mathbb{R}^{n_z}$ et où l'opération du produit Redheffer $\star$ et du produit Kronecker $\otimes$ sont respectivement définies dans la sous-section 2.5.3, page 74 et l'annexe C, page 138 du chapitre 3.

Le système (1.2) représente un système global de grande dimension. Tout au long de cette étude, nous considérons la stabilité globale et locale ainsi que la performance globale et locale. Le sous-système local considéré correspondant est représenté sur

la Figure 1.16 et décrit par :

$$\begin{array}{ccc} \left[\begin{array}{c} z_{pi} \\ y_i \end{array}\right] & = & \overbrace{\left[\begin{array}{cc} \widetilde{H}_{11} & \widetilde{H}_{12} \\ \widetilde{H}_{21} & \widetilde{H}_{22} \end{array}\right]}^{\widetilde{H}} \left[\begin{array}{c} w_{pi} \\ u_i \end{array}\right] \\ u_i & = & K \qquad\quad y_i \end{array} \tag{1.3}$$

avec $w_{pi}(t) \in \mathbb{R}^{n_{wp}}$, $z_{pi}(t) \in \mathbb{R}^{n_{zp}}$, $y_i(t) \in \mathbb{R}^{n_y}$, $u_i(t) \in \mathbb{R}^{n_u}$.

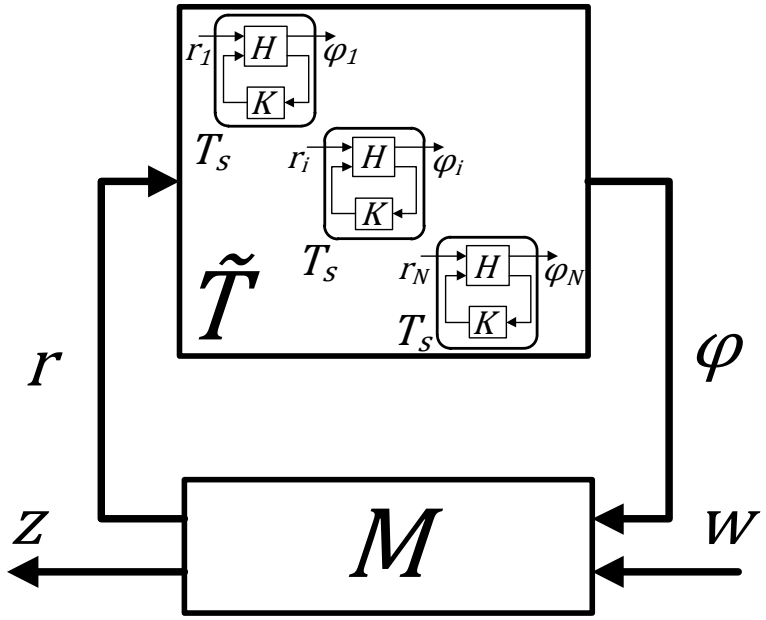


Figure 1.15: Système de grande dimension LTI global considéré

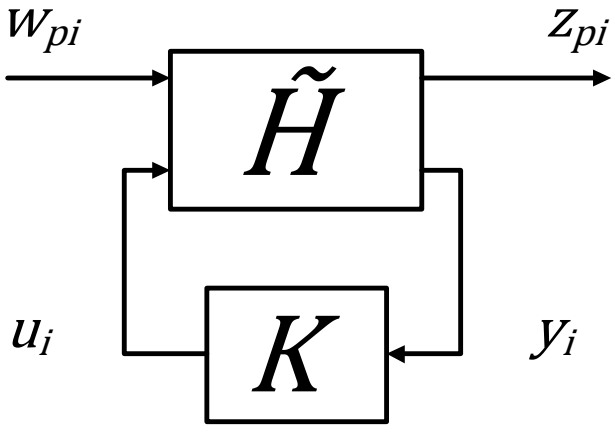


Figure 1.16: Sous-système LTI local considéré

Il est très important de distinguer ces propriétés pour les deux cas. La ***stabilité locale*** correspond à la stabilité d'un sous-système indépendant (1.3) sans autre sous-système et sans l'interconnexion (stabilité d'un nœud séparé) tandis que la ***stabilité***

globale correspond à la stabilité du système complet de grande dimension défini par (1.2). La ***performance locale*** est évaluée pour un sous-système (1.3) séparé de l'autre partie du réseau, augmenté par les entrées et les sorties de performance w_{pi} et z_{pi} respectivement. La ***performance globale*** est évaluée pour le système complet de grande dimension (1.2) avec les entrées w et les sorties z globales de performance correspondantes. Ces entrées et sorties définissent les dimensions et la structure des opérateurs LTIs $\widetilde{H}$ et M respectivement.

Par la suite, les problème généraux de la synthèse et d'analyse peuvent être formulés comme suit.

Problème 1.3 (Problème de la synthèse)**.** *Étant donné un système LTI H et un système LTI M décrivant l'interconnexion, trouver un système LTI K tel que :*

1. *Chaque sous-système* (1.3) *représenté sur la Figure 1.16 séparé soit stabilisé ;*
2. *Le système global* (1.2) *représenté sur la Figure 1.15 soit stabilisé ;*
3. *Les spécifications de la performance locale soient assurées ;*
4. *Les spécifications de la performance globale soient assurées.*

Problème 1.4 (Problème d'analyse)**.** *Étant donné un sous-système local et un système global incertains dont la description est respectivement donnée dans la section 2.6 et la section 5.6, tester si la stabilité et les spécifications de performance pour ces systèmes assurées par la loi de commande synthétisée (voir le Problème 1.3) sont respectées pour toutes les incertitudes possibles.*

2 Conception automatique d'une ADPLL locale

Introduction

Dans le chapitre précédent, nous avons vu que la boucle à verrouillage de phase ou Phase Locked Loop (PLL) est un dispositif clef des réseaux actifs de distribution du signal d'horloge. Dans ce chapitre, nous allons étudier le comportement dynamique d'une PLL indépendante du réseau, ses blocs fonctionnels ainsi que les spécifications de performance désirées. Comme nous allons voir, une PLL est naturellement modélisée par un système non-linéaire [Raz96] ce qui complique de manière significative l'analyse de son comportement ainsi que sa conception. En outre, la complexité devient encore plus importante du fait des dispersions technologiques de paramètres de la PLL inévitables durant le processus de sa fabrication. Nous considérons alors les trois modèles suivants décrivant le comportement dynamique d'une PLL avec différents niveaux de précision : un modèle simplifié LTI pour la conception efficace de la loi de commande garantissant *a priori* la performance de la PLL, un modèle plus complet incluant les incertitudes sur ses paramètres pour l'analyse mathématique de la performance ; un modèle le plus complet pour la réalisation de simulations :

Modèle de synthèse : un modèle simplifié LTI nominal traduisant un comportement le plus significatif d'une PLL autour d'un point de fonctionnement. Il est aussi appelé dans la littérature de Microélectronique le modèle « petits signaux ». Ce modèle va être utilisé pour réaliser la synthèse de la loi de commande efficace en utilisant les méthodes avancées d'automatique fréquentielle (méthode H_∞) assurant la stabilité et un niveau de performance spécifié.

Modèle d'analyse : un modèle LTI incertain regroupant en plus du Modèle de synthèse les incertitudes paramétriques possibles de la PLL. Ce modèle est nécessaire pour effectuer l'analyse efficace de la stabilité et de la performance obtenues après la synthèse de la loi de commande permettant de tester si ces propriétés sont conservées en dépit de dispersions technologiques.

Modèle de simulation : un modèle représentant le comportement de la PLL réelle avec tous les détails possibles. Il est utilisé par la suite pour réaliser les simulations temporelles nécessaires pour valider l'efficacité de la loi de commande obtenue sur le procédé réel.

Pour conclure ce chapitre, nous verrons que si l'on interconnecte les PLLs ainsi synthétisées dans un réseau avec une topologie quelconque, même dans le cas de

modèles simples LTI (Modèles de synthèse) on ne conserve pas forcément les mêmes propriétés de performance, voire même de stabilité. Il manque certaines conditions supplémentaires nécessaires pour établir le lien entre les propriétés locales et globales.

2.1 Généralités sur les PLLs

La boucle à verrouillage de phase a été développée par Henri de Bellescize [Bel32] en 1932. C'est seulement quarante ans plus tard que ce système électronique a commencé à être massivement utilisé. Les PLLs sont extrêmement répandues dans tous les domaines d'application du contrôle automatique de la phase et/ou de la fréquence tels qu'en communication, contrôle de systèmes, mesure et instrumentation, horlogerie de microprocesseurs, *etc.*. Leur principe de fonctionnement consiste à asservir la phase d'un oscillateur de fréquence variable sur une source extérieure de référence. La phase du signal périodique de l'oscillateur est en général contrôlée par une tension ou un mot numérique et c'est pour ces raisons qu'il est appelé respectivement oscillateur contrôlé en tension (*VCO*) ou *oscillateur contrôlé par un mot numérique* (que nous appellerons *DCO* pour *« Digitally Controlled Oscillator »*).

Un diagramme fonctionnel de base d'une PLL identique à celui de la Figure 1.2 est représenté sur la Figure 2.1. Le signal du *VCO Loc* après la division de fréquence *Div* est comparé au signal de référence extérieure Ref à l'aide d'un comparateur de phase *PD*. La sortie du comparateur de phase est en général constituée de deux composantes : la première, en basses fréquences, est proportionnelle à la différence de phases des deux signaux appliqués, et le deuxième, en hautes fréquences, est un signal parasite déterministe (appelé « *ripple* » dans la littérature anglo-saxonne) dont les caractéristiques peuvent varier avec la technologie de *PD* employée. La sortie de *PD* est ensuite envoyée à l'entrée du filtre F remplissant deux fonctions : la première, l'atténuation et le filtrage du ripple, la deuxième, la réalisation de la loi de commande du *VCO* assurant la synchronisation de la PLL avec le signal de référence (en fréquence et/ou en phase), c'est-à-dire la stabilité de la boucle fermée et un niveau de performance suffisant dont les caractéristiques seront présenter par la suite.

Définition 2.1. Deux signaux sont dits *synchronisés en fréquence* quand ils ont la même fréquence et que la différence de phase est constante. Ils sont *synchronisés en fréquence et en phase* si, en plus, la différence de phase est égale à zéro.

Le bloc $1/d$ de division de fréquence permet de garder un certain rapport en fréquence entre le signal de référence Ref et le signal local *Loc* issus du *VCO* tout en respectant la synchronisation entre le signal divisé *Div* et la référence Ref.

Il existe différents types de PLL [Mar04] : analogique, mixte ou entièrement numérique.

La ***PLL analogique*** opère sur des signaux périodiques de nature analogique et dont tous les éléments sont des composants analogiques. Le *VCO* est un oscillateur

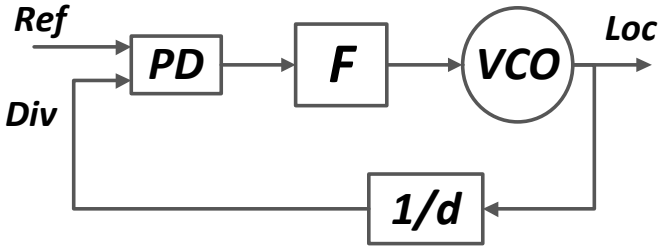


Figure 2.1: Diagramme fonctionnel d'une PLL

quasi-harmonique ou un oscillateur de relaxation [1] ; le comparateur de phase et le diviseur de fréquence sont réalisés sur des multiplexeurs analogiques.

La ***PLL mixte*** opère des signaux de forme carrée « 0 » et « 1 » (ou « ±1 ») : le *VCO* est donc un oscillateur de relaxation, et contrairement aux PLL analogiques, le comparateur de phase et le diviseur de fréquence sont réalisés avec des éléments digitaux : bascules, logique numérique, *etc.*. Le filtre reste toutefois un filtre analogique délivrant un signal de tension analogique de commande du *VCO*.

La ***PLL entièrement numérique*** ou ***All Digital PLL*** (ADPLL en Anglais) est une boucle à verrouillage de phase dont tous les éléments internes y compris le filtre sont à base de circuits numériques [2]. L'oscillateur d'une ADPLL, vue la nature numérique du signal de commande à son entrée, est un *DCO*.

Les réseaux de distribution du signal d'horloge actifs décrits dans [PN95, GC00] utilisent des PLLs mixtes. Bien que ces architectures soient fiable et remplissent la fonction de distribution active du signal d'horloge, elles souffrent tout de même de trois défauts principaux : occupation de la place sur silicium, sensibilité au bruit, difficulté d'intégration des fonctions analogiques et digitales ainsi que le manque d'outils de CAO [3] associés. En effet, de par leur nature analogique, les filtres occupent beaucoup de place sur la puce, ce qui n'est pas le cas des filtres numériques. Par ailleurs, les signaux analogiques au sein de chaque PLL sont plus sensibles au bruit que les mêmes signaux numériques, surtout dans un environnement bruité de microprocesseurs : bruits d'alimentation, perturbations venant de blocs fonctionnels du circuit, échauffements. En ce qui concerne le troisième point sur les outils CAO, il est bien connu que les microprocesseurs modernes sont essentiellement composés d'éléments numériques et tout le flux de leur conception est associé à des outils de CAO. Il est donc logique et préférable pour une intégration aisée que le système de distribution du signal d'horloge puisse être lui aussi conçu, fabriqué et testé par les mêmes outils de CAO. Pour les raisons mentionnées précédemment, dans notre étude, nous considérons donc un réseau de distribution du signal d'horloge actif composé d'ADPLLs. Un bon état de l'art sur le sujet présentant les avantages et les

1. Un oscillateur quasi-harmonique fournit à sa sortie un signal quasi-sinusoïdal ; tandis que l'oscillateur de relaxation fournit un signal qui peut être approché par un signal en forme carrée.

2. L'adjectif « entièrement numérique », toutefois, n'est pas employé ici pour souligner la nature des signaux mêmes. En réalité, comme nous allons voir par la suite, les ADPLLs possèdent les deux types de signaux : analogique et numérique avec les convertisseurs correspondants.

3. Conception Assistée par Ordinateur ou Computer-Aided Design (CAD) en anglais.

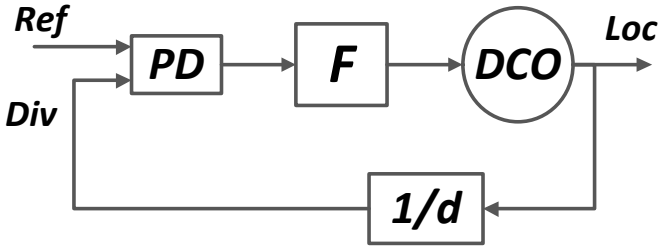


Figure 2.2: Diagramme fonctionnel d'une ADPLL

inconvénients d'ADPLLs peut être trouvée dans [Sta11].

Le diagramme fonctionnel d'une ADPLL est représenté sur la Figure 2.2. Il est équivalent à celui de la Figure 2.1, à la différence que tous les blocs sont numériques et que le filtre a besoin d'un signal d'horloge externe pour effectuer les calculs internes. Étant donné que l'ADPLL fait partie d'un système qui génère le signal d'horloge, il existe aucun autre signal d'horloge à notre disposition. L'idée est donc d'utiliser le signal de référence extérieur ou le signal issu du *DCO* local comme signal d'horloge pour l'échantillonnage du filtre.

Par la suite, nous présentons avec plus de détails chaque bloc de l'ADPLL utilisée dans le réseau actif de distribution d'horloge pour cette étude. Une description complète d'un point de vue microélectronicien de l'ADPLL est donnée dans [SB06].

2.1.1 Comparateur de phases

Le comparateur de phases mesure un déphasage entre deux signaux à son entrée. Sa sortie est une valeur numérique dont l'amplitude est proportionnelle au déphasage des deux signaux. En réalité, le comparateur de phase d'une ADPLL remplit une fonction supplémentaire : la conversion d'une valeur analogique du « déphasage » vers « un mot numérique d'erreur » (signal *Err* dont nous appellerons l'unité de mesure « *u.e.* » ce qui signifie « unités d'erreur »). Le comparateur de phase utilisé dans l'architecture de l'ADPLL est souvent appelé dans la littérature un comparateur de phase et de fréquence (en anglais Phase Frequency Detector *PFD*). En effet, dans le cas où la différence de phase instantanée est égale à un multiple de 2π, et compte tenu de la nature cyclique de la phase ($\Delta\varphi = 0$ modulo 2π), un comparateur de phase conventionnel génèrerait un signal d'erreur nul à sa sortie. Dans l'architecture considérée ici, le comparateur de phase utilisé fournit un signal d'erreur détectant en temps réel la différence de fréquences.

Le *PFD* est composé de trois parties (voir la Figure 2.3) : *comparateur de phase bang-bang*, *convertisseur temps numérique TDC* (en anglais *Time to Digital Converter*) et un *Encodeur*.

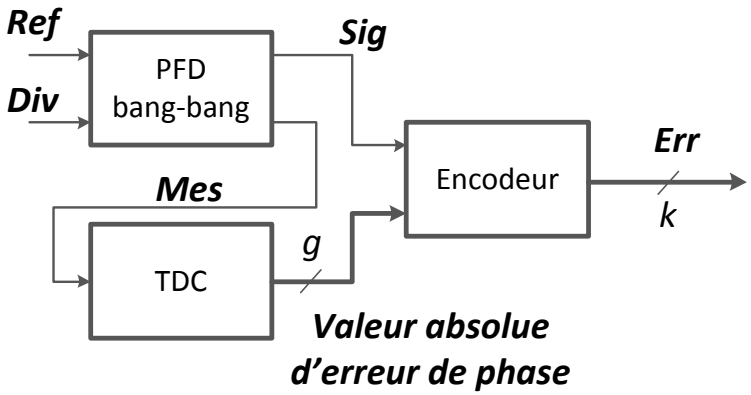


Figure 2.3: Diagramme de blocs du comparateur de phase pour l'architecture choisie

Comparateur de phase bang-bang

Le comparateur de phase bang-bang produit uniquement l'information sur le signe de la différence de phase (fréquence) des deux signaux appliqués à son entrée. Les signaux peuvent être représentés par des événements, par exemple par leurs fronts montants. Nous allons appeler les événements associés aux signaux Ref et Div respectivement *« ref »* et *« div »*.

En parlant de façon simplifiée, le comparateur de phase bang-bang « décide » quel événement arrive en premier à partir d'un temps initial. Par convention, la sortie est un signal binaire « 1 » ou « 0 » qui signifie *« div* est arrivé en premier » ou *« ref* est arrivé en premier ». Le comparateur de phase bang-bang est une machine d'états dont les états sont représentés sur la Figure 2.4 et qui a donc 2 bits de mémoire interne. Nous allons appeler ces bits par *« SIGNE »* et *« MESURE »* ainsi que les signaux correspondants par *« Sig »* et *« Mes »*. Le bit *SIGNE* est la sortie de comparateur de phase bang-bang portant l'information sur le signe de la différence de phases instantanée, le bit *MESURE* porte l'information sur la valeur absolue de cette différence. Le paragraphe suivant décrit le fonctionnement de ce comparateur.

Fonctionnement Initialement, le bit *MESURE* est égal à zéro et le bit *SIGNE* est indéterminé. Quand l'un des deux événements arrive, le bit *MESURE* prend la valeur 1 et le bit *SIGNE* devient 0 ou 1 si l'évènement arrivé est *div* ou *ref* respectivement. Ensuite, quand le bit *MESURE = 1* et *SIGNE = 0* (respectivement *SIGNE = 1*), la machine d'état « attend » l'événement *ref* (respectivement *div*) en ignorant tous les événements *div* (respectivement *ref*) qui peuvent apparaître à l'entrée de comparateur bang-bang. Quand l'événement attendu *ref* (respectivement *div*) arrive, le bit *SIGNE* reste inchangé tandis que le bit *MESURE* passe à zéro signifiant la fin d'un cycle de mesure. Par la suite, un nouveau cycle identique à celui décrit précédemment commence avec l'arrivée d'un nouvel événement à l'entrée du *PFD* bang-bang.

Dans la littérature, il existe différentes méthodes d'implémentation du comparateur

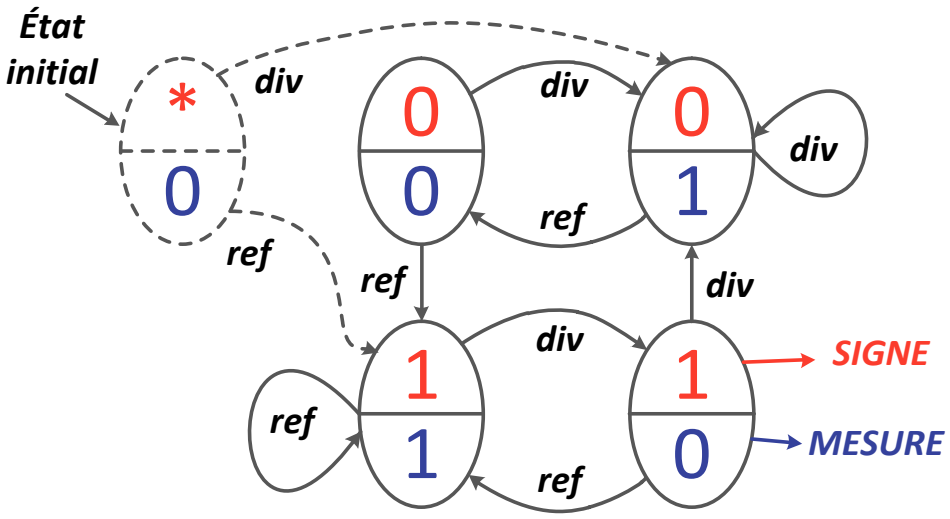


Figure 2.4: Machine d'états du comparateur de phase bang-bang

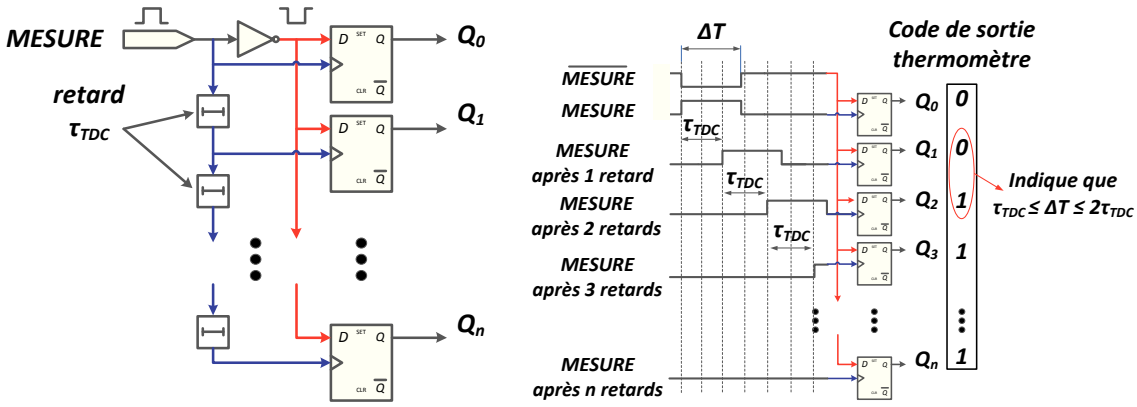
de phase bang-bang réalisant la machine d'états décrite précédemment. Étant donné que le but de cette étude est l'application de méthodes d'Automatique pour la synthèse de réseaux de distribution du signal d'horloge, nous ne rentrerons pas dans les détails de la réalisation d'un tel comparateur de phase, mais nous retiendrons seulement son fonctionnement global. Le lecteur intéressé, toutefois, peut consulter les travaux de [RTE+07, SB06, Sta11] où différentes implémentations d'un *PFD* bang-bang sont décrites.

Convertisseur temps numérique

Si le comparateur de phase bang-bang permet de déterminer l'ordre d'arrivée des événements, le convertisseur temps numérique ou *TDC* a pour but de quantifier l'intervalle de temps entre ces deux évènements. En regardant plus en détails, le fonctionnement d'un comparateur bang-bang, on peut remarquer que nous avons déjà à notre disposition un signal permettant de réaliser cette quantification. En effet, le bit *MESURE* est égal à 1 durant toute une période du temps entre deux événements. En mesurant la durée de cette période, on peut quantifier la différence de temps entre deux événements et donc la différence de phase absolue des deux signaux appliqués en entrée de la PLL.

Le principe de mesure le plus simple utilisé dans notre étude est l'envoi du signal Mes correspondant au bit *MESURE* dans une chaîne de retards élémentaires (ligne bleue sur la Figure 2.5a) et l'observation de la profondeur de sa propagation durant l'intervalle de temps à mesurer (voir la Figure 2.5b). Ceci est réalisé de la manière suivante. Au même instant que le signal correspondant au bit *MESURE* est envoyé dans une chaîne de retards, son inverse est également envoyé vers l'entrée D des bascules[4] D [Hof04]. L'entrée de chaque inverseur élémentaire est envoyé à

4. Une bascule est un circuit logique réalisant un opérateur logique avec plusieurs entrées et



(a) Implémentation d'un TDC **(b)** Fonctionnement

Figure 2.5: Implémentation possible et fonctionnement de convertisseur temps numérique TDC

l'entrée *RESET* de chaque bascule *D* (voir la Figure 2.5a). Quand le front montant de l'inverse au signal correspondant au bit *MESURE* apparaît à l'entrée des bascules, seules les bascules où le signal correspondant au bit *MESURE* n'est pas encore arrivé, changent leur état initial de 0 à 1. Les bascules restantes, atteintes par le front montant du signal correspondant au bit *MESURE*, étant dans un état désactivé, gardent leur valeur initiale 0. La limite de transition entre les valeurs 0 et 1 indique la profondeur de la propagation du signal correspondant au bit *MESURE* comme le niveau de pression dans un thermomètre classique d'où le nom du code « thermomètre » réservé pour le code en sortie de ce type d'architecture. La valeur nominale des retards et leur quantité déterminent ainsi la précision minimale τ_{TDC} et le niveau de saturation E_{sat} du comparateur de phase.

Par la suite, le code thermomètre est regroupé avec le code du signe d'erreur de phase par l'Encodeur de la Figure 2.3 afin d'obtenir à la sortie du comparateur de phase un code numérique *ERREUR* de k bits variant dans la plage $\pm E_{sat}$ correspondant au signal Err.

La précision de l'architecture retenue de *TDC* est assez sensible aux variations de température. En effet, les variations de température font varier la valeur des retards élémentaires, ce qui provoque une estimation imprécise de la différence de phase. Il existe plusieurs autres types d'architecture de *TDC* [LR04], parmi lesquels les plus complexes sont robustes aux variations de température. Étant donné que le réseau complet de distribution d'horloge consiste en plusieurs ADPLLs avec plusieurs *TDC*s chacune, il est souhaitable d'utiliser des *TDC*s les plus simples possibles. Par ailleurs, l'influence de la précision d'estimation de la différence de phase sur le résultat final de synchronisation sera prise en compte dans les modèles d'analyse et de simulation. En effet, le paramètre le plus important pour assurer la stabilité et la synchronisation du réseau est le signe de la différence de phase. Ce dernier est déterminé par le comparateur de phase bang-bang simple avec une grande précision [RTE+07]. Le

plusieurs sorties prenant un niveau logique 0 ou 1. Le changement d'état de la sortie est déterminé par les signaux appliqués aux entrées et le type d'opérateur réalisé par la bascule.

changement de retards élémentaires étant lent, il provoque une variation du gain du *PD*s, ce qui peut être modélisé par une incertitude paramétrique. On pourra toujours tester *a posteriori* la stabilité et la dégradation de la performance du réseau impliquée par cette variation sur le modèle d'analyse. Pour toutes ces raisons, on retient l'architecture de *TDC* présentée sur Figure 2.5. De la même manière que pour le comparateur de phase bang-bang, nous n'entrerons pas dans les détails de la réalisation et de l'implémentation du *TDC* et de l'encodeur. Une description plus détaillée peut être trouvée dans [LR04, JZA+11].

Le comportement global entrée-sortie du comparateur de phase utilisé dans l'architecture d'ADPLL choisie est présenté sur la Figure 2.6. Ainsi pour une différence de phase faible, $\Delta\varphi < \tau_{TDC}$, la sortie du comparateur de phase bang-bang 1 ou 0 est encodée par une valeur $+1\,u.e.$ ou $-1\,u.e.$ respectivement. En réalité, la Figure 2.6 représente la réponse en régime établi du *PD* à la différence de phase de deux signaux appliqués en entrée. Le régime établi est obtenu une fois le cycle complet de mesure achevé, c'est-à-dire quand les deux événements ref et div sont arrivés.

Étant donné que la phase est définie modulo 2π, la valeur absolue de la sortie de *PFD* l'est aussi. Ceci est illustré par les répétitions cycliques d'un même motif sur la Figure 2.6. Le signe de l'erreur de phase n'est pas cyclique contrairement à la valeur absolue de l'erreur de phase [RTE+07]. C'est pour cette raison que la réponse statique est symétrique par rapport à l'origine et non par rapport à l'axe des ordonnées. Comme dit précédemment, la valeur du retard élémentaire utilisée dans la chaîne de retards du *TDC* détermine la sensibilité ou la précision minimale τ_{TDC} du comparateur de phase. Le nombre des retards et de bits du *TDC* détermine la valeur maximale de sa sortie, c'est-à-dire la valeur de la saturation E_{sat}. Pour les erreurs de phase inférieures au retard élémentaire $\Delta\varphi < \tau_{TDC}$, la valeur absolue n'est pas quantifiée par le *TDC* puisqu'elle sort de sa limite de précision et le comportement du *PD* est identique à celui d'un comparateur de phase bang-bang avec une précision « infinie » [RTE+07] (voir la zone bang-bang sur la Figure 2.6). Idéalement, quand l'erreur de phase tend vers zéro, la sortie du comparateur de phase oscille entre « +1 » et « -1 » avec une fréquence infinie. En réalité, il existe différents phénomènes provoquant l'apparition de retards et d'hystérésis à paramètres variables et aléatoires qui limitent la fréquence des oscillations bang-bang. L'étude [LKR04] montre qu'en prenant en compte la probabilité de variation des retards et des hystérésis, le comportement du comparateur de phase bang-bang autour de zéro peut être modélisé par une fonction linéaire avec un gain constant.

2.1.2 Filtre

Le filtre, ou autrement dit le correcteur, dans l'architecture ADPLL est un bloc numérique qui réalise la commande du système. C'est un élément de réglage dont le choix (synthèse) est effectué afin de garantir 1) la stabilité, 2) un certain niveau de performance. Un exemple de filtre PI numérique avec les coefficients Ki et Kp pour les branches « intégrale » et « proportionnelle » respectivement est présenté dans le domaine de Laplace sur la Figure 2.7. Les critères de performance à satisfaire ainsi que la synthèse du correcteur associé sont décrits dans la section 2.5, page 60.

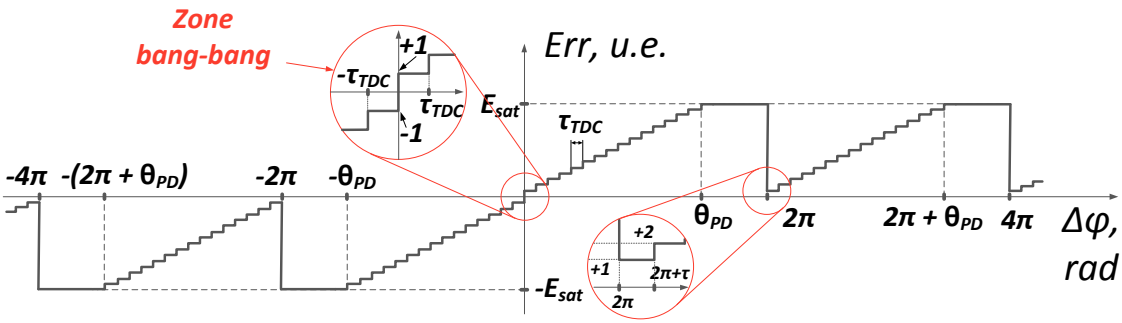


Figure 2.6: Réponse statique entrée-sortie du PFD

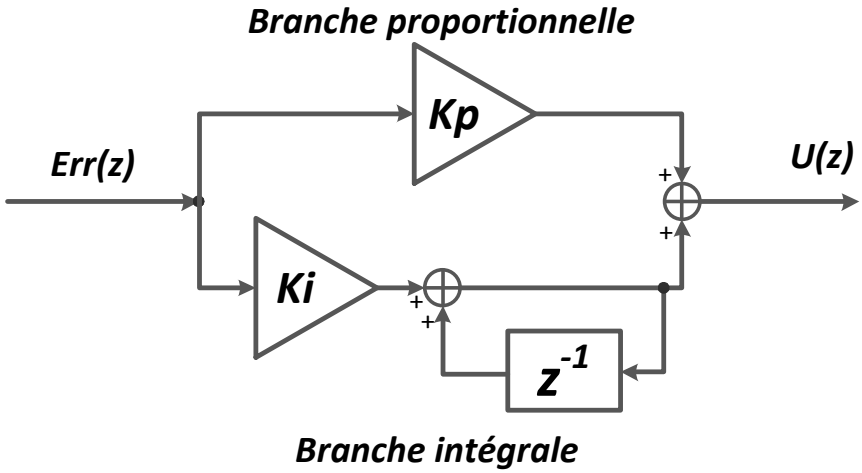


Figure 2.7: Filtre PI numérique

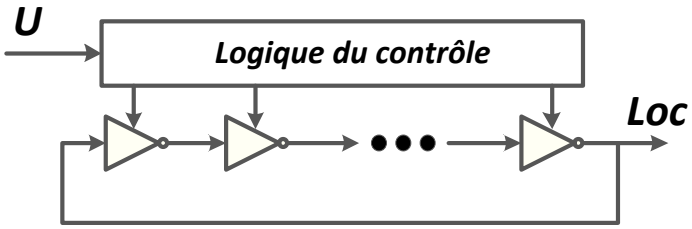


Figure 2.8: Schéma fonctionnel de l'oscillateur en anneau utilisé

2.1.3 Oscillateur contrôlé par un mot numérique

L'oscillateur utilisé dans l'ADPLL est un oscillateur en anneau contrôlé par un mot numérique provenant du filtre. Le schéma fonctionnel simple d'un oscillateur en anneau est représenté sur la Figure 2.8. Il consiste en un nombre impair d'inverseurs à trois états connectés en boucle. Chaque inverseur peut être activé ou désactivé par la logique de contrôle pilotée par un signal de commande extérieur U venant du filtre. Les unités de mesure de commande du filtre sont *u.c.* signifiant « unités de commande ». Le nombre d'inverseurs activés détermine la fréquence d'oscillation de la sortie du *DCO*. Plus le nombre d'inverseurs effectifs dans la boucle est grand, plus la fréquence d'oscillation de la sortie *Loc* est importante. Pour plus de détails sur la réalisation d'un oscillateur en anneau avec une matrice d'inverseurs en deux dimensions ainsi que son implémentation pratique, le lecteur est invité à consulter [RTE+07]. Dans notre étude, nous allons nous contenter du comportement simplifié global de l'oscillateur commandé par un mot numérique. La réponse statique de celui-ci, c'est-à-dire la fréquence de la sortie en fonction du code en entrée, est représentée sur la Figure 2.9. f_0 désigne la fréquence centrale du *DCO*, sa fréquence propre d'oscillation avec le code nul en entrée, f_{max} et f_{min} représentent la fréquence maximale et minimale atteignable et μ_{max} et μ_{min} les codes de la commande correspondants.

2.1.4 Diviseur de fréquence

Le choix de l'architecture du diviseur de fréquence est dicté comme dans les cas précédents par sa facilité d'implémentation. On choisit ainsi un diviseur réalisé par une connexion en cascade de bascules D effectuant la division de la fréquence par un nombre d multiple de deux. Il existe également des diviseurs de fréquence programmables permettant de modifier la fréquence d'horloge générée en temps réel. Ceci est très intéressant dans un contexte de réduction de la consommation de la puce. En effet, on a vu dans le chapitre précédent qu'une diminution de la fréquence du signal d'horloge baisse aussi la consommation liée aux impédances parasites. Pour les Systems On Chip (SOCs) modernes pour lesquels on adapte la fréquence de fonctionnement de différents modes d'opération de la puce selon la charge de travail, le changement du facteur de division est très important. Plus de détails sur l'implémentation de diviseurs simples ainsi que de diviseurs programmables peuvent être

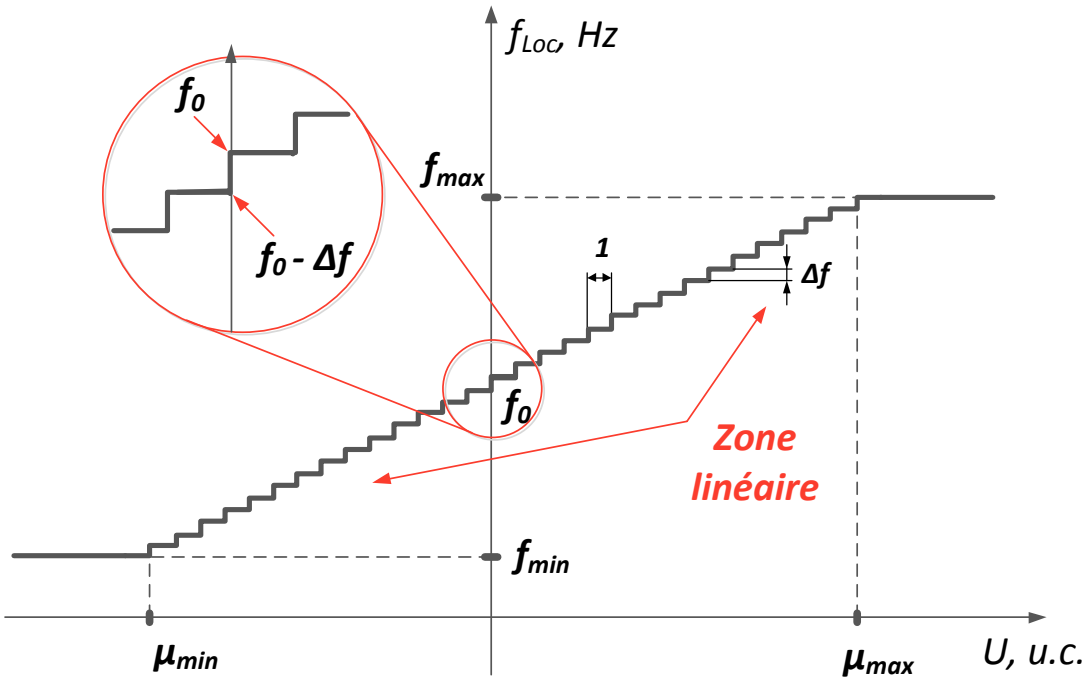


Figure 2.9: Caractéristique statique de l'oscillateur en anneau

retrouvés dans [Raz96, Roh97, Raz98].

2.2 Modélisation d'une ADPLL

Les paramètres les plus importants de l'ADPLL tels que la fréquence centrale du *DCO*, le facteur de division de fréquence, les retards élémentaire de *PD etc.* doivent être choisis selon les applications. Ces paramètres permettent de réaliser un choix optimal d'architecture de chaque composant de l'ADPLL, ainsi que d'autres valeurs paramétriques. Il reste néanmoins un composant à déterminer, le filtre. Étant un élément de réglage relativement souple (car programmable) réalisant la commande de l'ADPLL, son choix détermine le comportement globale de celle-ci. Comme cela a été mentionné auparavant, d'un point de vue d'Automatique, une ADPLL est une boucle de rétroaction et le filtre est un composant réalisant la loi de commande. Ses caractéristiques fixent le comportement dynamique de l'ADPLL.

Le comportement dynamique d'une ADPLL est complexe et fortement non-linéaire (voir les Figure 2.6 et Figure 2.9). Néanmoins, l'approche par linéarisation de son comportement dans le domaine de phase simplifie considérablement son étude et sa conception [Roh97, Kro03, KCV08]. Ainsi, comme cela a été mentionné dans l'introduction, on linéarise l'ADPLL autour d'un point de fonctionnement : on établit donc un modèle « petits signaux », autour duquel le comportement est quasi-linéaire et où la contribution de la partie du comportement non-linéaire est négligeable. En se basant sur ce modèle linéaire dans le domaine de phase, une analyse et une

synthèse efficaces peuvent être effectuées, garantissant la stabilité et les propriétés de performance désirées du modèle. Ensuite, une analyse non-linéaire peut être réalisée afin d'obtenir un domaine de validité du modèle linéaire, c'est-à-dire un domaine où la loi de commande obtenue est efficace non seulement sur le modèle linéaire mais aussi sur le procédé réel.

Cette section a pour but de décrire dans le domaine de phase le modèle de chaque bloc ainsi que le modèle global de l'ADPLL utilisé dans notre étude. Ce modèle est ensuite linéarisé et est utilisé, dans la section prochaine, comme base pour la synthèse de la loi de commande.

2.2.1 Comparateur de phase

Le comparateur de phase fait intervenir quatre non-linéarités principales (voir la Figure 2.6) : le caractère cyclique de la phase, la saturation, le comportement bang-bang autour du zéro et l'effet de quantification car, comme cela a déjà été évoqué, le comparateur de phase remplit une fonction supplémentaire de Conversion Analogique Numérique (*CAN*) du signal analogique de la différence de phase vers un signal numérique *Err*.

Nous allons modéliser les deux premières non-linéarités par une fonction non-linéaire Ψ déterminée comme dans (2.1) et Figure 2.10 (voir la courbe rouge en trait interrompu mixte) :

$$\Psi(\Delta\varphi) = \begin{cases} 0, & |\Delta\varphi| \leq \theta_{PD}; \\ K_{PD}(\Delta\varphi - \theta_{PD}), & 2\pi l + \theta_{PD} < \Delta\varphi < 2\pi(l+1); \\ 2\pi l \cdot K_{PD}, & 2\pi(l+1) \leq \Delta\varphi \leq 2\pi(l+1) + \theta_{PD}; \\ K_{PD}(\Delta\varphi + \theta_{PD}), & -2\pi(l+1) < \Delta\varphi < -2\pi l - \theta_{PD}; \\ -2\pi l \cdot K_{PD}, & -2\pi(l+1) - \theta_{PD} \leq \Delta\varphi \leq -2\pi(l+1); \end{cases} \quad , l \in \mathbb{Z}. \tag{2.1}$$

Les effets de la quantification ainsi que des oscillations aléatoires bang-bang sont modélisés par un signal aléatoire de réalisation temporelle bornée à la sortie du comparateur de phase b_{PD} : $|b_{PD}| \leq 1\,u.e.$ dont la Densité Spectrale de Puissance (*DSP*) est importante essentiellement en hautes fréquences. Il peut être modélisé par la convolution temporelle d'un filtre H_{PD} et d'un bruit blanc, soit dans le cas présent, par la convolution d'un filtre passe-haut et la réalisation temporelle d'un bruit blanc b (voir la Figure 2.11).

La partie linéaire du comportement du *PD* est modélisée dans le domaine de phase par un bloc de soustraction suivi d'un simple gain K_{PD} et d'un bloc d'échantillonnage (Figure 2.11 et Figure 2.10, courbe bleue plein). La différence entre le gain linéaire du *PD* et la fonction non-linéaire Ψ est représentée à titre informatif par la fonction $\widetilde{\Psi}$ sur la Figure 2.11 et Figure 2.10 (courbe verte en pointillé).

Les variations des retards élémentaires dûes aux dispersions technologiques ainsi qu'à une variation de la température peuvent être modélisées, comme cela a été

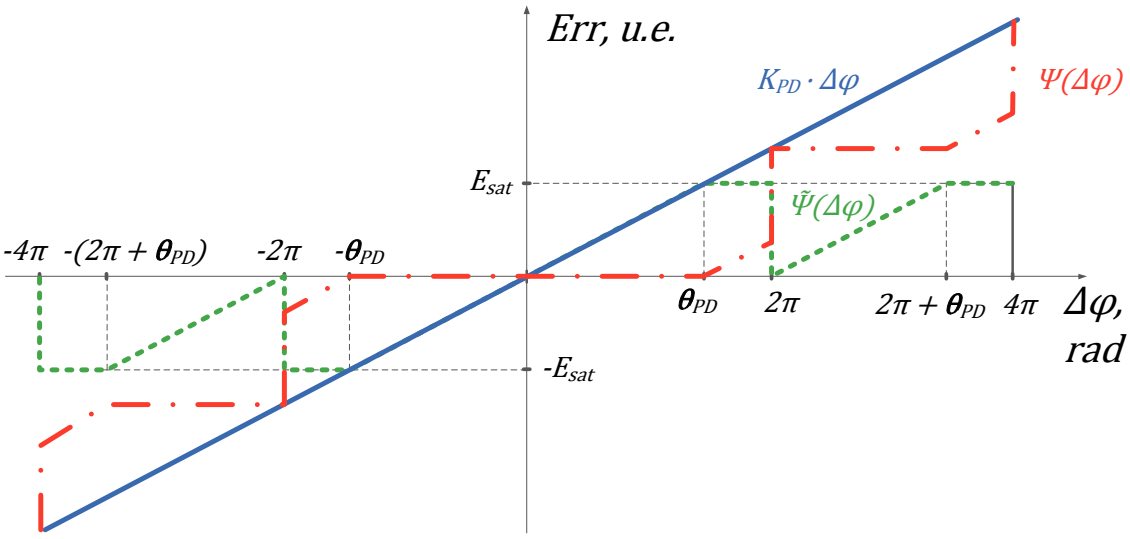


Figure 2.10: Caractéristique du modèle du *PD* dans le domaine de phase

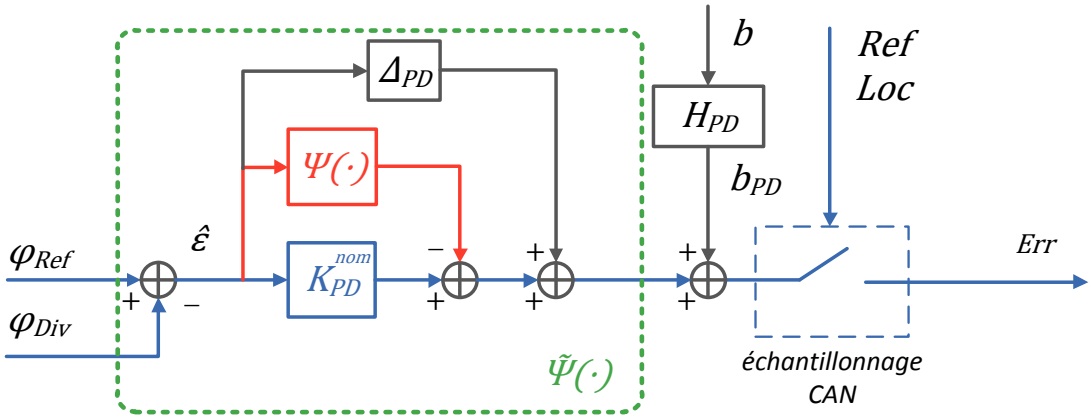


Figure 2.11: Modèle non-linéaire dans le domaine de phase du *PD*

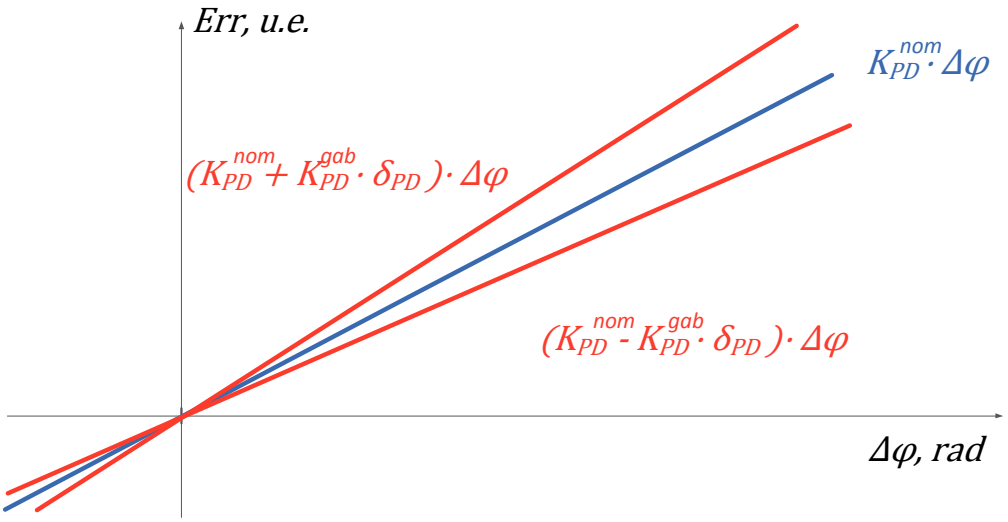


Figure 2.12: Influence d'incertitude réelle sur le gain de *PD*

évoqué dans la section précédente, par une incertitude réelle sur le gain linéaire de *PD* (voir la Figure 2.12), $\Delta_{PD} = K_{PD}^{gab} \cdot \delta_{PD}$:

$$\begin{aligned} K_{PD} &= K_{PD}^{nom} + \Delta_{PD} = K_{PD}^{nom} + K_{PD}^{gab} \cdot \delta_{PD}; \\ K_{PD} &\in \left[K_{PD}^{nom} - K_{PD}^{gab};\, K_{PD}^{nom} + K_{PD}^{gab}\right], \end{aligned} \tag{2.2}$$

où K_{PD}^{nom} est la valeur nominale du gain du *PD*, K_{PD}^{gab} est le gabarit de l'incertitude, δ_{PD} est l'incertitude réelle scalaire normalisée : $|\delta_{PD}| < 1$.

Finalement, on obtient un modèle non-linéaire dans le domaine de phase pour le comparateur de phase représenté sur la Figure 2.11. Le modèle linéaire nominal est obtenu par la linéarisation du modèle de la Figure 2.11 autour du point de fonctionnement où la différence de phase tend vers zéro. Dans ce cas-là $\Psi \equiv 0$ et $\Delta \equiv 0$ et on obtient un modèle nominal simplifié linéaire du *PD* dans le domaine de phase (voir la Figure 2.13).

2.2.2 Filtre

Le filtre ou correcteur utilisé dans l'architecture ADPLL considéré est un filtre discret et son modèle est représenté par une fonction de transfert discrète $F(z)$. Les coefficients du filtre sont choisis afin de garantir la stabilité et un certain niveau de performance. Il est important de mentionner que l'implémentation du filtre provoque parfois des imprécisions sur les valeurs numériques des coefficients [JZA+11]. Ainsi, dans l'exemple du filtre PI représenté sur la Figure 2.7, page 41, les coefficients proportionnel et intégral vont être modélisés comme la somme d'un paramètre réel nominal et d'un paramètre réel incertain :

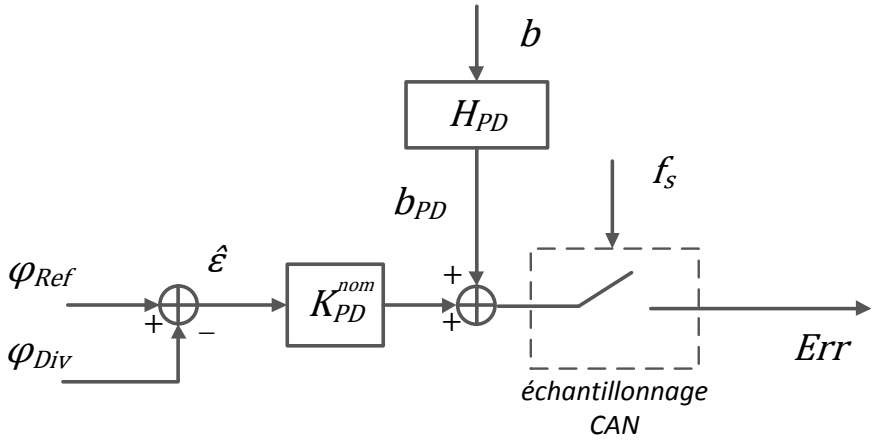


Figure 2.13: Modèle linéaire dans le domaine de phase du *PD*

$$\begin{aligned} K_P &= K_P^{nom} + \overbrace{K_P^{gab}\delta_P}^{\Delta_P}; \\ K_I &= K_I^{nom} + \overbrace{K_I^{gab}\delta_I}^{\Delta_I}, \end{aligned} \tag{2.3}$$

qui implique que

$$\begin{aligned} K_P &\in \left[K_P^{nom} - K_P^{gab};\, K_P^{nom} + K_P^{gab}\right]; \\ K_I &\in \left[K_I^{nom} - K_I^{gab};\, K_I^{nom} + K_I^{gab}\right], \end{aligned} \tag{2.4}$$

où K_P^{nom}, K_I^{nom} sont les valeurs nominales ou désirées des coefficients du filtre, K_P^{gab}, K_I^{gab}, leurs gabarit d'incertitudes respectives et δ_P, δ_I les incertitudes réelles normalisées : $|\delta_P| < 1, |\delta_I| < 1$. La même approche est utilisée pour des filtres plus complexes avec un nombre de coefficients plus important.

2.2.3 Oscillateur

L'oscillateur de l'architecture ADPLL considéré est un *DCO*. Il remplit donc une fonction supplémentaire de conversion du signal numérique de commande du filtre vers un signal analogique oscillant. Nous allons ainsi modéliser le *DCO* comme un *Convertisseur Numérique Analogique* (*CNA*) suivi d'un *VCO* comme indiqué sur la Figure 2.14.

Dans le domaine de phase, un signal périodique est représenté selon sa phase dont l'évolution est décrite par une rampe. Pour un oscillateur commandé, la pente dé-

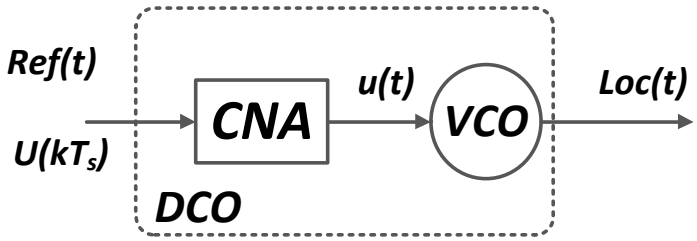


Figure 2.14: Modélisation du *DCO*

pend de la commande appliquée à son entrée :

$$\varphi_{Loc}(t) = \omega_0 t + K_{VCO} \int_{-\infty}^{t} u(t)\, dt, \tag{2.5}$$

où $\omega_0 = 2\pi f_0$ est la pulsation centrale d'oscillateur, K_{VCO} est son gain et u est la commande à son entrée.

La pente instantanée de la rampe correspond à la fréquence instantanée de l'oscillateur :

$$\omega_{Loc}(t) = \omega_0 + K_{VCO} u(t) \ . \tag{2.6}$$

L'oscillateur est donc idéalement modélisé dans le domaine de phase par un gain K_{VCO} suivi d'un intégrateur pur à l'entrée duquel la commande et la pulsation centrale sont appliquées. On suppose donc que l'oscillateur travaille proche de sa fréquence centrale dans la zone linéaire [5] de sa caractéristique présentée sur la Figure 2.9, page 43.

Bruit de phase ou jitter Bien que les *DCO*s en anneau soient faciles à intégrer, ils souffrent d'un bruit de phase important. En effet, le bruit causé par les composants électroniques ou les signaux extérieurs (commande, alimentation) peut entraîner une distorsion de la fréquence et de l'amplitude du signal généré en sortie. Dans la plupart des cas, la distorsion en amplitude est négligeable ou peu importante, et seule la distorsion de la fréquence est considérée. Elle peut également être vue comme une variation aléatoire de la période (pour les *VCO*s) ou de l'écart de temps de fronts montants et descendants par rapport à leur position idéale (pour les *DCO*s). Le

5. Sans prendre en compte les effets de quantification, qui vont être modélisés par la suite par un bruit blanc.

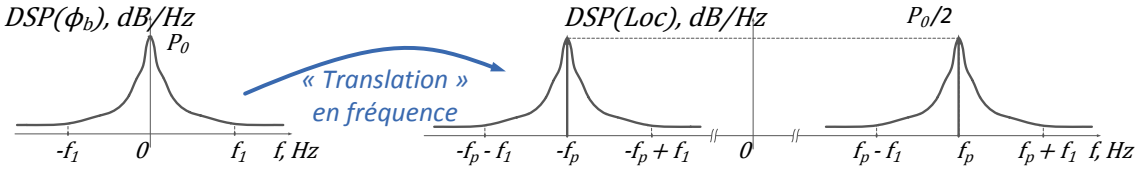


Figure 2.15: « Translation » en fréquence de la densité spectrale de puissance du bruit de phase

signal de sortie de l'oscillateur nominal sinusoïdal devient alors :

$$Loc\,(t) = A_{Loc} cos\,(\omega_p t + \phi_b\,(t))\,, \tag{2.7}$$

où A_{Loc} est l'amplitude du signal généré par l'oscillateur, ω_p la pulsation de sa porteuse (la pulsation sur laquelle l'oscillateur est synchronisé) et $\phi_b\,(t)$ représente la variation aléatoire de la période avec une certaine propriété spectrale.

La fonction $\phi_b\,(t)$ est appelée aussi le « *bruit de phase* » ou, dans les termes du chapitre 1, le jitter. Pour des petites valeurs de distorsion $|\phi_b\,(t)| \ll 1\,rad$ (ce qui est le cas pour les oscillateurs modernes), on obtient :

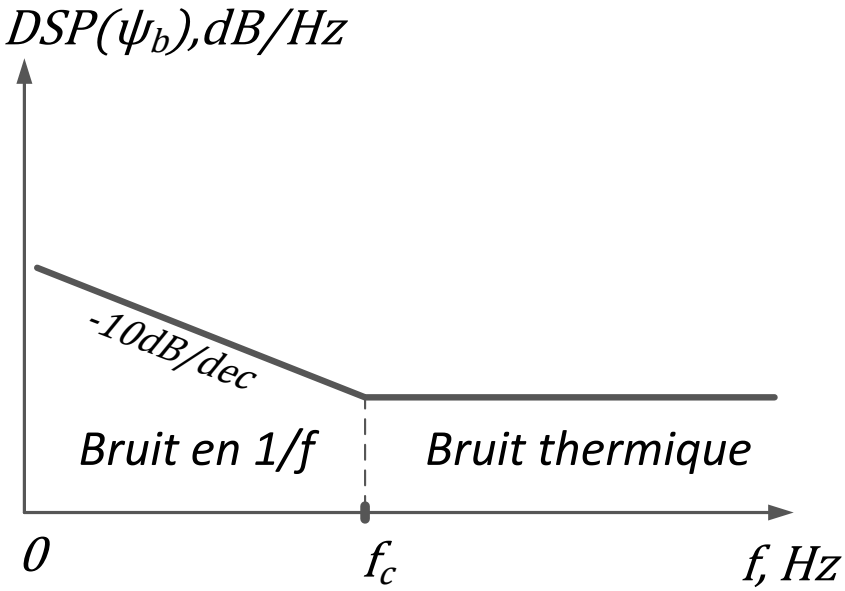
$$Loc\,(t) \approx A_{Loc} cos\,(\omega_p t) + A_{Loc}\phi_b\,(t)\,sin\,(\omega_p t)\,. \tag{2.8}$$

L'équation (2.8) montre que la *DSP* du bruit de phase grâce à la propriété de modulation de la transformée de Fourier est dédoublée avec un facteur $\frac{1}{2}$ et ramenée aux pulsations $\pm\omega_p$ (ou de la même manière aux fréquences $\pm f_p$ en Hz) ce qui est représenté sur la Figure 2.15. En outre, étant donné que les signaux traités sont réels et que donc le module de leur transformée de Fourier est symétrique par rapport à l'axe imaginaire, le spectre du bruit des oscillateurs est souvent représenté d'un seul côté et non en fonction de la fréquence absolue f mais en fonction de l'écart de celle-ci par rapport à la fréquence de la porteuse : $\Delta f = f - f_p$. La fréquence Δf est aussi appelée offset de fréquence.

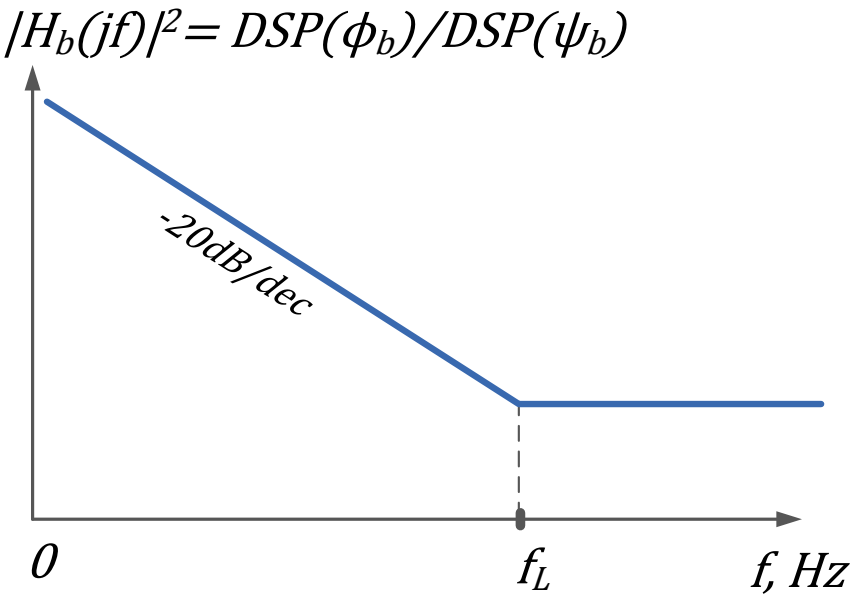
Pour comprendre les différentes sources de bruit de phase d'oscillateur, examinons le bruit d'un transistor, le composant principal élémentaire faisant partie de l'architecture de *DCO* choisie. Le bruit d'un transistor ψ_b est composé de deux contributions principales [Raz98] :

- Bruit thermique ou bruit de résistance ou bruit de Johnson d'après le nom du physicien qui l'a mis en évidence. Ce bruit est responsable du plancher de bruit en hautes fréquences (voir la Figure 2.16a) ;
- Bruit de Flicker ou bruit en $1/f$ ou bruit de basses fréquences ou bruit de scintillement. Il peut avoir différentes origines comme des impuretés dans le matériau libérant de manière aléatoire des porteurs de charge. Ce bruit se traduit par une descente de sa *DSP* en basses fréquences (BF) suivant une pente de $-10\,dB/dec$ jusqu'à la fréquence de coupure appelée la fréquence de corner f_c.

La puissance spectrale du bruit d'un transistor typique S_{ψ_b} en fonction de la fréquence est représentée sur la Figure 2.16a.



(a) La densité spectrale de puissance du bruit d'un transistor



(b) Réponse fréquentielle en amplitude au carré de la fonction de transfert du bruit de phase

Figure 2.16: *DSP* du bruit d'un transistor et la réponse fréquentielle au carré du filtre du bruit

L'architecture d'oscillateur en anneau est réalisée par un bouclage d'inverseurs dont chacun est représenté par une combinaison de transistors. Le bruit de chaque transistor ayant un spectre semblable à celui de la Figure 2.16a est converti à la sortie de l'oscillateur en un bruit de phase pour l'oscillateur ϕ_b. Ce phénomène de conversion du bruit ayant diverses origines a été évoqué pour la première fois par [Lee66]. Les auteurs de [Roh97, Rub10] montrent que les pentes du profil du bruit du transistor sont diminuées de $20dB/dec$ pour une gamme d'offset de fréquences plus petites que la fréquence caractéristique dite de Leeson f_L déterminée par :

$$f_L = \frac{f_p}{2Q} \ , \tag{2.9}$$

où Q et f_p sont respectivement le facteur de qualité du résonateur chargé et sa fréquence propre.

L'expression du profil de bruit est donné par la formule empirique de Leeson dont la validité peut être démontrée par un calcul rigoureux [Roh97, Rub10] :

$$S_{\phi_b}(f) = \left(1 + \frac{f_L^2}{f^2}\right) S_{\psi_b}(f) \ , \tag{2.10}$$

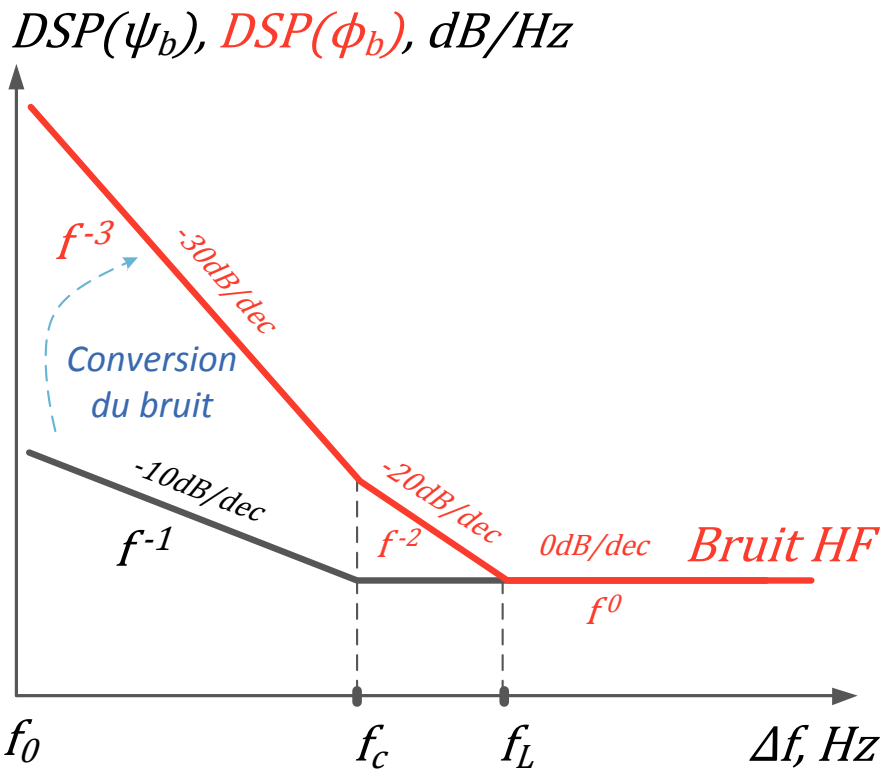
où $S_{\phi_b}(f)$ et $S_{\psi_b}(f)$ sont respectivement les *DSP* du bruit de phase et du bruit de transistor.

Le terme entre parenthèses de (2.10) exprime l'effet de Leeson, qui peut être vu comme une mise en forme du bruit des transistors par un filtre linéaire $H_b(s) = \frac{2\pi f_L}{s}\left(\frac{s}{2\pi f_L} + 1\right)$ d'ordre 1 avec un intégrateur pur et de fréquence de coupure f_L. Ce filtre est souvent appelé le filtre du bruit de phase dont la réponse fréquentielle est représentée sur la Figure 2.16b. La Figure 2.17 montre le phénomène de conversion du bruit qui en prenant en compte la partie symétrique de la *DSP* résulte en un profil du bruit décrit selon la Figure 2.15.

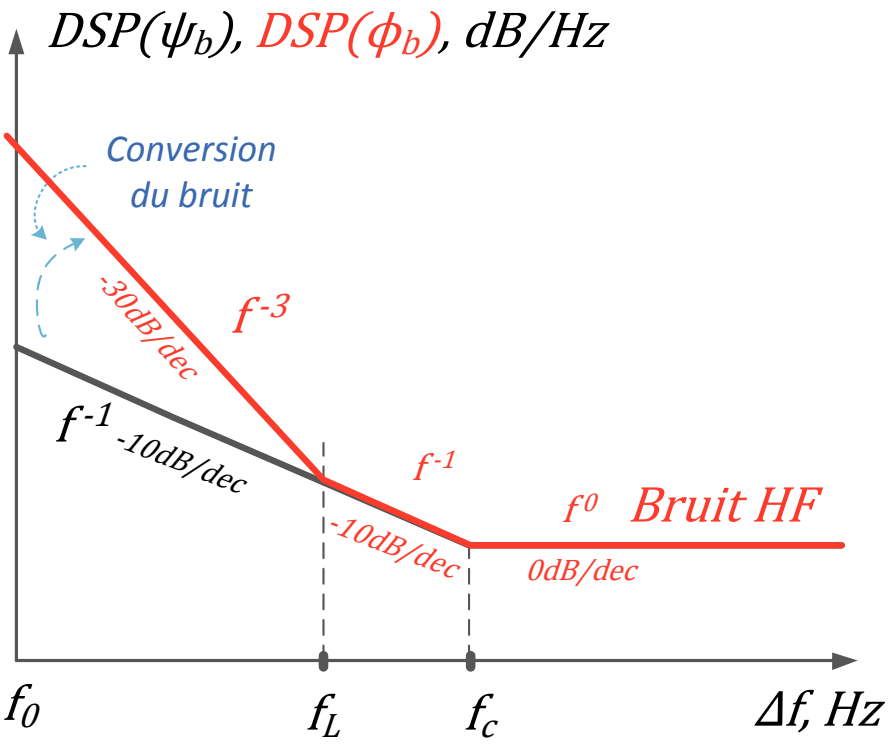
Cette modélisation du bruit de l'oscillateur est évidemment simplifiée mais suffisante pour bien comprendre les objectifs de performance à atteindre. Cela nous permettra d'écrire un cahier de charge pour la synthèse d'une ADPLL ainsi que de le transformer dans un langage mathématique nécessaire pour l'application des méthodes de synthèse.

Quantification Un autre point important concernant l'oscillateur utilisé dans l'architecture choisie est l'effet de quantification. Le *CNA* du *DCO* est modélisé comme un bloqueur d'ordre 0 à la sortie duquel s'ajoute un bruit blanc b_q modélisant le phénomène de quantification. Ce bruit de quantification est un bruit blanc dans le domaine des fréquences $[-f_s/2, f_s/2]$ où f_s est une fréquence d'échantillonnage (c'est-à-dire soit la fréquence du signal de la référence Ref soit la fréquence du signal local synchronisé Loc).

Incertitudes Le dernier point important à modéliser sont les variations du gain linéaire du *VCO* K_{VCO}. En réalité, les dispersions technologiques ou les variations



(a) $f_c < f_L$



(b) $f_L < f_c$

Figure 2.17: Principe de conversion du bruit dans l'oscillateur, l'effet Lesson

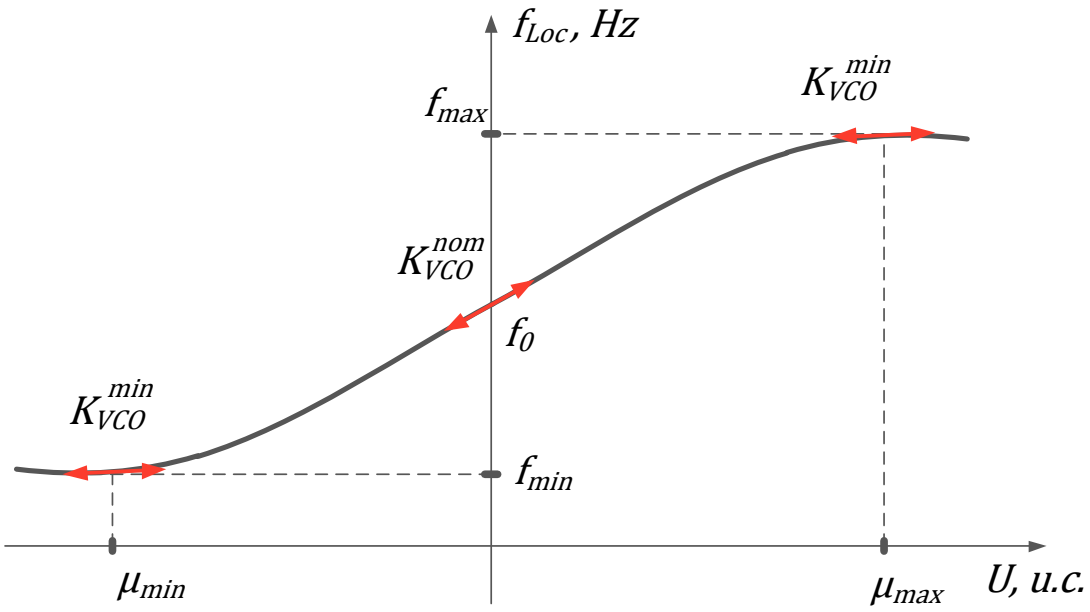


Figure 2.18: Un exemple de distorsion du gain linéaire du *VCO* provoquée par la variation des délais élémentaires

environnementales (température, alimentation) provoquent une variation des retards des inverseurs élémentaires du *DCO*. Cela implique deux phénomènes principaux : un changement du gain linéaire équivalent et une distorsion de la caractéristique linéaire de l'oscillateur (voir la Figure 2.18). On propose de modéliser ces variations comme dans les cas précédents par une incertitude réelle Δ_{VCO} sur le gain du *VCO* :

$$\begin{aligned} K_{VCO} &= K_{VCO}^{nom} + \Delta_{VCO} = K_{VCO}^{nom} + K_{VCO}^{gab} \cdot \delta_{VCO}; \\ K_{VCO} &\in \left[K_{VCO}^{nom} - K_{VCO}^{gab};\, K_{VCO}^{nom} + K_{VCO}^{gab}\right], \end{aligned} \tag{2.11}$$

où K_{VCO}^{nom} est la valeur nominale du gain du *VCO*, K_{VCO}^{gab} est le gabarit d'incertitude et δ_{VCO} est l'incertitude réelle scalaire normalisée : $|\delta_{VCO}| < 1$.

Le modèle complet de l'oscillateur ainsi obtenu est représenté sur la Figure 2.19.

2.2.4 Diviseur de fréquence

Le modèle de diviseur de fréquence dans le domaine de phase est un simple gain égal au facteur de division :

$$K_d = 1/d. \tag{2.12}$$

La division de fréquence est réalisée par une architecture simple de mise en série de bascules *D*. Le nombre de bascules détermine le facteur de division qui est un mul-

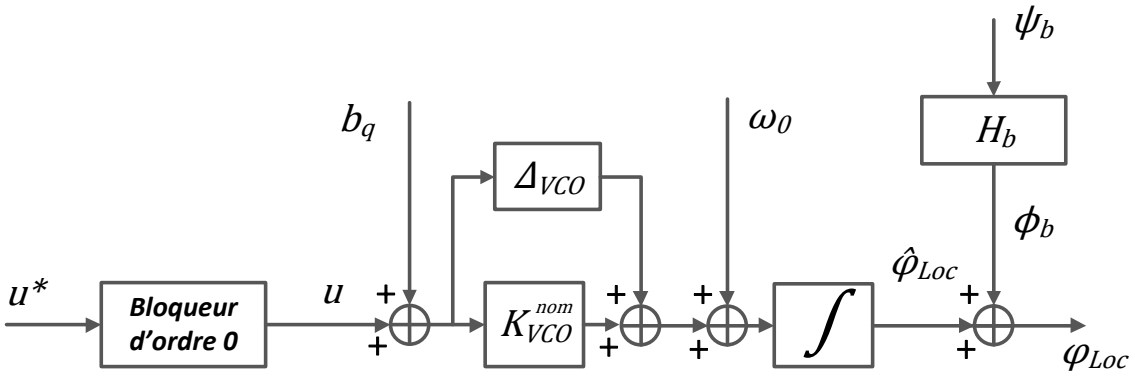


Figure 2.19: Modèle linéaire du DCO

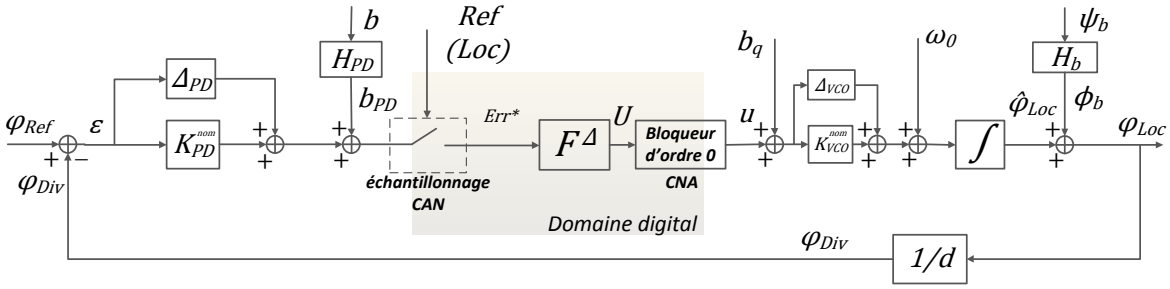


Figure 2.20: Modèle linéaire de l'ADPLL (Modèle d'analyse hybride)

tiple d'une puissance de deux $d = 2^l, l \in \mathbb{Z}$. On peut donc pratiquement implémenter la valeur idéale du gain de division de fréquence K_d.

2.2.5 Modèle complet de l'ADPLL

Le modèle complet linéaire de l'ADPLL dans le domaine de phase est ainsi représenté sur la Figure 2.20. Ceci est un Modèle d'analyse hybride. Les blocs Δ représentent les incertitudes paramétriques du modèle. L'indice Δ en expression du filtre F_Δ est également utilisé pour souligner que le modèle du filtre lui aussi possède des incertitudes et étant donné que l'on ne connait pas la structure du filtre (il n'est pas encore synthétisé) on ne les représente pas explicitement. Le même modèle nominal (Modèle de synthèse) est représenté sur la Figure 2.21. Le modèle nominal est composé donc de deux parties : une partie digitale ou discrète (correcteur, affiché en gris sur la Figure 2.21) et une partie continue (reste du schéma). A travers ce document, on va considérer la configuration des paramètres des ADPLLs pour l'implémentation sur une carte FPGA (Field-Programmable Gate Array) qui est résumée dans le Tableau 2.1.

Paramètre	**Configuration FPGA**
Fréquence de la référence	$f_{Ref} = 50.04\,kHz$
Nombre de bits à la sortie de TDC	$g = 4\,bits$
Nombre de bits à la sortie de PD	$k = 5\,bits$
Sensibilité minimale de PD	$\tau_{TDC} = 149.88\,ns$
Niveau de saturation (nombre d'inverseurs) de TDC	$15\,u.e.$
Différence de phase correspondante à la saturation	$\theta_{PD} = 2.25\,\mu s$
Gain linéaire nominal du PD	$K_{PD}^{nom} = 21.22\,u.e./rad$
Le gabarit d'incertitude du gain du PD	$K_{PD}^{gab} = 0.2K_{PD}^{nom}(20\%)$
Fréquence centrale du DCO	$f_0 = 250.14\,kHz$
Dispersion de la fréquence centrale du DCO	$f_0^{\triangle} = 0.3f_0(30\%)$
Fréquence minimale du DCO	$f_{min} = 139.6\,KHz$
Valeur du code du DCO correspondante	$\mu_{min} = 0\,u.c.$
Fréquence maximale du DCO	$f_{max} = 326\,KHz$
Valeur du code du DCO correspondante	$\mu_{min} = +1023\,u.c.$
Gain linéaire nominale du DCO	$K_{VCO}^{nom} = 1.006 \cdot 10^3\,rad/\,(s \cdot u.c.)$
Le gabarit d'incertitude du gain du DCO	$K_{VCO}^{gab} = 0.3K_{VCO}^{nom}\ (30\%)$
Division de fréquence	$K_d = 1/4$

Table 2.1: Configurations des paramètres selon le mode de réalisation de l'ADPLL

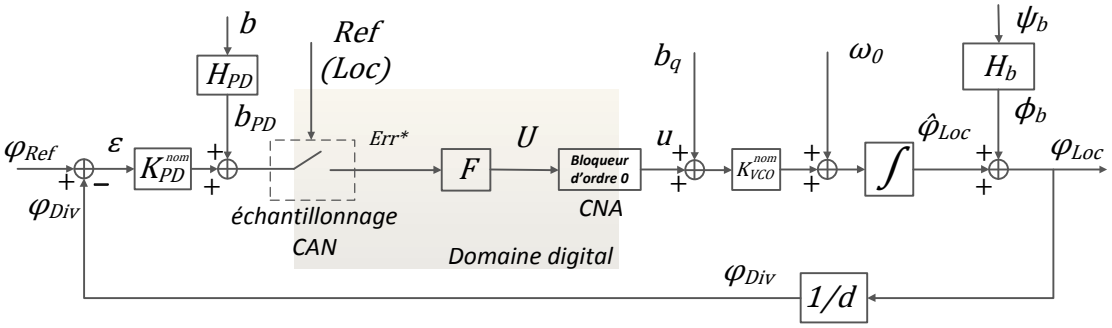


Figure 2.21: Modèle linéaire nominale de l'ADPLL (Modèle de synthèse hybride)

2.3 Les aspects d'auto-échantillonnage

Une propriété importante de l'ADPLL utilisée dans notre architecture est la propriété de l'*auto-échantillonnage*. En effet, étant donné que le comparateur de phase, utilisé dans l'architecture retenue, met à jour la valeur de sa sortie seulement après qu'un cycle de mesure soit achevé (c'est-à-dire après que les deux fronts montant de signaux Ref et Dev soient passés, voir la sous-section 2.1.1, page 36), le filtre F pour calculer la valeur du signal de commande à l'instant donné du temps, en fonction de l'ordre d'apparition des évènements (ref ou div), n'utilise pas les valeurs du signal à son entrée aux mêmes instants du temps. En effet, supposons que le filtre numérique F de l'ADPLL est échantillonné par les fronts montants (les évènements) du signal de référence Ref à l'instant $kT_{Ref} = \frac{k}{f_{Ref}}$, $k \in \mathbb{Z}$. Dans ce cas-là, si à l'instant kT_{Ref} le premier évènement dans le cycle de mesure en cours est l'évènement ref, le filtre utilisera la valeur du signal d'erreur à son entrée à l'instant $(k-1)\,T_{Ref}$ (donc retardé à une période T_{Ref} du signal Ref) puisque la valeur d'erreur de l'instant en cours kT_{Ref} n'est pas encore prête à la sortie du comparateur de phase (le cycle de mesure n'est pas encore fini). En revanche, si à l'instant kT_{Ref} le premier évènement dans le cycle de mesure en cours est l'évènement div, le filtre utilisera la valeur du signal d'erreur à son entrée au même instant kT_{Ref} puisque la valeur d'erreur de l'instant en cours kT_{Ref} est déjà prête à la sortie du comparateur de phase (le cycle de mesure est déjà fini [6]).

Cela implique qu'en fonction de la succession des évènements à l'entrée de l'ADPLL, elle se comporte comme deux systèmes différents : un système avec un retard supplémentaire l'autre sans ce retard supplémentaire dans la chaîne de la boucle ouverte. Ceci est un système auto-échantillonné hybride qui nécessite une étude adaptée de son comportement dynamique, comme par exemple les travaux de [AJO+10, AJGC10].

Cependant, le fait d'avoir utilisé comme signal d'échantillonnage du filtre le signal hautes fréquences Loc issu de l'ADPLL, divise par d le retard supplémentaire d'auto-échantillonnage, où d est le facteur de division de fréquence de l'ADPLL. En outre,

6. On suppose que le temps nécessaire au parcours du signal d'erreur de phase entre la sortie de comparateur de phase et l'entrée du filtre F est négligeable ou inférieur au temps de parcours de la même distance par le signal de référence Ref même.

les études [AJGC11, AJJ+11] montrent que lorsqu'il s'agit du réseau d'ADPLLs interconnectées auto-échantillonnées, son comportement peut être approché par un réseau de modèles d'ADPLLs linéaires. Cette approximation est plus proche de la réalité dans le cas de réseaux autonomes, c'est-à-dire sans référence extérieure. Sans rentrer plus en détails sur ce sujet, on considère dans la suite que cette approximation a été réalisée et on obtient les deux modèles échantillonnés classiques de l'ADPLL représentés sur la Figure 2.20 et la Figure 2.21, page 56. Par la suite, nous allons nous focaliser sur les aspects d'échantillonnage en méthodes de synthèse continues.

2.4 Méthodes de synthèse continues de correcteurs numériques

D'un point de vue Automatique, une ADPLL est une boucle fermée divisée en deux parties : le filtre ou le correcteur réalisant la fonction de commande, et le procédé couvrant toutes les autres fonctions. Le correcteur exerce la commande sur le procédé afin de garantir que les signaux en entrée et en sortie possèdent certaines propriétés. Étant donné que le correcteur est numérique et que le procédé est continu, l'ADPLL ainsi que ces modèles de synthèse et d'analyse obtenus (voir les Figure 2.20 et Figure 2.21) sont hybrides. Il s'agit donc de la commande d'un système continu par un correcteur discret. Malheureusement, les méthodes usuelles d'Automatique pour la synthèse de correcteur se concentrent généralement sur deux approches :

– l'approche « tout discret » : synthèse d'un correcteur discret pour un procédé discret ;
– l'approche « tout continu » : synthèse d'un correcteur continu pour un procédé continu.

Ces deux approches ne traitent donc pas de notre problème hybride. Cela est dû au fait que ce problème de commande de systèmes hybrides est toujours un problème ouvert et difficile car il n'existe pas à ce jour de solution efficace. Par ailleurs, l'application des méthodes de l'automatique fréquentielle, ce qui sera motivé par la suite, nécessite un modèle continu et fournit un correcteur analogique (fonction de transfert continue) ne permettant pas de répondre également à notre problème. Une façon de procéder est de mettre au point sur le modèle continu un correcteur continu puis de l'approcher par une fonction de transfert discrète. L'inconvénient d'une telle approche est de ne pas prendre en compte de façon directe lors de l'étape de synthèse du correcteur le fait que le système sera échantillonné-bloqué (voir la Figure 2.21). Le correcteur numérique obtenu peut présenter des performances dégradées par rapport au correcteur analogique initial en fonction du choix de la période d'échantillonnage. Par la suite, il est important de travailler sur un modèle échantillonné-bloqué.

Une façon de contourner le problème est de transformer un modèle discret en un modèle équivalent ayant les propriétés d'un modèle continu. Ce dernier sera appelé dans la suite « modèle continu équivalent ». Par équivalent, il faut comprendre que :

– le modèle continu équivalent est stable si et seulement si le modèle discret l'est ;
– les réponses fréquentielles des deux modèles coïncident au moins aux basses pulsations.

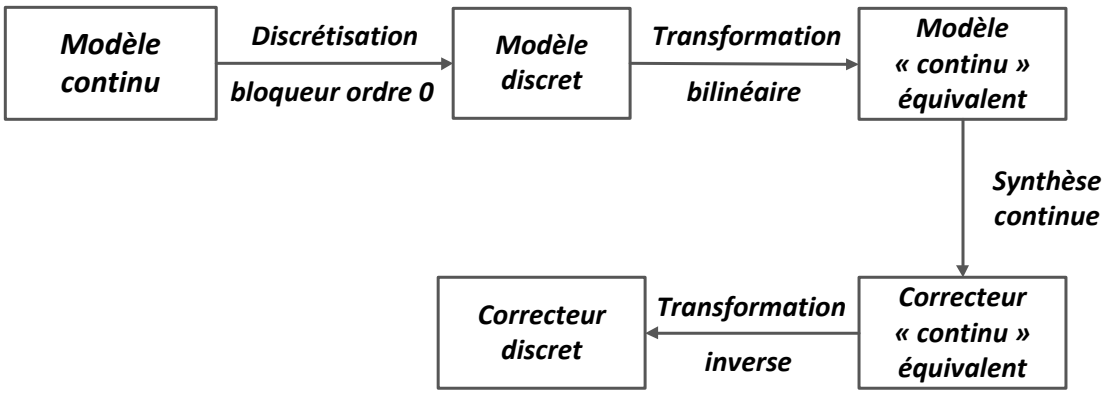


Figure 2.22: Synthèse continue d'un correcteur discret pour un système continu

Enfin, le modèle continu équivalent aura l'aspect d'un modèle continu s'il peut s'écrire comme une fonction rationnelle à coefficients réels d'une variable complexe notée w jouant le rôle de la variable de Laplace s associé à la fonction de transfert du modèle continu.

La transformation qui accomplit cela est la transformation bilinéaire ou transformation de Tustin. Elle consiste à remplacer [FPE86] dans la fonction de transfert discrète la variable z par une expression rationnelle en w :

$$z = \frac{1 + \frac{T_s}{2}w}{1 - \frac{T_s}{2}w}. \tag{2.13}$$

La réponse fréquentielle de la fonction de transfert correspondante est obtenue en remplaçant w par $j\nu$ où ν est l'analogue de la pulsation ω. Elles sont reliées par :

$$\nu = \frac{2}{T_s} tan \frac{\omega T_s}{2}. \tag{2.14}$$

Cette transformation va être utilisée pour synthétiser un correcteur discret pour un système continu à commander en utilisant les méthodes de synthèse continue (voir [SF08]) qui seront détaillées par la suite. Le modèle discret correspondant à un système échantillonné-bloqué est calculé de façon à disposer d'un modèle prenant en compte l'échantillonnage et le bloquage. A partir de ce modèle discret, un modèle continu équivalent est obtenu en appliquant (2.13). En se basant sur ce modèle, un correcteur "continu" (équivalent à un correcteur discret) peut être mis au point par des méthodes de synthèse continue. Le correcteur numérique correspondant est alors obtenu par la transformation inverse à (2.13). Cette démarche est représentée schématiquement sur la Figure 2.22.

En appliquant les transformations nécessaires décrites précédemment, on obtient le modèle continu équivalent incertain (Modèle d'analyse) et le modèle continu nominal (Modèle de synthèse) représentés sur la Figure 2.23 et la Figure 2.24 respectivement où l'étoile à côté de l'intégrateur est utilisée pour rappeler que c'est un intégrateur équivalent à l'intégrateur continu prenant en compte l'échantillonnage et le bloquage.

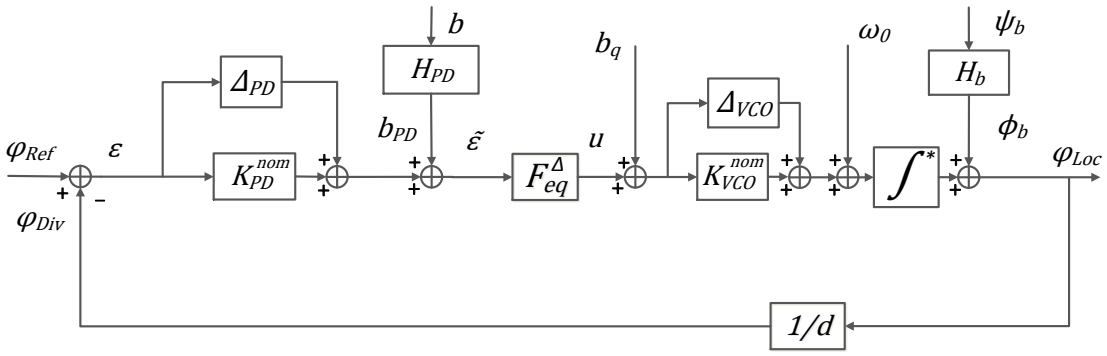


Figure 2.23: Modèle linéaire continu équivalent de l'ADPLL (Modèle d'analyse continu)

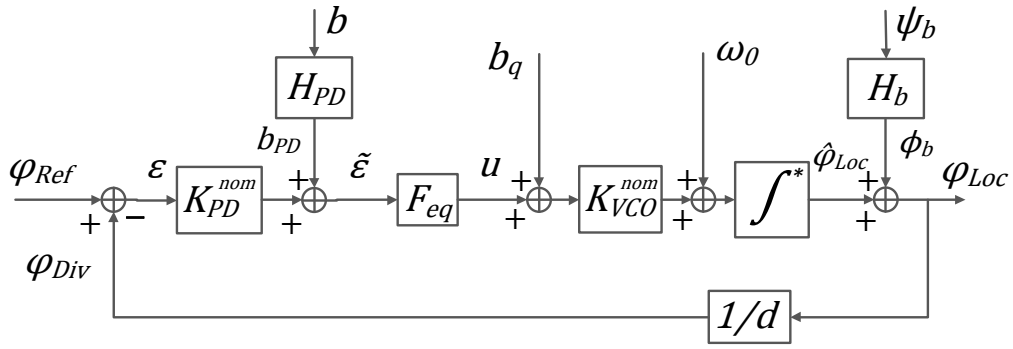


Figure 2.24: Modèle nominal linéaire continu équivalent de l'ADPLL (Modèle de synthèse continu)

En regroupant les signaux de perturbation en entrée du comparateur de phase et en entrée et sortie de l'oscillateur (2.15), on obtient un modèle équivalent transformé décrit sur la Figure 2.25.

$$\begin{aligned} q_{PD} &= b_{PD}/K_{PD}^{nom}; \\ q_{VCO} &= b_q + \omega_0/K_{VCO}; \\ q_{VCO}^{b} &= \phi_b/d. \end{aligned} \tag{2.15}$$

Sur la Figure 2.25, on regroupe le filtre avec le gain du *PD* pour obtenir un sous-système K que nous allons appeler par la suite le correcteur (souligné par les traits tirés rouges fin sur la Figure 2.25). En procédant de la même manière, on regroupe l'intégrateur du *VCO* avec son gain K_{VCO} et le gain de diviseur de fréquence $1/d$ pour obtenir le sous-système G (entourée par les traits tirés verts sur la Figure 2.25) que nous allons appeler par la suite le procédé.

On obtient ainsi un modèle simple linéaire de l'ADPLL sous forme d'un rebouclage de deux sous-systèmes : *(i)* le procédé G, déterminé par l'application (voit le Tableau 2.1), *(ii)* le correcteur, dont la forme et les paramètres (sauf le gain de *PD*)

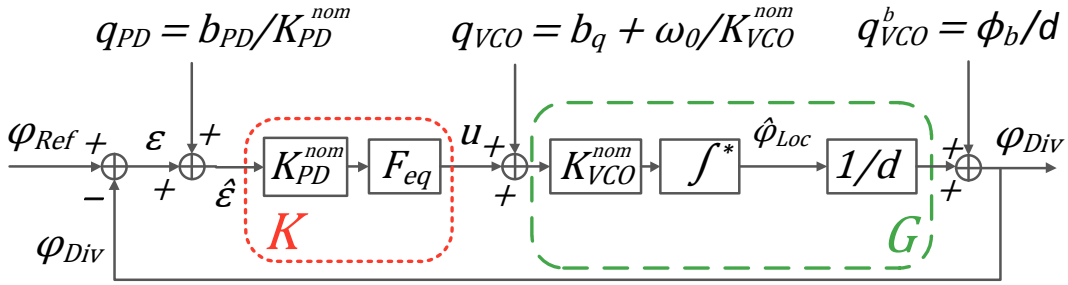


Figure 2.25: Modèle transformé de la Figure 2.24

reste à choisir par la suite. La section suivante présente une des méthodes de synthèse pertinente pour notre application : la synthèse fréquentielle avancée H_∞.

2.5 Synthèse de la loi de commande d'une ADPLL

Synthétiser un correcteur ou autrement dit une loi de commande signifie trouver une forme de correcteur, c'est-à-dire sa fonction de transfert $K(s)$, et déterminer les valeurs numériques de ses coefficients en se basant sur certains objectifs à atteindre (c'est-à-dire un cahier des charges à respecter). Comme cela a été évoqué dans la section précédente, nous allons utiliser la méthode de synthèse H_∞ pour réaliser cette synthèse. Les raisons principales de ce choix de la méthode se résument en quatre points :

1. La méthode de synthèse H_∞ est une méthode de synthèse dans le domaine fréquentiel, ce qui est très important du point de vue de l'application. En effet, les méthodes classiques de la conception d'une ou des ADPLLs en Microélectronique se basent essentiellement sur les méthodes fréquentielles classiques [Roh97, Raz98]. Le but de la thèse est de développer une méthode de synthèse de la loi de commande qui soit cohérente avec les méthodes de conception en Microélectronique et qui soit compréhensible par les ingénieurs travaillant dans ce domaine.
2. Comme cela a déjà été mentionné au début de ce chapitre (page 35), les méthodes de conception en Microélectronique avancée sont basées sur les outils CAO. La méthode de synthèse H_∞ permet de poser le problème de la conception comme un problème d'optimisation selon un critère mathématique fixé par un cahier des charges [SP05, SF08]. Le critère pouvant être mis sous forme d'optimisation convexe [GA94], le problème admet alors une résolution numérique efficace [NY83, BEFB94, BTN01]. Ainsi, la méthode H_∞ est particulièrement bien adaptée pour une insertion facile dans le flux des outils existants de la CAO en Microélectronique.
3. La méthode de synthèse H_∞ est une méthode puissante de synthèse fréquentielle avancée qui se base sur des résultats théoriques rigoureux ayant une connexion directe avec les méthodes et concepts en Automatique avancée telles

que : la Théorie de la dissipativité, l'approche entré-sortie, la robustesse, la μ-analyse. Cette connexion permet potentiellement d'élargir le champ d'application de la méthode de synthèse de la loi de commande d'une ADPLL simple vers une synthèse de la loi de commande pour un réseau d'ADPLLs interconnectées. La résolution de ce dernier problème, ne pouvant pas être accomplie par les méthodes de Microélectronique et de l'Automatique classique (ce qui sera illustré dans un exemple à la fin de ce chapitre), constitue la contribution principale de cette thèse.

4. Le dernier point concerne les perspectives futures. En effet, étant en relation étroite avec des méthodes et concepts de l'Automatique avancée cités précédemment, la synthèse H_∞ permet d'étendre, en perspective, le savoir faire vers des problèmes plus généraux et complexes englobant les aspect non-linéaires, temps variants ainsi que les retards et de la différence de sous-systèmes du réseau.

Le cahier des charges d'un point de vue mathématique est souvent exprimé sous la forme d'une caractéristique temporelle désirée des signaux d'entrée et de sortie du système. La méthode de synthèse H_∞ permet une formalisation élégante et simple du cahier de charge par un critère mathématique à optimiser. Il est nécessaire ainsi, avant de commencer la synthèse, d'établir le cahier des charges pour notre application.

2.5.1 Cahier des charges pour la synthèse d'une ADPLL

Stabilité

Un des points du cahier des charges le plus important à respecter est la stabilité. Étant donné que notre système est en général décrit par un modèle non-linéaire, ils existent plusieurs définitions de sa stabilité. Une des définitions la plus utilisée est la stabilité au sens de Lyapunov, c'est-à-dire la stabilité de l'un des points d'équilibre du système. Une autre définition de la stabilité d'un système est la stabilité au sens entrée-sortie que nous allons utiliser dans la présente étude :

Définition 2.2 (Stabilité au sens entrée-sortie)**.** Un système dynamique est un système *stable au sens entrée-sortie* (*input-output stability* en anglais), si pour tous les signaux d'entrées à énergie bornée, tous les signaux de sorties du système sont également d'énergie bornée.

Il est important de noter que la propriété de stabilité au sens entrée-sortie (comme tous les autres) est une propriété intrinsèque du système et ne dépend pas des signaux appliqués en entrée. Quand il s'agit d'un système composé d'une interconnexion de sous-systèmes (comme par exemple sur la Figure 2.25) il est utile de donner une définition de la *stabilité interne* d'un système :

Définition 2.3 (Stabilité interne)**.** Un système est dite *stable de façon interne* si toutes les fonctions de transfert du système en boucle fermée produisent des sorties à énergie bornée à partir d'entrées à énergie bornée.

Spécifications	Commentaires
Synchronisation	Suivi de référence (rampe), erreur statique nulle $\varepsilon_{stat} = 0$
Rapidité	Fréquence de coupure ω_S^c de $T_{\varphi_{Ref} \to \varepsilon}$: $\omega_S^c > \omega_{gab}$
Filtrage du bruit	Filtrer les HF de q_{PD}, les BF de q_{VCO} et q_{VCO}^b
Rejet de perturbations	Perturbations en entrée du *VCO* q_{VCO}
Limitation de la commande	Commande modérée

Table 2.2: Propriétés de Performance d'une ADPLL désirées

Ceci est une définition de la stabilité interne des systèmes linéaires qui peut être facilement étendue au cas des systèmes non-linéaires en remplaçant les fonction de transfert par les opérateurs correspondants (voir l'Annexe D du chapitre 3, page 143).

Pour un système linéaire toutes les définitions de stabilité coïncident (stabilité au sens de Lyapunov, au sens d'entrée-sortie, *etc.*). Par ailleurs, si un système linéarisé autour d'un point de fonctionnement est stable, le système d'origine non-linéaire est stable autour de ce point de fonctionnement [Kha92]. On s'intéresse alors à la synthèse d'un système, dont la linéarisation autour d'un point de fonctionnement est stable. Une première exigence est alors la stabilité du modèle linéaire de la Figure 2.25.

Performance

La stabilité seule ne garantit pas forcément un bon comportement de l'ADPLL (ou bien de son réseau entier). On fixe alors d'autres propriétés qui garantiront un bon fonctionnement du système complet. Nous allons appeler ces propriétés les spécifications de performance. Les propriétés de performance désirées d'une ADPLL sont résumées dans le Tableau 2.2.

Synchronisation Pour notre application, la propriété la plus importante est la synchronisation. L'ADPLL doit être capable de se synchroniser en fréquence et en phase avec un signal périodique extérieur Ref. Dans le domaine de phase, cette exigence est exprimée sous forme d'un suivi de la phase de référence φ_{Ref} (voir la Figure 2.25). L'ADPLL doit donc être capable de suivre une rampe à son entrée [Roh97, Kro03, KCV08]. Le suivi est d'autant mieux assuré que l'erreur $\varepsilon = \varphi_{Ref} - \varphi_{Div}$ est proche de zéro. Le signal d'erreur ε est caractérisé d'une part par son *régime permanent* et d'autre part, par son *régime transitoire*. La synchronisation concerne le régime permanent et est donc assurée par une minimisation de l'erreur de suivi de la référence : $\lim_{t \to \infty} (\varepsilon(t)) \to 0$.

Rapidité La rapidité (ou le temps de réponse du système) est aussi liée au signal d'erreur de suivi $\varepsilon(t)$, et concerne plutôt le régime transitoire. En réalité, de manière

qualitative, elle est reliée à la fréquence de coupure ω_S^c de la fonction de transfert $T_{\varphi_{Ref}\to\varepsilon}$ qui associe le signal de référence φ_{Ref} et le signal d'erreur de son suivi ε. Plus la fréquence de coupure est importante, plus la décroissance de l'erreur de suivi est rapide. On fixe pour objectif alors de garantir une fréquence de coupure supérieure à ω_{gab}.

Filtrage du bruit Comme nous l'avons vu dans la section précédente, l'architecture réelle de l'ADPLL possède différentes sources de bruit. Les trois sources les plus importantes sont le bruit à la sortie du *PD* $b_{_{PD}}$ dû à la quantification et les effets bang-bang généralement en hautes fréquences (HF), le bruit à l'entrée du *VCO* dû à la quantification b_q et le bruit à la sortie du *VCO* causé par le bruit de phase de l'oscillateur ϕ_b en basses fréquences (BF) (voir la Figure 2.24 et la Figure 2.25). Nous allons fixer comme objectif de les filtrer au maximum dans la plage de fréquences où cela est possible.

Rejet de perturbation Dans le cas où la pulsation centrale à l'entrée du *VCO* ω_0 correspond à la pulsation instantanée de la référence ω_{Ref} à l'entrée de l'ADPLL, elle peut être considérée comme une pré-commande virtuelle nécessaire pour une synchronisation rapide en fréquence de l'ADPLL. La situation où les deux pulsations ne sont pas égales peut être modélisée par un signal de différence de pulsations ε_ω à l'entrée du *VCO* tel que : $\omega_0 = \omega_{Ref} + \varepsilon_\omega$. Cette situation peut se produire durant un régime transitoire en présence d'une différence initiale entre deux pulsations et/ou de perturbations provenant de l'alimentation et du changement de la température influant sur la pulsation centrale du *VCO* ce qui s'exprime par une perturbation en basses fréquences à son entrée. C'est donc un signal de perturbation à rejeter ε_ω et le système complet doit être capable ainsi de le rejeter.

Limitation de la commande La limitation de la commande générée par le filtre est motivée par le fait qu'il n'est pas toujours possible de réaliser pratiquement la commande sur une étendue dynamique importante ce qui se traduit pratiquement par une amplitude trop élevée. Ainsi, dans le contexte de l'ADPLL, la valeur maximale de commande est nécessairement limitée sur une certain nombre de bits. En général, on essaie de minimiser le nombre de bits afin de minimiser le nombre d'opérations de calcul et le nombre de connexions, ce qui permet au final de diminuer la consommation. Ainsi, il n'est pas envisageable de réaliser une amplitude de commande trop élevée obligeant à prendre en compte un nombre de bits trop important. Par ailleurs, la commande est principalement sollicitée en HF à cause du bruit HF à la sortie du *PD* $q_{_{PD}}$. Il faut donc limiter le gain de la fonction de transfert correspondante dans cette plage de fréquences.

2.5.2 Cahier des charges sous forme d'un critère mathématique

En regardant plus en détails le schéma-bloc de la Figure 2.25, nous pouvons remarquer qu'il est identique au schéma classique de commande d'un système G bouclé par

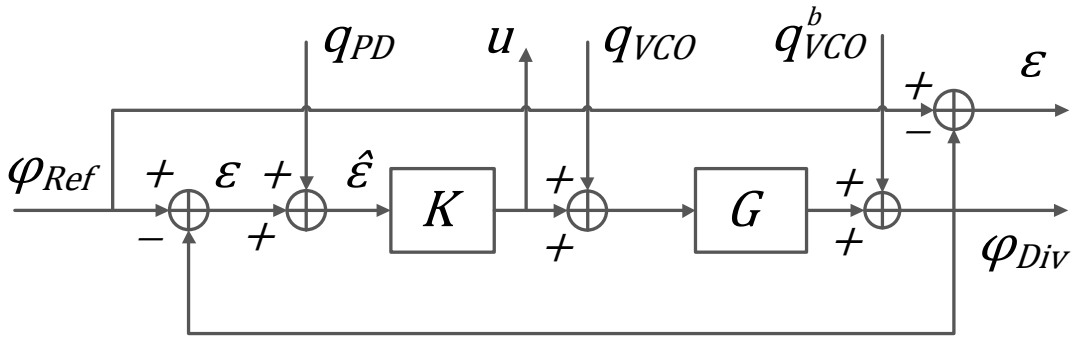


Figure 2.26: Schéma classique de commande d'un système boucle fermée

un correcteur K (voir la Figure 2.26). On voit naturellement apparaître les signaux d'entrée décrits dans le cahier des charges :

– la consigne $\varphi_{_{Ref}}$;
– les perturbations et bruits $q_{_{VCO}}$, $q^b_{_{VCO}}$ et $q_{_{PD}}$

ainsi que les signaux de sortie :

– la sortie de l'ADPLL $\varphi_{_{Div}}$;
– l'erreur de suivi de référence $\varepsilon = \varphi_{_{Ref}} - \varphi_{_{Div}}$;
– la commande u délivrée par le correcteur K.

L'approche naturelle pour décrire les spécifications de performance sous forme mathématique consiste à imposer une contrainte à respecter pour les signaux temporels (voir la Figure 2.27 pour un exemple). Les signaux temporels ne doivent pas sortir de la zone de cette contrainte afin que les spécifications de performance soient satisfaites.

Cependant, le critère mathématique de performance utilisé dans la méthode H_∞ est exprimé dans le domaine fréquentiel plutôt que dans le domaine temporel. La raison principale de cette « traduction » des caractéristiques temporelles en caractéristiques fréquentielles est une grande simplicité d'exploitation d'un point de vue mathématique de ces dernières [SF08]. Par la suite, nous allons ainsi utiliser les spécifications fréquentielles. Il est donc nécessaire de « traduire » les spécifications de performance du domaine temporel dans le domaine fréquentiel.

Notons $T_{x\to y}(s)$ la fonction de transfert du signal d'entrée x vers le signal de sortie y. On a alors les relations suivantes entre les sorties et les entrées du système de la

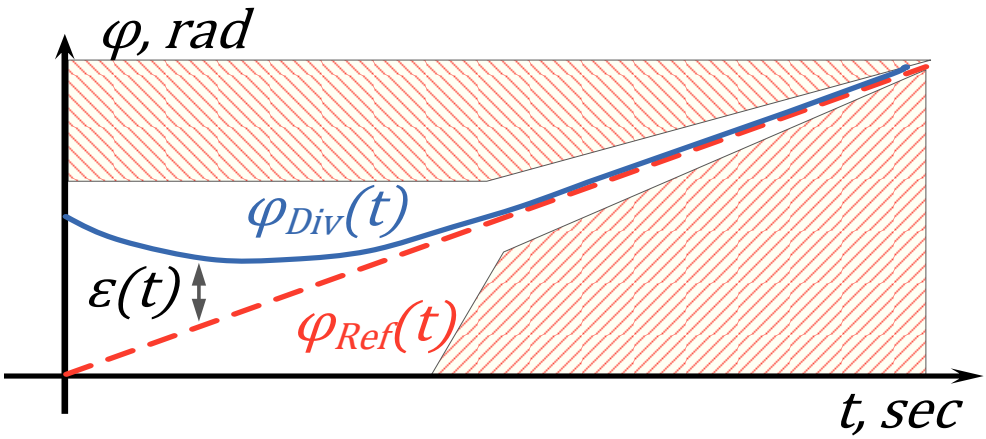


Figure 2.27: Un exemple de la contrainte temporelle d'une spécification de performance à atteindre. La zone rouge décrit une zone à ne pas franchir

Figure 2.26 :

$$\begin{aligned}
\varphi_{Div}(s) &= T_{\varphi_{Ref}\to\varphi_{Div}}(s)\,\varphi_{Ref}(s) + T_{q_{PD}\to\varphi_{Div}}(s)\,q_{PD}(s) + \\
&+ T_{q_{VCO}\to\varphi_{Div}}(s)\,q_{VCO}(s) + T_{q^b_{VCO}\to\varphi_{Div}}(s)\,q^b_{VCO}(s); \\
\varepsilon(s) &= T_{\varphi_{Ref}\to\varepsilon}(s)\,\varphi_{Ref}(s) + T_{q_{PD}\to\varepsilon}(s)\,q_{PD}(s) + \\
&+ T_{q_{VCO}\to\varepsilon}(s)\,q_{VCO}(s) + T_{q^b_{VCO}\to\varepsilon}(s)\,q^b_{VCO}(s); \\
u(s) &= T_{\varphi_{Ref}\to u}(s)\,\varphi_{Ref}(s) + T_{q_{PD}\to u}(s)\,q_{PD}(s) + \\
&+ T_{q_{VCO}\to u}(s)\,q_{VCO}(s) + T_{q^b_{VCO}\to u}(s)\,q^b_{VCO}(s),
\end{aligned} \tag{2.16}$$

où $T_{\varphi_{Ref}\to\varepsilon}(s) = \frac{1}{1+G(s)K(s)}$ souvent notée $S(s)$ est appelée ***fonction de sensibilité*** ;

$T_{\varphi_{Ref}\to\varphi_{Div}}(s) = \frac{G(s)K(s)}{1+G(s)K(s)}$ souvent notée $T(s)$ est appelée *fonction de transmission* car elle relie l'entrée de consigne φ_{Ref} à la sortie φ_{Div}. Elle est aussi nommée ***fonction de sensibilité complémentaire*** car on a la relation :

$$S(s) + T(s) = \frac{1}{1+G(s)K(s)} + \frac{G(s)K(s)}{1+G(s)K(s)} = 1;$$

Spécification	Fonction de transfert concernée
Synchronisation	$S(s)$
Rapidité	$S(s)$
Filtrage du bruit	$S(s)$, $GS(s)$, $T(s)$
Rejet de perturbations	$GS(s)$
Limitation de la commande	$KS(s)$,$T(s)$

Table 2.3: Critères de performance et fonctions de transfert concernées

$$
\begin{aligned}
T_{q_{PD}\to\varphi_{Div}}(s) &= T(s); & T_{q^b_{VCO}\to\varepsilon}(s) &= -S(s);\\
T_{q_{VCO}\to\varphi_{Div}}(s) &= G(s)S(s) = GS(s); & T_{\varphi_{Ref}\to u}(s) &= K(s)S(s) = KS(s);\\
T_{q^b_{VCO}\to\varphi_{Div}}(s) &= S(s); & T_{q_{PD}\to u}(s) &= KS(s);\\
T_{q_{PD}\to\varepsilon}(s) &= -T(s); & T_{q_{VCO}\to u}(s) &= -T(s);\\
T_{q_{VCO}\to\varepsilon}(s) &= -GS(s); & T_{q^b_{VCO}\to u}(s) &= -KS(s).
\end{aligned}
$$

Il est possible d'étudier chacune des spécifications de performance indépendamment les unes des autres. Ceci est pleinement justifié par *le Théorème de Superposition* qui est valable ici car le modèle que nous étudions est linéaire stationnaire [SF08]. Ce théorème permet d'affirmer que l'influence de plusieurs signaux d'entrée non nuls sur les signaux de sortie est égale à la somme des influences de chacun des signaux d'entrée pris individuellement. Chaque spécification peut donc être étudiée indépendamment en examinant les fonctions de transfert reliant les signaux d'entrée et de sortie concernés.

Stabilité

La description mathématique de la stabilité interne du modèle linéaire de l'ADPLL peut être exprimée comme la stabilité de toutes les fonctions de transfert internes de la boucle : $S(s)$, $T(s)$, $KS(s)$, $GS(s)$ (voir la Figure 2.26). Il est d'autre part possible de démontrer qu'il y a stabilité interne si et seulement si la fonction de transfert S est stable et s'il n'y a pas de compensations pôles-zéros instables entre le système G et le correcteur K.

Des indicateurs classiques de stabilité d'un système bouclé sont les marges de stabilité : *marge de gain* et *marge de phase*. Ces indicateur de stabilité sont utilisés dans le cas de systèmes monovariables (Single-Input Single-Output, SISO en anglais). Il est important de rappeler que l'objectif principale de cette thèse est le développement de la méthode de synthèse d'un réseau d'ADPLLs couplées, c'est-à-dire un système

global multivariable à plusieurs entrées et à plusieurs sorties (Multi-Input Multi-Output, MIMO en anglais). Dans le cas d'un système multivariable, les marges de stabilité classiques perdent leur sens. Néanmoins, une généralisation de ces dernières est possible couvrant le cas des systèmes MIMO. Il s'agit de la ***marge de module*** Δ_M qui pour le cas de systèmes monovariables dans le diagramme de Nyquist peut s'exprimer comme la distance minimale entre le point critique $(-1,\, 0)$ et le tracé de la fonction de transfert en boucle ouverte $K(j\omega)\, G(j\omega)$:

$$\Delta_M = \inf_{\omega} \left|1 + K(j\omega)\, G(j\omega)\right|. \tag{2.17}$$

La dernière expression peut être facilement réécrite à l'aide de la fonction de sensibilité S :

$$\Delta_M = \inf_{\omega} \left|1 + K(j\omega)\, G(j\omega)\right| = \frac{1}{\sup_{\omega} \left|\left(1 + K(j\omega)\, G(j\omega)\right)^{-1}\right|} = \left(\sup_{\omega} \left|S(j\omega)\right|\right)^{-1}. \tag{2.18}$$

Il est donc possible en fixant une borne supérieure MS sur la fonction de sensibilité S de fixer une borne inférieure sur la marge de module Δ_M :

$$\sup_{\omega} \left|S(j\omega)\right| \leq MS \Leftrightarrow \Delta_M \geq 1/MS. \tag{2.19}$$

En général pour garantir une bonne marge de module, on fixe la borne supérieure $MS \approx 8\, dB$.

La partie gauche de (2.19) est en réalité la norme H_∞ de la fonction de transfert S qui est un scalaire que l'on va désigner par la suite par $\|S\|_\infty$. Pour le cas plus général des systèmes MIMO, c'est-à-dire pour la matrice des fonctions de transfert, il est aussi possible de définir la marge de module sous forme :

$$\Delta_M = \frac{1}{\|S\|_\infty}. \tag{2.20}$$

Dans ce cas-là, la norme H_∞ est définie au sens plus large pour les systèmes MIMO :

Définition 2.4 (Norme H_∞)**.** *Pour une matrice de fonctions de transfert stables $H(s)$ la norme H_∞ est définie par :*

$$\|H\|_\infty = \sup_{Re(s)>0} \overline{\sigma}\left(H(s)\right) = \sup_{\omega \in [0,+\infty]} \overline{\sigma}\left(H(j\omega)\right), \tag{2.21}$$

où $Re(s)$ désigne la partie réelle du nombre complexe s et $\overline{\sigma}(A)$ la valeur singulière maximale de la matrice complexe A définie comme suit :

$$\overline{\sigma}(A) = \max\left(\sigma_1, \sigma_2, \ldots, \sigma_N\right), \tag{2.22}$$

où σ_i sont les N valeurs singulières de la matrice A qui sont définie comme les racines carrées des valeurs propres de la matrice A^*A, c'est-à-dire $\sigma_i = \sqrt{\lambda_i(A^*A)}$.

Pour le cas simple monovariable, du fait qu'une fonction de transfert scalaire a une

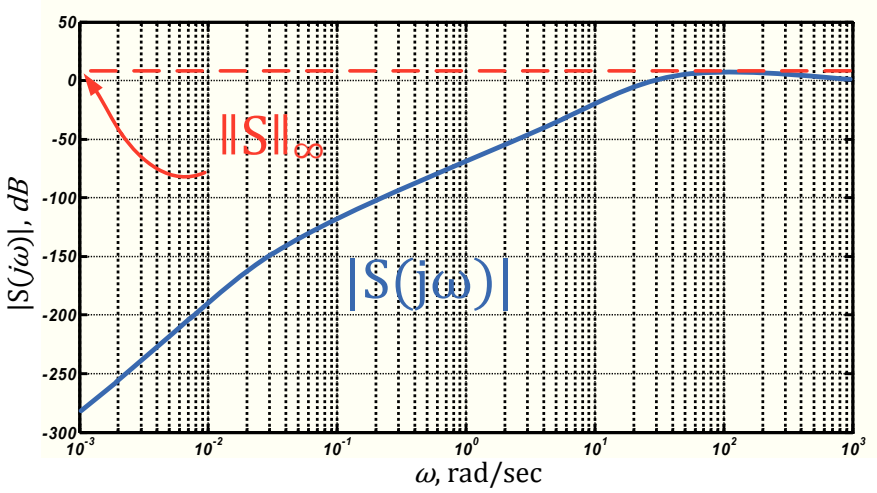


Figure 2.28: Illustration de la norme H_∞ d'une fonction de transfert S

seule valeur singulière égale à son gain $\sqrt{H(j\omega)^* H(j\omega)}$, la norme H_∞ indique la valeur maximale du gain de la fonction de transfert (voir la Figure 2.28).

Propriétés de la norme H_∞ Les propriétés suivantes de la norme H_∞ seront utiles dans la suite de ce document.

La norme H_∞ de la mise en série de deux (matrices de) fonctions de transfert est bornée par le produit des normes H_∞ de chacune des fonctions de transfert :

$$\|G(s)\,H(s)\|_\infty \leq \|G(s)\|_\infty \|H(s)\|_\infty . \tag{2.23}$$

La norme H_∞ d'une matrice de fonctions de transfert est supérieure à la norme H_∞ de chacun de ses éléments :

$$\begin{gathered} \left\| \begin{bmatrix} G(s) \\ H(s) \end{bmatrix} \right\|_\infty \geq \|G(s)\|_\infty ; \qquad \left\| \begin{bmatrix} G(s) \\ H(s) \end{bmatrix} \right\|_\infty \geq \|H(s)\|_\infty ; \\ \left\| \begin{bmatrix} G(s) & H(s) \end{bmatrix} \right\|_\infty \geq \|G(s)\|_\infty ; \quad \left\| \begin{bmatrix} G(s) & H(s) \end{bmatrix} \right\|_\infty \geq \|H(s)\|_\infty . \end{gathered} \tag{2.24}$$

Performance

Spécifications sur la fonction de sensibilité S : Comme mentionné précédemment, la synchronisation de l'oscillateur interne de l'ADPLL avec l'oscillateur extérieur est assurée par le suivi d'une rampe à l'entrée de l'ADPLL. Pour pouvoir exprimer cette spécification sous forme d'un critère fréquentiel, on fait appel au théorème de la valeur finale.

Théorème 2.5 (Théorème de la valeur finale)**.** *Si tous les pôles de $s\varepsilon(s)$ sont dans le demi plan de gauche (axe imaginaire exclu) alors*

$$\lim_{t\to+\infty} \varepsilon(t) = \lim_{s\to 0} s\varepsilon(s) . \tag{2.25}$$

Par application de ce Théorème 2.5 on a :

$$\lim_{t\to+\infty} \varepsilon(t) = \lim_{s\to 0} s\varepsilon(s) = \lim_{s\to 0} sT_{\varphi_{Ref}\to\varepsilon}(s)\,\varphi_{Ref}(s). \tag{2.26}$$

La transformée de Laplace d'une rampe est $\varphi_{Ref}(s) = \frac{\omega_{Ref}}{s^2}$ où $\omega_{Ref} \in \mathbb{R}$ est la pente de la rampe, on obtient alors :

$$\lim_{t\to+\infty} \varepsilon(t) = \omega_{Ref} \lim_{s\to 0} \frac{T_{\varphi_{Ref}\to\varepsilon}(s)}{s}. \tag{2.27}$$

Pour assurer que l'erreur de suivi de trajectoires $\varepsilon(t)$ tende vers 0 quand t tend vers l'infini, la fonction de transfert $T_{\varphi_{Ref}\to\varepsilon}(s)$ doit donc contenir au moins deux zéros en zéro : $T_{\varphi_{Ref}\to\varepsilon}(s) = s^k T'_{\varphi_{Ref}\to\varepsilon}(s)$ avec $k \geq 2$ et tous ses pôles doivent être à partie réelle strictement négative. Le premier zéro est utilisé pour la compensation du pôle en zéro de $s\varepsilon(s)$. Les conditions du Théorème 2.5 sont donc satisfaites et l'expression (2.27) est donc valable. Le deuxième zéro en zéro compense alors l'erreur de trainage.

L'exigence de deux zéros en zéro de la fonction de transfert $T_{\varphi_{Ref}\to\varepsilon}(s)$ peut s'exprimer comme « la fonction de transfert $T_{\varphi_{Ref}\to\varepsilon}(s)$, ou de manière équivalente la fonction de sensibilité $S(s)$, a une pente d'au moins $+40dB/dec$ en basses fréquences ».

Nous allons procéder de la même manière pour la contrainte de rapidité. En règle générale, le *temps de réponse* évolue de manière inversement proportionnelle à la fréquence de coupure ω_S^c de la fonction de sensibilité $S(s)$. En outre, plus l'atténuation du module de la fonction de sensibilité $S(s)$ est importante en BF, plus le système est rapide en poursuite c'est-à-dire plus le temps de réponse est petit [SP05, SF08]. Nous allons utiliser par la suite cette relation qualitative qui dans le cas de systèmes simples du deuxième ordre peut être évaluée quantitativement [SF08], pour fixer la contrainte de performance de la rapidité de l'ADPLL.

Une autre spécification concernant la fonction de sensibilité S est la spécification de filtrage du bruit du *VCO* regroupé dans le signal q^b_{VCO} sur le signal de sortie de l'ADPLL φ_{Div} (voir la Figure 2.25, page 60 et la Figure 2.26, page 64). Il s'agit notamment de diminuer l'influence du bruit à la sortie de l'oscillateur ϕ_b sur le signal de sortie de l'ADPLL φ_{Div}, la fonction de transfert concernée en boucle fermé est la fonction de sensibilité S (voir le Tableau 2.3, page 66).

En se basant sur la section 2.2, on rappelle que le niveau de bruit de l'oscillateur ϕ_b est important en basses fréquences. Il est donc préférable d'avoir un faible gain de la fonction de sensibilité S en BF pour non seulement respecter les spécifications de synchronisation et de rapidité mais aussi pour assurer le filtrage du bruit de l'oscillateur. Concernant la contribution du bruit de *PD* q_{PD}en hautes fréquences, il n'est pas possible de diminuer son niveau sur la sortie $\widehat{\varepsilon}$ car, indépendamment du correcteur, le gain de la fonction de sensibilité S reliant ces deux signaux en HF est égale à 1. En effet, en HF le gain de la boucle ouverte est en général faible, c'est-à-dire négligeable par rapport à 1. D'un côté, c'est lié au fait qu'habituellement le procédé

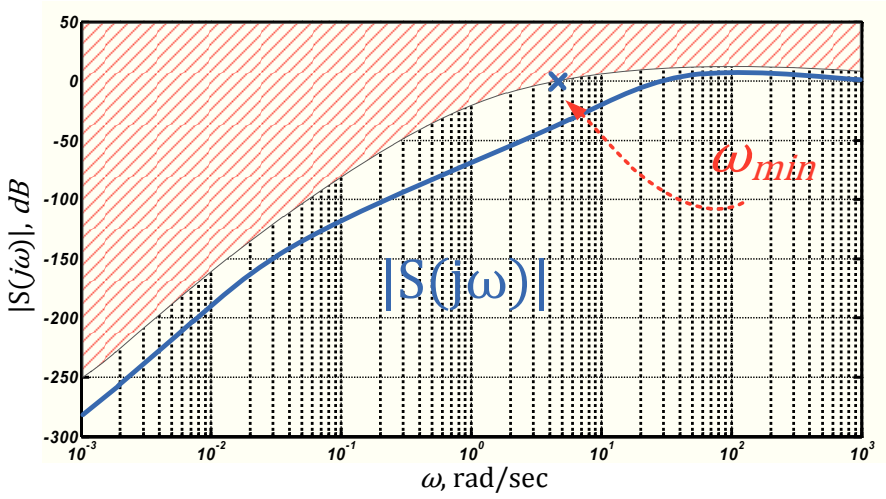


Figure 2.29: Un exemple de contrainte fréquentielle à respecter

G a un faible gain en HF (comme par exemple dans l'application de la commande de l'ADPLL, le procédé contient une action d'intégration pure $G = \frac{K_G}{s} G'$). D'un autre côté, la spécification de limitation de commande selon le cahier des charges fixe un gain faible du correcteur dans cette plage de fréquences. Nous avons donc la relation approchée pour les HFs :

$$|S(j\omega)| = \left| \frac{1}{1 + G(j\omega) K(j\omega)} \right| \approx 1. \tag{2.28}$$

C'est une limitation possible sur les spécifications que l'on peut exiger. Pour plus de détails sur cette limitation ainsi que toutes les autres, le lecteur est invité à consulter [SP05].

Au final, on obtient une contrainte fréquentielle sur la fonction de sensibilité S (voir la Figure 2.29) à respecter pour garantir la performance visée (voir le Tableau 2.3). Il s'agit maintenant de déterminer une méthode de synthèse pour satisfaire cette contrainte fréquentielle. La réponse à cette problématique réside dans l'utilisation de fonctions de pondération, un outil clef en commande H_∞.

Soient deux fonctions de transfert stables $W_{Ref}(s)$ et $W_\varepsilon(s)$ appelées fonctions de pondération telles que leur produit inverse fixe la contrainte fréquentielle à satisfaire (voir la Figure 2.30). Sous cette hypothèse, en cas des systèmes SISO, la fonction de sensibilité S respecte la contrainte fréquentielle qui s'écrit :

$$|S(j\omega)| \leq \left| \left(W_\varepsilon(j\omega) W_{Ref}(j\omega) \right)^{-1} \right| = \left| (W_S(j\omega))^{-1} \right|, \forall \omega \in \mathbb{R}. \tag{2.29}$$

L'expression (2.29) pour le cas MIMO peut se réécrire sous la forme suivante :

$$\overline{\sigma} \left(W_\varepsilon(j\omega) S(j\omega) W_{Ref}(j\omega) \right) \leq 1, \forall \omega \in \mathbb{R} \Leftrightarrow \left\| W_\varepsilon S W_{Ref} \right\|_\infty \leq 1. \tag{2.30}$$

Cette méthode permettant d'imposer la forme désirée pour les fonctions transfert en

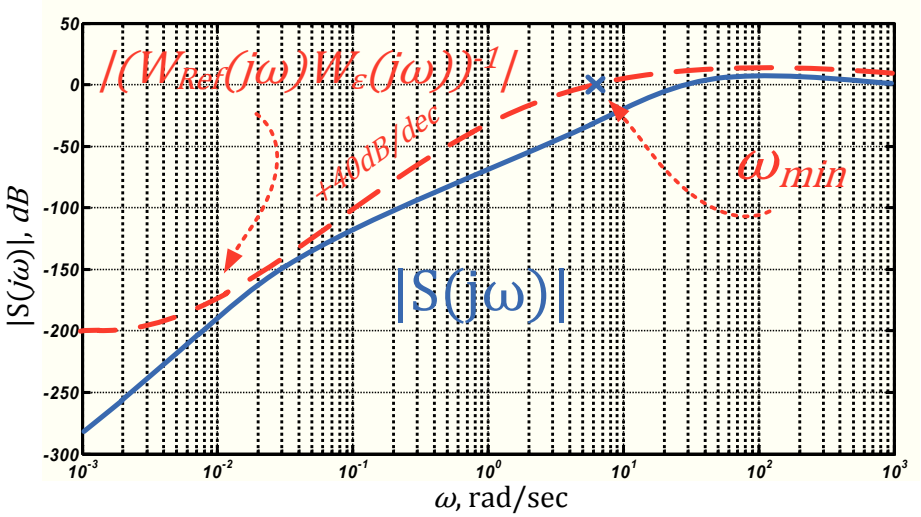


Figure 2.30: Un exemple d'utilisation de fonctions de pondération pour la réalisation de la contrainte fréquentielle

boucle fermée grâce à l'utilisation de fonctions de pondération est appelée la méthode de modelage (ou loop shaping en anglais) de fonctions de transfert en boucle fermé.

Mentionnons que la contrainte fréquentielle en BF $+40dB/dec$ signifie qu'une ou deux fonctions de pondération aient un intégrateur pur. Ceci est en contradiction avec l'exigence de stabilité des fonctions de pondération étant donné que l'intégrateur est un système instable. En pratique, pour contourner cette limitation, à la place d'intégrateurs purs on peut utiliser des quasi-intégrateurs, c'est-à-dire des fonctions de transfert avec une constante de temps très élevée ou de manière équivalente la pulsation de coupure ω_c très faible (voir la Figure 2.31). L'erreur statique ε_{stat} ou de trainage ε_{train} provoquée en poursuite dans ce cas-là reste négligeable.

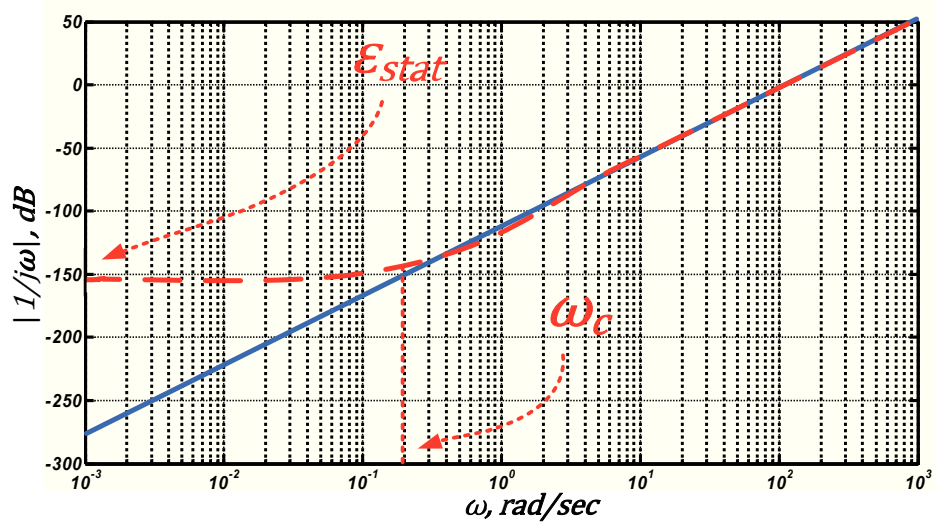


Figure 2.31: Quasi-intégrateur

Spécifications sur les fonctions de sensibilité T, KS, GS : De la même façon, on formalise les spécifications de performance sous forme d'une contrainte fréquentielle sur les autres fonctions de transfert en boucle fermée, à savoir $T(s)$, $KS(s)$ et $GS(s)$. On ne rentre pas dans les détails de cette formalisation étant donné que la présente étude ne porte pas sur la méthode de synthèse H_∞ même, mais sur son application au réseau d'ADPLLs. Néanmoins, les détails nécessaires peuvent être retrouvés dans [SP05, SF08]. On obtient donc les contraintes fréquentielles suivantes :

$$\begin{aligned} &\left\| W_{Div} T W_{Ref} \right\|_\infty \leq 1 \Leftrightarrow \\ &|T(j\omega)| \leq \left| \left(W_{Div}(j\omega) W_{Ref}(j\omega) \right)^{-1} \right| = \left| \left(W_T(j\omega) \right)^{-1} \right|, \forall \omega \in \mathbb{R}; \end{aligned} \tag{2.31}$$

$$\begin{aligned} &\left\| W_u K S W_{Ref} \right\|_\infty \leq 1 \Leftrightarrow \\ &|KS(j\omega)| \leq \left| \left(W_u(j\omega) W_{Ref}(j\omega) \right)^{-1} \right| = \left| \left(W_{KS}(j\omega) \right)^{-1} \right|, \forall \omega \in \mathbb{R}; \end{aligned} \tag{2.32}$$

$$\begin{aligned} &\left\| W_{Div} G S W_{VCO} \right\|_\infty \leq 1 \Leftrightarrow \\ &|GS(j\omega)| \leq \left| \left(W_{Div}(j\omega) W_{VCO}(j\omega) \right)^{-1} \right| = \left| \left(W_{GS}(j\omega) \right)^{-1} \right|, \forall \omega \in \mathbb{R}. \end{aligned} \tag{2.33}$$

Le cahier des charges du Tableau 2.2, page 62 est ainsi une mise en forme des contraintes (2.30)- (2.33) à respecter. Elles peuvent être facilement exprimées comme un critère mathématique d'optimisation. Pour cela, imaginons un système multivariable $H(s)$ reliant deux signaux en entrée $\left[\varphi_{Ref}, q_{VCO}\right]^T$ avec les signaux en sortie $[\varepsilon, u]^T$ (voir la Figure 2.32) qui est défini par [7] :

$$\begin{bmatrix} \varepsilon(s) \\ u(s) \end{bmatrix} = \overbrace{\begin{bmatrix} S(s) & -GS(s) \\ KS(s) & -T(s) \end{bmatrix}}^{\Phi(s)} \begin{bmatrix} \varphi_{Ref}(s) \\ q_{VCO}(s) \end{bmatrix}. \tag{2.34}$$

Soient les signaux :

$$\begin{aligned} \widetilde{\varphi}_{Ref}(s) &= W_{Ref}^{-1}(s) \varphi_{Ref}(s); \qquad \widetilde{u}(s) = W_u(s) u(s); \\ \widetilde{q}_{VCO}(s) &= W_{VCO}^{-1}(s) q_{VCO}(s); \qquad \widetilde{\varepsilon}(s) = W_\varepsilon(s) \varepsilon(s), \end{aligned} \tag{2.35}$$

7. Les signaux q_{PD}, q_{VCO}^b et φ_{Div} ont été omis étant donné qu'ils apparaissent dans l'expression entrée-sortie de la boucle (2.16) représentée sur la Figure 2.26, page 64 avec les mêmes fonctions de transfert déjà utilisées dans (2.34).

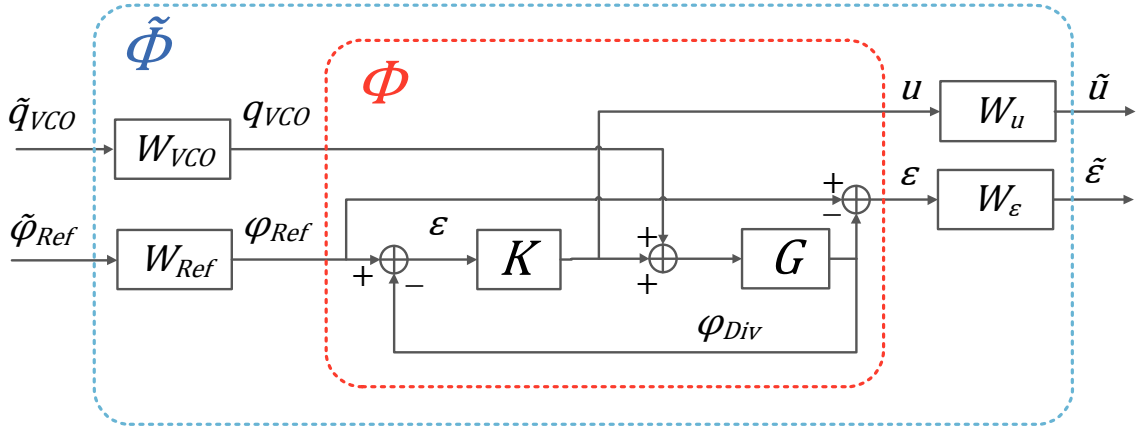


Figure 2.32: Système multivariable H

l'expression (2.34) peut donc être reformulée sous la forme suivante :

$$\begin{bmatrix} \widetilde{\varepsilon}(s) \\ \widetilde{u}(s) \end{bmatrix} = \underbrace{\begin{bmatrix} W_\varepsilon(s) S(s) W_{Ref}(s) & -W_\varepsilon(s) GS(s) W_{VCO}(s) \\ W_u(s) KS(s) W_{Ref}(s) & -W_u(s) T(s) W_{VCO}(s) \end{bmatrix}}_{\widetilde{\Phi}(s)} \begin{bmatrix} \widetilde{\varphi}_{Ref}(s) \\ \widetilde{q}_{VCO}(s) \end{bmatrix}. \quad (2.36)$$

Les spécifications de performance peuvent alors être exprimées sous forme de la minimisation d'une norme H_∞ de la matrice de fonction de transfert $\widetilde{\Phi}(s)$:

$$\begin{aligned} &\min_{K(s)} \gamma \\ &\left\| \widetilde{\Phi} \right\|_\infty \leq \gamma. \end{aligned} \quad (2.37)$$

Le critère exprimé dans (2.37) porte le nom du « critère 4 blocs » dont de plus amples détails peuvent être trouvés dans [Fon95].

Si la contrainte du problème d'optimisation (2.37) est satisfaite, en prenant en compte les propriétés de la norme H_∞ (2.23) et (2.24) on obtient :

$$\begin{aligned} \left\| W_\varepsilon(s) S(s) W_{Ref}(s) \right\|_\infty &\leq \gamma; \\ \left\| W_\varepsilon(s) GS(s) W_{VCO}(s) \right\|_\infty &\leq \gamma; \\ \left\| W_u(s) KS(s) W_{Ref}(s) \right\|_\infty &\leq \gamma; \\ \left\| W_u(s) T(s) W_{VCO}(s) \right\|_\infty &\leq \gamma. \end{aligned} \quad (2.38)$$

Si en plus on a

$$\gamma \leq 1, \quad (2.39)$$

les spécifications (2.30)-(2.33) sont vérifiées par (2.38).

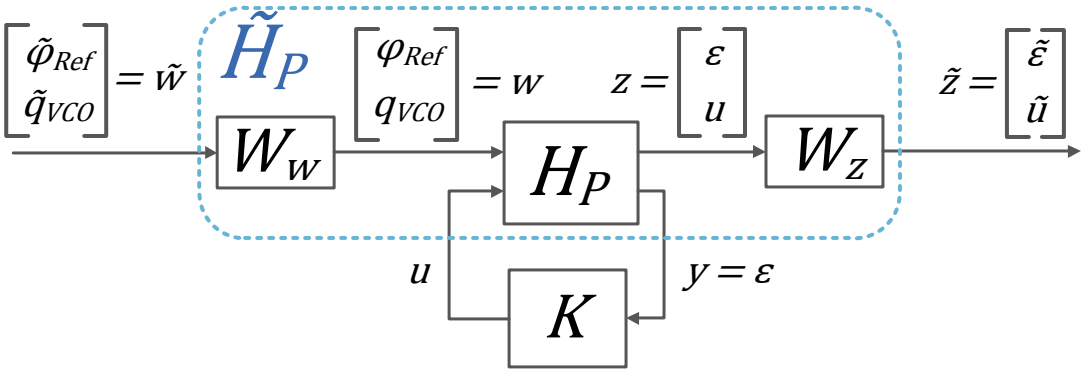


Figure 2.33: Modèle sous la forme du problème H_∞-standard

2.5.3 Mise en forme standard d'une synthèse H_∞

Le schéma représenté sur la Figure 2.32 après quelques transformations peut être mis sous forme canonique d'une interconnexion du correcteur K avec un sous-système H_P appelé le procédé (ou « plant » en anglais d'où l'indice P) à commander (voir la Figure 2.33). Le procédé dans notre cas d'application est défini par l'expression suivante :

$$H_P(s) = \begin{bmatrix} 1 & -G(s) & -G(s) \\ 0 & 0 & 1 \\ 1 & -G(s) & -G(s) \end{bmatrix}. \tag{2.40}$$

En ajoutant les matrices de pondération ayant pour fonctions de transfert W_w et W_z telles que :

$$W_w(s) = \begin{bmatrix} W_{Ref}(s) & 0 \\ 0 & W_{VCO}(s) \end{bmatrix}, \text{ et } W_z(s) = \begin{bmatrix} W_\varepsilon(s) & 0 \\ 0 & W_u(s) \end{bmatrix}, \tag{2.41}$$

on obtient une interconnexion du correcteur avec le procédé augmenté $\widetilde{H_P}(s)$ dont la fonction de transfert entre les nouvelles entrées $\tilde{w}$ et sorties $\tilde{z}$ extérieures est définie par :

$$T_{\tilde{w}\to\tilde{z}}(s) = \widetilde{H_P}(s) \star K(s) = W_z(s)\left(H_P(s) \star K(s)\right) W_w(s), \tag{2.42}$$

où $\star$ désigne le *produit Redheffer* (ou *star product* en anglais) dont la définition est rappelée ci-dessous.

Définition 2.6. Le *produit de Redheffer* ou le *produit étoile* de la matrice de fonctions de transfert $H(s)$ partitionnée en 4 éléments :

$$H(s) = \begin{bmatrix} H_{11}(s) & H_{12}(s) \\ H_{21}(s) & H_{22}(s) \end{bmatrix},$$

par une matrice de fonctions de transfert $G(s)$ partitionnée en 4 éléments :

$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix},$$

noté $H(s) \star G(s)$ est défini par la matrice de fonctions de transfert suivante (en omettant la variable de Laplace s) :

$$H \star G = \begin{bmatrix} H_{11} + H_{12}G_{11}(I - H_{22}G_{11})^{-1} H_{21} & H_{12}(I - G_{11}H_{22})^{-1} G_{12} \\ G_{21}(I - H_{22}G_{22})^{-1} H_{21} & G_{22} + G_{21}H_{22}(I - G_{11}H_{22})^{-1} G_{12} \end{bmatrix}. \quad (2.43)$$

Dans le cas particulier où $G_{12} = 0$, $G_{21} = 0$, $G_{22} = 0$ et $G = G_{11}$, on a :

$$H(s) \star G(s) = H_{11}(s) + H_{12}(s) G_{11}(s) (I - H_{22}(s) G_{11}(s))^{-1} H_{21}(s), \quad (2.44)$$

et dans le cas où $H_{12} = 0$, $H_{21} = 0$, $H_{22} = 0$ et $H = H_{11}$, on a :

$$H(s) \star G(s) = G_{22}(s) + G_{21}(s) H(s) (I - G_{11}(s) H(s))^{-1} G_{12}(s). \quad (2.45)$$

La transformation (2.44) est une *transformation fractionnelle linéaire inférieure* (en anglais *Lower (linear) Fractional Transformation, LFT*). Elle est représentée sur la Figure 2.33 avec $H(s) = H_P(s)$ et $G(s) = K(s)$. A son tour la transformation (2.45) est une *transformation fractionnelle linéaire supérieure* (en anglais *Upper (linear) Fractional Transformation, UFT*).

Par la suite nous allons appeler la forme canonique de la Figure 2.33 la forme H_∞-standard. La raison pour laquelle on la désigne de cette manière est le fait que la Figure 2.33 décrit le problème standard de la synthèse H_∞. En effet, trouver un correcteur garantissant la stabilité du système de la Figure 2.26 et les spécifications de performance décrites dans le Tableau 2.2, page 62 peut se résumer par la formulation ci-dessous.

Définition 2.7 (Problème H_∞-standard)**.** Étant donné $\gamma > 0$, existe-il un correcteur K tel que :

(i) le système bouclé $\widetilde{H_P}(s) \star K(s)$ de la Figure 2.33 est stable ;

(ii) la norme H_∞ de la fonction de transfert entre les entrées $\widetilde{w}$ et sorties $\widetilde{z}$ extérieures est inférieur à $\gamma \in \mathbb{R}$:

$$\left\| \widetilde{H_P}(s) \star K(s) \right\|_\infty < \gamma.$$

Si oui, construire un correcteur K assurant pour le système en boucle fermée les deux propriétés précédentes.

Il existe deux méthodes de résolution du problème exprimé dans la Définition 2.7 appelé la synthèse H_∞ : *(i)* par une série d'équations de Riccati [DGKF89] et *(ii)* par la méthode d'optimisation sous contrainte *Inégalité Matricielle Affine* (ou LMI de l'anglais *Linear Matrix Inequality*) [GA94]. Ces résultats peuvent être retrouvés également dans [DF99, Duc03, SP05, Sco06]. Nos travaux se basent sur la méthode de résolution du problème H_∞-standard par l'optimisation sous contraintes LMIs exigeant moins d'hypothèses et étant ainsi plus générale. Une brève discussion sur l'optimisation LMI un outils clef de la méthode choisie est reportée dans l'annexe A, page 102 de ce chapitre. La solution même du problème H_∞-standard est discutée dans l'annexe B, page 106. Ici, on donne seulement un algorithme de la synthèse de la loi de commande K pour le modèle présenté sur les Figures 2.25, page 60 et 2.26, page 64 garantissant la stabilité et la performance décrit dans le Tableau 2.2 :

Algorithme 2.1 Synthèse H_∞ standard

1. Choisir les fonctions de pondérations en se basant sur les spécifications de performance (voir le Tableau 2.2, page 62) ;
2. Poser le problème sous forme standard (voir la Figure 2.33 et la Définition 2.7) ;
3. Résoudre le problème standard H_∞ (voir le Problème 2.7) en minimisant la valeur de γ (voir l'Annexe B, page 106 de ce chapitre) ;
4. Si la valeur de γ résolvant le problème standard H_∞ (voir le Problème 2.7) n'a pas été trouvée, la stabilité n'est pas donc garantie, il faut changer les fonctions de pondération en fixant un autre ensemble de spécifications de performance et revenir au point 2 du présent algorithme ;
5. Si $\gamma > 1$, la stabilité est garantie mais la performance non, on peut changer les fonctions de pondération en fixant un autre ensemble de spécifications de performance et revenir au point 2 du présent algorithme ;
6. Si $0 < \gamma \leq 1$, le correcteur assurant la stabilité et la performance est synthétisé, fin de l'algorithme.

2.5.4 Exemple numérique de la synthèse d'une loi de commande d'une ADPLL locale

Pour illustrer l'application de la méthode H_∞ pour la synthèse de la loi de commande, on considère le Modèle de synthèse d'ADPLL (voir page 33) de la Figure 2.24 avec les valeurs numériques de la Configuration FPGA présentée dans le Tableau 2.1, page 55. Après quelques itérations de l'Algorithme 2.1, on obtient les résultats suivants.

1. Les spécifications de performance (voir le Tableau 2.2, page 62) sont mises sous la forme de contraintes fréquentielles sur les modules des réponses fréquentielles des fonctions de transfert en boucle fermée du modèle (voir le Tableau 2.4, page 79) réalisées par les modules des réponses fréquentielles des fonctions de

pondération représentées sur la Figure 2.34 dont l'expression est donnée par :

$$\begin{aligned} W_{S}(s) &= W_{\varepsilon}(s) W_{Ref}(s) = \frac{0.36(s+1085)}{s+0.24}; \\ W_{GS}(s) &= W_{\varepsilon}(s) W_{VCO}(s) = \frac{2.5(s+1085)}{s+0.24}; \\ W_{KS}(s) &= W_{u}(s) W_{Ref}(s) = 0.091; \\ W_{T}(s) &= W_{u}(s) W_{VCO}(s) = 0.625. \end{aligned} \tag{2.46}$$

Le choix de ces dernières a été fait en se basant sur la sous-section 2.5.1, page 61. Néanmoins il est nécessaire de faire quelques remarques concernant les valeurs numériques et leur réalisation pratique.

a) La contrainte fréquentielle de $+40dB/dec$ en BF sur la fonction de sensibilité S est remplacée par une contrainte de $+20dB/dec$ sur la fonction de transfert en boucle fermée GS. En effet, comme cela est expliqué dans l'annexe B, page 106, l'ordre du correcteur K obtenu après une synthèse H_∞ dépend de l'ordre du procédé augmenté $\tilde{P}$ (il est, en générale, égale à son ordre). Ainsi, les fonctions de pondération faisant partie du procédé augmenté contribuent à son ordre global. On a donc intérêt à utiliser des fonctions de pondération d'ordre minimal pour minimiser l'ordre du correcteur. Aussi, pour réaliser la contrainte de $+40dB/dec$ en BF sur la fonction de sensibilité S, il est possible de l'imposer directement par la fonction de pondération W_S. Dans ce cas-là, son ordre est égal à 2 au minimum. Il existe cependant une autre solution. On remarque que le procédé G contient une action d'intégration pure $\frac{K_G}{s}$, c'est-à-dire que son module en BF représente une pente de $+20dB/dec$. La fonction de sensibilité S peut se réécrire sous forme de $S = G^{-1}GS$. En imposant ainsi une contrainte de $+20dB/dec$ en BF de la fonction de transfert GS : $|GS(j\omega)| \leq |W_{GS}(j\omega)|$ on impose automatiquement la contrainte de $+20 + 20 = +40dB/dec$ en BF sur la fonction de sensibilité S : $|S(j\omega)| \leq |G^{-1}(j\omega) W_{GS}(j\omega)|$. Pour réaliser la contrainte de $+20dB/dec$ en BF une fonction de pondération W_{GS} de l'ordre 1 peut être utilisée.

b) La marge de gain d'un système composé d'un procédé sous forme d'un intégrateur pur et d'un correcteur PI est infinie. Cependant, il ne faut pas oublier que le bloc G inclus dans le modèle de l'ADPLL considéré n'est pas un intégrateur pur. Hormis l'action d'intégration, il regroupe aussi la dynamique issue de l'échantillonnage et du bloquage comme expliqué dans la section 2.4, page 57. C'est pour ces raisons que la marge de gain ne sera pas forcement infinie. Néanmoins, la contrainte fréquentielle maximale de $8dB$ sur le module de la fonction de sensibilité S fixe des marges de stabilité minimales tout à fait acceptables, correspondantes à la marge de module $\Delta_M^{min} \approx 0.4$ (voir la section 2.5.1, page 61).

c) La contrainte sur la pulsation de coupure minimale $\omega_{min} = 420\, rad/sec$ de

la fonction de sensibilité S fixe un temps de montée maximum de l'ordre de $\tau_m^{max} = 7.1 \cdot 10^{-3}\, sec$ [8] soit à peu près 35 périodes du signal d'horloge généré par l'ADPLL selon la Configuration FPGA du Tableau 2.1, page 55.

d) La contrainte de $+20\, dB/dec$ sur la fonction de transfert GS assure le rejet de signaux de perturbation en BF en entrée du *VCO*, le module maximal de $-8\, dB$ assure l'atténuation de signaux de perturbation sur toute la plage de fréquences en entrée du *VCO*.

e) Pour mieux respecter l'exigence sur la limitation de la commande, il est préférable d'imposer une contrainte supplémentaire de $-20\, dB/dec$ en HF sur le module de la fonction de transfert KS. Cela s'exprime par une action de filtrage supplémentaire en HF du correcteur et donc par l'augmentation de son ordre. Cependant, étant donné que l'objectif final est d'appliquer une loi de commande sur un réseau d'ADPLLs, le principe de la structure de commande la plus simple possible, pour des raisons d'implémentation pratique, est adopté dans ce travail. On va voir par la suite qu'un correcteur PI est un bon compromis pour assurer la synchronisation du réseau tout en garantissant un niveau de commande tolérable même si on n'a pas une pente de $-20\, dB/dec$ en HF pour la fonction de transfert KS. On impose alors, aussi pour le cas d'une seule ADPLL, l'ordre du correcteur égale à 1 et une valeur maximale du module de fonction de transfert KS égale à environ $20\, dB$ sans filtrage supplémentaire en HF.

f) Concernant la fonction de sensibilité complémentaire T, elle permet, grâce à sa propriété structurelle (un gain de l'ordre de 1 dans les BFs et faible dans les HFs) et une contrainte de $4\, dB$ maximale, d'atténuer l'influence du bruit en HF du comparateur de phase q_{PD} sur le signal de l'erreur de suivi ε et de limiter la commande du correcteur u excitée par la perturbation à l'entrée du *VCO* q_{VCO}, voir la Figure 2.26, page 64.

2. Le problème sous forme standard est représenté par la Figure 2.33, page 74.
3. La synthèse H_∞ fournit le correcteur d'ordre 1 sous la forme $K(s) = \frac{0.43(s+440.9)}{s+0.24}$ avec la valeur de γ correspondante : $\gamma = 0.92$. C'est un correcteur à retard de phase souvent utilisé dans le passé à la place du correcteur PI. On remarque que le pôle de la fonction de transfert du correcteur, $p_{_K} = -0.24$, est négligeable par rapport à son zéro, $z_{_K} = -440.9$, et est lié au fait d'avoir approximé l'intégration pur de la fonction de pondération W_ε par un presque-intégrateur (voir la Figure 2.31, page 71). On propose donc d'approximer ce pôle par un pôle en zéro pur $p_0 = 0$, ce qui n'influera pas sur la satisfaction des contraintes de performance au final. Le correcteur discret, équivalent au correcteur continu, est un correcteur PI avec la structure représentée sur la Figure 2.7, page 41 avec les coefficients numériques $K_p = 4.27 \cdot 10^{-1}$ et $K_i = 3.8 \cdot 10^{-3}$. Les fonctions de

8. Ce temps de montée correspond au temps de montée d'une fonction de transfert canonique d'ordre deux sous forme $T = \frac{\omega_0}{s^2+2\zeta\omega_0 s+\omega_0^2}$ ce qui n'est pas forcement le cas pour une ADPLL considérée dans cette étude. On utilise cependant cet indicateur pour avoir l'ordre de grandeur du temps de montée et donc du temps de réponse du système.

Fonctions de transfert en boucle fermée	Contraintes concernées réalisée par W_S, W_{GS}, W_{KS}, W_T respectivement
S	$+40dB/dec$ en BF, la pulsation de coupure minimale $\omega_{min} = 420\,rad/sec$, le temps de réponse maximale correspondant $\tau_m^{max} \approx 7 \cdot 10^{-3}\,sec$, gain maximal de $8\,dB$
GS	$+20dB/dec$ en BF, gain maximal de $-8\,dB$
KS	Gain maximal de $20.8dB$
T	Gain maximal de $4.1dB$

Table 2.4: Contraintes fréquentielles sur les fonctions de transfert en boucle fermée du modèle

transfert en boucle fermée obtenues sont représentées sur la Figure 2.35. On peut donc conclure que les contraintes fréquentielles fixées sont respectées, le modèle nominal de la boucle à verrouillage de phase est donc stable et possède le niveau de la performance demandé. Les marges de stabilité associées sont respectées : marge du module $\Delta_M \approx 0.98$, marge de gain $MG \approx 33\,dB$, marge de phase $MP \approx 78°$ ce qui est confirmé par le lieu de Nichols représenté sur la Figure 2.36.

2.6 Robustesse vis-à-vis d'incertitudes paramétriques

Dans la section précédente, nous avons décrit un algorithme de synthèse de la loi de commande d'un modèle nominal garantissant la stabilité et un niveau de performance désiré. Néanmoins, en pratique, comme cela est décrit par exemple dans la section 2.2, page 43, il existe toujours des incertitudes sur les paramètres nominaux dûes aux dispersions technologiques de la réalisation pratique du système et dûes aux erreurs de modélisation. Il est donc nécessaire d'assurer la stabilité et un niveau de performance non seulement pour un modèle nominal du système (Modèle de synthèse, voir page 33) mais aussi pour toute une famille de modèles le décrivant et prenant en compte ces incertitudes (Modèle d'analyse, voir page 33).

De manière habituelle, les concepteurs en électronique utilisent deux méthodes d'analyse du comportement du modèle d'un système électronique incertain, toutes les deux se basant sur les simulations : les simulations « corner » et les simulations Monte-Carlo [RK08, Kra98, Kra06]. Les simulations selon la première méthode sont réalisées pour chaque valeur extrême de chaque paramètre incertain du modèle d'où le nom de la méthode « corner » (coin en français). Une limitation de cette méthode d'analyse est que les problèmes de stabilité ou de dégradation de performance ne se produisent pas forcement pour les valeurs extrêmes de paramètres incertains. En pratique, ils ont lieu pour des valeurs de paramètres incertain à l'intérieur de leur plage de variation. La méthode Monte-Carlo à son tour permet de réaliser les simulations du système sur un maillage important de points en prenant en compte

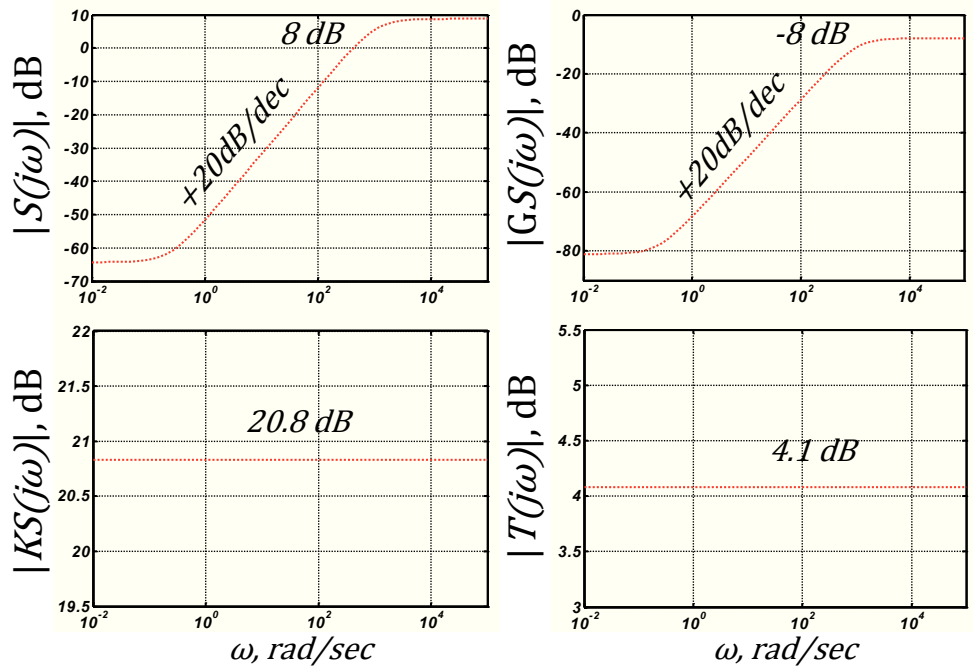


Figure 2.34: Les contraintes correspondantes aux fonctions de pondération choisies pour l'ADPLL locale

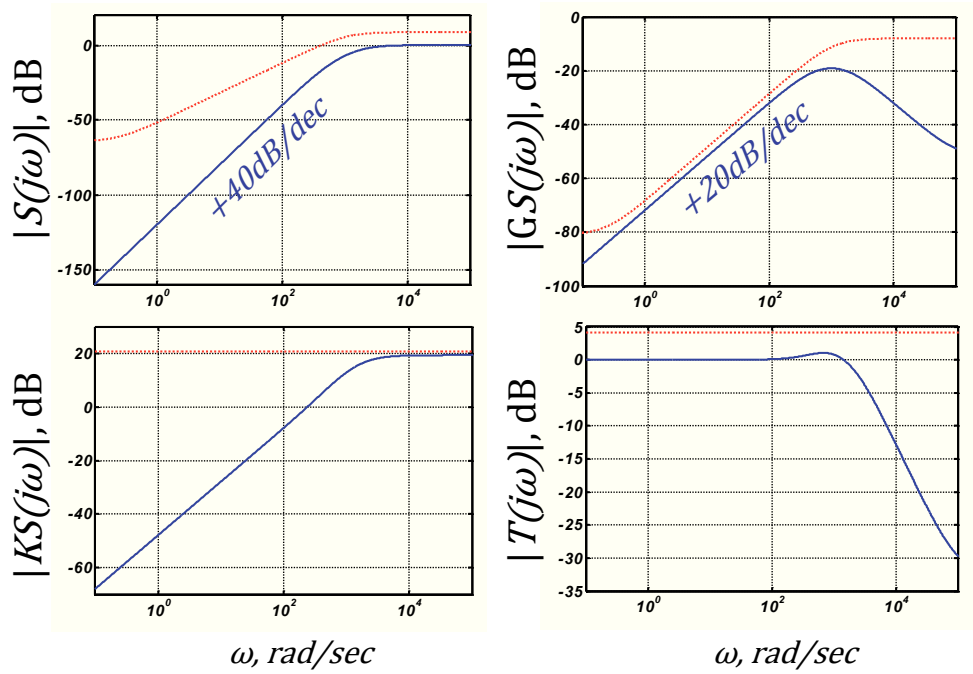


Figure 2.35: Les fonctions de transfert en boucle fermée obtenues après la synthèse H_∞

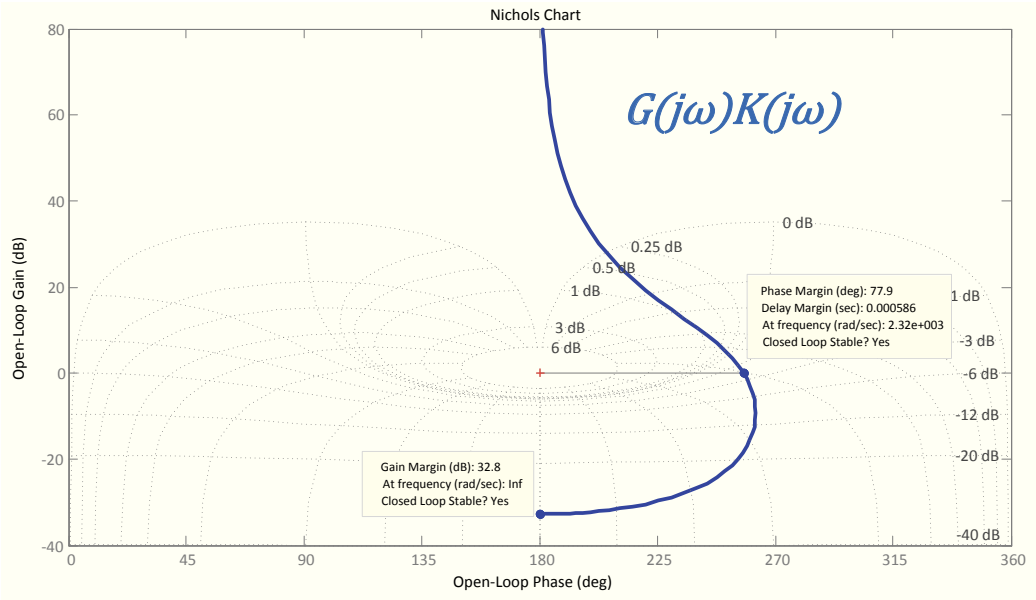


Figure 2.36: Le lieu de Nichols de la boucle ouverte obtenue

leur probabilité. Une grande limitation de cette méthode est le temps important nécessaire pour réaliser les simulations pour des modèles de systèmes complexes. Le problème devient d'autant plus important quand il s'agit d'un modèle du réseau de systèmes complexes interconnectés tel que le réseau d'ADPLLs interconnectées considéré dans cette étude.

Pour ces raisons nous proposons de combiner la méthode de simulations Monte-Carlo avec un outil avancé de la μ-analyse [SP05]. En plus d'être une méthode fréquentielle, elle présente l'intérêt s'exprimer sous la forme d'un problème d'optimisation d'un critère mathématique. Comme dans la cas de la méthode de synthèse de la loi de commande H_∞ (voir la page 60), elle peut alors être intégré dans le flux de CAO existant en Microélectronique.

Nous proposons ainsi d'appliquer la méthode Monte-Carlo pour les sous-systèmes élémentaires du réseau tels que le comparateur de phase (PD), le Filtre (F), l'oscillateur contrôlé en tension (VCO) comme représenté sur la Figure 2.2, page 36 afin de déterminer les plages de variation des paramètres incertains de chacun de ces blocs (voir les équations (2.2), page 46, (2.3) et (2.4), page 47, (2.11), page 53). Étant donné que tous les sous-systèmes sont identiques et donc que leurs incertitudes ont la même plage de variation, et comme la complexité des sous-systèmes élémentaires est faible, cette étape peut être accomplie de manière efficace.

Ensuite, l'application de la μ-analyse permet de tester si la stabilité et/ou la performance initialement garanties dans le cas nominal sont toujours conservées dans le cas incertain du modèle de l'ADPLL seule ou du modèle du réseau d'ADPLLs interconnectées pour toutes les valeurs de paramètres incertains possibles dans les plages déterminées par les simulations Monte-Carlo.

Nous supposons que la première étape de la méthode d'analyse proposée c'est-à-dire l'étape de simulations Monte-Carlo est réalisée. Dans cette étude, nous limitons

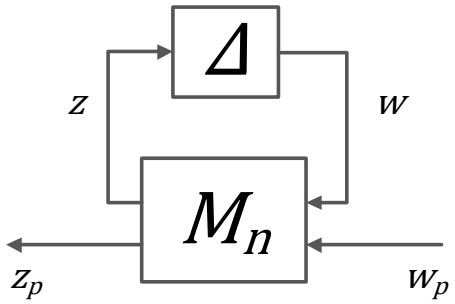


Figure 2.37: Structure $\Delta \star M_n$ standard de μ-analyse

notre attention à la méthode de la μ-analyse. Les résultats sont généralement pris de [SF08]. Tout d'abord, on donne une brève description générale de l'outil en énonçant les définitions et les théorèmes nécessaires à sa compréhension. Ensuite nous appliquerons à titre d'exemple la μ-analyse au cas de la commande d'une ADPLL. Les questions concernant l'application de la μ-analyse pour un réseau d'ADPLLs seront abordées dans le Chapitre 5.

2.6.1 Valeur singulière structurée et son calcul

Soit un modèle incertain mis en forme de deux blocs : *(i)* le bloc incertain Δ, englobant toutes les incertitudes possibles du modèle ; *(ii)* le bloc nominal M_n, décrivant le modèle nominal (voir la Figure 2.37). Cette forme qui couvre toute une famille de modèles décrivant le système réel, va être appelée par la suite une *structure standard* $(\Delta \star M_n)$ de μ-analyse. Ici, le bloc Δ est une matrice bloc diagonale, contenant sur sa diagonale les blocs incertains normalisés $\Delta^i : \bar{\sigma}(\Delta^i) < 1$. Les blocs incertains normalisés Δ^i forment une structure $\underline{\Delta}$ de l'incertitude globale

$$\underline{\Delta} = \left\{ \Delta \in \mathbb{C}^{k\times k} / \Delta = \begin{bmatrix} \Delta^1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \Delta^m \end{bmatrix} \text{ avec } k = \sum_{i=1}^{m} k_i \right\}, \tag{2.47}$$

et couvrent un large spectre d'incertitudes possibles du modèle dont les plus intéressantes sont :

– la dynamique non modélisée ou négligée qui est prise en compte par une incertitude dynamique complexe $\Delta^i = \Delta^i_d$, avec $\Delta^i_d \in \mathbb{C}^{k_i \times k_i}$;
– le paramètre incertain physique qui est pris en compte par une incertitude paramétrique réelle $\Delta^j = \delta_j I_{k_j}$, avec $\delta_j \in \mathbb{R}$.

Le bloc du modèle nominal M_n définit les valeurs numériques nominales des paramètres, la « taille » des incertitudes, l'interconnexion entre les signaux d'incertitude des entrées w et des sorties z ainsi qu'entre les signaux d'entrée w_p et de sortie z_p décrivant la performance. C'est une matrice dynamique et donc complexe dans le domaine fréquentiel.

Il est nécessaire par la suite, pour pouvoir appliquer le résultat principal de la μ-analyse, d'introduire la définition de la valeur singulière structurée.

Définition 2.8. Soit la matrice complexe $A \in \mathbb{C}^{k\times k}$. La ***valeur singulière structurée*** de A par rapport à la structure $\underline{\mathbf{\Delta}}$, notée $\mu_{\underline{\mathbf{\Delta}}}$, est définie par :

$$\mu_{\underline{\Delta}}(A) = \begin{cases} 0, & si\ \forall \Delta \in \underline{\Delta},\ det\,(I - \Delta A) \neq 0 \\ \dfrac{1}{\underset{\Delta\in\underline{\Delta}}{inf}\ (\bar{\sigma}(\Delta)/det(I-\Delta A)=0)}, & autrement. \end{cases} \tag{2.48}$$

La valeur singulière structurée $\mu_{\underline{\mathbf{\Delta}}}$ est souvent appelée indicateur μ et est une généralisation de la valeur singulière non structurée d'une matrice complexe. En effet, dans le cas où la structure $\underline{\mathbf{\Delta}}$ a la forme particulière d'une matrice complexe de blocs pleins : $\underline{\mathbf{\Delta}} = \left\{\Delta \in \mathbb{C}^{k\times k}\right\}$ (comme cela est défini dans la Définition 2.4, page 67), on a alors $\mu_{\underline{\mathbf{\Delta}}}(A) = \overline{\sigma}(A)$. Sinon $\forall \underline{\mathbf{\Delta}}$, on a la relation suivante :

$$\mu_{\underline{\mathbf{\Delta}}}(A) \leq \overline{\sigma}(A). \tag{2.49}$$

Pour le cas particulier d'une structure Δ et de la matrice A de dimensions 1×1, l'indicateur μ est le simple module du scalaire complexe A.

2.6.2 μ-analyse

Par la suite, nous allons procéder à l'analyse de la robustesse du système de la manière suivante : nous allons tout d'abord tester la stabilité du modèle incertain en appliquant la méthode de μ-analyse, ensuite nous montrerons comment aborder les questions de performance par le même outil d'analyse. Il s'agit de trouver une incertitude minimale, c'est-à-dire l'incertitude de la plus petite taille, pour laquelle la boucle de la Figure 2.37 est déstabilisée et/ou le niveau désiré de performance est dépassé.

Analyse de la stabilité robuste avec l'indicateur μ

Pour tester la stabilité, nous supprimons les entrées et sorties extérieures w_p/z_p décrivant la performance. Le schéma simplifié de la Figure 2.38 est ainsi obtenu avec un nouveau modèle nominal M_0.

Nous avons maintenant tous les éléments pour énoncer le résultat principal de la μ-analyse appelé le théorème de petit gain structuré.

Théorème 2.9 (Théorème du petit gain structuré)**.** *Soit la famille de systèmes bouclés* $(M_0,\ \Delta)$*, où* $M_0(s)$ *est une matrice de fonctions de transfert stable et où* $\Delta(j\omega) \in \underline{\mathbf{\Delta}}$ *est un bloc stable d'incertitudes. La famille de systèmes bouclés* $(M_0,\ \Delta)$ *est stable pour tout* Δ*, tel que* $\overline{\sigma}(\Delta(j\omega)) < \beta$ *si et seulement si*

$$\forall \omega \in [0,\ +\infty], \quad \mu_{\underline{\mathbf{\Delta}}}(M_0(j\omega)) \leq \frac{1}{\beta}. \tag{2.50}$$

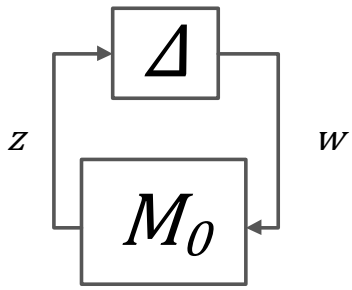


Figure 2.38: Structure ΔM standard de la μ-analyse pour la stabilité robuste

Démonstration. La démonstration de ce théorème peut être retrouvée dans le Théorème 8.6 de [SP05] □

On appelle μ-analyse l'analyse de la robustesse d'un système par la mise en œuvre de ce théorème.

Remarque 2.10. Le résultat de ce théorème est une extension du résultat du Théorème du petit gain qui, à son tour, est une extension au cas d'une incertitude dynamique du critère de stabilité de Nyquist multivariable. En effet, dans le cas des systèmes SISO, les conditions du Théorème 2.9 impliquent que le gain de la boucle ouverte $|M_0(j\omega)\Delta(j\omega)| < 1$ est donc par l'application du critère de Nyquist, la boucle fermée est toujours stable. D'un autre côté si la condition (2.50) n'est pas satisfaite, étant donné que M_0 et Δ sont stables, on peut toujours trouver une incertitude dont le gain est inférieur à β, mais pour lequel le point critique $-1+0j$ est entouré par le lieu de Nyquist de la boucle ouverte $M_0(j\omega)\Delta(j\omega)$.

Remarque 2.11. On peut rechercher la taille de la plus petite incertitude $\Delta \in \underline{\mathbf{\Delta}}$ pour laquelle le système $(M_0,\ \Delta)$ est déstabilisé. Ceci revient à définir le plus grand ensemble d'incertitudes Δ pour lequel la famille de systèmes $(M_0,\ \Delta)$ reste stable. Cette taille est donnée par

$$\frac{1}{\sup_{\omega\in[0,\,+\infty]} \mu_{\underline{\mathbf{\Delta}}}(M(j\omega))}. \tag{2.51}$$

En effet, d'après le théorème du petit gain structuré, rechercher le plus grand ensemble d'incertitudes Δ pour lequel la famille de systèmes (M, Δ) reste stable revient à rechercher le plus grand β tel que la condition (2.50) soit satisfaite. Dans ce cas-là, la famille est stable pour tous les Δ tels que $\|\Delta\|_\infty < \beta$. D'après (2.50), le plus grand β pour lequel la condition (2.50) est satisfaite est donné par la condition (2.51). De façon pratique, l'indicateur $\mu_{\underline{\mathbf{\Delta}}}$ sera calculé pour les matrices $M_0(j\omega)$ pour un ensemble fini de pulsations ω_i (incluant la pulsation 0) puis représenté graphiquement en fonction de la pulsation ω. L'inverse de la valeur maximale correspond à la taille de la plus grande incertitude que peut tolérer le système bouclé avant de devenir instable.

Comme nous pouvons le constater d'après le Théorème 2.9 que le calcul de l'indicateur μ pour une structure d'incertitude et une matrice complexe $M_0(j\omega)$ données

pour $\omega \in [0, +\infty]$ est la base de la μ-analyse expliquant le nom donné à cette méthode. Il y a deux difficultés principales liées à ce calcul.

La première difficulté est qu'il n'existe pas à ce jour d'algorithmes efficaces permettant de calculer la valeur singulière structurée d'une matrice quelconque. En général, le calcul « exact » de l'indicateur μ pour une matrice complexe donnée A est un problème NP-difficile [9] qui peut se formuler comme le problème d'optimisation suivant :

$$\max_{Q \in \mathcal{Q}} \rho_R(QA) = \mu_{\underline{\mathbf{\Delta}}}(A), \tag{2.52}$$

avec

$$\mathcal{Q} = \left\{ \Delta,\ \Delta \in \underline{\mathbf{\Delta}},\ \ \Delta^{i*}\Delta^i = I_{k_i},\quad i \in \{1, \ldots, q\},\ \ \delta_j \in]-1,\ 1[,\quad j \in \{1, \ldots, r\} \right\}.$$

Il est cependant possible de calculer une borne supérieure et une borne inférieure de l'indicateur μ par des algorithmes efficaces. Par exemple, la borne supérieure la plus couramment utilisée est obtenue par la résolution d'un problème d'optimisation convexe présentée par la suite.

Une autre difficulté est que selon le Théorème 2.9 l'indicateur $\mu_{\underline{\mathbf{\Delta}}}(M_0(j\omega))$ doit être calculé pour toutes les pulsations ω, soit un nombre infini de pulsations. En pratique, tout comme par exemple pour le tracé d'un diagramme de Bode, on se restreint à une gamme de pulsation intéressante (déterminée par exemple en regardant la localisation des pôles et des zéros du modèle nominal M_n) et à la pulsation $\omega = 0$. De plus, l'approche la plus couramment utilisée est d'effectuer un échantillonnage ou un maillage sur cet intervalle de pulsations afin de se ramener au calcul de l'indicateur en un nombre fini de points. Le danger de cette approche est de mal évaluer la valeur maximale de l'indicateur μ en ne choisissant pas correctement le maillage.

Calcul de la borne supérieure et inférieure de μ

La borne supérieure de $\mu(A)$ est obtenue par la résolution du problème d'optimisation suivant [SF08] :

$$\begin{aligned} \mu^{sup}_{\underline{\mathbf{\Delta}}}(A) \ = \ & \min_{\substack{D \in \mathcal{D} \\ G \in \mathcal{G} \\ \beta \in \mathbb{R}^+}} \beta \\ & \text{tel que} \quad A^*DA + (GA - A^*G) - \beta^2 D \leq 0, \end{aligned}$$

9. Une brève discussion sur la complexité de problèmes est reporté en l'annexe A de ce chapitre.

avec

$$\mathcal{D} = \left\{ D \in \mathbb{C}^{k\times k} \ / \ D = \begin{bmatrix} d_1 I_{k_1} & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & d_q I_{k_q} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & D_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & D_r \end{bmatrix} \text{ avec } \left\{ \begin{array}{lll} d_i \in \mathbb{R}, & d_i > 0, & i \in \{1,\ldots,q\} \\ D_j \in \mathbb{C}^{k_j \times k_j}, & D_j = D_j^* > 0, & j \in \{1,\ldots,r\} \end{array} \right\} \right\}.$$

et

$$\mathcal{G} = \left\{ G \in \mathbb{C}^{k\times k} \ / \ G = \begin{bmatrix} 0_{k_1} & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & 0_{k_q} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & G_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & G_r \end{bmatrix} \text{ avec } \left\{ G_j \in \mathbb{C}^{k_j \times k_j}, \ G_j = -G_j^* \ \ j \in \{1,\ldots,r\} \right\} \right\}.$$

où $q + r = m$ et les premiers blocs de D et G correspondent aux incertitudes dynamiques, les deuxièmes blocs de D et G correspondent aux incertitudes paramétriques. Ceci est un problème d'optimisation sous contraintes LMI qui peut être résolu de manière efficace (voire l'annexe A de ce chapitre, page 102).

Il y a deux propriétés importantes de la borne supérieure de μ :

Propriété 1 $\mu_{\underline{\mathbf{\Delta}}}(A) \leq \mu^{sup}_{\underline{\mathbf{\Delta}}}(A)$.

Propriété 2 Si $2r + q \leq 3$ alors $\mu_{\underline{\mathbf{\Delta}}}(A) = \mu^{sup}_{\underline{\mathbf{\Delta}}}(A)$ sinon $\mu_{\underline{\mathbf{\Delta}}}(A) < \mu^{sup}_{\underline{\mathbf{\Delta}}}(A)$.

Concernant la borne inférieure de μ, on calcule un maximum local du problème d'optimisation (2.52). Par suite, une borne inférieure $\mu^{inf}_{\underline{\mathbf{\Delta}}}(A)$ de $\mu_{\underline{\mathbf{\Delta}}}(A)$ est alors obtenue.

Remarque 2.12. On peut s'interroger sur l'utilité des bornes inférieure et supérieure. La borne supérieure permet de garantir la stabilité robuste de façon suffisante. En effet, d'après la Propriété 1 de la borne supérieure, on a :

$$\mu^{sup}_{\underline{\mathbf{\Delta}}}(A) \leq \frac{1}{\beta} \Rightarrow \mu_{\underline{\mathbf{\Delta}}}(A) \leq \frac{1}{\beta}.$$

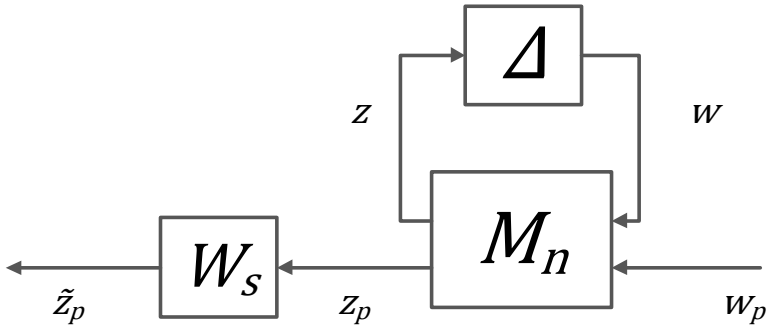


Figure 2.39: Structure ΔM standard de la μ-analyse pour la performance robuste

L'intérêt de la borne inférieure est qu'elle permet d'obtenir un encadrement de $\mu_{\underline{\mathbf{\Delta}}}(A) : \mu_{\underline{\mathbf{\Delta}}}^{inf}(A) \leq \mu_{\underline{\mathbf{\Delta}}}(A) \leq \mu_{\underline{\mathbf{\Delta}}}^{sup}(A)$. Si $\mu_{\underline{\mathbf{\Delta}}}^{sup}(A) - \mu_{\underline{\mathbf{\Delta}}}^{inf}(A)$ est faible alors $\mu_{\underline{\mathbf{\Delta}}}^{sup}(A) \simeq \mu_{\underline{\mathbf{\Delta}}}(A)$.

Analyse de la performance robuste avec l'indicateur μ

On considère dans cette section le problème de la robustesse en performance (plus général que la stabilité robuste), c'est-à-dire que l'on veut que l'une des fonctions de transfert (par exemple la fonction de sensibilité S_Δ [10]) vérifie une contrainte fréquentielle donnée par une pondération W_S ($\|W_S S_\Delta\|_\infty \leq 1$) pour la famille de systèmes interconnectés (voir la Figure 2.39).

Pour faire cela, on propose de réécrire le problème de la performance robuste sous la forme d'un problème équivalent de stabilité robuste. Par l'équivalence, il faut comprendre que la condition de stabilité robuste équivalente est satisfaite si et seulement si la condition de performance robuste est satisfaite et par conséquent la condition de stabilité robuste du problème d'origine (voir la Figure 2.39).

Soit Δ_p une fonction de transfert stable telle que $\|\Delta_p\|_\infty < 1$. Imaginons une famille de modèles représentée sur la Figure 2.40 où le bloc $\widetilde{\Delta}$ joue le rôle du bloc augmenté d'incertitudes et le système $\widetilde{M}_n$ est le système nominal augmenté.

D'après le théorème du petit gain structuré (voir le Théorème 2.9), en introduisant la structure d'incertitude suivante :

$$\widetilde{\underline{\mathbf{\Delta}}} = \left\{ \widetilde{\Delta} \ / \ \widetilde{\Delta} = \begin{bmatrix} \Delta_p & 0 \\ 0 & \Delta \end{bmatrix} \text{ avec } \Delta \in \underline{\mathbf{\Delta}} \text{ et } \Delta_p \in \mathbb{C}^{n_p \times n_p} \right\},$$

la famille de systèmes est stable pour toute fonction de transfert Δ stable et telle que $\|\Delta\|_\infty < 1$ avec $\Delta(j\omega) \in \underline{\mathbf{\Delta}}$ et pour toute fonction de transfert $\Delta_p(j\omega) \in \mathbb{C}^{n_p \times n_p}$, stable et telle que $\|\Delta_p\|_\infty < 1$ si et seulement si

$$\mu_{\widetilde{\underline{\mathbf{\Delta}}}}\left(\widetilde{M}_n(j\omega)\right) \leq 1.$$

10. L'indice Δ est utilisé ici pour mettre l'accent sur la dépendance de S en Δ.

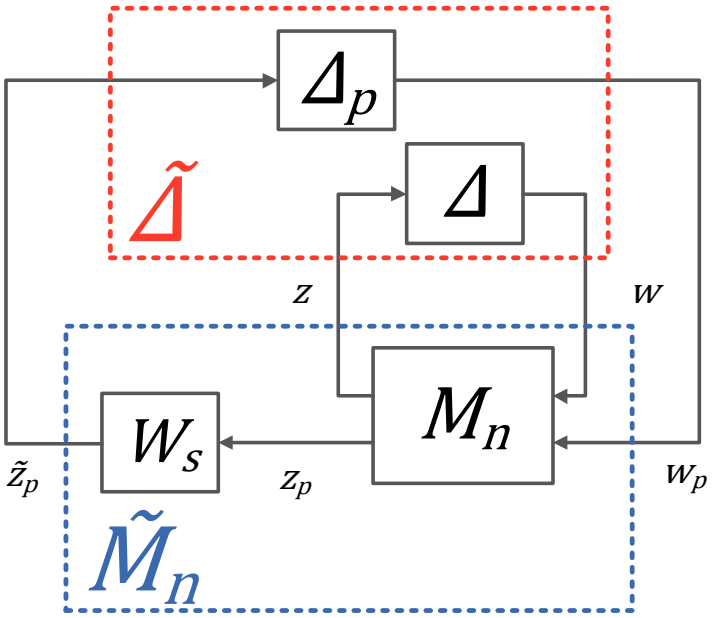


Figure 2.40: Structure ΔM équivalente de la performance robuste

D'après la discussion précédente, cette condition est équivalente à $\|W_s S_\Delta\|_\infty \leq 1$ pour toute fonction de transfert Δ stable et telle que $\|\Delta\|_\infty < 1$ avec $\Delta(j\omega) \in \underline{\mathbf{\Delta}}$ soit la condition de performance robuste.

2.6.3 Application de la μ-analyse au cas d'une ADPLL

Supposons que la synthèse de la loi de commande fournit un filtre PI avec deux paramètres : le gain de la branche proportionnelle K_P et le gain de la branche intégrale K_I (voir la version numérique du filtre sur la Figure 2.7, page 41). D'après la section 2.2, page 43, en pratique, les paramètres du filtre, les gains du *PD* et du *VCO* sont réalisés avec une certaine incertitude. La famille des modèles continus incertains est ainsi représentée sur la Figure 2.41 (Modèle d'analyse voir la page 33). On trouve quatre paramètres réels incertains que l'on peut regrouper par la suite dans un bloc incertain global Δ. La μ-analyse décrite dans la section précédente peut alors être appliquée.

Cependant, afin de réduire le nombre d'incertitudes à prendre en compte et donc la complexité du problème d'analyse, on propose de faire un changement de paramètres :

$$K = K_{PD} \cdot K_{VCO} \cdot \frac{1}{d} \cdot K_P = K_0 (1 + K_d \delta_K); \qquad a = \frac{K_I}{K_P} = a_0 (1 + a_d \delta_a), \quad (2.53)$$

avec

$$K_0 = K_{PD}^{nom} \cdot K_{VCO}^{nom} \cdot \frac{1}{d} \cdot K_P^{nom}; \qquad a_0 = \frac{K_I^{nom}}{K_P^{nom}}; \quad (2.54)$$

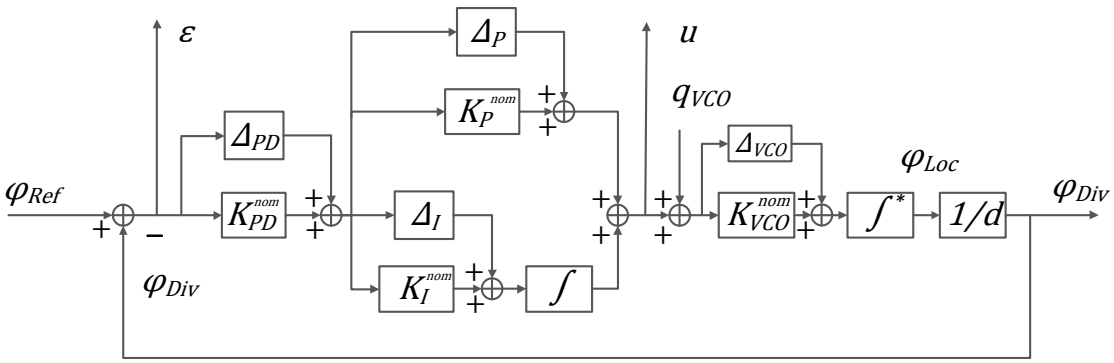


Figure 2.41: Famille des modèles continus incertains dans le domaine de phase de l'ADPLL

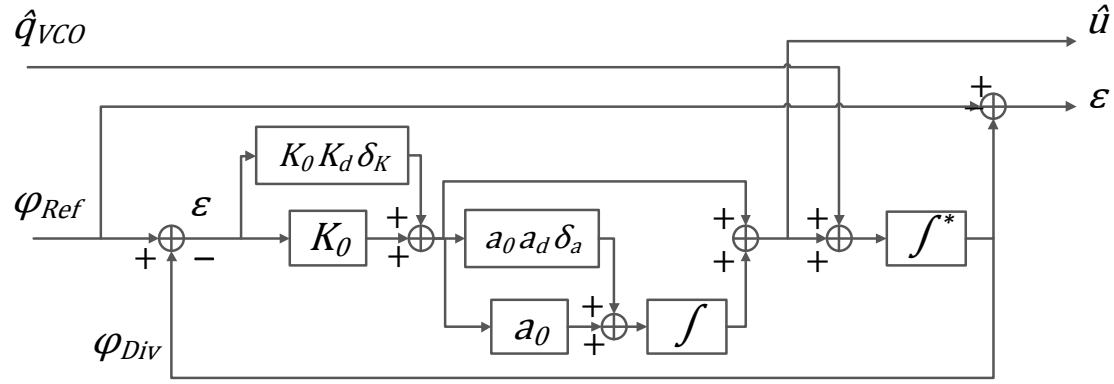


Figure 2.42: Schéma simplifié de la famille des modèles continus incertains dans le domaine de phase de l'ADPLL

$$
\begin{aligned}
K_d &= \left(1+\frac{K_{PD}^{gab}}{K_{PD}^{nom}}\right)\left(1+\frac{K_{VCO}^{gab}}{K_{VCO}^{nom}}\right)\left(1+\frac{K_P^{gab}}{K_P^{nom}}\right)-1;\\
a_d &= \frac{\left(1+\dfrac{K_I^{gab}}{K_I^{nom}}\right)}{\left(1+\dfrac{K_P^{gab}}{K_P^{nom}}\right)}-1,
\end{aligned}
\tag{2.55}
$$

et $\delta_K, \delta_a \in \mathbb{R} : |\delta_K| < 1,\ |\delta_a| < 1$.

Avec ces nouvelles notations, on obtient deux paramètres réels incertains et donc le schéma équivalent de la Figure 2.42.

En utilisant les équations (2.53)–(2.55), le schéma de la Figure 2.42 est transformé en un schéma équivalent de la forme $\Delta \star M_n$ standard de la μ-analyse, voir la Figure 2.43.

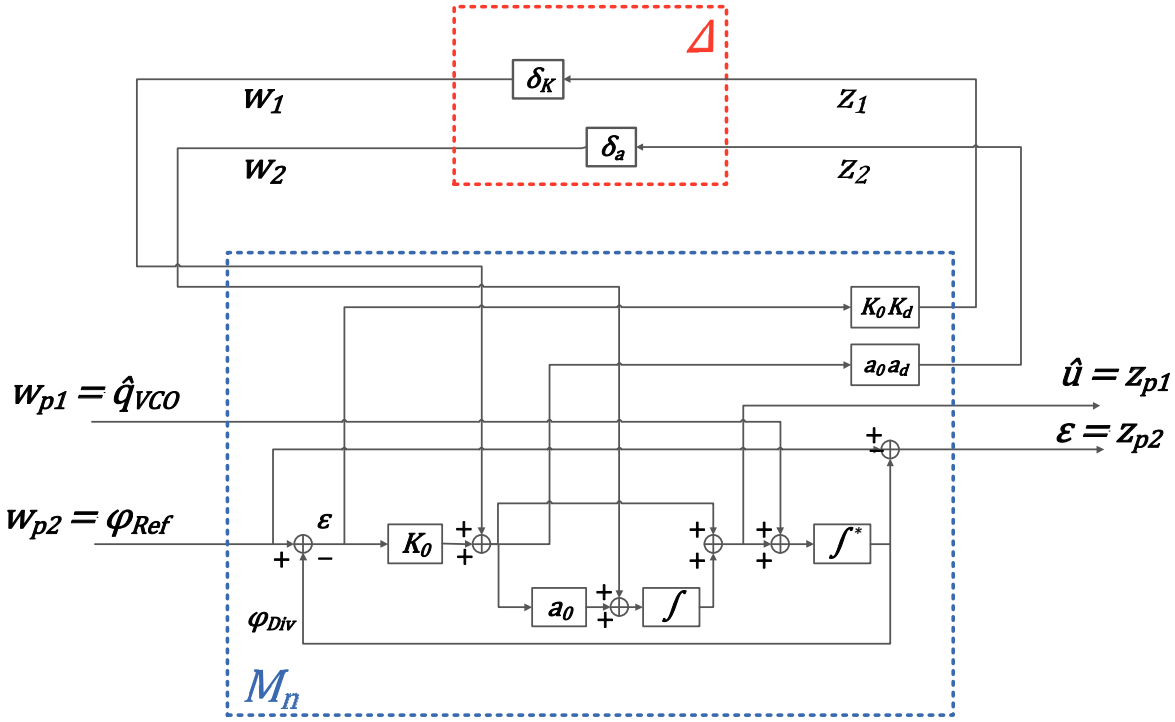


Figure 2.43: Schéma $\Delta \star M_n$ standard de la μ-analyse dans le domaine de phase de l'ADPLL

On applique maintenant la procédure de μ-analyse permettant de déterminer la taille maximale des incertitudes c'est-à-dire les valeurs de K_0K_d et a_0a_d pour lesquelles le système reste stable et/ou a un niveau de performance désiré. Il est intéressant de noter que les gains du *VCO* et du *PD* ont un certain niveau d'incertitude fixé par leur mode de réalisation et l'influence de l'environnement extérieur (voir la section 2.2, page 43). Aussi, on peut fixer une précision sur les paramètres du correcteur (K_P et K_I par exemple pour un correcteur PI) et donc leur incertitude en choisissant un nombre de bits dédiés à leur codage. Il est ainsi possible, d'après le calcul de μ-analyse, de trouver la taille maximale d'incertitudes possibles sur les paramètres K et a garantissant la stabilité et/ou la performance, et de déduire ensuite une précision minimale à respecter pour les paramètres du filtre correcteur (K_P^{gab} et K_I^{gab} pour l'exemple du filtre PI). La dernière partie de cette section montre un exemple numérique d'une telle approche.

2.6.4 Exemple numérique de la μ-analyse

Pour appliquer la μ-analyse, on transforme le Modèle d'analyse de l'ADPLL sous la forme décrite par la Figure 2.43. Étant donné que le respect des performances implique nécessairement la stabilité du modèle, on applique la méthode de μ-analyse pour tester directement le niveau de performance désiré. Étant entendu que la fonction de sensibilité S est la fonction de transfert la plus importante pour l'évaluation de performance (voir les Tableaux 2.2, page 62, 2.3, page 66 et 2.4, page 79 ainsi que la Figure 2.35) on considère le schéma équivalent de la Figure 2.40. En calculant

l'indicateur μ fréquence par fréquence comme expliqué au début de la section 2.6.2 pour différents rapports entre les gabarits des incertitudes sur les paramètres K et a, on trouve ainsi leurs valeurs maximales :

$$\begin{array}{c} K \in [K_0\,(1-K_d)\,;\,K_0\,(1+K_d)] \\ a \in [a_0\,(1-a_d)\,;\,a_0\,(1+a_d)] \end{array} \text{ avec } \begin{array}{l} K_d = 0.8; \\ a_d = 0.5; \end{array}$$

En considérant les valeurs numériques d'incertitudes sur le gain du *VCO* et du *PD* (voir le Tableau 2.1, page 55 de la Configuration FPGA) et en utilisant les transformations (2.53)–(2.55) on obtient les valeurs suivantes d'incertitude maximale sur les paramètres du correcteur PI numérique respectant la stabilité et le niveau de performance désiré :

$$\begin{array}{l} K_P^{*gab} = \pm 0.66 \cdot 10^{-1} \text{ soit } \pm 15.4\%; \\ K_I^{*gab} = \pm 2.8 \cdot 10^{-3} \text{ soit } \pm 73\%. \end{array} \tag{2.56}$$

Les tracés des bornes inférieures et supérieures correspondantes sont représentés sur la Figure 2.44.

La Figure 2.45 confirme que la contrainte assurant le niveau de performance désiré reste respecté en dépit des incertitudes sur les paramètres réels pour lesquels plusieurs valeurs ont été choisies dans l'ensemble des valeurs définies dans (2.56) et le Tableau 2.1, page 55.

2.7 Simulations

Pour confirmer le résultat obtenu, des simulations temporelles ont été effectuées sur les différents modèles de l'ADPLL. La Figure 2.46 montre l'évolution des signaux d'erreur de suivi ε et de la commande u au cours du temps issues de simulations du Modèle de synthèse (voir la Figure 2.25, page 60) sans bruits extérieurs c'est-à-dire un modèle simple LTI de l'ADPLL (voir la page 33 pour plus de détails) sans signaux b, b_q, ψ. Elle est partagé en trois parties :

- durant la partie I, seul le signal échelon unitaire est appliqué à l'entrée de l'ADPLL (φ_{Ref}), ce qui permet d'évaluer le temps de montée et donc le temps de réponse [11] ;
- durant la partie II, la perturbation q_{VCO} sous forme d'un échelon unitaire est appliquée à l'entrée du procédé pour confirmer son rejet ;
- durant la partie III, une rampe unitaire est appliquée à l'entrée de l'ADPLL (φ_{Ref}) pour confirmer la synchronisation de l'ADPLL avec une source périodique extérieure.

Les mêmes simulations sur le Modèle de synthèse mais en présence de bruits b, b_q, ψ sont représentées sur la Figure 2.47.

Étant donné que le bruit de phase du *VCO* ϕ_b est la source de bruit la plus importante dans l'ADPLL, les simulations du Modèle de synthèse montrant sa mise en

11. Le temps de montée est, par ailleurs, inférieur au temps de montée indicatif de la fonction de transfert canonique de deuxième ordre $T = \frac{\omega_0}{s^2+2\zeta\omega_0 s+\omega_0^2}$ $(7.1 \cdot 10^{-3}\, sec)$.

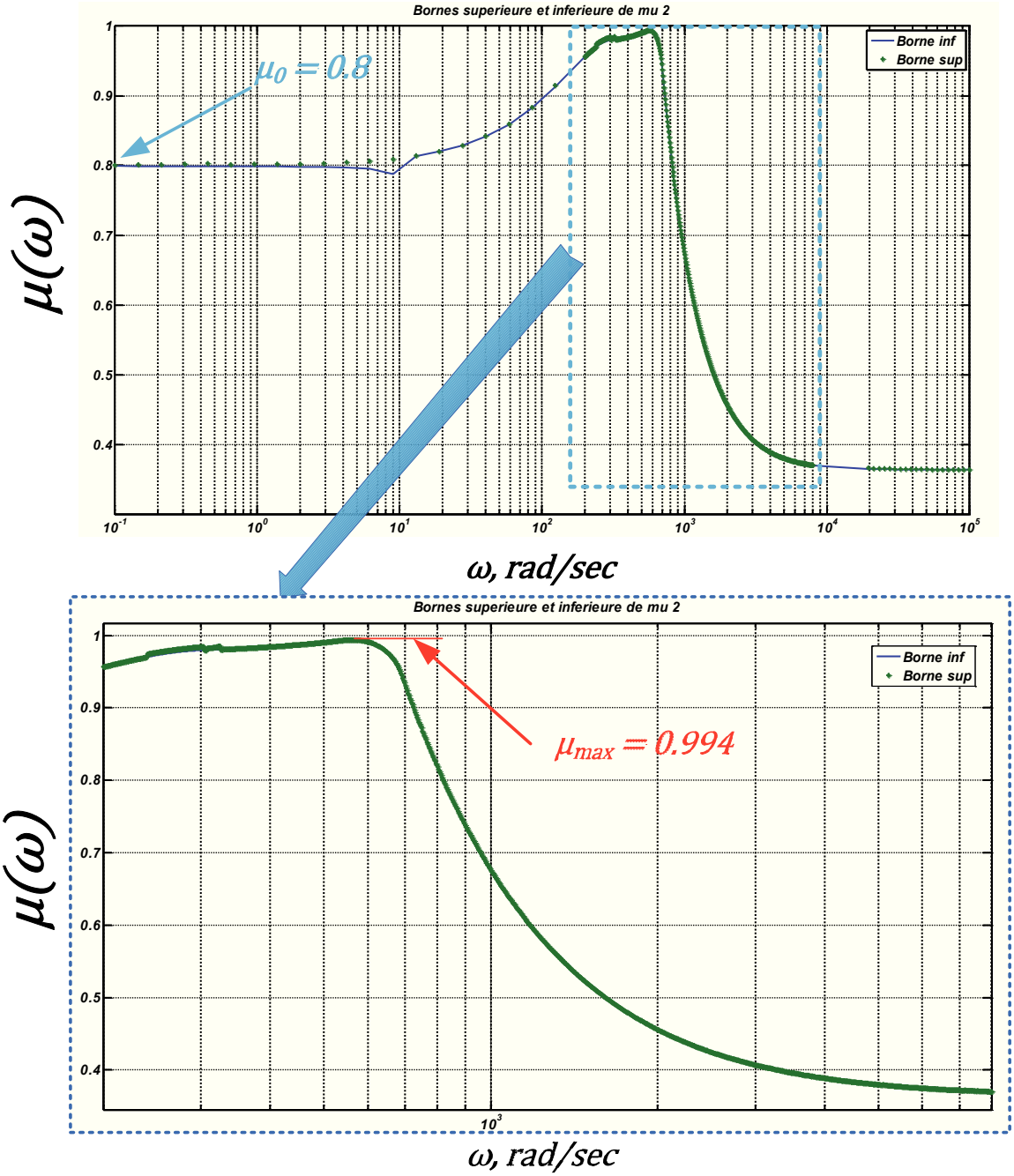


Figure 2.44: Les bornes inférieure (ligne bleue) et supérieure (points verts) de l'indicateur μ de l'exemple numérique considéré

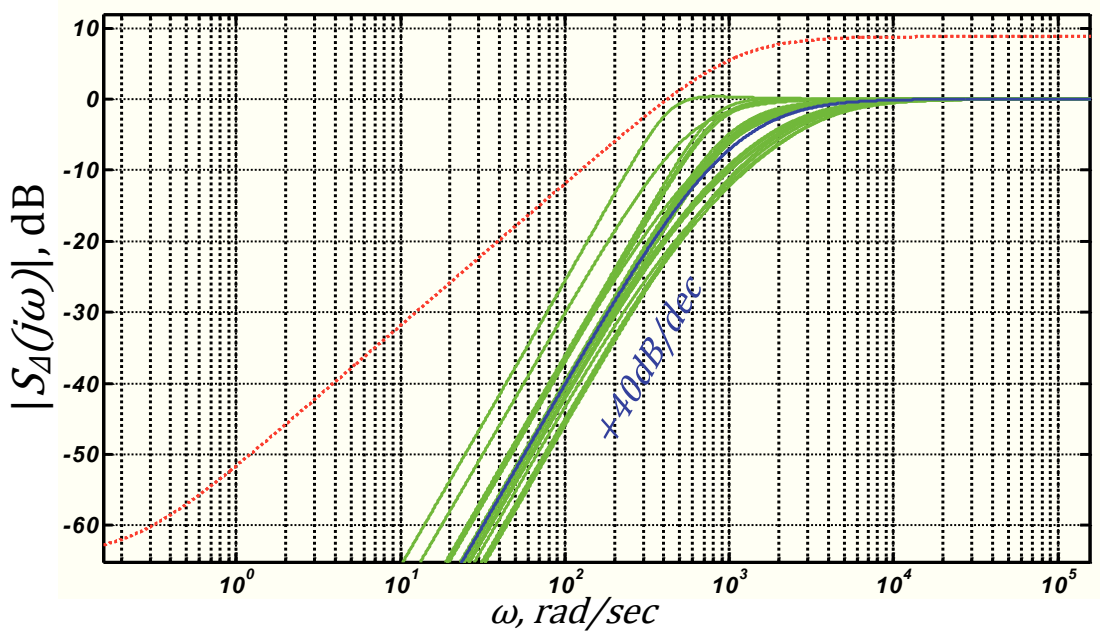


Figure 2.45: La fonction de sensibilité S_Δ incertaine pour différentes valeurs d'incertitudes choisies aléatoirement (en vert) ainsi que la fonction de sensibilité S nominale (en bleue) et la contrainte fréquentielle (en rouge pointillé)

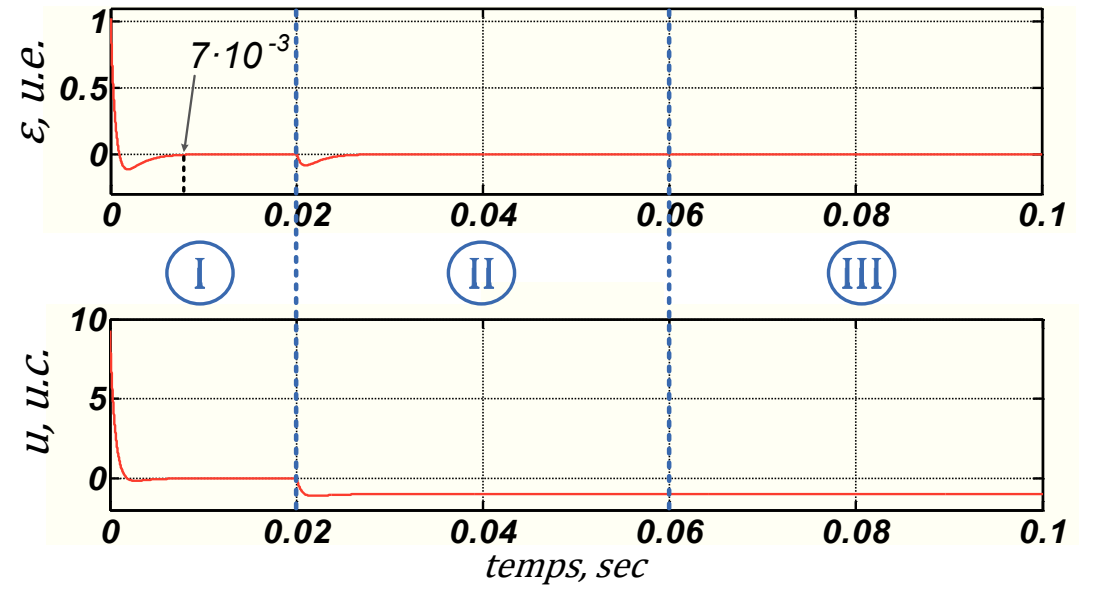


Figure 2.46: Simulations temporelles du Modèle de synthèse de l'ADPLL sans le bruit extérieur, l'évolution de l'erreur de suivi ε (première sous-figure) et de la commande u (deuxième sous-figure) au cours du temps.

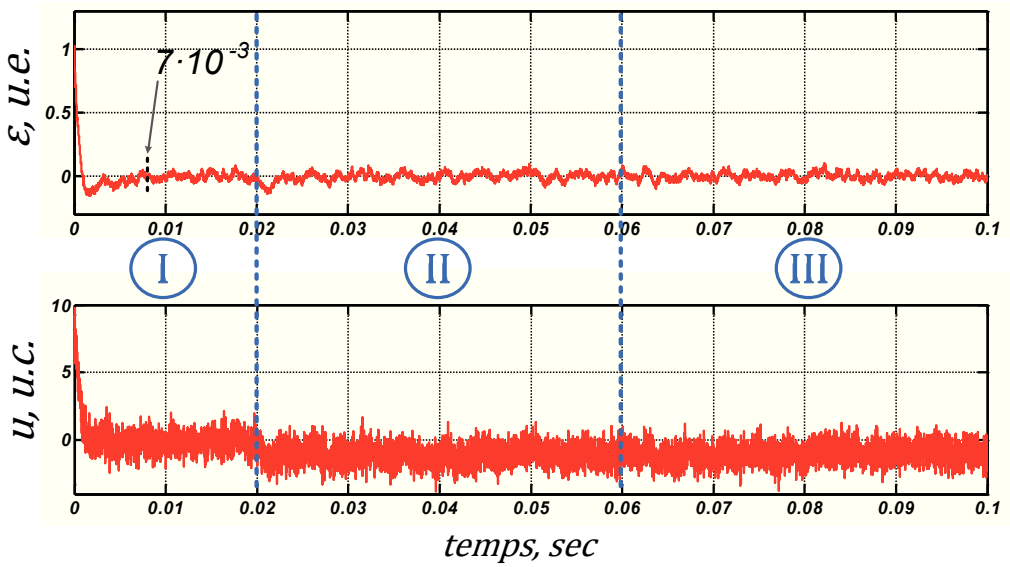


Figure 2.47: Simulations temporelles du Modèle de synthèse de l'ADPLL avec le bruit extérieur, l'évolution de l'erreur de suivi ε (première sous-figure) et de la commande u (deuxième sous-figure) au cours du temps.

forme ont été réalisées avec un seul signal de bruit ϕ_b appliqué à la sortie du *VCO*. La Figure 2.48 illustre les résultats obtenus. La courbe rouge correspond à la densité spectrale de puissance du bruit de phase ϕ_b. Comme montré dans la section 2.2, page 43, son profil en BF représente une pente de $-30\,dB/dec$. Pour la réalisation d'un tel profil un filtre générateur a été utilisé. La réponse fréquentielle du filtre grâce à une technique de combinaison de filtres linéaires ou différentiation fractionnelle [LMS+04, LML+06] possède une pente de $-30\,dB/dec$ en BF. La courbe bleue sur la Figure 2.48 correspond à la densité spectrale du signal d'erreur de suivi ε. La mise en forme du bruit est parfaitement conforme au module de la fonction de sensibilité S reliant ces deux signaux (voir la Figure 2.35).

Pour confirmer que les hypothèses prises durant la modélisation sont valables autour du point de fonctionnement et pour valider l'efficacité de la loi de commande obtenue sur le procédé réel des simulations sur le Modèle de simulation incluant les non-linéarités, l'effet de quantification, des incertitudes (voir la section 2.2) ont été réalisées. Les résultats de simulations sont représentées sur les Figures 2.49 et 2.50. La Figure 2.49 présente l'évolution de la fréquence instantanée f de l'ADPLL et de l'erreur de suivi ε (courbes bleues) au cours du temps pendant deux étapes : la première étape illustre la synchronisation du signal à la sortie de l'ADPLL et d'un signal de référence à son entrée dans le cas où leurs fréquences instantanées initiales ne sont pas identiques ; la deuxième étape illustre le rejet de perturbations à l'entrée du *VCO* sous forme d'un échelon dont l'amplitude correspond à la différence de fréquences initiales. La courbe verte correspond à la fréquence du signal de référence Ref appliqué à l'entrée de l'ADPLL. Les fluctuations en régime permanent sont dûes à la nature « bang-bang » du comparateur de phase utilisé (voir la sous-section 2.1.1, page 36) et aux différentes sources de bruit (bruit de phase du *VCO*, les effets de quantification, *etc.*).

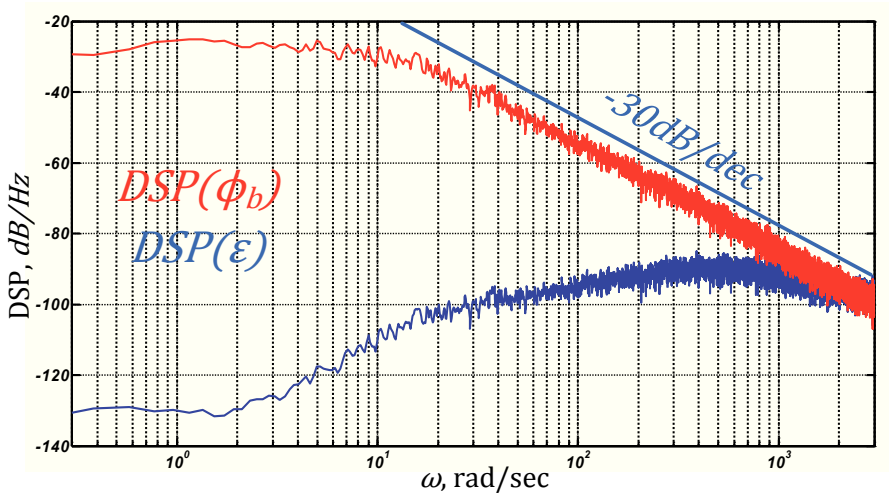


Figure 2.48: La mise en forme du bruit de phase ϕ_b par la fonction de sensibilité S

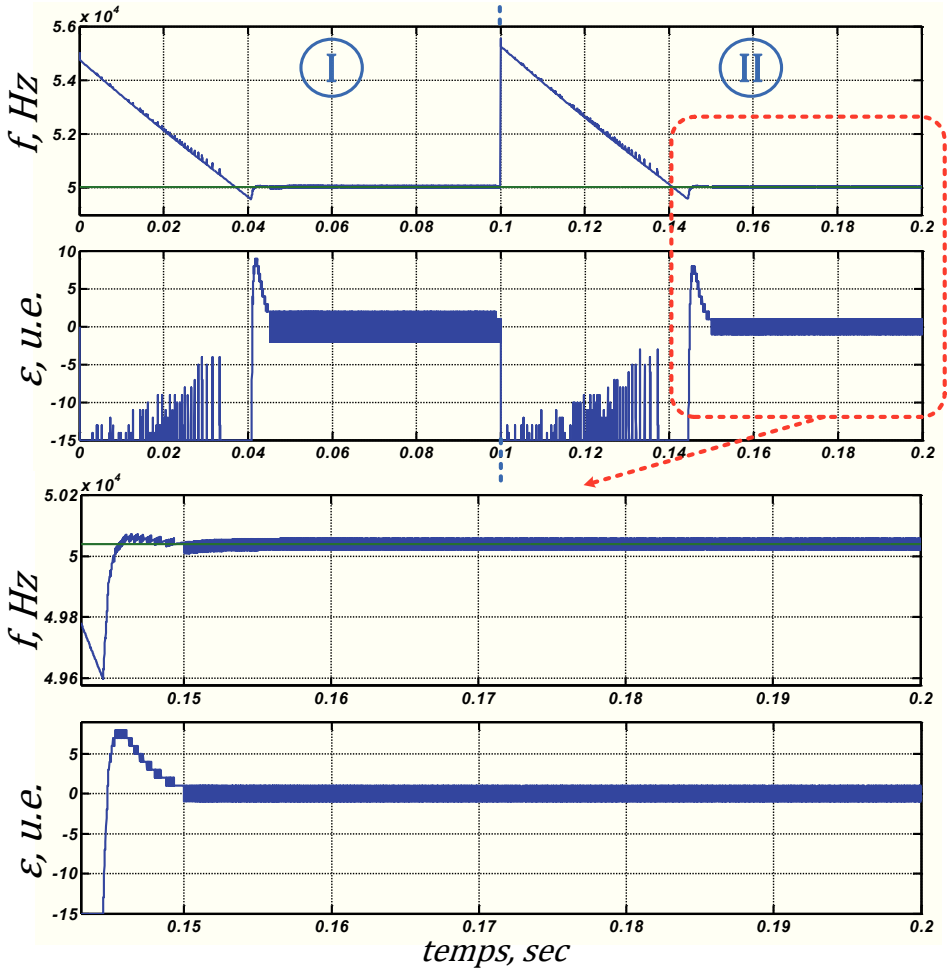


Figure 2.49: Simulations temporelles du Modèle de simulation de l'ADPLL, l'évolution de la fréquence instantané de l'ADPLL f et de l'erreur de suivi ε au cours du temps.

La Figure 2.50 présente l'évolution au cours du temps des signaux d'horloge *(i)* appliqué à l'entrée de l'ADPLL Ref et *(ii)* générés par l'ADPLL Div [12]. On remarque que les signaux sont synchronisés (avec une certaine erreur) en phase et en fréquence après une période transitoire : la loi de commande est donc efficace pour le Modèle de simulation le plus proche du procédé réel.

2.8 D'une ADPLL locale au réseau d'ADPLLs global

Dans les sections précédentes, nous avons synthétisé une loi de commande pour une seule ADPLL selon l'Algorithme 2.1, page 76. Ensuite, nous avons appliqué la μ-analyse pour étudier la robustesse vis-à-vis de différentes incertitudes paramétriques possibles. Étant donné que l'objectif final de ce travail de thèse est la synthèse d'un réseau d'ADPLLs, par la suite, nous allons interconnecter les ADPLLs identiques ainsi obtenues selon une topologie quelconque et analyser le comportement nominal de tel réseau. Aussi, nous allons voir que non seulement la performance garantie pour une seule ADPLL ne l'est plus forcément pour un réseau d'ADPLLs interconnectées, mais aussi que la stabilité, pour certaines topologies d'interconnexion, est remise en question.

2.8.1 Interconnexion en réseau

Il est intéressant d'examiner le comportement du réseau entier composé d'un modèle identique d'ADPLL obtenue lors de la synthèse locale. Par l'analogie avec la Figure 1.14, page 26, on obtient sur la Figure 2.51 un modèle linéaire dans le domaine de phase d'une d'ADPLLs à entrées multiples. Les entrées supplémentaires servent à réaliser les interconnections entre les ADPLLs faisant partie du réseau. On se focalise uniquement dans cette sous-section sur un modèle nominal linéaire invariant d'ADPLL. En fonction de la topologie du réseau et de la position des ADPLLs au sein de ce réseau, le nombre d'entrées m_i peut être différent. Le système de la Figure 2.51 est équivalent au système représenté sur la Figure 2.52 : c'est un nœud de l'ADPLL où on voit apparaître le même sous-système et les mêmes signaux que dans le cas d'une ADPLL élémentaire (voir la Figure 2.25, page 60).

Imaginons maintenant que les ADPLLs soient interconnectées selon les deux topologies *(a)* et *(b)* représentées graphiquement sur la Figure 2.53. Sur la Figure 2.52, chaque rond bleu désigne un nœud du réseau de ADPLLs, et les flèches modélisent les échanges d'information de phase entre les nœuds. Les deux réseaux ont une entrée extérieure de référence connectée sur le premier nœud. On suppose également que chaque nœud du réseau d'ADPLLs est synthétisé selon la méthode H_∞ présentée précédemment.

12. Il s'agit bien du signal à fréquence divisée Div issus du diviseur de fréquence, le signal généré par le *VCO* du haute fréquence Loc est cependant synchronisé avec le signal à fréquence divisée Div.

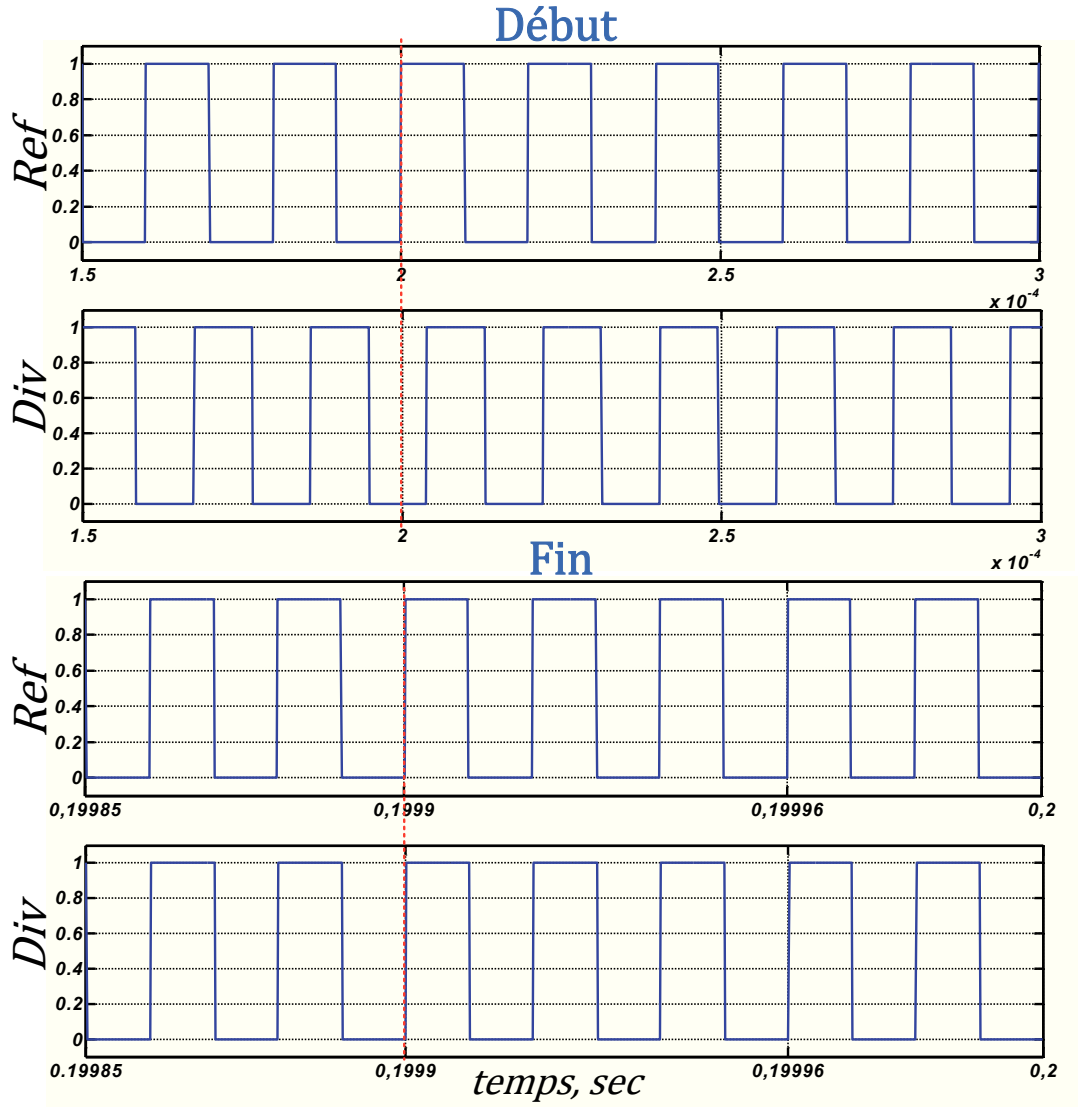


Figure 2.50: Simulations temporelles du Modèle de simulation de l'ADPLL, l'évolution au cours du temps de signaux d'horloge *(i)* appliqué à l'entrée de l'ADPLL *Ref* et *(ii)* généré par l'ADPLL *Div*.

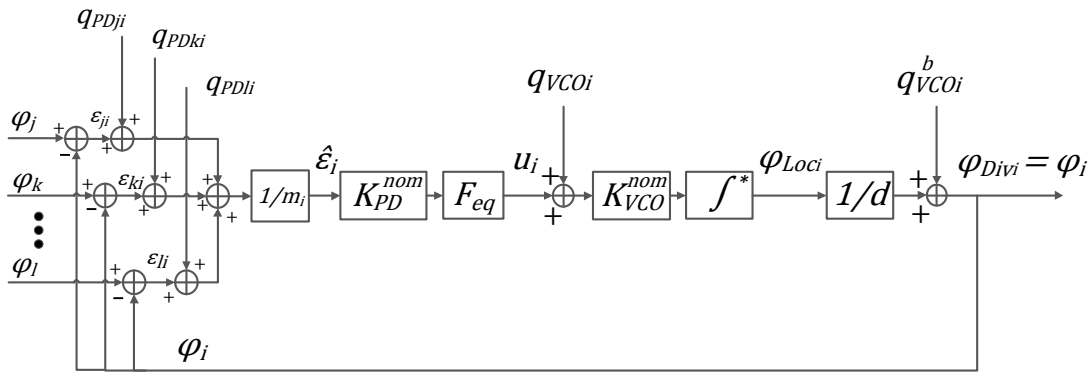


Figure 2.51: Modèle linéaire continue d'une PLL à entrées multiples

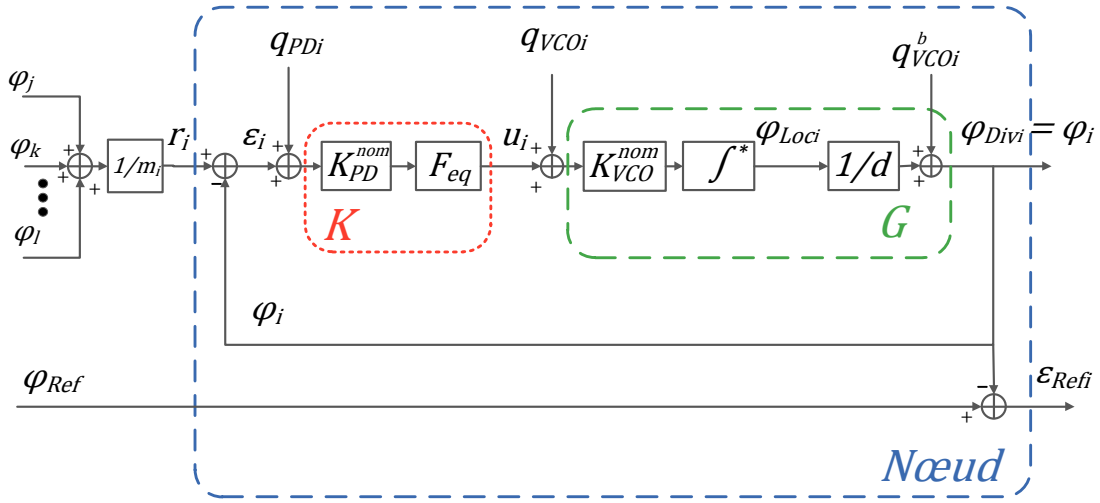


Figure 2.52: Modèle équivalent transformé du modèle de la Figure 2.51

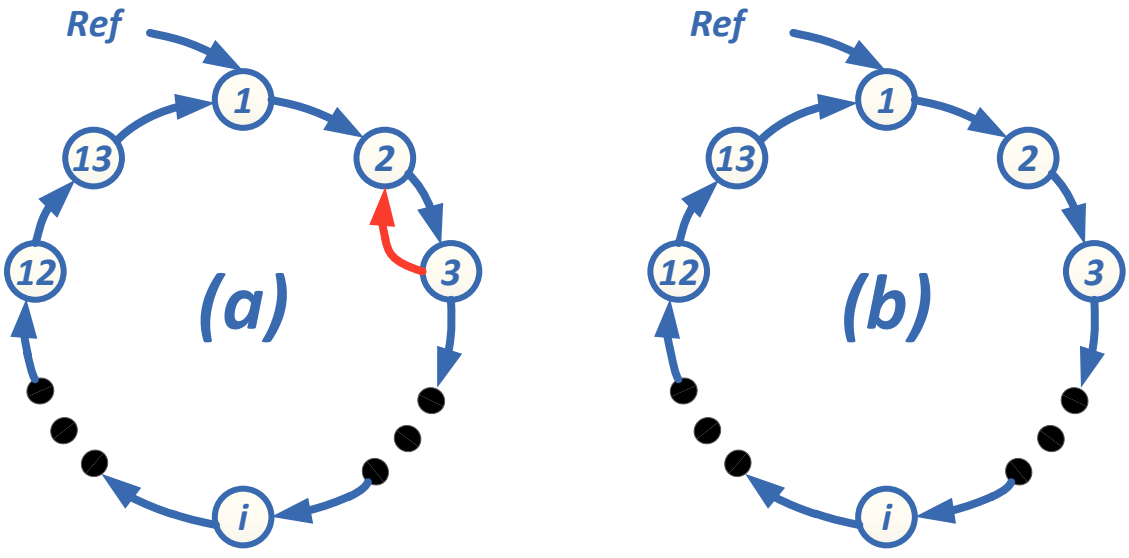


Figure 2.53: Deux topologies d'interconnexion considérées

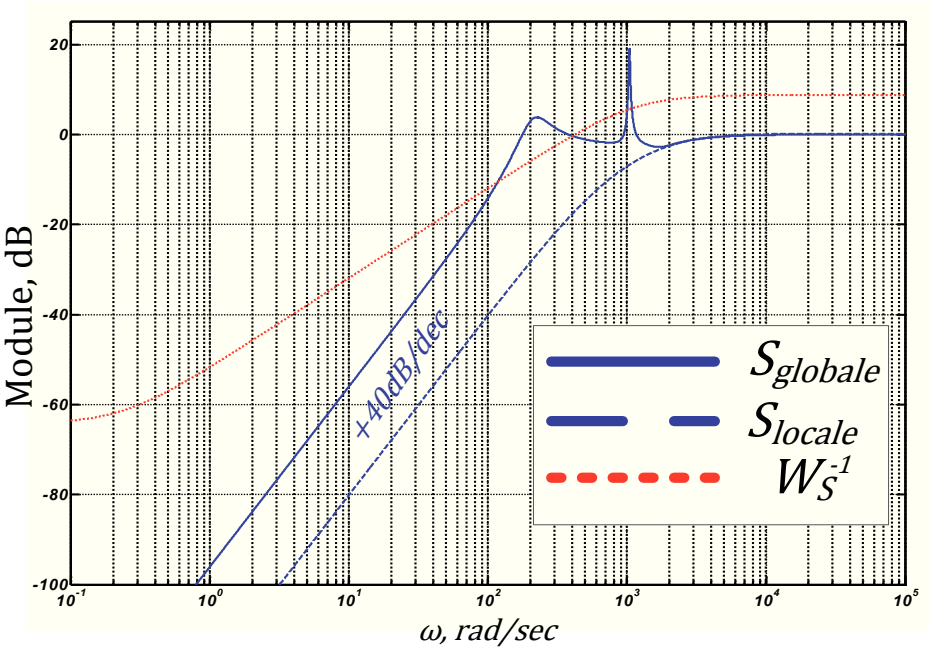


Figure 2.54: Comparaison des fonctions de sensibilité dans le cas d'une PLL élémentaire et du réseau d'ADPLLs décrit selon la topologie (a)

Pour tester la stabilité interne du réseau, on construit tout d'abord une représentation d'états du modèle du système global :

$$\begin{array}{rcl} \dot{x} & = & A^a_{glob}x \quad + \quad B^a_{glob}u \\ y & = & C^a_{glob}x \quad + \quad D^a_{glob}u \end{array} .$$

Ensuite, on vérifie si la partie réelle de toutes les valeurs propres de la matrice d'évolution d'états obtenue A^a_{glob} est négative. Ceci est équivalent à vérifier si la partie réelle de la valeur propre de la matrice d'états présentant la plus grande partie réelle est négative. Pour, la topologie (a), la valeur propre de la matrice d'états présentant la plus grande partie réelle est égale à :

$$\lambda_{max}\left(A^a_{glob}\right) = \max_i Re\left(\lambda_i\left(A^a_{glob}\right)\right) = -8.02. \tag{2.57}$$

Le réseau de topologie (a) est donc bien stable. En revanche en comparant la fonction de transfert entre l'entrée de référence et l'erreur de suivi de cette référence par le premier nœud $S_{globale}$ avec la fonction de sensibilité obtenue pour une PLL unique élémentaire (voir la Figure 2.54), on remarque que la performance est dégradée considérablement. En effet, la contrainte n'est plus respectée, la valeur de la pulsation de coupure a diminué et cela conduit à une augmentation du temps de réponse du système. Par ailleurs, on voit apparaître un pic important d'environ $19dB$ qui contribue à diminuer la marge du module et donc à s'approcher de la limite de stabilité du réseau.

Maintenant, si l'on surprime le lien entre le nœud 3 et le nœud 2 (voir la flèche rouge de la topologie (a) sur la partie gauche de la Figure 2.53) on obtient la topologie (b)

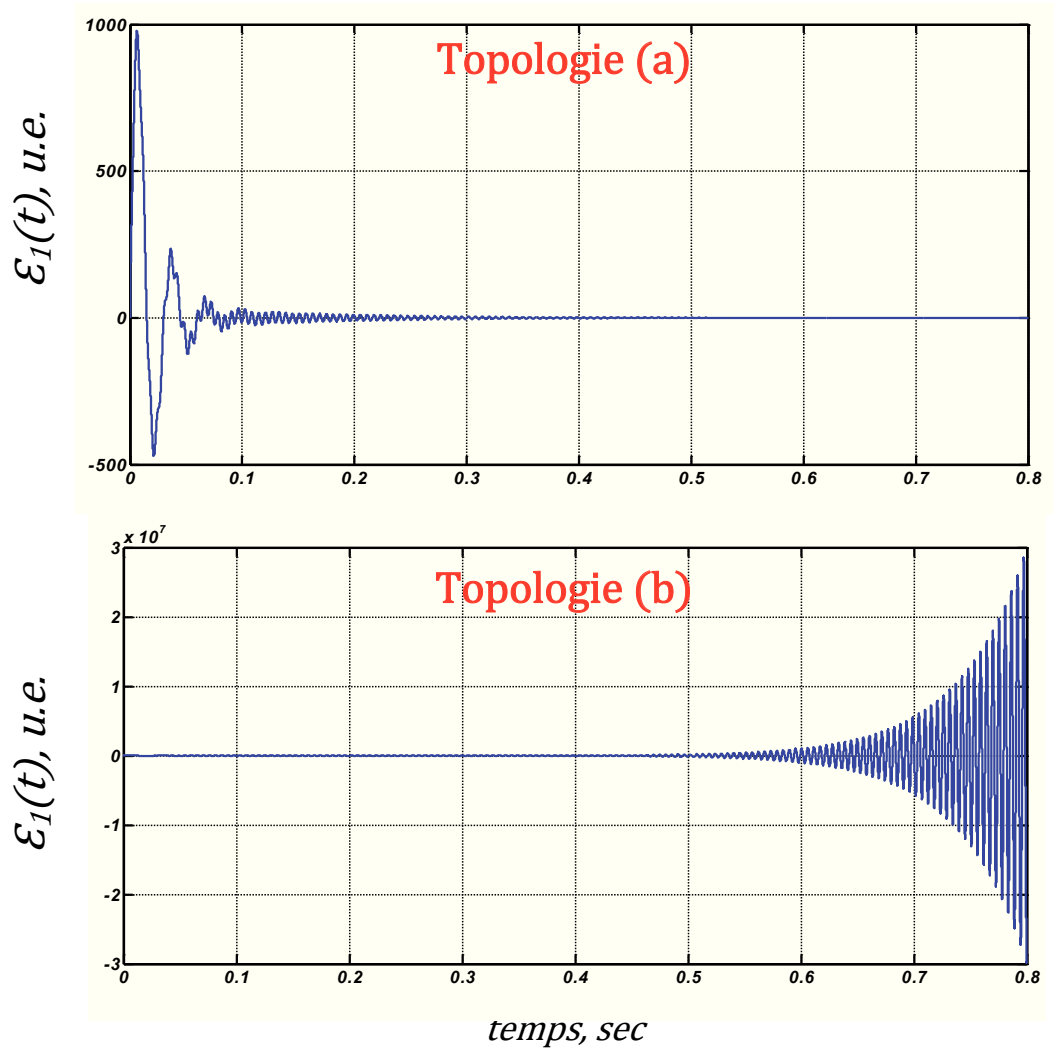


Figure 2.55: Simulations temporelles des deux topologies de réseau décrites sur la Figure 2.53 ; Évolution de l'erreur l'erreur de suivi $\varepsilon_1 = \varphi_{Ref} - \varphi_1$ pour le premier nœud du réseau

représentée sur la Figure 2.53 avec :

$$\lambda_{max}\left(A^b_{glob}\right) = \max_i Re\left(\lambda_i\left(A^b_{glob}\right)\right) = +16.7. \tag{2.58}$$

Le réseau devient donc instable bien que chaque ADPLL élémentaire du réseau soit stable avec de bonnes marges de stabilité. Les simulations temporelles des deux topologies (voir l'évolution au cours du temps de l'erreur de suivit du premier nœud $\varepsilon_1 = \varphi_{Ref} - \varphi_1$ de chaque topologie sur la Figure 2.55) confirment ces résultats.

En conclusion, on ne peut pas négliger l'influence des interconnexions sur la dynamique globale du système dans un réseau d'ADPLLs même dans le cas simple de systèmes LTIs. Pour assurer certaines propriétés globales du réseau, il est donc nécessaire de les prendre en compte durant la procédure de synthèse.

Conclusion

Les exemples numériques présentés à la fin de ce chapitre montrent clairement, que l'approche logique et simple évoquée à la fin du chapitre 1 de synthèse locale ne convient pas toujours. Il est plus intéressant de pouvoir garantir certaines propriété globales du réseau d'ADPLLs lors de sa conception. Nous ne pouvons pas, dans ce cas-là, nous concentrer seulement sur la synthèse d'une ADPLL locale sans prendre en compte les propriétés à assurer pour un réseau global. D'un autre côté, les conditions assurant les propriétés globales du réseau complet peuvent être complexes et le problème de synthèse de la loi de commande les respectant est extrêmement difficile à résoudre. Dans ce cas-là, on peut perdre l'avantage de la simplicité de la synthèse de loi de commande locale. Or d'un point de vue pratique, nous sommes intéressés par une synthèse de loi de commande ainsi que par des algorithmes de réalisation simples et efficaces. C'est pour ces raisons que nous nous tournons vers la première idée importante retenue lors de la discussion sur l'approche locale simpliste proposée en Microélectronique de conception d'un réseau d'ADPLLs interconnectées (voir la page 26). Il s'agit de l'idée de la facilité de synthèse locale de la loi de commande. En effet, comme nous le verrons dans les chapitres qui suivent, il est possible de garder la même démarche simple de synthèse locale mais avec des conditions locales supplémentaires garantissant les propriétés globales désirées. Notre objectif est alors de transformer les conditions globales issues de propriétés globales désirées (stabilité, performance) en des conditions locales qui peuvent être satisfaites par la suite par une synthèse de la loi de commande locale simple bien maîtrisée en général. Néanmoins, ces questions sortent du cadre d'étude de la Microélectronique standard et on est amené à faire appel aux outils et méthodes de l'Automatique classique et moderne. Cependant avant de résoudre le problème répondant à l'objectif posé dans le chapitre 4 on se propose dans le chapitre 3 d'examiner les méthodes préexistantes de l'Automatique qui peuvent permettre d'aborder ce problème.

2.9 Annexe A : Optimisation LMI

Après une définition formelle du problème d'optimisation, on présente dans une brève discussion les problèmes d'optimisation convexe ou quasi-convexe ainsi que les algorithmes pour les résoudre efficacement. Enfin, nous définirons trois problèmes d'optimisation sous contrainte LMI généralement considérés dans la littérature et utilisés dans ce document.

Définition 2.13 (Problème d'optimisation)**.** *Un problème d'optimisation est défini de la manière suivante.*

Soient n_ξ un entier naturel non nul, $\mathbb{D} \subseteq \mathbb{R}^{n_\xi}$ un ensemble non vide et

$$\begin{array}{rccc} f & : & \mathbb{R}^{n_\xi} & \rightarrow & \mathbb{R} \\ & & \xi & \mapsto & f(\xi) \end{array}$$

Alors un problème d'optimisation s'écrit :

$$\min_{\xi \in \mathbb{D}} f(\xi)$$

où
- *$\mathbb{D}$ est appelé l'ensemble des contraintes ;*
- *$\xi = \begin{bmatrix} \xi_1 & \cdots & \xi_{n_\xi} \end{bmatrix}^T$ est appelé le vecteur des variables de décision ;*
- *f est appelée la fonction de coût.*

On dit que :
- *$\xi_{opt} \in \mathbb{D}$ est un optimum local s'il existe un voisinage ouvert $\mathbb{U}$ de ξ_{opt} tel que, pour tout $\xi \in \mathbb{U} \cap \mathbb{D}$, on a $f(\xi_{opt}) \leq f(\xi)$;*
- *$\xi_{opt} \in \mathbb{D}$ est un optimum global si, pour tout $\xi \in \mathbb{D}$, on a $f(\xi_{opt}) \leq f(\xi)$.*

Résoudre le problème d'optimisation signifie trouver un optimum global (et la valeur de la fonction de coût f en ce point).

On distingue différentes classes de problème d'optimisation en fonction des propriétés de f et de $\mathbb{D}$. Deux classes importantes de problèmes sont les problèmes d'optimisation convexes et les problèmes d'optimisation quasi–convexes.

Définition 2.14. *Un problème d'optimisation est dit convexe si*
- *$\mathbb{D}$ est un ensemble convexe : pour tout $\xi_1 \in \mathbb{D}$, pour tout $\xi_2 \in \mathbb{D}$, pour tout $\lambda \in [0,1]$, on a $(\lambda\xi_1 + (1-\lambda)\xi_2) \in \mathbb{D}$;*
- *f est une fonction convexe : pour tout $\xi_1 \in \mathbb{D}$, pour tout $\xi_2 \in \mathbb{D}$, pour tout $\lambda \in [0,1]$, on a $f(\lambda\xi_1 + (1-\lambda)\xi_2) \leq \lambda f(\xi_1) + (1-\lambda) f(\xi_2)$.*

Un problème d'optimisation est dit quasi–convexe si
- *$\mathbb{D}$ est un ensemble convexe : pour tout $\xi_1 \in \mathbb{D}$, pour tout $\xi_2 \in \mathbb{D}$, pour tout $\lambda \in [0,1]$, on a $(\lambda\xi_1 + (1-\lambda)\xi_2) \in \mathbb{D}$;*
- *f est une fonction quasi-convexe : pour tout $\xi_1 \in \mathbb{D}$, pour tout $\xi_2 \in \mathbb{D}$, pour tout $\lambda \in [0,1]$, on a $f(\lambda\xi_1 + (1-\lambda)\xi_2) \leq \max\{f(\xi_1), f(\xi_2)\}$.*

Les deux classes de problèmes d'optimisation convexe mentionnées ci-dessus admettent une résolution numérique efficace. En effet, la convexité implique que tout optimum local est en réalité un optimum global et donc le problème de trouver un optimum global se réduit à un problème de trouver un optimum local. Ce dernier type de problème est « facile » à résoudre numériquement [BTN01], *i.e.* il existe au moins un algorithme de résolution, dit « efficace », dont le temps de résolution (temps de calcul nécessaire à cet algorithme pour se terminer) est « raisonnable ». Un algorithme de résolution permet effectivement de résoudre un problème d'optimisation particulier, mais surtout il permet de résoudre tous les problèmes d'optimisation de même classe. Dans ce cadre, une mesure usuelle du temps de résolution d'un algorithme est son évolution en fonction de la « taille » du problème d'optimisation qu'il résout. Un algorithme est dit efficace si l'évolution de son temps de résolution est bornée par une fonction polynomiale de la taille du problème d'optimisation. On dit aussi dans ce cas-là que le problème est P-difficile. Dans le cas contraire, l'algorithme est dit non-efficace et le problème d'optimisation correspondant est dit NP-difficile. Nous ne développerons pas plus cet aspect. Les lecteurs intéressés pourront se référer au livre [GJ79] pour une introduction à la complexité algorithmique.

Du fait qu'ils peuvent être résolus efficacement, les problèmes d'optimisation convexes et quasi–convexes sont donc appréciés. Des sous–problèmes importants sont les problèmes d'optimisation sous contraintes LMI. D'une part, ces problèmes apparaissent actuellement comme des problèmes importants pour lesquels des algorithmes de résolution efficaces ont été programmés dans les logiciels de calcul scientifique généraux comme `Matlab` [GNLC95] ou `Scilab`. D'autre part, les problèmes d'optimisation sous contrainte LMI ont d'importantes applications en Sciences de l'Ingénieur (en Automatique [BEFB94] ou dans d'autres domaines [BTN01]). Ils permettent par exemple de résoudre le problème H_∞-standard formulé dans la Définition 2.7 ainsi que d'autres problèmes qui vont apparaître tout au long de ce manuscrit.

Définition 2.15 (Contrainte LMI)**.** *Soient n_ξ un entier naturel non nul et $F_i \in \mathbb{R}^{m\times m}$, $i = 0, \cdots, n_\xi$, des matrices symétriques. Une contrainte LMI est définie comme l'ensemble de vecteurs suivant*

$$\left\{\xi \in \mathbb{R}^{n_\xi} \mid F(\xi) > 0\right\}, \tag{2.59}$$

avec

$$F(\xi) \triangleq F_0 + \sum_{i=1}^{n_\xi} \xi_i F_i. \tag{2.60}$$

Les matrices F_i, $i = 0, \cdots, n_\xi$, sont appelées les données et le vecteur ξ est appelé le vecteur des variables de décision de la contrainte LMI.

Dans l'expression (2.59), $F(\xi) > 0$ signifie que la matrice $F(\xi)$ est définie positive (de la même façon, $F(\xi) < 0$ signifie que $F(\xi)$ est définie négative). Une contrainte est une contrainte LMI lorsque ξ apparaît de façon affine dans $F(\xi)$. Dans la suite de ce document, l'expression $F(\xi) > 0$ sera aussi appelée contrainte LMI.

Si, dans la Définition 2.13, l'ensemble des contraintes $\mathbb{D}$ est défini par l'expression (2.60), le problème d'optimisation est alors appelé un problème d'optimisation sous contrainte LMI. Ces problèmes seront écrits plus simplement de la manière suivante :

$$\begin{aligned} \min_{\xi \in \mathbb{R}^{n_\xi}} \quad & f(\xi) \\ \text{tel que} \quad & F(\xi) > 0. \end{aligned} \tag{2.61}$$

Il est ainsi possible de définir plusieurs problèmes selon le choix de f. Deux choix sont généralement considérés : f est linéaire ou f est la valeur propre généralisée de deux matrices. Un autre problème important est le problème de faisabilité qui consiste à trouver $\xi \in \mathbb{R}^{n_\xi}$ tel que $F(\xi) > 0$. En résumé, on a les trois problèmes suivants.

Problème de faisabilité Tester s'il existe $\xi \in \mathbb{R}^{n_\xi}$ tel que $F(\xi) > 0$, et si oui, déterminer un tel $\xi \in \mathbb{R}^{n_\xi}$:

$$\begin{aligned} \text{trouver} \quad & \xi \in \mathbb{R}^{n_\xi} \\ \text{tel que} \quad & F(\xi) > 0. \end{aligned} \tag{2.62}$$

Le problème de faisabilité peut se formuler comme un problème d'optimisation au sens de la Définition 2.13. Il revient au problème d'optimisation suivant : $\min_{(\xi,t)\in\mathbb{D}} t$ avec $\mathbb{D} = \{(\xi, t) \,|F(\xi) + tI > 0\}$. Si la valeur minimale de t est strictement positive, alors le Problème de faisabilité (2.62) n'a pas de solution ; si elle négative ou nulle, alors le Problème de faisabilité (2.62) a une solution donnée par la valeur de ξ_{opt} pour laquelle (ξ_{opt}, t) est un optimum global.

Problème de minimisation d'une fonction de coût linéaire Tester s'il existe $\xi \in \mathbb{R}^{n_\xi}$ tel que $F(\xi) > 0$, et si oui, déterminer $\xi \in \mathbb{R}^{n_\xi}$ tel que $F(\xi) > 0$ et qui minimise $c^T\xi$ où $c \in \mathbb{R}^{n_\xi}$ est un vecteur donné :

$$\begin{aligned} \min_{\xi \in \mathbb{R}^{n_\xi}} \quad & c^T\xi \\ \text{tel que} \quad & F(\xi) > 0. \end{aligned} \tag{2.63}$$

Problème de minimisation de la valeur propre généralisée maximale Tester s'il existe $\xi \in \mathbb{R}^{n_\xi}$ tel que $F(\xi) > 0$ et $H(\xi) > 0$ où $H(\xi)$ est de la forme (2.60), et si oui, déterminer $\xi \in \mathbb{R}^{n_\xi}$ tel que $F(\xi) > 0$ et $H(\xi) > 0$ et qui minimise la valeur propre généralisée maximale $\lambda_{max}(G(\xi), F(\xi))$ de $F(\xi)$ et $G(\xi)$ où $G(\xi)$ est de la forme (2.60) et où $\lambda_{max}(G(\xi), F(\xi))$ est la valeur minimale de λ pour laquelle

$\lambda F(\xi) - G(\xi)$ est définie positive [13] :

$$\begin{aligned} \min_{\xi \in \mathbb{R}^{n_\xi}} \quad & \lambda_{max}\left(G(\xi), F(\xi)\right) \\ \text{tel que} \quad & F(\xi) > 0 \text{ et } H(\xi) > 0. \end{aligned} \tag{2.64}$$

Remarque 2.16. Lorsqu'il y a plusieurs contraintes LMI $F_i(\xi) > 0$, $i = 1, \cdots, n_F$ (comme dans le Problème (2.64)), il est toujours possible de les réécrire sous la forme d'une seule contrainte :

$$F(\xi) \triangleq \begin{bmatrix} F_1(\xi) & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & F_{n_F}(\xi) \end{bmatrix} > 0.$$

Il n'y a donc pas de perte de généralité à définir les problèmes d'optimisation sous contrainte LMI avec une seule contrainte LMI.

Jusqu'à présent, nous avons considéré une contrainte LMI avec un vecteur de variables de décision scalaires. En Automatique, il est plus courant et naturel d'utiliser des variables de décision matricielles. L'expression (2.64) peut se récrire (et inversement) sous la forme :

$$F(\xi) \triangleq F_c + \sum_{i=1}^{n_\mathcal{R}} F_i^g \mathcal{R}_i(\xi) F_i^d + \left(F_c + \sum_{i=1}^{n_\mathcal{R}} F_i^g \mathcal{R}_i(\xi) F_i^d \right)^T, \tag{2.65}$$

où $F_c \in \mathbb{R}^{m \times m}$, $F_i^g \in \mathbb{R}^{m \times a_i}$ et $F_i^d \in \mathbb{R}^{b_i \times m}$ sont données et où les $\mathcal{R}_i(\xi) \in \mathbb{R}^{a_i \times b_i}$, $i = 1, \cdots, n_\mathcal{R}$, linéaires et possiblement structurées en ξ, sont les variables de décision matricielles. Par exemple, l'expression (2.65) peut être trouvée à partir de l'expression (2.60) en choisissant $F_c = \frac{1}{2}F_0$, F_i^g et F_i^d telles que $F_i^g F_i^d = \frac{1}{2}F_i$ et $\mathcal{R}_i(\xi) = \xi_i I$.

13. $\lambda_{max}(A, B)$ est aussi définie comme la plus grande valeur propre de $B^{-1/2}AB^{-1/2}$ et n'est définie que dans le cas où B est définie positive. Dans le problème d'optimisation (2.64), $F(\xi) > 0$ est une contrainte du problème ; ce problème est donc bien posé.

2.10 Annexe B : Résolution du problème H_∞ standard par la méthode d'optimisation sous contrainte LMI

Dans cette annexe, nous allons présenter les grandes lignes de la démonstration de la solution du problème H_∞-standard (voir la Définition 2.7 page 75) généralement reprises de [Sco06]. Soit le système $\widetilde{H_P}$ interconnecté avec le système K comme représenté sur la Figure 2.33 page 74 et répété pour faciliter la compréhension sur la Figure 2.56 où :
– le vecteur des sorties commandées $\widetilde{z}(t)$ est de dimension p_z ;
– le vecteur des entrées de critère $\widetilde{w}(t)$ est de dimension m_w ;
– le vecteur des signaux mesurées $y(t)$ est de dimension p_y ;
– le vecteur des entrées de commande $u(t)$ est de dimension m_u.
Le système $\widetilde{H_P}$ admet pour équations d'état :

$$\begin{array}{rclcrclcl} \dot{x}(t) & = & Ax(t) & + & B_w\widetilde{w}(t) & + & B_u u(t)\,; & & \\ \widetilde{z}(t) & = & C_z x(t) & + & D_{zw}\widetilde{w}(t) & + & D_{zu}u(t)\,; & & (2.66) \\ y(t) & = & C_y x(t) & + & D_{yw}\widetilde{w}(t) & + & D_{yu}u(t)\,. & & \end{array}$$

L'entier n est l'ordre de la représentation d'état. Dans le domaine de Laplace, les équations du système se réécrivent :

$$\widetilde{H_P}(s) = \begin{bmatrix} \widetilde{H}_{zw}(s) & \widetilde{H}_{zu}(s) \\ \widetilde{H}_{yw}(s) & \widetilde{H}_{yu}(s) \end{bmatrix} = \begin{bmatrix} D_{zw} & D_{zu} \\ D_{yw} & D_{yu} \end{bmatrix} + \begin{bmatrix} C_z \\ C_y \end{bmatrix} (sI-A)^{-1} \begin{bmatrix} B_w & B_u \end{bmatrix};$$

$$\begin{bmatrix} \widetilde{z}(s) \\ y(s) \end{bmatrix} = \widetilde{H_P}(s) \begin{bmatrix} \widetilde{w}(s) \\ u(s) \end{bmatrix} \quad \text{et} \quad u(s) = K(s)\,y(s)\,,$$

soit $\widetilde{z}(s) = H_P(s) \star K(s)\,\widetilde{w}(s)$

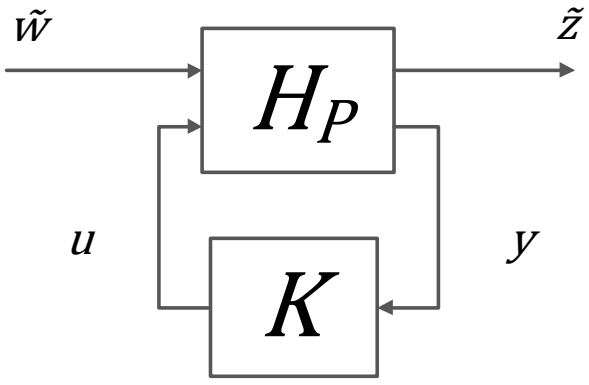


Figure 2.56: Problème sous forme H_∞-standard

Par soucis de simplicité, seul le cas du retour d'état où « l'information complète » (c'est-à-dire, que le correcteur à accès à tout l'état du système : $y(t) = x(t)$ soit $C_y = I$, $D_{yw} = 0$ et $D_{yu} = 0$) sera considéré en détails. Le cas général nécessite des développements techniques assez lourds qui n'apportent rien à la compréhension des idées sous-jacentes. Sa présentation complète peut être trouvée dans [GA94]. Ici, on se concentre seulement sur les principes généraux de la démonstration.

Synthèse H_∞ par retour d'états

Dans le cas du retour d'état, le système $\widetilde{H_P}$ admet pour équations d'état :

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_w\widetilde{w}(t) + B_u u(t); \\ \widetilde{z}(t) &= C_z x(t) + D_{zw}\widetilde{w}(t) + D_{zu}u(t); \\ y(t) &= x(t). \end{aligned} \tag{2.67}$$

On recherche une loi de commande par retour d'état (c'est-à-dire une matrice $K \in \mathbb{R}^{p\times n}$) telle que $u(t) = Kx(t)$ assure la résolution du problème H_∞-standard (voir la Définition 2.7).

Les développements qui suivent sont basés sur le lemme suivant.

Lemme 2.17 (Réel borné)**.** *Soit un système linéaire stationnaire multivariable stable H décrit par une représentation d'état minimale :*

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bw(t); \\ z(t) &= Cx(t) + Dw(t), \end{aligned}$$

où $x(t)$ est l'état du système, $w(t)$ l'entrée et $z(t)$ est la sortie. Alors $\|H\|_\infty \leq \gamma$ si et seulement s'il existe une matrice P telle que

$$\begin{cases} P > 0; \\ \begin{bmatrix} A^TP + PA + C^TC & PB + C^TD \\ B^TP + D^TC & D^TD - \gamma^2 I \end{bmatrix} < 0. \end{cases} \tag{2.68}$$

Le système en boucle fermée (voir la Figure 2.56) admet pour équations d'état :

$$\begin{aligned} \dot{x}(t) &= (A + B_uK)x(t) + B_w\widetilde{w}(t); \\ \widetilde{z}(t) &= (C_z + D_{zu}K)x(t) + D_{zw}\widetilde{w}(t). \end{aligned}$$

En appliquant le lemme 2.17 au système en boucle fermée, il existe un correcteur par retour d'état s'il existe une matrice K et une matrice symétrique P telles que $P > 0$ et :

$$\begin{bmatrix} (A+B_uK)^T P + P(A+B_uK) + (C_z+D_{zu}K)^T(C_z+D_{zu}K) & PB_w + (C_z+D_{zu}K)^T D_{zw} \\ B_w^T\underline{P} + D_{zw}^T(C_z+D_{zu}K) & D_{zw}^TD_{zw} - \gamma^2 I \end{bmatrix} < 0. \tag{2.69}$$

Les variables matricielles d'optimisation sont P et K. Cette inégalité n'étant pas affine en P et en K, elle ne définit pas une LMI. Néanmoins, par une série de transformations, nous allons montrer que l'on peut obtenir une contrainte LMI équivalente après un changement de variables adéquat.

L'inégalité (2.69) peut se réécrire :

$$\begin{bmatrix} A^TP + PA + K^TB_u^TP + PB_uK & PB_w \\ B_w^TP & -\gamma^2 I \end{bmatrix} + \begin{bmatrix} (C_z + D_{zu}K)^T \\ D_{zw}^T \end{bmatrix} \begin{bmatrix} C_z + D_{zu}K & D_{zw} \end{bmatrix} \leq 0. \quad (2.70)$$

Par la suite, nous allons avoir besoin du lemme suivant.

Lemme 2.18 (Lemme de Schur modifié)**.** *Soit une matrice symétrique partionnée :*

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix},$$

où $\mathbf{A}$ *et* $\mathbf{C}$ *ont des matrices carrées, avec* $\mathbf{A}$ *définie positive. Cette matrice est définie positive si et seulement si* $\mathbf{C} - \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B}$ *est semi définie positive.*

L'inégalité (2.70) peut alors se réécrire :

$$\underbrace{\begin{bmatrix} A^TP + PA + K^TB_u^TP + PB_uK & PB_w \\ B_w^TP & -\gamma^2 I \end{bmatrix}}_{\mathbf{C}} - \underbrace{\begin{bmatrix} (C_z + D_{zu}K)^T \\ D_{zw}^T \end{bmatrix}}_{\mathbf{B}^T} \underbrace{(-I)}_{\mathbf{A}^{-1}} \underbrace{\begin{bmatrix} C_z + D_{zu}K & D_{zw} \end{bmatrix}}_{\mathbf{B}} < 0. \quad (2.71)$$

Par application du lemme de Schur modifié, on a alors l'inégalité (2.71) qui est équivalente à :

$$\begin{bmatrix} A^TP + PA + K^TB_u^TP + PB_uK & PB_w & (C_z + D_{zu}K)^T \\ B_w^TP & -\gamma^2 I & D_{zw}^T \\ C_z + D_{zu}K & D_{zw} & -I \end{bmatrix} < 0. \quad (2.72)$$

L'intérêt de cette transformation est que, dans l'inégalité obtenue, il ne reste plus que deux termes bilinéaires : PB_uK et $K^TB_u^TP$. On va utiliser une nouvelle transformation puis un changement de variable de façon à se ramener à une inégalité affine. Pour cela, on va se baser sur la propriété suivante :

Propriété $\forall \mathbf{T}$ inversible $\in \mathbb{R}^{n\times n}$, $\mathbf{A} < 0 \quad \Leftrightarrow \quad \mathbf{T}^T\mathbf{A}\mathbf{T} < 0.$

En appliquant cette propriété avec :

$$\mathbf{T} = \begin{bmatrix} P^{-1} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \quad \text{et} \quad \mathbf{A} = \begin{bmatrix} A^TP + PA + K^TB_u^TP + PB_uK & PB_w & (C_z + D_{zu}K)^T \\ B_w^TP & -\gamma^2 I & D_{zw}^T \\ C_z + D_{zu}K & D_{zw} & -I \end{bmatrix},$$

on obtient :

$$\begin{bmatrix} P^{-1}A^T + AP^{-1} + P^{-1}K^TB_u^T + B_uKP^{-1} & B_w & P^{-1}\left(C_z^T + K^TD_{zu}^T\right) \\ B_w^T & -\gamma^2 I & D_{zw}^T \\ C_zP^{-1} + D_{zu}KP^{-1} & D_{zw} & -I \end{bmatrix} < 0. \quad (2.73)$$

En posant $Q = P^{-1} \in \mathbb{R}^{n\times n}$ et $Y = KP^{-1} \in \mathbb{R}^{p\times n}$, on obtient :

$$\begin{bmatrix} QA^T + AQ + Y^TB_u^T + B_uY & B_w & QC_z^T + Y^TD_{zu}^T \\ B_w^T & -\gamma^2 I & D_{zw}^T \\ C_zQ + D_{zu}Y & D_{zw} & -I \end{bmatrix} < 0. \quad (2.74)$$

D'autre part, on a $P > 0 \Leftrightarrow Q > 0$.

Ce changement de variables est bien posé car la fonction qui relie (P, K) et (Q, Y) est une bijection. De plus, les inégalités (2.74) et $Q > 0$ sont bien affines en les variables Q et Y. Rechercher Q et Y tels que $Q > 0$ et (2.74) est donc un problème de faisabilité LMI tel que posé dans la section 2.9.

En conclusion, la recherche d'une loi de commande $u(t) = Kx(t)$ assurant la stabilité de la boucle fermée et une norme H_∞ entre $\widetilde{w}$ et $\widetilde{z}$ inférieure à γ s'obtient de la façon suivante :

1. Trouver Q et Y tels que les inégalités $Q > 0$ et (2.74) soient satisfaites (problème de faisabilité) ;
2. Poser $P = Q^{-1}$ et $K = YP$.

Existence d'un correcteur H_∞

On peut être intéressé par déterminer s'il existe un correcteur H_∞ pour un γ donné, sans le calculer explicitement. Pour cela, on dispose du résultat suivant.

Lemme 2.19 (Lemme d'élimination)**.** *Soit* $\mathbf{G} = \mathbf{G}^T \in \mathbb{R}^{n\times n}$, $\mathbf{U} \in \mathbb{R}^{r\times n}$ *avec* $Rang(\mathbf{U}) = r < n$ *et* $\mathbf{V} \in \mathbb{R}^{s\times n}$ *avec* $Rang(\mathbf{V}) = s < n$. *Alors, il existe une matrice* $\underline{\mathbf{K}} \in \mathbb{R}^{r\times s}$ *telle que l'inégalité suivante est vérifiée :*

$$\mathbf{G} + \mathbf{U}^T\underline{\mathbf{K}}\mathbf{V} + \mathbf{V}^T\underline{\mathbf{K}}^T\mathbf{U} < 0$$

si et seulement si :

$$\mathbf{U}_\perp^T\mathbf{G}\mathbf{U}_\perp < 0 \quad et \quad \mathbf{V}_\perp^T\mathbf{G}\mathbf{V}_\perp < 0,$$

avec

– $\mathbf{U}_\perp$ *engendre le noyau de l'application linéaire associée à la matrice*[14] $\mathbf{U}$;

– $\mathbf{V}_\perp$ *engendre le noyau de l'application linéaire associée à la matrice* $\mathbf{V}$.

14. Les vecteurs colonnes de $\mathbf{U}_\perp$ forment une base du noyau.

Ce résultat est remarquable puisqu'il donne une condition nécessaire et suffisante permettant de tester l'existence d'une matrice $\underline{\mathbf{K}}$ telle que la matrice $\mathbf{G}+\mathbf{U}^T\underline{\mathbf{K}}^T\mathbf{V}+\mathbf{V}^T\underline{\mathbf{K}}\mathbf{U}$ est (semi) définie négative, sans rechercher $\underline{\mathbf{K}}$ explicitement.

L'inégalité (2.74) peut se réécrire :

$$\underbrace{\begin{bmatrix} QA^T+AQ & B_w & QC_z^T \\ B_w^T & -\gamma^2 I & D_{zw}^T \\ C_zQ & D_{zw} & -I \end{bmatrix}}_{\mathbf{G}} + \underbrace{\begin{bmatrix} B_u \\ 0 \\ D_{zu} \end{bmatrix}}_{\mathbf{U}^T} \underbrace{Y}_{\underline{\mathbf{K}}} \underbrace{\begin{bmatrix} I & 0 & 0 \end{bmatrix}}_{\mathbf{V}} + \begin{bmatrix} I \\ 0 \\ 0 \end{bmatrix} Y^T \begin{bmatrix} B_u^T & 0 & D_{zu}^T \end{bmatrix} < 0. \quad (2.75)$$

En appliquant le Lemme 2.19, l'inégalité (2.75) est équivalente à

$$\mathbf{U}_\perp^T\mathbf{G}\mathbf{U}_\perp < 0 \Leftrightarrow \begin{bmatrix} B_u^T & 0 & D_{zu}^T \end{bmatrix}_\perp^T \begin{bmatrix} QA^T+AQ & B_w & QC_z^T \\ B_w^T & -\gamma^2 I & D_{zw}^T \\ C_zQ & D_{zw} & -I \end{bmatrix} \begin{bmatrix} B_u^T & 0 & D_{zu}^T \end{bmatrix}_\perp < 0$$

et à

$$\mathbf{V}_\perp^T\mathbf{G}\mathbf{V}_\perp < 0 \Leftrightarrow \begin{bmatrix} I & 0 & 0 \end{bmatrix}_\perp^T \begin{bmatrix} QA^T+AQ & B_w & QC_z^T \\ B_w^T & -\gamma^2 I & D_{zw}^T \\ C_zQ & D_{zw} & -I \end{bmatrix} \begin{bmatrix} I & 0 & 0 \end{bmatrix}_\perp < 0.$$

Comme

$$\begin{bmatrix} I & 0 & 0 \end{bmatrix}_\perp = \begin{bmatrix} 0 & 0 \\ I & 0 \\ 0 & I \end{bmatrix}$$

on a :

$$\mathbf{V}_\perp^T\mathbf{G}\mathbf{V}_\perp < 0 \Leftrightarrow \begin{bmatrix} -\gamma^2 I & D_{zw}^T \\ D_{zw} & -I \end{bmatrix} < 0 \Leftrightarrow D_{zw}^T D_{zw} < \gamma^2 I.$$

En résumé, une condition nécessaire et suffisante d'existence d'un correcteur par retour d'état assurant que

1. le système bouclé est stable ;
2. la norme H_∞ de la fonction de transfert en boucle fermée entre l'entrée $\widetilde{w}$ et la sortie $\widetilde{z}$ est inférieure à γ

est donnée par

1. $D_{zw}^T D_{zw} < \gamma^2 I$ (valeur singulière maximale de D_{zw} inférieure ou égale à γ) ;

2. l'existence d'une matrice Q telle que $Q > 0$ et

$$\left[\begin{array}{ccc} B_u^T & 0 & D_{zu}^T \end{array}\right]_\perp^T \left[\begin{array}{ccc} QA^T + AQ & B_w & QC_z^T \\ B_w^T & -\gamma^2 I & D_{zw}^T \\ C_z Q & D_{zw} & -I \end{array}\right] \left[\begin{array}{ccc} B_u^T & 0 & D_{zu}^T \end{array}\right]_\perp < 0.$$

On a donc un problème de faisabilité LMI dans lequel le correcteur n'est pas recherché explicitement.

Synthèse H_∞ par retour de sortie

Le théorème suivant donne la solution du problème de la Définition 2.7 dans le cas où tout l'état n'est pas mesuré.

Théorème 2.20. *Soit le système $\widetilde{H_P}$ d'ordre n défini par* (2.66)*. Étant donné $\gamma > 0$, il existe un correcteur K d'ordre inférieur ou égal à n, défini par la représentation d'état :*

$$\begin{array}{rcl} \dot{x}_K(t) & = & A_K x_K(t) + B_K y(t); \\ u(t) & = & C_K x(t) + D_K y(t), \end{array}$$

telle que

(i) le système bouclé $\widetilde{H_P} \star K$ soit asymptotiquement stable
(tous les pôles du système en boucle fermée sont à partie réelle strictement négative)

(ii) $\|\widetilde{H_P} \star K\|_\infty < \gamma$

si et seulement si il existe deux matrices P et Q telles que :

$$\left[\begin{array}{ccc} B_u^T & 0 & D_{zu}^T \end{array}\right]_\perp^T \left[\begin{array}{ccc} QA^T + AQ & B_w & QC_z^T \\ B_w^T & -\gamma I & D_{zw}^T \\ C_z Q & D_{zw} & -\gamma I \end{array}\right] \left[\begin{array}{ccc} B_u^T & 0 & D_{zu}^T \end{array}\right]_\perp < 0; \quad (2.76)$$

$$\left[\begin{array}{ccc} C_y & D_{yw} & 0 \end{array}\right]_\perp^T \left[\begin{array}{ccc} A^T P + PA & PB_w & C_z^T \\ B_w^T P & -\gamma I & D_{zw}^T \\ C_z & D_{zw} & -\gamma I \end{array}\right] \left[\begin{array}{ccc} C_y & D_{yw} & 0 \end{array}\right]_\perp < 0 \quad (2.77)$$

et

$$\begin{bmatrix} P & I \\ I & Q \end{bmatrix} > 0. \tag{2.78}$$

Tester l'existence d'un correcteur qui assure un "niveau de performance" γ donné est un problème de faisabilité LMI. Tester l'existence d'un correcteur qui assure le plus petit "niveau de performance" γ peut s'écrire comme un problème de minimisation d'un coût linéaire sous contraintes LMI (voir la section 2.9).

Principe de la démonstration Comme dans le cas du retour d'état, le résultat s'obtient en appliquant le lemme réel borné au système en boucle fermée en prenant soin de faire apparaître dans sa représentation d'état les matrices d'état du correcteur réunies dans une matrice par blocs :

$$\begin{aligned}
\begin{bmatrix} \dot{x}(t) \\ \dot{x}_K(t) \end{bmatrix} &= \left(\underbrace{\begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}}_{\widehat{A}} + \underbrace{\begin{bmatrix} 0 & B_u \\ I & 0 \end{bmatrix}}_{\widehat{B}_u} \underbrace{\begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix}}_{\widehat{K}} \underbrace{\begin{bmatrix} 0 & I \\ C_y & 0 \end{bmatrix}}_{\widehat{C}_y} \right) \underbrace{\begin{bmatrix} x(t) \\ x_K(t) \end{bmatrix}}_{\widehat{x}(t)} \quad \cdots \\
&\quad + \left(\underbrace{\begin{bmatrix} B_w \\ 0 \end{bmatrix}}_{\widehat{B}_w} + \begin{bmatrix} 0 & B_u \\ I & 0 \end{bmatrix} \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix} \underbrace{\begin{bmatrix} 0 \\ D_{yw} \end{bmatrix}}_{\widehat{D}_{yw}} \right) \widetilde{w}(t); \\
\widetilde{z}(t) &= \left(\underbrace{\begin{bmatrix} C_z & 0 \\ 0 & 0 \end{bmatrix}}_{\widehat{C}_z} + \underbrace{\begin{bmatrix} 0 & D_{zu} \end{bmatrix}}_{\widehat{D}_{zu}} \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix} \begin{bmatrix} 0 & I \\ C_y & 0 \end{bmatrix} \right) \begin{bmatrix} x(t) \\ x_K(t) \end{bmatrix} \quad \cdots \\
&\quad + \left(D_{zw} + \begin{bmatrix} 0 & D_{zu} \end{bmatrix} \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix} \begin{bmatrix} 0 \\ D_{yw} \end{bmatrix} \right) w(t).
\end{aligned}$$

Ceci peut s'interpréter comme la commande d'un système $\widehat{H_P}$ augmenté de n_K intégrateurs :

$$\begin{aligned}
\dot{\widehat{x}}(t) &= \widehat{A}\widehat{x}(t) + \widehat{B}_w\widetilde{w}(t) + \widehat{B}_u\widehat{u}(t); \\
\widetilde{z}(t) &= \widehat{C}_z\widehat{x}(t) + D_{zw}\widetilde{w}(t) + \widehat{D}_{zu}\widehat{u}(t); \\
\widehat{y}(t) &= \widehat{C}_y\widehat{x}(t) + \widehat{D}_{yw}\widetilde{w}(t)
\end{aligned}$$

par un correcteur par retour statique de la sortie :

$$\widehat{u}(t) = \widehat{K}\widehat{y}(t).$$

L'étape suivante consiste à appliquer le lemme réel borné (énonce page 107). On obtient alors une formulation similaire à la formulation 2.72 : il existe une matrice $\widehat{P}$ définie positive telle que :

$$\begin{bmatrix} \widehat{A}^T\widehat{P} + \widehat{P}\widehat{A} + \widehat{C}_y^T\widehat{K}^T\widehat{B}_u^T\widehat{P} + \widehat{P}\widehat{B}_u\widehat{K}\widehat{C}_y & \widehat{P}\widehat{B}_w & (\widehat{C}_z + \widehat{D}_{zu}\widehat{K}\widehat{C}_y)^T \\ \widehat{B}_w^T\widehat{P} & -\gamma^2 I & (\widehat{D}_{zw} + \widehat{D}_{zu}\widehat{K}\widehat{D}_{yw})^T \\ \widehat{C}_z + \widehat{D}_{zu}\widehat{K}\widehat{C}_y & \widehat{D}_{zw} + \widehat{D}_{zu}\widehat{K}\widehat{D}_{yw} & -I \end{bmatrix} < 0 \qquad (2.79)$$

Par application du lemme d'élimination, on obtient les conditions (2.76) et (2.77) dans lesquelles on a P et Q telles que :

$$\widehat{P} = \begin{bmatrix} P & \star \\ \star & \star \end{bmatrix} \quad \text{et} \quad \widehat{P}^{-1} = \begin{bmatrix} Q & \star \\ \star & \star \end{bmatrix}$$

La condition (2.78) garantit que $\widehat{P} > 0$.

Ayant déterminé P et Q satisfaisant les conditions du Théorème 2.20, il est possible de reconstruire le correcteur K résolvant le problème H_∞-standard de la Définition 2.7 avec $\gamma > 0$. En réalité, on est intéressé que par la solution où $0 < \gamma \leq 1$, nous permettant de garantir la performance (hormis la stabilité du modèle) selon les conditions (2.39) et (2.37).

Une réalisation de l'algorithme de la résolution du problème H_∞-standard décrit dans cette annexe peut être trouvée dans la boite à outils de `Matlab` : Robust Control Toolbox.

3 Systèmes en réseau de grande dimension

Introduction

Dans les chapitres précédents, nous avons vu que le réseau actif de distribution d'horloge est un système de grande dimension complexe, composé de sous-systèmes interconnectés. Les sous-systèmes sont des ADPLLs identiques et l'interconnexion est réalisée par un échange d'information sur la différence de phase relative des ADPLLs voisines. On dit aussi dans ce cas-là, que les ADPLLs sont couplées et que si l'information transmise dans le réseau est la différence de phase, elles sont couplées dans le domaine des phases. Nous avons également vu, dans le chapitre précédent, que la synthèse d'une telle ADPLL après un choix préalable des paramètres principaux de ses sous-blocs élémentaires (VCO, PD, diviseur de fréquence) et la linéarisation du modèle consiste en la synthèse d'une loi de commande appropriée c'est-à-dire en un choix de correcteur qui assure la stabilité et la performance désirées. Un exemple d'une telle synthèse du point de vue Automaticien a été également présenté dans le chapitre précédent.

Pour un réseau complet d'ADPLLs identiques couplées et distribuées sur une puce, le problème de sa synthèse est composé comme pour la synthèse d'une ADPLL unique des mêmes étapes principales. On retrouve également un choix préalable des paramètres principaux des sous-blocs d'ADPLLs et une linéarisation du modèle complet, ainsi que la synthèse de la loi de commande. Mais contrairement au cas d'une ADPLL locale, le choix d'une structure d'interconnexion (c'est-à-dire le choix d'une topologie d'échange de l'information) ainsi qu'un choix approprié de correcteurs locaux de chaque PLL assurant non seulement le comportement désiré d'une seule ADPLL mais aussi les propriétés désirées du comportement du réseau complet sont nécessaires.

Dans la présente étude, on suppose que les paramètres d'ADPLLs, mis à part les paramètres du filtre, sont choisis au préalable (voir le Tableau 2.1 page 55) fixant ainsi le procédé à commander P de chaque ADPLL. On considère également que la topologie d'interconnexion est fixée par l'application. En réalisant la linéarisation du réseau complet d'ADPLLs (selon l'approche décrite dans le chapitre 2) le problème de la synthèse du réseau d'ADPLLs consiste alors en la synthèse d'une loi de commande réalisée par des correcteurs locaux assurant la stabilité et un niveau de performance désiré du modèle global ainsi obtenu du réseau d'ADPLLs en dépit de leur couplage. On a montré dans l'exemple numérique à la fin du chapitre précédent qu'une utilisation naïve de la synthèse locale du correcteur d'une ADPLL ne pre-

nant pas en compte les aspects d'interconnexion globale ne permet pas de résoudre le problème de synthèse considérée.

Le problème de la synthèse de la loi de commande d'un réseau d'ADPLLs peut se généraliser alors en un problème de synthèse de la loi de commande d'un système de grande dimension $\sum_{glob}$ composé de sous-systèmes identiques locaux $\sum_{loc}$ comprenant deux parties : *(i)* le correcteur local K réalisant la loi de commande et *(ii)* le procédé local à commander P. La loi de commande doit assurer, en plus des propriétés locales de sous-systèmes, la stabilité et un niveau de performance globaux désirés du système de grande dimension $\sum_{glob}$. Cette description générale du problème de la synthèse couvre non seulement le cas d'application considérée dans la thèse mais aussi un large éventail d'autres applications de la commande de systèmes de grande dimension : réseau de satellites, de robots, de véhicules mobiles, de capteurs, d'oscillateur, de MEMS/NEMS *etc.* ; touchant différent domaines des Sciences de l'ingénieur tel que les Télécommunications, la Microélectronique, la Robotique, l'Aéronautique, le Transport, la Biologie, *etc.*. On se propose d'élaborer une méthode plus générale de résolution de ce problème. A titre d'illustration, nous démontrerons les résultats obtenus sur un exemple de synchronisation d'un réseau actif de distribution d'horloge.

La résolution directe du problème formulé dans le cas général vue sa grande taille et son caractère décentralisé, n'est pas une tâche simple et il n'existe pas à ce jour d'algorithmes efficaces pour sa réalisation. On propose alors en se basant sur des connaissances sur l'interconnexion de transformer les conditions issues de propriétés désirées du système global $\sum_{glob}$ en des conditions locales à satisfaire par les sous-systèmes locaux $\sum_{loc}$. Ceci permet de non seulement simplifier la difficulté du problème d'origine mais aussi de tirer profit des outils existants efficaces de synthèse locale de la commande tout en les généralisant pour le cas de la commande de système de grande dimension.

Ce problème du lien « global-local » a été déjà étudié dans la littérature par les deux approches différentes suivantes : une approche de Commande Décentralisée et une approche de Systèmes Multi-Agents. Ces approches étudient deux différents problèmes :

– Commande Décentralisée : la commande par les différents correcteurs K_i de différents sous-systèmes interconnectés H_i
– Systèmes Multi-Agents : la commande par N correcteurs identiques K des N sous-systèmes identiques G [1].

Dans l'optique de notre application de la commande de réseau d'ADPLLs identiques interconnectées, chaque approche apporte une contribution intéressante. Cependant, aucune d'elles ne peut être directement appliquée de manière efficace dans la résolution du problème général de la synthèse évoquée précédemment. Dans ce chapitre, nous allons examiner en détails ces deux approches et les raisons pour lesquelles elles ne peuvent pas être appliquées directement pour résoudre le problème considéré. Nous allons aborder tout d'abord l'approche dite « Multi-Agents ». Ensuite,

1. Généralement, les sous-systèmes selon l'approche de systèmes Multi-Agents sont identiques, ils existent, néanmoins, comme nous allons voir par la suite, quelques extensions au cas hétérogène c'est-à-dire au cas où les sous-systèmes sont différents. Dans tous le cas les correcteurs restent les mêmes pour tous les sous-systèmes.

nous allons étudier les méthodes de Commande Décentralisée qui, pour une raison ou pour une autre, sont souvent négligées par les auteurs de l'approche Multi-Agents. Une combinaison des méthodes de Commande Décentralisés datant de la fin des années 70, avec les outils modernes d'optimisation convexe LMI peut en effet apporter une grande contribution pour la résolution de certains problèmes de commande de systèmes Multi-Agents. Cette thèse par ailleurs, est une démonstration possible de l'efficacité d'une telle démarche.

3.1 Systèmes Multi-Agents

3.1.1 Définition

Étant donné que l'étude des systèmes Multi-Agents est un domaine relativement récent, il existe plusieurs définitions d'un système Multi-Agents. Dans notre étude, nous allons utiliser la définition suivante issue de [Woo01].

Définition 3.1 (Système Multi-Agents)**.** Un *Système Multi-Agents* (ou *Multi-Agent System*, *MAS* en anglais) est un système composé de sous-systèmes intelligents (appelés agents) interagissant les uns avec les autres à travers un échange d'informations. Chaque agent effectue certaines opérations en se basant sur l'information localement disponible afin d'atteindre les objectifs désirés globaux.

En général, les auteurs de l'approche MAS considèrent que les agents sont identiques comme dans le cas du réseau d'ADPLLs interconnectées pour la synthèse de signaux d'horloge synchrones. Le lien entre les propriétés globales et locales est établi grâce à la Théorie des graphes et à la Théorie algébrique des matrices modélisant l'interconnexion. Un petit aperçu sur les définitions et les résultats principaux de ces deux théories est donné dans l'Annexe C, page 143, de ce chapitre. Ici, nous nous concentrons plutôt sur une description détaillée du résultat principal de stabilité de l'approche de MAS [FM04] qui est une des bases d'une grande majorité des autres résultats pour les MASs. Il permet également de faire une démonstration du lien « global-local » dont le principe est utilisé par la suite dans les résultats ultérieurs de systèmes Multi-Agents, que ce soient les résultats d'analyse ou de synthèse. Ces résultats sont discutés à la fin de la présente sous-section.

3.1.2 Stabilité du système Multi-Agents

Comme mentionné précédemment, un des résultats principaux de l'approche de MAS est le résultat de stabilité donné par [FM04]. Les auteurs de [FM04] considèrent un réseau de N agents LTIs dont chacun est composé de deux sous-systèmes : le procédé G et le correcteur K. Les agents sont supposés identiques et le comportement de leurs sous-systèmes est décrit par les deux représentations d'états suivantes :

$$G : \begin{cases} \dot{x}_i &= Ax_i + Bu_i \\ y_i &= Cx_i \end{cases} \tag{3.1}$$

et

$$K : \begin{cases} \dot{v}_i &= K_A v_i + K_B \varepsilon_i \\ u_i &= K_C v_i + K_D \varepsilon_i \end{cases} \tag{3.2}$$

où $x_i(t) \in \mathbb{R}^n$, $v_i(t) \in \mathbb{R}^s$, $u_i(t) \in \mathbb{R}^m$, $\varepsilon_i(t) \in \mathbb{R}^l$, et $y_i(t) \in \mathbb{R}^l$ sont les états, les signaux d'entrée et les signaux de sortie du procédé G et du correcteur K respectivement.

Les agents sont interconnectés selon une topologie constante définie par un graphe $\mathcal{G}(\mathcal{A}, \mathcal{V})$ (voir l'Annexe C page 143 de ce chapitre pour plus de détails) grâce aux signaux d'erreurs ε_i à l'entrée de correcteurs :

$$\varepsilon_i = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} (y_j - y_i) \tag{3.3}$$

où $\mathcal{N}_i$ est un ensemble des agents voisins de l'agent i : $\mathcal{N}_i = \{j \in \mathcal{V} : (j,i) \in \mathcal{A}\}$ et $|\mathcal{N}_i|$ désigne la cardinalité de l'ensemble $\mathcal{N}_i$.

En notant les vecteurs x, v, u, ε et y des concaténations de vecteurs correspondants de chaque agent et en utilisant le produit de Kronecker, on obtient le système global suivant :

$$\sum_{glob} : \begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} \tag{3.4}$$

où

$$\begin{aligned} A_{11} &= I_N \otimes A - L \otimes B K_D C \\ A_{12} &= I_N \otimes B K_C \\ A_{21} &= -L \otimes K_B C \\ A_{22} &= I_N \otimes K_A \end{aligned} \tag{3.5}$$

et où L est la matrice décrivant l'interconnexion des agents définie à partir de l'équation (3.3). Dans [FM04], la matrice utilisée pour décrire les interconnexions entre les agents est une matrice Laplacienne normalisée définie dans la Définition 3.15, page 137 de l'Annexe C de ce chapitre. Dans l'optique qui nous intéresse, nous formulons le résultat plus général en utilisant une matrice d'interconnexion L quelconque.

Théorème 3.2. *Le correcteur local K (3.2) stabilise le réseau global de MAS (3.4) si et seulement si il stabilise simultanément un ensemble de N sous-systèmes transformés indépendants définis par :*

$$\sum_i : \begin{cases} \dot{x}_i &= A x_i + B u_i \\ y_i &= \lambda_i C x_i \end{cases} \tag{3.6}$$

où λ_i sont les valeurs propres de la matrice d'interconnexion L.

Démonstration. La démonstration utilise un changement de base dans la représentation d'états (3.4) qui définit la transformation de Schur de la matrice d'interconnexion L (voir le Théorème 3.18 page 140 de l'Annexe C) en transformant le système (3.4) en un système équivalent dont chaque matrice est soit diagonale soit triangulaire supérieure. La stabilité du système triangulaire supérieur obtenu est équivalente à la stabilité des sous-systèmes définis par les blocs diagonaux. Chaque sous-système diagonal a un gain différent à la sortie du procédé (3.6) qui est défini par les valeurs propres de la matrice d'interconnexion L. En conclusion, le système global (3.4) est stabilisé par les mêmes correcteurs K (3.2) si et seulement si les N sous-systèmes transformés (3.6) sont stabilisés par le même correcteur K (3.2). Pour obtenir plus de détails sur la démonstration, le lecteur peut consulter le Théorème 3 de [FM04]. □

Remarque 3.3. Le Théorème 3.2 montre que la stabilité du système de Multi-Agents interconnectés est équivalente à la stabilité simultanée de N sous-systèmes transformés indépendants dont la différence est définie par les valeurs propres de la matrice d'interconnexion, et notamment de la matrice Laplacienne. Ce résultat d'analyse peut être vu comme une généralisation du critère de Nyquist pour les MASs. En effet, comme dans le critère de Nyquist classique, le nombre d'encerclement du point critique par le lieu de Nyquist de la boucle ouverte $G(j\omega)K(j\omega)$ est testé. Cependant contrairement au critère de Nyquist classique, le point critique $-1+0j$ est remplacé par un ensemble des points critiques définies par $-1/\lambda_i$. Ce qui est représenté graphiquement sur la Figure 3.1.

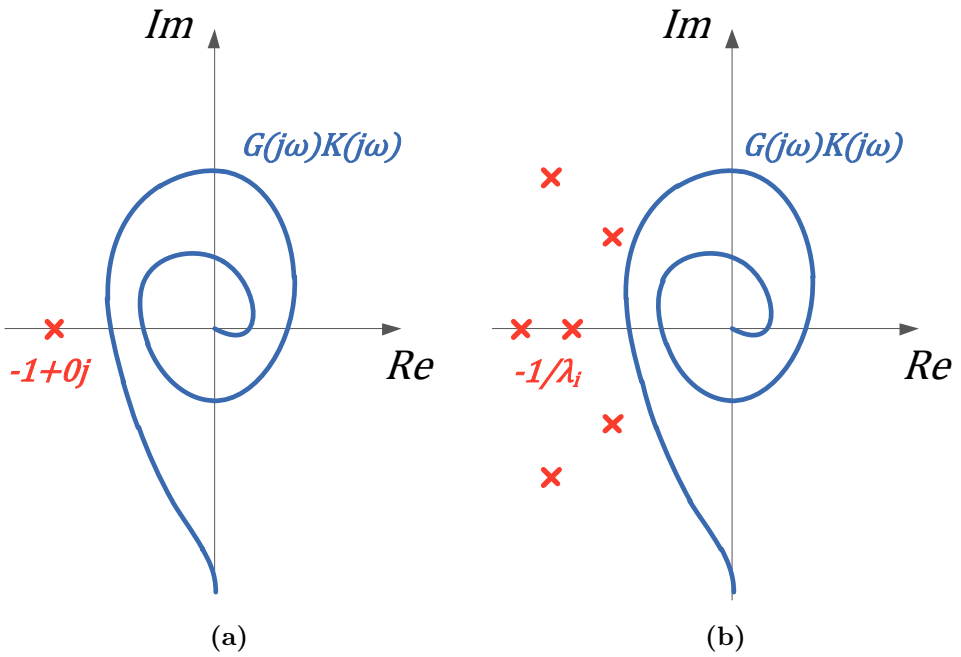


Figure 3.1: Critère de Nyquist classique (a) et généralisé pour les systèmes Multi-Agents (b)

L'idée la plus importante du résultat de Théorème 3.2 commune pour tous les autres résultats discutés par la suite, établie un lien entre les comportements global-local

du MAS. En effet, elle peut être formulée de la manière suivante : une propriété globale (stabilité globale du MAS) est transformée en des propriétés locales (stabilités simultanées) avec une information supplémentaire sur l'interconnexion (les valeurs propres de la matrice Laplacienne). Il est important de mentionner ici, qu'un sous-système local transformé du Théorème 3.2 et sa propriété de stabilité ne sont pas équivalents au sous-système local et sa propriété de stabilité définie au sens de (1.3) et la Figure 1.16, page 30. Selon le Théorème 3.2, les sous-systèmes transformés sont composés d'un rebouclage du système (3.6) avec le correcteur K et donc ils contiennent une partie dégénérée du réseau d'interconnexion c'est-à-dire les valeurs propres de la matrice d'interconnexion. L'adjectif « local » signifie ici que la propriété est satisfaite pour un sous-système transformé indépendamment des autres sous-systèmes transformés.

De manière générale, une propriété locale peut être assurée beaucoup plus facilement qu'une propriété globale. Cependant, contrairement à l'approche classique locale qui consiste à négliger les interconnexions, cette information supplémentaire délivrée sous la forme des valeurs propres de la matrice d'interconnexion est utilisée pour garantir que la propriété locale est aussi conservée dans le cas global. Nous retiendrons cette idée de la transformation « globale-locale » pour une utilisation ultérieure.

Afin de faire une démonstration du résultat discuté, revenons à nos exemples numériques de la section 2.8, page 96. On applique le Théorème 3.2 pour une ADPLL synthétisée dans la section 2.8 et deux topologies d'interconnexion représentées sur la Figure 2.53, page 98. Les matrices d'interconnexion correspondantes sont :

$$L_a = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & -0.5 \\ -0.5 & 1 & -0.5 & 0 & \cdots & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & -1 & 1 \end{bmatrix}, \; L_b = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & -0.5 \\ -1 & 1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & -1 & 1 \end{bmatrix}$$

La Figure 3.2 montre les lieux de Nyquist du réseau d'ADPLLs pour les deux topologies considérées. Les résultats obtenus confirment les simulations de la section 2.8. En effet, le lieu de Nyquist de la boucle ouverte $G(j\omega)K(j\omega)$ n'encercle pas les points critiques (représentés par les cercles bleus) associés à la topologie (a), le système global est donc bien stable. D'un autre côté, le même lieu encercle les deux points critiques (représentés par les étoiles rouges) correspondant à la topologie (b). Ceci est mis en évidence sur la partie agrandie du lieu et confirmé par les simulations effectuées dans la section 2.8.

Assez souvent dans la littérature des systèmes Multi-Agents, les auteurs traitent des problèmes de consensus ou de synchronisation des agents. Ces problèmes sont pertinents dans le cas de notre application. On dit que le consensus d'un réseau d'agents est atteint si leurs états ou leurs sorties après une période transitoire converge asymp-

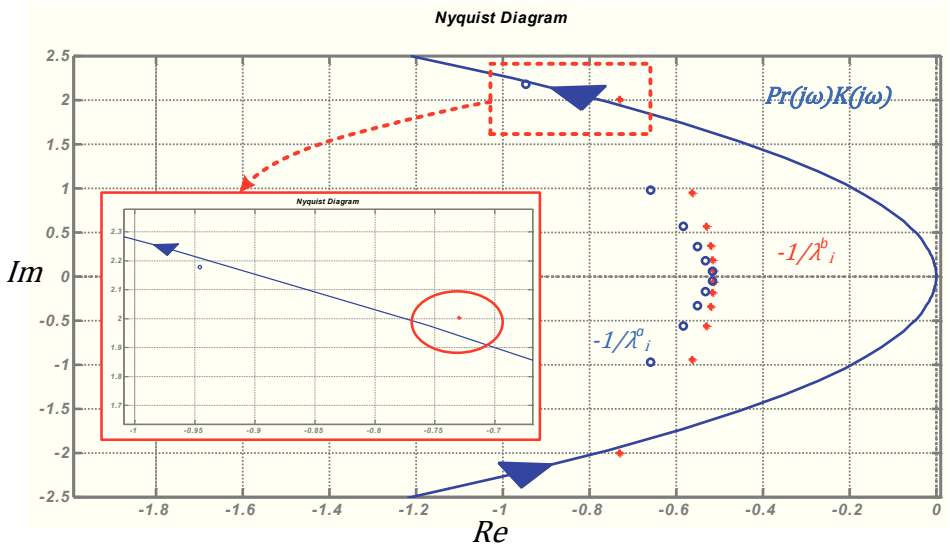


Figure 3.2: Lieux de Nyquist d'un réseau d'ADPLLs pour les deux topologies de la Figure 2.53

totiquement vers la même valeur [2]. On dit également que le réseau est synchronisé si certains événements au sein des agents se produisent aux mêmes instants du temps [3]. Les deux types de problèmes évoqués sont souvent transformés en un problème de stabilité équivalent [WKSA08, Wie10] et donc le résultat de [FM04] peut être appliqué.

Cependant, les propriétés de synchronisation ou de consensus sont des propriétés de performance du système. En effet, elles traduisent une capacité du système global ou des sous-systèmes locaux de suivre une ou plusieurs références. Il est donc nécessaire hormis la stabilité de garantir certaines conditions supplémentaires :

1. La première condition est la condition de connexité du graphe décrivant l'interconnexion du réseau de Multi-Agents : pour pouvoir atteindre le consensus ou la synchronisation du réseau de Multi-Agents, le graphe doit être connexe (voir Lemme 3.5, Théorème 3.11 et Théorème 3.15 de [Wie10] par exemple). Cette condition semble être logique car on ne peut pas assurer la synchronisation de la partie du réseau qui est déconnectée des autres au sens qu'elle ne reçoit pas l'information nécessaire à la synchronisation ou au consensus. Ceci est vrai même pour le cas où le système global est stable.
2. La deuxième condition concerne les aspects dynamiques du système. En effet, la stabilité et la connexité seules ne sont pas suffisantes pour garantir qu'un système soit capable de suivre une référence à son entrée. Même si le système est stable, après un régime transitoire, il peut posséder une certaine erreur permanente non nulle (erreur statique constante non nulle ou bien erreur de trainage non nulle, *etc.*). Un exemple d'une telle spécification de performance (pour le suivi d'une rampe) complémentaire à la stabilité est formulé dans la

2. Ou éventuellement vers des valeurs espacées l'une par rapport à l'autre par une valeur désirée.
3. Ou éventuellement aux instants du temps espacés l'un par rapport à l'autre par une valeur désirée.

section 2.5, page 60 au moment de la synthèse d'une loi de commande pour une ADPLL. En revanche, contrairement au cas local considéré dans la section 2.5, les contraintes fréquentielles associées doivent être satisfaites pour des fonctions de transfert globales. Malheureusement, cette exigence, rarement mise en évidence, est souvent négligée ou assurée indirectement par les conditions techniques formulées dans le domaine de MAS.

Le résultat de [FM04] a donné lieu à un grande nombre de résultats intéressants dans le domaine de Multi-Agents utilisant la même transformation. Les deux sous-sections suivantes donnent un aperçu des résultats principaux sur ce sujet. La plupart de ces résultats sont des résultats d'Analyse. Néanmoins il existe quelques résultats de Synthèse de loi de commande pour certains cas particuliers.

3.1.3 Résultats d'analyse de MASs

Parmi les résultats d'analyse de systèmes Multi-Agents, il y a beaucoup de résultats d'analyse de la stabilité du système global qui traitent un ensemble supplémentaire de problèmes tel que l'interconnexion non-linéaire, l'influence de retards dans le réseau, la commutation de l'interconnexion, *etc.*. Concernant l'analyse de la performance du système global, le taux de convergence vers un état stable est généralement la seule exigence considérée. Le Tableau 3.1 permet d'avoir une vision globale en regroupant quelques résultats importants sur ce sujet.

Publications	**Résultat**	**Basé sur**	**Commentaires**
[OSFM07, Mur07, FM04, RBA05]	Stabilité, taux de convergence	Théorie des graphes, Algèbre matricielle	Certaines classes de sous-systèmes NL, hétérogénéité, retards, topologie variant dans le temps
[CS05, CS06, CS07, Arc07, BAW08, HAW07]		Passivité, $\mathcal{L}_2$-gain	
[JKF07, KJF09]		Caractérisation *IQC*	
[Mor04]	Stabilité	Propriété de contraction	Topologie variant dans le temps, retards
[TM09, TM10]	Performance exprimée sous forme de fonctions de transferts globales	Théorie des graphes, Règle direct de Mason	Considérations sur le choix de la topologie du réseau

Table 3.1: L'état de l'art des résultats d'Analyse de l'approche de MASs

Les trois premières lignes du Tableau 3.1 traitent des questions d'analyse de la stabilité et du taux de convergence d'une certaine classe de systèmes Multi-Agents non-linéaires avec des retards dans le réseau, une commutation et une variation de

la topologie d'interconnexion dans le temps. Ces méthodes se basent principalement sur la Théorie des Graphes et l'Algèbre matricielle associée, sur les propriétés de passivité ou de $\mathcal{L}_2$-gain des agents ou bien sur une caractérisation plus générale sous forme de Contraintes Intégrales Quadratiques (Integral Quadratic Constraint, IQC en anglais). Un résultat intéressant d'analyse de la stabilité de MASs dont la topologie varie dans le temps avec de retards possibles est le travail de [Mor04] qui se base sur la propriété de contraction. L'auteur construit une fonction de Lyapunov qui prouve la stabilité du système avec une telle propriété.

Concernant le résultat d'analyse de la performance, un de résultats les plus généraux est celui de [TM09, TM10]. Les auteurs utilisent la règle directe de Mason qui permet de faire le calcul de fonctions de transfert globales du système. La forme de ces fonctions de transfert respectant un certain gabarit fréquentiel fixe un comportement temporel et donc un niveau de performance du système. En combinant les méthodes de la Théorie des graphes et le calcul des fonctions de transfert globales les auteurs donnent des limitations sur le niveau de performance maximal atteignable induites par le réseau d'interconnexion. Ainsi, certaines considérations sur le choix de la topologie d'interconnexion d'agents pour l'amélioration de la performance sont également données.

Cependant étant donné que ce sont les résultats d'analyse, les méthodes du Tableau 3.1 ne peuvent pas être utilisées dans la résolution de notre problème à savoir la conception de la loi de commande pour un réseau d'ADPLLs interconnectées. Ils peuvent néanmoins être utilisés pour analyser le comportement du Modèle C (voir la page 33) non-linéaire une fois que le système LTI global est synthétisé.

3.1.4 Résultats de synthèse de MASs

Il existe, cependant, quelques résultats qui traitent de la question de la synthèse de lois de commande pour les systèmes Multi-Agents. Nous nous intéressons ici seulement aux résultats de synthèse de la loi de commande linéaire pour un système linéaire temps invariant. Les travaux principaux sur ce sujet sont résumés dans le Tableau 3.2.

Les auteurs de la première publication [PW09] en se basant sur le résultat de la stabilisation simultanée de [FM04] et sur le Théorème 3.20 de Geršgorin sur la caractérisation spectrale de la matrice d'interconnexion représenté dans l'Annexe C page 143 de ce chapitre, donnent une condition de la stabilité du MAS robuste pour tous les types de topologies possibles. Ensuite, ils proposent de satisfaire cette condition en utilisant la commande H_∞ standard. Cependant, la condition à vérifier est très forte (conservative) et ne peut pas être exploitée de manière efficace. En effet, en appliquant directement la méthode proposée dans [PW09] sur le réseau d'ADPLLs implique la contrainte que la norme H_∞ de la fonction de sensibilité complémentaire T (définie dans la section 2.5, page 60) soit inférieure à 1. Il est bien connu [SP05] que la valeur maximale du module de la fonction de sensibilité T est supérieure ou égale à 1. Nous ne pouvons pas donc utiliser directement ce résultat.

Publications	**Propriétés assurées**	**Méthode de calcul**	**Limitations**
[PW09]	Stabilité robuste pour toutes les topologies possibles	Synthèse H_∞ standard basée sur la stabilisation simultanée	Très conservative, pas de résultat numérique satisfaisant
[WKSA08, Wie10]	Stabilité et taux de convergence	Optimisation LMI	Retour d'état statique, tous les états doivent être disponibles pour le correcteur
[SS09, WLH09, Wie10, WA10]		Pas de méthode générale de calcul	Commande basée sur l'approche d'observateurs, l'information supplémentaire doit être transmise
[ZOIK09b, ZOIK09a, ZOIK10, IZF96, ZIF01]		Optimisation BMI	Optimisation non convexe, point d'initialisation nécessaire
[LJDY09, LJ10]	Stabilité et performance par minimisation de la norme H_∞ d'une fonction de transfert spécifique	Optimisation LMI	Interconnexion symétrique, résultat conservatif

Table 3.2: L'état de l'art des résultats de Synthèse de l'approche de MASs

Une autre approche de la synthèse poursuit l'idée de la stabilisation simultanée proposée par [FM04]. Les auteurs de [WKSA08, Wie10] utilisent la commande par retour d'état statique comme une solution du problème d'optimisation sous contraintes d'inégalités matricielles affines (LMIs). Ce résultat couvre seulement le cas où tous les états relatifs des agents communiquant entre eux sont disponibles pour leurs correcteurs c'est-à-dire commande par le retour d'état statique. Autrement, la commande basée sur l'utilisation d'observateurs est proposée dans [SS09, WLH09, Wie10, WA10]. Cependant, les auteurs de cette approche ne proposent pas de méthodes efficaces de calcul de la loi de commande, la méthode ne garantit pas non plus des critères de performance sauf l'exigence sur le temps de convergence. En plus, pour la réalisation de commande proposée dans [SS09, WLH09, Wie10, WA10] un réseau de communication d'agents capable de transmettre une information arbitraire (les états des correcteurs et observateurs) est nécessaire ce qui peut être contraignant selon l'application. Dans l'application considérée dans la présente thèse, seuls les si-

gnaux de sorties d'oscillateur (ou de manière équivalente leurs phases respectives) peuvent être transmis.

Le quatrième résultat couvrant le cas de la synthèse de la loi de commande est la méthode basée sur l'optimisation sous contraintes d'inégalités matricielles bilinéaires (ou Bilinear Matrix Inequalities, BMI en anglais) de [ZOIK09b, ZOIK09a, ZOIK10]. Les auteurs résolvent le problème d'optimisation sous contraintes BMI par la méthode de l'Homotopie initialement proposée dans [IZF96] et développée dans [ZIF01]. Malheureusement, ce problème d'optimisation n'est pas un problème d'optimisation convexe et donc l'algorithme de résolution n'est pas efficace.

Un autre résultat de synthèse de la loi de commande dynamique garantissant la stabilité et un niveau de performance après minimisation de la norme H_∞ de la fonction de transfert globale avec des entrées et des sorties spécifiques, est le résultat proposé dans [LJDY09, LJ10]. Cependant les auteurs de [LJDY09, LJ10] considèrent une topologie d'interconnexion symétrique et une spécification de performance concernant le rejet de perturbation en entrée de chaque agents sur une sortie spécifique. Si la première hypothèse peut être relaxée en considérant une version augmentée complexe des LMIs utilisées dans [LJDY09, LJ10], la deuxième hypothèse limite le choix des signaux en entrée et en sortie nécessaire pour assurer les spécifications de performance plus générales dépendant de l'application. En outre, la solution présentée dans [LJDY09, LJ10] semble être fortement conservative. En effet, la matrice Lyapunov commune à tous les sous-systèmes est choisie avec une structure *bloc-diagonale* ce qui est nécessaire afin de reconstruire les matrices d'états du correcteur local dynamique.

L'étude de l'état de l'art de l'approche des systèmes Multi-Agents a démontré la possibilité de transformation des conditions globales assurant certaines propriétés globales vers des conditions locales nécessaires à satisfaire pour garantir les propriétés globales. Cependant, dans ce domaine, il n'existe pas actuellement de méthodes de synthèse de loi de commande décentralisée qui puissent être appliquées de manière efficace pour assurer la stabilité et un niveau de performance générale du système global. Pour ces raisons, nous nous tournons vers l'approche de Commande Décentralisée afin de résoudre le problème.

3.2 Commande Décentralisée

3.2.1 Définitions

Contrairement à l'approche des systèmes Multi-Agents, les problèmes de la synthèse de loi de commande efficace garantissant à la fois la stabilité globale et un niveau de performance ne se limitant pas à la convergence simple ont été étudiés dans le domaine de la commande décentralisée de systèmes de grande dimension [SVAS78, Lun91, LM76, MH78, HS94, SD01]. Un système de grande dimension est un système qui peut être modélisé par une interconnexion de sous-systèmes élémentaires représentés sur la Figure 3.3. Le bloc $\widetilde{T}$ représente un ensemble de sous-systèmes T_i et le bloc M leur interconnexion. La commande décentralisée consiste en

une synthèse de lois de commande locales réalisées par les correcteurs locaux K_i de chaque sous-système, remplissant le deux objectif globaux : la stabilité et un niveau de performance du système global de grande dimension.

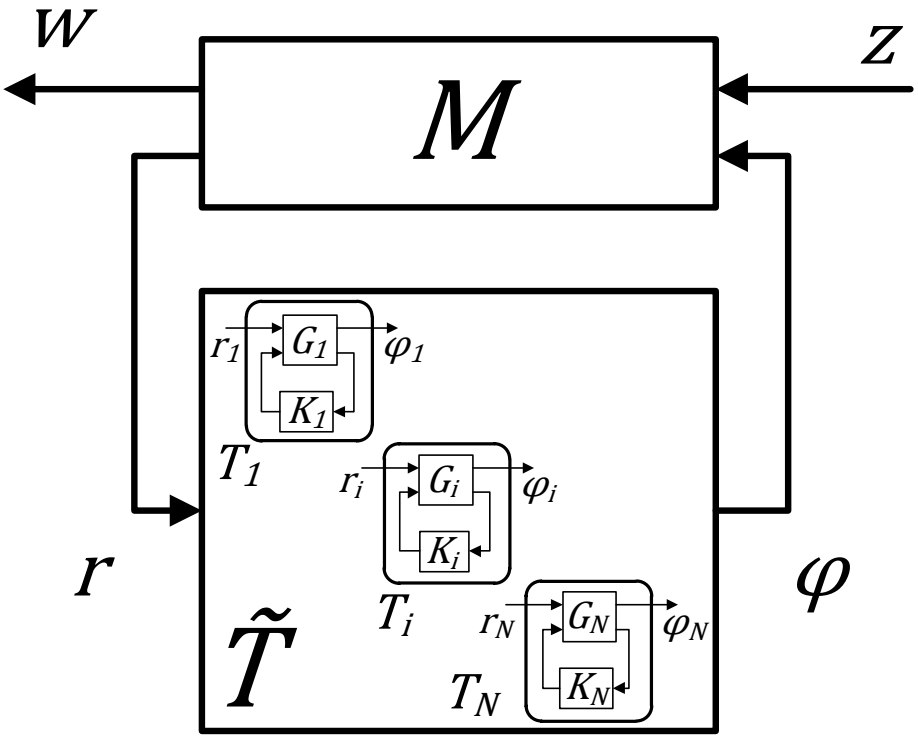


Figure 3.3: Système de grande dimension

Ils existent plusieurs approches permettant de résoudre le problème de commande décentralisée. L'une des approches la plus prometteuse, sur quelle se base l'étude de cette thèse, est l'approche entrée-sortie. Ce choix est motivé par le fait que cette approche est une extension naturelle de la commande H_∞ dont les raisons d'utilisation pour l'application de la synchronisation d'ADPLLs ont été mises en évidence dans le chapitre 2, page 60.

L'approche entrée-sortie a commencé à se développer au début des années 60 et continue son développement jusqu'à nos jours. Selon cette approche, un système dynamique est décrit non pas par sa représentation interne (des équations différentielles, représentation d'états) mais par une relation ou une contrainte entre ses signaux d'entrée et de sortie. On dit aussi dans ce cas-là que le système est caractérisé par cette contrainte que nous allons appeler par la suite la dissipativité. Quand il s'agit d'une étude de systèmes à travers leurs signaux d'entrée et de sortie, il est indispensable de les modéliser mathématiquement et de définir les opérations mathématiques associées. L'Annexe D, page 143 de ce chapitre donne un aperçu sur les différentes définitions de ce sujet.

Il y a beaucoup de résultats de la synthèse de la loi de commande décentralisée se basant sur une des deux caractérisations de dissipativité particulières : *(i)* la *passivité* ou *(ii)* $\mathcal{L}_2$*-gain*. Il existe, néanmoins, une caractérisation de dissipativité plus générale qui a été utilisée pour la première fois dans l'analyse de la stabilité

d'un système de grande dimension dans [MH78]. Étant donné sa généralité, nous allons utiliser cette caractérisation de dissipativité dans la suite de la thèse :

Définition 3.4. (Dissipativité) Un système G avec l'entrée u et la sortie y est ***strictement*** $\{X, Y, Z\}-$***dissipatif***, s'ils existent un scalaire réel $\varepsilon > 0$, des matrices réelles $X = X^T$, Y, $Z = Z^T$ de dimension appropriée telles que la matrice $\begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix}$ soit de rang plein, et pour tout $\tau > 0$ et tous signaux u et y tels que $y = G(u)$ on ait :

$$\begin{aligned} \int_{-\infty}^{\tau} \begin{bmatrix} u(t) \\ y(t) \end{bmatrix}^T \begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix} \begin{bmatrix} u(t) \\ y(t) \end{bmatrix} dt \quad &\leq \quad \cdots \\ \cdots \quad &-\varepsilon \int_{-\infty}^{\tau} \begin{bmatrix} u(t) \\ y(t) \end{bmatrix}^T \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} u(t) \\ y(t) \end{bmatrix} dt \end{aligned} \tag{3.7}$$

si l'inégalité (3.7) est satisfaite avec $\varepsilon = 0$, le système est appelé tout simplement $\{X, Y, Z\}-$ ***dissipatif***.

La définition de la dissipativité (Définition 3.4) peut être utilisée aussi bien pour les systèmes linéaires que non-linéaires comme dans [Wil72, Kha92, SJK97]. En effet, la condition (3.7) (avec le produit scalaire tronqué $\langle u, y \rangle_\tau$ tel que défini dans l'annexe D de ce chapitre) peut être facilement transformée en la condition suivante :

$$\langle u, Xu \rangle_\tau + 2 \langle u, Yy \rangle_\tau + \langle y, Zy \rangle_\tau \leq -\varepsilon \left(\langle u, u \rangle_\tau + \langle y, y \rangle_\tau \right) \tag{3.8}$$

où on voit naturellement apparaître la *fonction de taux d'approvisionnement* w qui peut être définie en fonction de X, Y et Z comme :

$$w(t) = -u^T(t) Xu(t) - 2y^T(t) Yu(t) - y^T(t) Zy(t) \tag{3.9}$$

On peut démontrer [Wil76] que si un système ayant une représentation d'état de vecteur d'état x est $\{X, Y, Z\}-$dissipatif ou autrement dit dissipatif avec un taux d'approvisionnement (3.9), il existe alors une fonction $\Psi(x)$ telle que $\Psi(0) = 0$, $\Psi(x) \geq 0 \ \forall x$ et

$$\Psi(x(t_0)) + \int_{t_0}^{t_1} w(t)\, dt \geq \Psi(x(t_1)).$$

Dans le cas où deux signaux u et y sont d'énergie finie, en définissant un nouveau signal $f = \begin{bmatrix} u^T & y^T \end{bmatrix}^T$ la condition (3.7) peut être réécrite dans le domaine fréquentiel

après l'application du Théorème Parseval-Plancherel généralisé pour presque tout $\omega \in \mathbb{R}^+$:

$$\begin{aligned}\int_{-\infty}^{+\infty} \begin{bmatrix} u(j\omega) \\ y(j\omega) \end{bmatrix}^* \begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix} \begin{bmatrix} u(j\omega) \\ y(j\omega) \end{bmatrix} d\omega &\leq \cdots \\ \cdots &\quad -\varepsilon \int_{-\infty}^{+\infty} \begin{bmatrix} u(j\omega) \\ y(j\omega) \end{bmatrix}^* \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} u(j\omega) \\ y(j\omega) \end{bmatrix} d\omega \end{aligned} \tag{3.10}$$

Si en plus le système G est un système LTI stable et causal, l'équation (3.10) peut être simplifiée pour presque tout $\omega \in \mathbb{R}^+$ en l'expression suivante :

$$\begin{bmatrix} I \\ G(j\omega) \end{bmatrix}^* \begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix} \begin{bmatrix} I \\ G(j\omega) \end{bmatrix} \leq -\varepsilon \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}, \tag{3.11}$$

Les deux cas particuliers de dissipativité mentionnés avant sont obtenus par un choix particulier de matrices X, Y et Z présenté ci-dessous :

1. Passivité : pour $X = 0$, $Y = -I$, $Z = 0$ et pour $\varepsilon = 0$ de (3.7), on obtient :

$$\int_{-\infty}^{\tau} u(t)\, y(t)\, dt \quad \geq \quad 0$$

   c'est-à-dire le produit scalaire de signaux en entrée et à la sortie du système passif est non-négatif.

2. $\mathcal{L}_2$-gain : pour $X = I$, $Y = 0$, $Z = -\gamma^2 I$ et pour $\varepsilon = 0$ de (3.7), on obtient :

$$\|G\|_{\mathcal{L}_2} \quad \leq \quad \gamma$$

   ou, autrement dit, la norme $\mathcal{L}_2$ du système G est bornée par ou égale à la valeur γ.

L'idée de base de l'approche de Commande Décentralisée peut se résumer de la manière suivante. Le système de grande dimension est mis sous forme d'interconnexion $M \star \widetilde{T}$ de deux systèmes : le système d'interconnexion M et le système $\widetilde{T}$ regroupant les sous-systèmes locaux T_i sur sa diagonale (voir la Figure 3.3). On cherche alors les conditions sur les signaux d'entrée et de sortie de chaque sous-système de $\widetilde{T}$ et de l'interconnexion M pour lesquelles la stabilité est garantie. Ensuite, ces conditions sont satisfaites par une synthèse appropriée des correcteurs locaux K_i de chaque sous-système locale T_i.

La recherche de conditions assurant la stabilité est réalisée à l'aide du principe de séparation des graphes [4]. Selon ce principe il est possible de démontrer la stabilité du système global $M \star \widetilde{T}$ en « séparant » les caractéristiques de dissipativité de chaque sous-bloc M ou $\widetilde{T}$ de l'interconnexion globale $M \star \widetilde{T}$. La démonstration de ce

4. Contrairement à la représentation graphique d'une interconnexion dans la Théorie de graphe, la notion d'un graphe dans le contexte de l'approche entrée-sortie n'est pas la même. Dans ce contexte, un graphe est une relation entre les signaux d'entrée et les signaux de sortie d'un opérateur.

principe avec la caractérisation de $\{X, Y, Z\}$ −dissipativité pour le cas de l'analyse de la stabilité d'un système de grande dimension peut être retrouvée dans [MH78]. Concernant le résultat de la synthèse de la loi de commande décentralisée assurant non seulement la stabilité mais aussi un niveau de performance désiré du système de grande dimension, le principe de séparation des graphes dans le cas de la caractérisation de $\{X, Y, Z\}$ −dissipativité est reporté dans [SD01]. Les deux sous-sections suivantes présentent le Théorème de séparation des graphes et son utilisation pour la commande décentralisée de systèmes de grande dimension.

3.2.2 Théorème de séparation des graphes

Un des résultats principaux utilisé dans l'approche entrée-sortie est le Théorème de séparation des graphes qui est en réalité une généralisation du Théorème de petit gain et du Théorème de la passivité. Il existe plusieurs versions de ce Théorème. Par exemple, une version avec une démonstration élégante et simple peut être trouvée dans [MH78].

Considérons un système représenté sous forme d'une interconnexion (3.12) de deux sous-systèmes G et Δ (voir la Figure 3.4) avec une entrée extérieure u_e :

$$\begin{aligned} y &= G(u) \\ u &= \Delta(y) + u_e \end{aligned} \tag{3.12}$$

avec u,$u_e \in \mathcal{U}_e^m$, $y \in \mathcal{U}_e^p$.

Dans la suite de ce document, nous allons avoir besoin de la définition suivante prise de [Sco97] :

Définition 3.5. L'interconnexion (3.12) est *bien posée* si le système est causal et que pour $\forall u_e$, $u_e \in \mathcal{L}_{2e}$, il existe une solution u, $y \in \mathcal{L}_{2e}$ et que cette solution est unique. Si en plus à un signal d'énergie fini à l'entrée u_e tous les signaux internes du système y et u sont d'énergie finie selon la Définition 2.3, page 61, le système est *stable de façon interne.*

Maintenant on peut formuler le Théorème de séparation des graphes.

Théorème 3.6. *Supposons que le système* (3.12) *représenté sur la Figure 3.4 soit bien posé. S'ils existent un scalaire réel $\varepsilon > 0$, des matrices réelles $X = X^T \leq 0$, Y, $Z = Z^T$ de dimensions appropriées telles que :*

(i) le sous-système G est strictement $\{X, Y, Z\}$ −dissipatif ;

(ii) le sous-système Δ est $\left\{-Z, -Y^T, -X\right\}$ −dissipatif,

le système global (3.12) *est alors stable au sens entrée-sortie de façon interne avec l'entrée u_e et les sorties y et u.*

Démonstration. La condition de dissipativité *(ii)* du Théorème 3.6 entraîne que pour u et y tels que $u = \Delta(y)$:

$$\langle (u - u_e), X(u - u_e) \rangle_\tau + 2 \langle (u - u_e), Yy \rangle_\tau + \langle y, Zy \rangle_\tau \geq 0$$

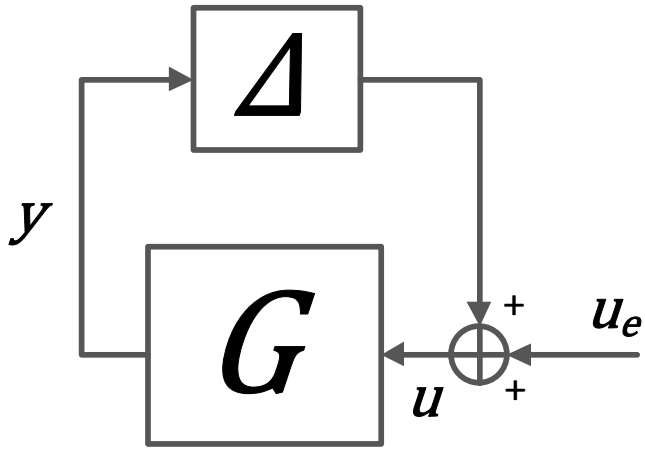


Figure 3.4: Interconnexion de deux sous-systèmes

soit

$$\langle u, Xu\rangle_\tau + 2\langle u, Yy\rangle_\tau + \langle y, Zy\rangle_\tau \geq 2\langle u_e, Yy\rangle_\tau + 2\langle u_e, Xu\rangle_\tau - \langle u_e, Xu_e\rangle_\tau . \quad (3.13)$$

La condition de dissipativité *(i)* du Théorème 3.6 implique à son tour qu'il existe un scalaire réel $\varepsilon > 0$ tel que pour u et y tels que $y = G(u)$:

$$\langle u, Xu\rangle_\tau + 2\langle u, Yy\rangle_\tau + \langle y, Zy\rangle_\tau \leq -\varepsilon(\langle u, u\rangle_\tau + \langle y, y\rangle_\tau)$$

soit

$$-\langle u, Xu\rangle_\tau - 2\langle u, Yy\rangle_\tau - \langle y, Zy\rangle_\tau \geq \varepsilon(\langle u, u\rangle_\tau + \langle y, y\rangle_\tau) \quad (3.14)$$

En sommant les deux conditions (3.13) et (3.14), nous obtenons donc pour tout τ :

$$\varepsilon\langle u, u\rangle_\tau + \varepsilon\langle y, y\rangle_\tau + 2\langle u_e, Yy\rangle_\tau + 2\langle u_e, Xu\rangle_\tau \leq \langle u_e, Xu_e\rangle_\tau \leq 0$$

car $X \leq 0$. En multipliant la dernière condition par $\varepsilon > 0$ et en utilisant la complétion des carrés, nous obtenons pour tout τ :

$$\langle(\varepsilon u + Xu_e), (\varepsilon u + Xu_e)\rangle_\tau + \left\langle\left(\varepsilon y + Y^T u_e\right), \left(\varepsilon y + Y^T u_e\right)\right\rangle_\tau \leq \left\langle Y^T u_e, Y^T u_e\right\rangle_\tau + \langle Xu_e, Xu_e\rangle_\tau$$

Ce qui implique, pour tout τ :

$$\begin{aligned} \|\varepsilon u + Xu_e\|_\tau &\leq \left\|Y^T u_e\right\|_\tau + \|Xu_e\|_\tau \\ \left\|\varepsilon y + Y^T u_e\right\|_\tau &\leq \left\|Y^T u_e\right\|_\tau + \|Xu_e\|_\tau \end{aligned}$$

Ainsi, d'après l'inégalité triangulaire $\|\varepsilon u\|_\tau \leq \|\varepsilon u + Xu_e\|_\tau + \|Xu_e\|_\tau$ et $\|\varepsilon y\|_\tau \leq$

$$\left\|\varepsilon y + Y^T u_e\right\|_\tau + \left\|Y^T u_e\right\|_\tau :$$

$$\begin{aligned} \|u\|_\tau &\leq \frac{1}{\varepsilon}\left(\left\|Y^T u_e\right\|_\tau + 2\left\|X u_e\right\|_\tau\right) \\ \|y\|_\tau &\leq \frac{1}{\varepsilon}\left(2\left\|Y^T u_e\right\|_\tau + \left\|X u_e\right\|_\tau\right) \end{aligned} \tag{3.15}$$

Donc si l'entrée $u_e \in \mathcal{L}_{2e}$ alors les signaux de sorties $y \in \mathcal{L}_{2e}$ et $u \in \mathcal{L}_{2e}$. Le système bouclé est donc stable de façon interne. □

Remarque 3.7. Il faut noter que les conditions du Théorème 3.6 s'applique aussi aux sous-systèmes non-linéaires dont les seules exigences sur la dynamique sont les conditions de dissipativité.

Remarque 3.8. Le résultat équivalent pour le cas avec deux signaux en entrée peut être retrouvé dans Lemme 3.2.1 de [Sco97]. Pour des raisons de clarté et de simplicité, nous avons exigé que la matrice de dissipativité X soit semi-définie négative ($X \leq 0$). Il est cependant possible de démontrer le même résultat sans exigences supplémentaires sur la définition des matrices de dissipativité (voir le Théorème 1 de [MH78]) où même dans la cas où les matrices X, Y, Z sont elles-mêmes des systèmes dynamiques (voir le Théorème 1 de [MR97]). Dans ce cas-là, la contrainte de dissipativité définie une IQC.

Remarque 3.9. Deux cas particuliers du Théorème 3.6 particulièrement intéressant sont le Théorème de passivité qui est facilement obtenu pour $X = 0$, $Y = -I$, $Z = 0$ et le Théorème de petit gain pour $X = -\gamma^2 I$, $Y = 0$, $Z = I$, $\gamma \in \mathbb{R}$.

3.2.3 Commande décentralisée des systèmes de grande dimension

Comme cela a été déjà noté, il est possible d'utiliser le Théorème de séparation des graphes (Théorème 3.6) pour synthétiser une loi de commande assurant la stabilité du système de grande dimension (voir la Figure 3.3). En effet, si l'on trouve une caractérisation de la dissipativité du sous-système M décrivant l'interconnexion entre les sous-systèmes locaux, il est possible d'assurer la stabilité du système global en assurant que le bloc $\widetilde{T}$ regroupant les sous-systèmes locaux T_i satisfasse la condition de dissipativité complémentaire. Ensuite, étant donné que le sous-système $\widetilde{T}$ est bloc-diagonal, il est facile de transformer la contrainte de dissipativité sur $\widetilde{T}$ (suffisante pour la stabilité) en un ensemble de contraintes de dissipativité sur T_i à satisfaire par les correcteurs locaux K_i afin d'assurer la stabilité globale.

Le problème devient maintenant le suivant : comment trouver les conditions de dissipativité qui conviennent et comment les assurer par des correcteurs locaux ? A l'époque des années 60-70, l'époque où le Théorème de séparation des graphes a été inventé, il n'existait pas encore de réponses à ces questions.

Grâce aux récents développements des outils d'optimisation convexe sous contraintes LMIs dont un aperçu est donné dans la sous-section 2.9 et de la synthèse H_∞

([GA94]), la résolution des problèmes posés est devenue possible. Comme cela a été déjà mentionné, une de méthodes la plus prometteuse de réalisation pour la synthèse de la loi de commande pour des systèmes de grande dimension, est la méthode proposée dans [SD01]. Les auteurs de [SD01] en utilisant l'optimisation LMI et la synthèse H_∞ proposent une méthode de synthèse de correcteurs locaux K_i (voir la Figure 3.3) assurant la stabilité globale du système. En outre, en exprimant la performance sous forme de stabilité équivalente (comme cela a été fait dans la sous-section 2.6.2 du Chapitre 2) les correcteurs synthétisés peuvent assurer aussi un niveau de performance globale du réseau sous forme d'une minimisation de la norme H_∞ de la fonction de transfert globale entre les entrées w et les sorties z globales externes.

Cependant, étant donné que les sous-systèmes considérés dans [SD01] ne sont pas forcément identiques, la complexité de la condition de la synthèse dépend fortement du nombre de sous-systèmes. Ceci amène potentiellement à un problème d'optimisation de grande taille étant donné que pour chaque sous-système considéré, un ensemble de conditions LMI doit être introduit. En outre, afin d'obtenir une méthode de synthèse efficace, l'approche proposée dans [SD01] se base sur des conditions seulement suffisantes, c'est-à-dire, des conditions donnant un résultat potentiellement conservatif.

Conclusion

Nous avons illustré dans ce chapitre les deux approches pour étudier les systèmes de grande dimension : les systèmes Multi-Agents et le Commande Décentralisée. Traitant à la base deux problèmes différents, ces deux approches, dans l'optique de notre application de la conception du réseau actif de distribution du signal d'horloge, semblent être toutes les deux intéressantes, ayant chacune leurs propres avantages et inconvénients.

Ainsi, l'approche des systèmes Multi-Agents, en utilisant de manière efficace l'information sur l'interconnexion et sur l'identité des sous-systèmes donne une solution au problème de la stabilité globale du système de grande dimension. En effet, comme cela a été discuté dans ce chapitre, la stabilité du système global est transformée en stabilité simultanée de sous-systèmes transformés comportant dans sa boucle de rétroaction des valeurs propres de la matrice d'interconnexion.

Cependant, il n'existe pas dans la littérature des systèmes Multi-Agents de méthodes de synthèse de loi de commande efficace garantissant en plus de la stabilité les spécifications générales de performance globale. Ceci est probablement lié au fait que dans cette approche la transformation des conditions globales vers des conditions locales se base sur la stabilisation simultanée. Il est bien connu que le problème de stabilisation simultanée pour au moins trois systèmes est un problème difficile et qu'il n'existe pas, et ne peut pas exister, de conditions nécessaires et à la fois suffisantes pour sa résolution. Une étude approfondie de cette question peut être trouvée dans [BG93].

D'un autre côté, l'approche de la Commande Décentralisée, grâce au Théorème de

séparation des graphes et à l'optimisation LMI, permet de manière efficace, sans faire appel à la stabilisation simultanée, et en dépit du fait que les conditions à satisfaire sont aussi seulement suffisantes, de synthétiser la loi de commande assurant aussi bien la stabilité globale que les spécifications générales de la performance globale.

Cependant, la complexité et le conservatisme des conditions pour la stabilité et la performance globales sont assez importants surtout pour le réseau avec un grand nombre de sous-systèmes. Les automaticiens traitant l'approche de la Commande Décentralisée n'ont jamais utilisé le fait qu'il soit possible de réduire la complexité ainsi que le conservatisme potentiel des conditions en utilisant l'information sur l'identité de sous-systèmes en réseau. Autrement dit, les sous-systèmes ont toujours été considérés comme des sous-systèmes différents avec des correcteurs différents.

Cependant, en pratique, selon l'application considérée, les sous-systèmes sont souvent identiques ($T_1 = T_2 = \cdots = T$, voir la Figure 3.3) dans le cas nominal ou au moins ceux qui ont la même structure. Ceci est également le cas pour un réseau d'ADPLLs interconnectées constituant le réseau actif de distribution du signal d'horloge. Cette information importante sur la structure des sous-systèmes peut être exploitée afin de réduire la complexité et le conservatisme de la synthèse de commandes décentralisées. Évidemment, il existe toujours des dispersions technologiques inévitables lors de la fabrication faisant de sort que les sous-systèmes ne soient jamais identiques. Néanmoins, le cas des dispersions peut toujours être pris en compte après la synthèse de la loi de commande dans le cas nominal, en appliquant les méthodes d'analyse de robustesse de modèle obtenu vis-à-vis de ces dispersions, comme cela a été déjà fait dans le cas d'une seule ADPLL (voir la section 2.6, chapitre 2).

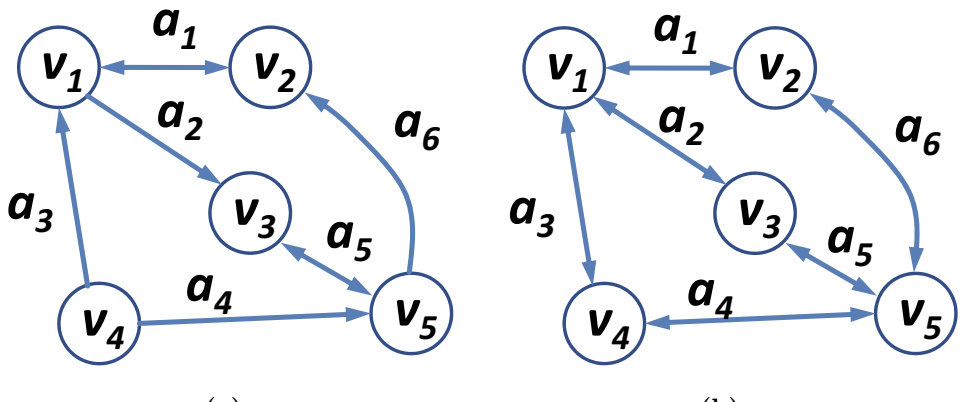
La mise en place et la réalisation d'une telle démarche appliquée à la synthèse de la loi de commande et à l'analyse de la robustesse d'un réseau d'ADPLLs sont présentés dans les chapitres suivants.

3.3 Annexe C : Théorie des Graphes et Théorie algébrique de matrices

3.3.1 Définitions générales

La Théorie des graphes est aujourd'hui largement utilisée comme un outil mathématique dans les différents domaines des Sciences, tels que : la Physique, la Chimie, l'Électronique, les Communications, le Transport, *etc.*. Son existence date de 1736 où Euler démontra à l'aide d'un « dessin » schématisé que le célèbre problème des ponts de Königsberg ne possède pas de solutions [Eul36]. Depuis lors, la Théorie des graphes est devenu une véritable science, indispensable pour l'étude des systèmes complexes interconnectés en réseau. Un système Multi-Agents est un exemple de système interconnecté en réseau. Dans cette annexe, on présente un petit aperçu sur des définitions et des résultats principaux de la Théorie des graphes. Une description plus complète de la Théorie algébrique des graphes peut être trouvée dans [Big94] et [HJ85]. Les définitions de cette sous-section sont généralement prises de [Maq03].

Définition 3.10. On appelle ***graphe orienté*** ou ***digraphe*** $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ la donnée qui consiste d'un ensemble $\mathcal{V} = \{v_1, v_2, ..., v_n\}$ dont les éléments sont appelés *sommets* et d'un ensemble $\mathcal{A} = \{a_1, a_2, ..., a_m\} \subset \mathcal{V} \times \mathcal{V}$ dont les éléments sont appelés *arcs* ou *arêtes*. Un arc a de l'ensemble $\mathcal{A}$ est défini par une paire *ordonnée* de sommets. Lorsque $a = (x, y)$, on dira que l'arc a va de x à y. On dit que x est *l'origine* (ou *extrémité initiale*) et y *l'extrémité* (*finale*) de a, que a est *sortant* en x et *incident* en y et que y est un *successeur* de x tandis que x est *prédécesseur* de y. On dit aussi que x et y sont *adjacents*. Le nombre de sommets est appelé *l'ordre* du graphe. Un graphe ayant la propriété que pour toutes les paires de sommets appartenant au graphe $(x, y) \in \mathcal{A}$ et que l'arc inversé, lui aussi, appartient au graphe $(y, x) \in \mathcal{A}$ est un ***graphe non-orienté***. La Fig.3.5 montre une représentation graphique des deux types de graphe : orienté (a) et non-orienté (b).



(a) (b)

Figure 3.5: Représentation graphique d'un graphe orienté (a) et non-orienté (b)

Définition 3.11. On appelle *degré sortant* ou *demi-degré extérieur* d'un sommet x le nombre d'arcs de la forme $a = (x, y)$ avec $\forall y \neq x$ et on le note : $d_s(x)$. On appelle *degré entrant* ou *demi-degré intérieur* d'un sommet x le nombre d'arcs de la forme $a = (y, x)$ avec $\forall y$, $y \neq x$ et on le note : $d_e(x)$. On appelle *degré* de x (ou *valence*) la somme du degré entrant et du degré sortant. Un sommet du graphe dont le degré entrant est nul est nommé un *sommet leader*.

Un graphe ne possédant pas de boucle ni d'arc parallèle (deux arcs distincts joignant la même paire de sommets) est appelé ***graphe simple*** ou ***1-graphe***. Un ***graphe complet*** est un graphe dont tous les sommets sont adjacents ou, autrement dit, dont tous les arcs possibles existent c'est-à-dire $\mathcal{A} = \mathcal{V} \times \mathcal{V}$.

Définition 3.12. Un *chemin* dans un digraphe est une suite de sommets reliés les uns aux autres par des arcs. La *longueur* du chemin est le nombre d'arcs utilisés, soit le nombre de sommets moins un. Un *chemin simple* ne peut pas visiter le même sommet plus d'une fois. Un *chemin fermé* ou tout simplement un *cycle* est un chemin ayant pour dernier sommet le premier. Un cycle possédant un seul sommet (le premier et donc le dernier) est appelé un *cycle local*. Un graphe dont chaque paire de sommets est connecté par au moins un chemin est appelé un ***graphe fortement connexe***. Un graphe qui a au moins un sommet qui peut être connecté aux autres sommets par au moins un chemin est un ***graphe connexe***. Pour une démonstration de ces propriétés de graphe, considérons le graphe représenté sur la Figure 3.6a. Ce graphe est connexe car le sommet v_6 est connecté par différents chemins à tous les autres sommets. En revanche, il n'est pas fortement connexe car il possède des sommets non connectés aux autres comme par exemple, le sommet v_5 qui n'est pas connecté aux sommets v_1, v_2, v_4 ni v_6. En ajoutant un arc a_9 représenté en rouge sur la Figure 3.6b le graphe obtenu devient fortement connexe car il est possible de trouver un chemin reliant tous les sommets du graphe. Un ***graphe apériodique*** est un graphe qui ne possède aucun cycle. Un graphe possédant k-cycles hormis ses cycles locaux est un graphe k-périodique.

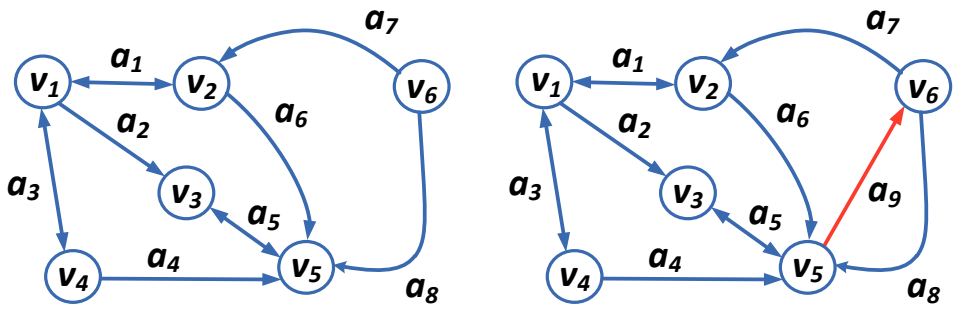


Figure 3.6: Représentation graphique d'un graphe connexe (a) et fortement connexe (b)

La relation entre la Théorie des graphes et la Théorie du contrôle repose sur les matrices associées au graphe.

Matrice d'Adjacence

Considérons un graphe $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ comportant n sommets.

Définition 3.13. La matrice d'*Adjacence* de $\mathcal{G}$ est égale à la matrice $\tilde{A} = [\tilde{a}_{ij}]$ de dimension $n \times n$ telle que :

$$\tilde{a}_{ij} = \begin{cases} 1 & \text{, si } (v_j, v_i) \in \mathcal{G} \text{ (c'est-à-dire } (v_j, v_i) \text{est un arc)} \\ 0 & \text{, sinon} \end{cases} \tag{3.16}$$

Selon le problème considéré, il est parfois plus intéressant d'utiliser la matrice d'Adjacence normalisée.

Définition 3.14. La matrice d'*Adjacence normalisée* de $\mathcal{G}$ est égale à la matrice $A = [a_{ij}]$ de dimension $n \times n$ telle que :

$$a_{ij} = \begin{cases} \frac{1}{d_e(v_i)} & \text{, si } (v_j, v_i) \in \mathcal{G} \\ 0 & \text{, sinon} \end{cases} \tag{3.17}$$

Cette définition de la matrice d'Adjacence adoptée dans notre étude est légèrement différente de celle retenue de la Théorie des graphes classique [Big94, HJ85]. Une matrice d'Adjacence classique (normalisée ou pas) est égale à la matrice transposée de celle définie en (3.16) ou (3.17). N'étant finalement pas différente sur le fond, la nouvelle définition permettra une modélisation et une représentation graphique du réseau de Multi-Agents et de son échange d'information plus claires et logiques du point de vue de l'Automatique. En effet, avec les définitions (3.16) ou (3.17), la situation où l'agent j envoie l'information à l'agent i est modélisée par un simple arc (v_j, v_i) orienté de l'agent j vers l'agent i.

Selon les Définitions 3.13 et 3.14, la somme des éléments de la $i^{ème}$ ligne de $\tilde{A}$ ou A est égale au degré entrant $d_e(v_i)$ du sommet x_i, ou à 1. La somme des éléments de la $j^{ème}$ colonne de $\tilde{A}$ est égale au degré sortant $d_s(v_j)$ du sommet x_j.

Un graphe non orienté possède une matrice d'Adjacence symétrique. L'absence de cycles locaux se traduit par une diagonale nulle de la matrice d'Adjacence. On supposera par la suite que les graphes considérés dans cette étude sont simples et n'ont pas de cycles locaux. Les matrices d'Adjacence des graphes de la Fig.3.5 sont

les suivantes :

$$\tilde{A}_a = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}, \; A_a = \begin{bmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \end{bmatrix},$$

$$\tilde{A}_b = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}, \; A_b = \begin{bmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\ \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \\ 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \end{bmatrix} \tag{3.18}$$

Matrice Laplacienne

Une autre matrice très utile pour l'étude de la dynamique des réseaux des Multi-Agents est la matrice dite Laplacienne.

Définition 3.15. La matrice *Laplacienne* $\tilde{L} = \left[\tilde{l}_{i,j}\right]$ est une matrice de dimension $n \times n$ associée au graphe définie par :

$$\tilde{L} = D - \tilde{A} \tag{3.19}$$

où

$$\tilde{l}_{ij} = \begin{cases} d_e(v_i) & \text{, si } i = j \\ -1 & \text{, si } (v_j, v_i) \in \mathcal{G} \text{ et } i \neq j \\ 0 & \text{, si } (v_j, v_i) \notin \mathcal{G} \text{ et } i \neq j \end{cases}$$

et $D = \text{diag}\left(d_e(v_1), d_e(v_2), ..., d_e(v_n)\right)$ est une matrice diagonale avec les degrés entrants des sommets correspondants sur sa diagonale.

La matrice *Laplacienne normalisée* $L = [l_{ij}]$ associée au graphe est une matrice de

dimension $n \times n$ définie par :

$$L = I - A \text{ ou}$$
$$l_{ij} = \begin{cases} 1 & , i = j \\ -\frac{1}{d_e(v_i)} & , \text{si } (v_j, v_i) \in \mathcal{G} \text{ et } i \neq j \\ 0 & , \text{si } (v_j, v_i) \notin \mathcal{G} \text{ et } i \neq j \end{cases} \tag{3.20}$$

La matrice Laplacienne d'un graphe possédant des leaders est définie de la même manière que dans la deuxième partie de (3.20) en annulant les termes diagonaux l_{ii} correspondant aux sommets leaders.

Deux matrices Laplaciennes définies par (3.19) et (3.20) correspondantes au graphe représenté sur la Fig.3.5a ont la forme suivante :

$$\tilde{L}_a = \begin{bmatrix} 2 & -1 & 0 & -1 & 0 \\ -1 & 2 & 0 & 0 & -1 \\ -1 & 0 & 2 & 0 & -1 \\ -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & -1 & 2 \end{bmatrix}, \quad L_a = \begin{bmatrix} 1 & -\frac{1}{2} & 0 & -\frac{1}{2} & 0 \\ -\frac{1}{2} & 1 & 0 & 0 & -\frac{1}{2} \\ -\frac{1}{2} & 0 & 1 & 0 & -\frac{1}{2} \\ -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{2} & -\frac{1}{2} & 1 \end{bmatrix} \tag{3.21}$$

Dans la partie précédente, nous avons vu qu'un graphe peut être caractérisé par les matrices associées. Les propriétés particulières de ces matrices peuvent être très utiles dans l'étude d'un réseau de Multi-Agents interconnectés. Dans la partie suivante, nous allons fournir quelques définitions et théorèmes nécessaires pour mener une telle étude. Plus de détails sur les définitions ainsi que les démonstrations des différents théorèmes peuvent être trouvés dans [Big94, HJ85, Wie10].

3.3.2 Produit de Kronecker

Le *produit de Kronecker* est très utile pour la modélisation d'interconnexions de systèmes identiques. Dans cette sous-section, on présentera sa définition et ses propriétés principales, pour plus des détails sur son utilisation et son application, le lecteur est invité à consulter [Bre78].

Définition 3.16. Le produit de Kronecker de deux matrices A et B, noté $A \otimes B$,

définit une matrice composée de sous-matrices $a_{ij}B$, c'est-à-dire :

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{bmatrix} \tag{3.22}$$

où a_{ij}, $i \in \{1, \ldots, n\}$, $j =\in \{1, \ldots, m\}$ représente $ij^{ème}$ élément de la matrice A de dimensions $m \times n$.

Les propriétés principales du produit de Kronecker pour les matrices réelles A, B, C et D de dimensions appropriées sont rassemblées ci-dessous.

– Le produit de Kronecker est un opérateur bilinéaire. Pour un scalaire réel $\alpha \in \mathbb{R}$:

$$\begin{aligned} A \otimes (\alpha B) &= \alpha (A \otimes B) \\ (\alpha A) \otimes B &= \alpha (A \otimes B). \end{aligned} \tag{3.23}$$

– Le produit de Kronecker est un opérateur distributif par rapport à l'addition :

$$\begin{aligned} (A + B) \otimes C &= (A \otimes C) + (B \otimes C) \\ A \otimes (B + C) &= (A \otimes B) + (A \otimes C). \end{aligned} \tag{3.24}$$

– Le produit de Kronecker est un opérateur associatif :

$$(A \otimes B) \otimes C = A \otimes (B \otimes C). \tag{3.25}$$

– Le produit de Kronecker est un opérateur non commutatif :

$$A \otimes B \neq B \otimes A. \tag{3.26}$$

– Transposée d'un produit de Kronecker :

$$(A \otimes B)^T = A^T \otimes B^T. \tag{3.27}$$

– Produit matriciel de produits de Kronecker :

$$(A \otimes B)(C \otimes D) = AC \otimes BD. \tag{3.28}$$

En particulier, on a pour $A \in \mathbb{R}^{m \times m}$:

$$\begin{aligned} &(A \otimes I_n)(I_m \otimes B) = (A \otimes B) = (I_m \otimes B)(A \otimes I_n), \text{ où } B \in \mathbb{R}^{n \times n} \\ &(A \otimes I_n)(I_m \otimes B) = (I_m \otimes B) A, \text{ où } B \in \mathbb{R}^{n \times 1} \\ &(I_m \otimes C)(A \otimes I_n) = A (I_m \otimes C), \text{ où } C \in \mathbb{R}^{1 \times n} \end{aligned} \tag{3.29}$$

– Pour les matrices A et B carrées et de rang plein :

$$(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}. \tag{3.30}$$

– Le déterminant du produit de Kronecker est donné par :

$$\det(A_{m\times m} \otimes B_{n\times n}) = \det(A)^n \det(B)^m. \tag{3.31}$$

– La trace des produits de Kronecker est :

$$\operatorname{trace}(A \otimes B) = \operatorname{trace}(A)\operatorname{trace}(B). \tag{3.32}$$

3.3.3 Propriétés algébriques de matrices

Définition 3.17. Une *matrice unitaire* U, possiblement complexe, est une matrice dont le produit avec sa transposée conjuguée est une matrice identité : $UU^* = U^*U = I$. Dans le cas d'une matrice unitaire réelle, elle est appelée *matrice orthogonale* et la transposée conjuguée dans la définition est remplacée simplement par la transposée $UU^T = U^TU = I$.

Théorème 3.18 (Théorème de Schur complexe de triangularisation)**.** *Pour une matrice carrée A (possiblement complexe) de dimensions $n\times n$ (désignation $A \in M_n$) avec les valeurs propres $\lambda_1, \lambda_2, \ldots, \lambda_n$ non ordonnées, il existe une matrice carrée unitaire U telle que $U \in M_n$ et :*

$$U^*AU = T = [t_{ij}]$$

est une matrice triangulaire supérieure, avec les éléments diagonaux $t_{ii} = \lambda_i, i = 1, \ldots, n$. Autrement dit, toutes les matrices carrées A sont équivalentes, à travers la transformation d'une matrice unitaire, à une matrice triangulaire supérieure dont les éléments diagonaux sont ses valeurs propres. De plus, si la matrice A est réelle (désignation $A \in M_n(\mathbb{R})$) et si toutes les valeurs propres de A sont réelles, la matrice U peut être choisie réelle et donc orthogonale.

Démonstration. La démonstration peut être retrouvée dans le Théorème 2.3.1 de [HJ85]. □

Théorème 3.19 (Théorème de Schur réel de triangularisation)**.** *Pour une matrice carrée réelle $A \in M_n(\mathbb{R})$, il existe une matrice réelle orthogonale $Q \in M_n(\mathbb{R})$ telle que :*

$$Q^TAQ = \begin{bmatrix} A_1 & & & * \\ & A_2 & & \\ & & \ddots & \\ 0 & & & A_k \end{bmatrix} \in M_n(\mathbb{R}),\ 1 \le k \le n$$

où chaque bloc A_i est soit un bloc d'une matrice réelle 1×1 soit un bloc d'une matrice réelle 2×2 avec une paire non réelle de valeurs propres complexes conjuguées.

Démonstration. La démonstration peut être retrouvée dans le Théorème 2.3.4 de [HJ85]. □

Le théorème suivant détermine l'emplacement des valeurs propres (la caractérisation spectrale) d'une matrice carrée.

Théorème 3.20 (Geršgorin)**.** *Pour une matrice $A=[a_{ij}]\in M_n$ donnée, avec*

$$R_i^{'}(A)\equiv \sum_{\substack{j=1\\ j\neq i}}^{n} |a_{ij}|,\ 1\leq i\leq n$$

désignant une ***somme absolue de la ligne avec l'élément diagonal supprimé*** *de A, ou*

$$C_j^{'}(A)\equiv \sum_{\substack{i=1\\ i\neq j}}^{n} |a_{ij}|,\ 1\leq j\leq n$$

désignant une ***somme absolue de la colonne avec l'élément diagonal supprimé*** *de A. Toutes les valeurs propres de A sont donc placées dans une réunion de n disques déterminées comme :*

$$\bigcup_{i=1}^{n}\left\{\mathfrak{z}\in\mathbb{C}:|\mathfrak{z}-a_{ii}|\leq R_i^{'}(A)\right\}\equiv\mathbb{G}(A)$$

ou comme :

$$\bigcup_{j=1}^{n}\left\{\mathfrak{z}\in\mathbb{C}:|\mathfrak{z}-a_{jj}|\leq C_j^{'}(A)\right\}\equiv\mathbb{G}\left(A^T\right)$$

En plus si une réunion de k de n disques forme une région déconnectée de tous les restes $n-k$ disques, alors exactement k valeurs propres de A correspondantes sont donc dans cette réunion.

La région $\mathbb{G}(A)$ est souvent appelée une *région de Geršgorin* (pour la lignes) de A, les disques individuels la formant sont appelés *les disques de Geršgorin* et leurs frontières, les *cercles de Geršgorin*.

Démonstration. La démonstration peut être retrouvée dans la Théorème 6.1.1 de [HJ85]. □

Le théorème suivant a pour but de démontrer la relation entre la propriété de la connexité d'un graphe avec la caractérisation spectrale de sa matrice Laplacienne.

Théorème 3.21. *Un graphe $\mathcal{G}$ est connexe si et seulement si la deuxième valeur propre $\lambda_2(L)$ de sa matrice Laplacienne a sa partie réelle strictement positive : $\mathfrak{Re}(\lambda_2(L)) > 0$, où la deuxième valeur propre, par convention, est une valeur propre dont la partie réelle est la deuxième plus petite valeur parmi les autres.*

Démonstration. La démonstration peut être retrouvée dans le Corollaire 2.14 de [Wie10]. □

3.4 Annexe D : Opérations mathématiques sur les signaux

Dans ce qui suit, on adopte les notations et les définitions présentées dans [Jön01] et [Sco97].

Un signal est modélisé par une fonction dépendante du temps. Il est donc nécessaire de définir les espaces fonctionnels et les opérateurs. Soit $\mathcal{U}$ un espace fonctionnel dont les éléments sont de fonctions réelles $f : \mathbb{R} \to \mathbb{R}$ et sur lequel on peut définir le *produit scalaire* de deux éléments f et g comme :

$$\langle f, g \rangle = \int_{-\infty}^{+\infty} f(t)\, g(t)\, dt \tag{3.33}$$

On définit aussi un espace $\mathcal{U}^n$, l'espace de fonctions réelles $f : \mathbb{R} \to \mathbb{R}^n$ dont les images sont des vecteurs réels de dimension n, possédant le produit scalaire suivant :

$$\langle f, g \rangle = \sum_{i=1}^{n} \langle f_i, g_i \rangle = \sum_{i=1}^{n} \int_{-\infty}^{+\infty} f_i(t)\, g_i(t)\, dt = \int_{-\infty}^{+\infty} f^T(t)\, g(t)\, dt \tag{3.34}$$

Voici quelques propriétés utiles du produit scalaire qui vont être utilisées par la suite (où $f \in \mathcal{U}^n$, $g \in \mathcal{U}^m$, $f_1 \in \mathcal{U}^n$, $f_2 \in \mathcal{U}^n$, $\alpha \in \mathbb{R}$, $Y \in \mathbb{R}^{n \times m}$) :

1. $\langle f, g \rangle = \langle g, f \rangle$
2. $\langle \alpha f, g \rangle = \langle f, \alpha g \rangle = \alpha \langle f, g \rangle$
3. $\langle f, Yg \rangle = \left\langle Y^T f, g \right\rangle$
4. $\langle f_1 + f_2, g \rangle = \langle f_1, g \rangle + \langle f_2, g \rangle$

La *norme* de l'espace $\mathcal{U}^n$ est définie selon son produit scalaire de la manière suivante :

$$\|f\| = \sqrt{\langle f, f \rangle} \tag{3.35}$$

Pour tout $f \in \mathcal{U}^n$ et tout $\tau \in \mathbb{R}$ on peut définir *l'opérateur de troncature* f_τ comme :

$$f_\tau(t) = \begin{cases} f(t) & \text{pour } t < \tau \\ 0 & \text{autrement} \end{cases} \tag{3.36}$$

On définit ainsi le *produit scalaire tronqué* comme $\langle f, g \rangle_\tau = \langle f_\tau, g_\tau \rangle$, la *norme tronquée* associée comme $\|f\|_\tau = \sqrt{\langle f, f \rangle_\tau}$ ainsi que l'*espace étendu* comme $\mathcal{U}_e^n = \{u \mid u_\tau \in \mathcal{U}^n, \text{ pour } \forall \tau \in \mathbb{R}\}$.

Un *système* ou un *opérateur* G avec m entrées et p sorties suivant ces nouvelles notations est alors formellement défini comme une relation sur $\mathcal{U}_e^m \times \mathcal{U}_e^p$ qui associe

à un signal d'entrée $u \in \mathcal{U}_e^m$ un signal de sortie $y \in \mathcal{U}_e^p$. Un système est causal si pour $\forall \tau \in \mathbb{R}$ l'équation suivante est respectée :

$$(G(u_\tau))_\tau = (G(u))_\tau \tag{3.37}$$

La fonction $f \in \mathcal{U}_e^n$ ou signal qu'il représente est à énergie finie si l'inégalité suivante est satisfaite :

$$\|f\|_\tau^2 = \int_{-\infty}^{\tau} f^T(t) f(t) dt < \infty \tag{3.38}$$

L'espace de toutes les fonctions ayant une énergie finie définit dans le sens de (3.38) pour $\tau = +\infty$ est appelé l'espace $\mathcal{L}_2$ et pour $\forall \tau : \tau \in \mathbb{R}$ et $\tau < +\infty$ est appelée $\mathcal{L}_{2e}$.

La norme (3.35) est en réalité la norme $\mathcal{L}_2$ du signal. A partir de cette norme, on peut définir une norme pour un opérateur dite la *norme induite* $\mathcal{L}_2$ d'un système (ou d'un opérateur) G qui est définie de la manière suivante :

$$\|G\|_{\mathcal{L}_2} = \sup_{\substack{f \in \mathcal{L}_{2e} \\ f \neq 0}} \frac{\|G(f)\|}{\|f\|} \tag{3.39}$$

Néanmoins, il existe d'autres types de norme de signaux et norme induite d'opérateur tels que $\mathcal{L}_p$ où $p \in \{\mathbb{N}, \infty\}$ [Sco97, Jön01]. Cependant, nous avons choisi d'utiliser la norme induite $\mathcal{L}_2$ d'opérateur pour les deux raisons suivantes :

1. elle a une interprétation physique en terme énergétique entre les entrées et les sorties du système ;
2. dans le cas de systèmes linéaires, elle correspond à la norme H_∞ définie dans la Définition 2.4 page 67. Ceci permet de formuler un certain nombre de spécifications du cahier de charges sous la forme d'un problème d'optimisation sur une norme pondérée H_∞ (voir la section 2.5 page 60 et [Zam81, SP05]).

Définition 3.22. Un système est $\mathcal{L}_2$-*gain stable* ou stable au sens entrée-sortie (voir la Définition 2.2 page 61) si à un signal à énergie finie à l'entrée il associe un signal à énergie finie à la sortie. Si la norme induite $\mathcal{L}_2$ d'un système est bornée, le système est $\mathcal{L}_2$-gain stable.

4 Méthode de la synthèse de la loi de commande décentralisée

Introduction

Dans le premier chapitre de cette thèse, nous avons discuté de la problématique de génération et de distribution du signal d'horloge de SOCs synchrones. Nous avons mis en évidence les facteurs limitant liés aux architectures classiques ainsi que l'intérêt potentiel des architectures actives. Une conception d'un tel réseau actif de génération et de distribution du signal d'horloge est un problème complexe et délicat. Ainsi certaines précautions sur les interconnections et sur les échanges d'information existant au sein du réseau doivent être prises en compte lors de sa réalisation. En effet, une utilisation naïve de la méthode de conception « locale » ne prenant pas en compte les aspects d'interconnexion globale peut apporter des résultats insuffisants. Ceci est illustré dans l'exemple du chapitre 2 où un problème de conception d'une ADPLL, c'est-à-dire un sous-système local du réseau, a été considéré.

D'un autre côté, un problème de conception du réseau entier considérant toute la complexité des interconnections peut être difficile et non efficace du point de vue de la réalisation pratique. Il est important donc de pouvoir transformer, si possible, les conditions globales en conditions locales qui peuvent être satisfaites par une synthèse locale de commande. Cette idée du lien « global-local » permet de simplifier le problème de départ et de donner, en outre, la possibilité de réutiliser les outils bien maitrisés et connus en général de la synthèse. Les exemples d'utilisation de cette idée dans l'état de l'art des systèmes Multi-Agents et de la Commande décentralisée ainsi que leurs limitations ont été présentés dans le chapitre 3. La raison principale des limitations rencontrées est le conservatisme inévitable pour la commande efficace de système de grande dimension tel qu'un réseau actif d'ADPLLs.

Ce chapitre présente une méthode originale de la synthèse de la loi de commande décentralisée assurant la stabilité d'un système de grande dimension composé de sous-systèmes identiques qui exploite de manière efficace l'idée du lien « global-local » et l'information de l'identité des sous-systèmes élémentaires. Cela permet d'avoir une méthode de synthèse de la loi de commande décentralisée à la fois efficace et exploitable en pratique toute en réduisant le conservatisme potentiel. En outre, la deuxième idée importante retenue de la conception classique « locale », à savoir une évaluation de la performance du réseau à travers l'évaluation de la « différence » de performance locale et globale, est utilisée afin de proposer une méthode élégante et originale de la synthèse de la loi de commande décentralisée assurant en plus de la stabilité un niveau de performance globale désirée.

Le chapitre est organisé de la manière suivante : la première section rappelle la formalisation du problème abordé, la section suivante montre la transformation des conditions globales assurant la stabilité et un niveau de performance globale en des conditions locales sur la dynamique d'un sous-système local et l'interconnexion. Ensuite, les deux sections suivantes présentent la méthode de synthèse de la loi de commande assurant ces conditions et donnent un algorithme de synthèse associé. A la fin du chapitre la méthode est appliquée à l'exemple numérique du réseau présenté à la fin du chapitre 2 page 96. La réalisation de l'algorithme de synthèse de la loi de commande décentralisée pour le réseau actif de distribution du signal d'horloge ainsi que différents aspects pratiques de son implémentation sont reportés dans le chapitre 5.

4.1 Formulation du problème

Considérons un système de grande dimension qui consiste en l'interconnexion de N sous-systèmes identiques T_s (voir la Figure 4.1). Pour des raisons de clarté et sans perte de généralité, on se place dans le cas d'un sous-système carré, c'est-à-dire un sous-système T_s avec le même nombre p d'entrées et de sorties. Chaque sous-système est LTI, causal et peut être divisé en deux parties : *(i)* une partie H du sous-système appelé le procédé à commander et *(ii)* une partie K réalisant la commande, appelé le correcteur. L'ensemble des sous-systèmes est décrit par un opérateur LTI global $\widetilde{T}$ et leur interconnexion par un opérateur LTI M :

$$\begin{array}{ccccc} \varphi & = & \overbrace{(I_N \otimes T_s)}^{\widetilde{T}} & & r \\ \left[\begin{array}{c} r \\ z \end{array}\right] & = & \underbrace{\left[\begin{array}{cc} M_{11} & M_{12} \\ M_{21} & M_{22} \end{array}\right]}_{M} & \left[\begin{array}{c} \varphi \\ w \end{array}\right] & \end{array} \tag{4.1}$$

avec $T_s = \underbrace{\left[\begin{array}{cc} H_{11} & H_{12} \\ H_{21} & H_{22} \end{array}\right]}_{H} \star K$, $r(t)$, $\varphi(t) \in \mathbb{R}^{pN}$, $w(t) \in \mathbb{R}^{n_w}$, $z(t) \in \mathbb{R}^{n_z}$.

Le système (4.1) représente un système global de grande dimension. Tout au long de ce chapitre, nous considérons la stabilité globale et locale ainsi que la performance globale et locale. Le sous-système considéré local correspondant est représenté sur la Figure 4.2 et décrit par :

$$\begin{array}{ccccc} \left[\begin{array}{c} z_{pi} \\ y_i \end{array}\right] & = & \overbrace{\left[\begin{array}{cc} \widetilde{H}_{11} & \widetilde{H}_{12} \\ \widetilde{H}_{21} & \widetilde{H}_{22} \end{array}\right]}^{\widetilde{H}} & \left[\begin{array}{c} w_{pi} \\ u_i \end{array}\right] & \\ u_i & = & K & & y_i \end{array} \tag{4.2}$$

avec $w_{pi}(t) \in \mathbb{R}^{n_{wp}}$, $z_{pi}(t) \in \mathbb{R}^{n_{zp}}$, $y_i(t) \in \mathbb{R}^{n_y}$, $u_i(t) \in \mathbb{R}^{n_u}$.

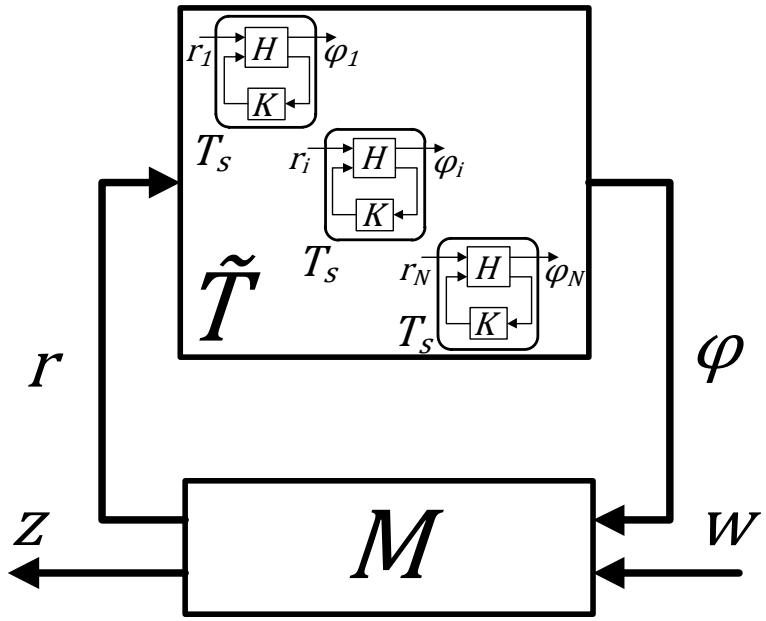


Figure 4.1: Système de grande dimension LTI global considéré

Il est très important de distinguer ces propriétés pour les deux cas. La ***stabilité locale*** correspond à la stabilité d'un sous-système indépendant (4.2) sans autres sous-systèmes et sans interconnexion (stabilité d'un nœud séparé) tandis que la ***stabilité globale*** correspond à la stabilité du système complet de grande dimension défini par (4.1). La ***performance locale*** est évaluée pour un sous-système (4.2) séparé du réseau, augmenté par les entrées et les sorties de performance w_{pi} et z_{pi} respectivement. La ***performance globale*** est évaluée pour le système complet de grande dimension (4.1) avec les entrées w et les sorties z globales de performances correspondantes. Ces entrées et sorties définissent les dimensions et la structure des opérateurs LTIs $\widetilde{H}$ et M respectivement.

Les spécifications de performance pour le cas global et le cas local sont exprimées sous forme de minimisation des normes H_∞ des fonctions de transfert $T_{w\to z} = \widetilde{T} \star M$ et $T_{w_{pi}\to z_{pi}} = \widetilde{H} \star K$ respectivement. Sous la condition que les deux systèmes soient stables, ceci permet d'imposer des contraintes fréquentielles réalisant les spécifications de performance générales semblables à celles exposées dans la section 2.5, page 60. En choisissant les entrées et les sorties appropriées ainsi que les contraintes fréquentielles concernées, nous pouvons aborder non seulement le problème de synchronisation d'ADPLLs considéré dans cette thèse mais aussi d'autres types de problèmes de systèmes Multi-Agents et de systèmes de grande dimension tels que : consensus, coopération, coordination, rendez-vous *etc.*.

Par la suite, nous nous concentrons sur le problème général suivant :

Problème 4.1 (Problème de la synthèse)**.** *Étant donné un système LTI H, et un système LTI M décrivant l'interconnexion, trouver un système LTI K tel qu'il :*

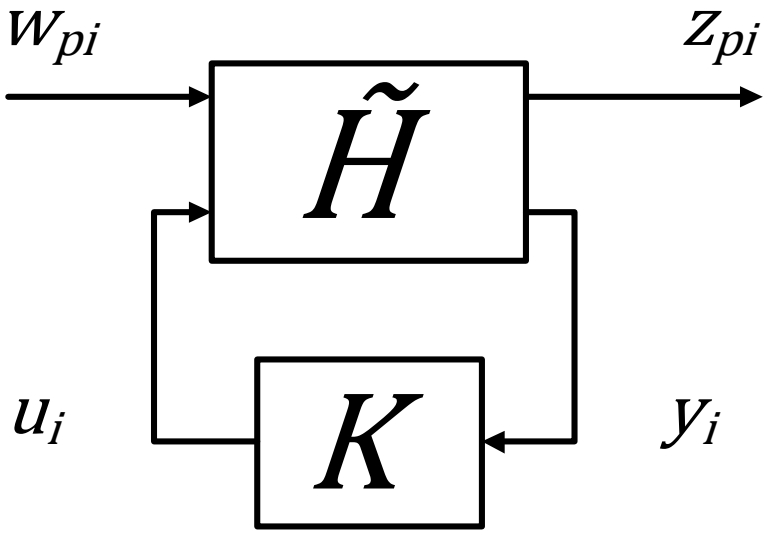


Figure 4.2: Sous-système LTI local considéré

1. *Stabilise chaque sous-système (*(4.2) *représenté sur la Figure 4.2) séparement ainsi que le système global (*(4.1) *représenté sur la Figure 4.1) ;*
2. *Assure les spécifications de performance locale par la minimization de la norme* H_∞ *de la fonction de transfert* $T_{w_{pi}\to z_{pi}} = \widetilde{H} \star K$ *entre les entrées* w_{pi} *et les sorties* z_{pi} *de performance ;*
3. *Assure les spécifications de performance globale par la minimization de la norme* H_∞ *de la fonction de transfert* $T_{w\to z} = \widetilde{T} \star M$ *entre les entrées* w *et les sorties* z *de performance.*

4.2 Les conditions locales pour la stabilité et la performance globales

Dans la première partie de cette sous-section, en se basant sur l'approche entrée-sortie et le principe de séparation des graphes (voir la sous-section 3.2 page 125), les conditions locales sur la dynamique d'un sous-système T_s et sur la dynamique de l'interconnexion M sont dérivées afin d'assurer la stabilité globale. Ensuite, ces conditions sont étendues pour assurer en plus de la stabilité les spécifications de performance globale.

Dans la suite du document, nous allons utiliser la caractérisation de dissipativité d'un opérateur dont la définition est donnée sur la page 127.

Nous allons utiliser souvent la notation de dissipativité courte suivante :

$$P \otimes \{X, Y, Z\}\text{-dissipatif} = \{P \otimes X, P \otimes Y, P \otimes Z\}\text{-dissipatif}$$

où la matrice P est une matrice réelle quelconque.

On rappelle que selon le principe de séparation (voir le Théorème 3.6 page 129) si le sous-système T_s (respectivement l'interconnexion M) est caractérisé(e) par une propriété de dissipativité, il est possible d'assurer la stabilité du système global en assurant la propriété de dissipativité complémentaire sur le système d'interconnexion M (respectivement le sous-système T_s). En utilisant ce résultat, la stabilité globale est transformée en conditions locales comme cela est résumé dans le théorème suivant :

Théorème 4.2. *Soit le système bien posé et causal décrit par l'équation* (4.1)*. Pour les matrices réelles données $X = X^T \leq 0$, Y, $Z = Z^T$ de dimensions $p \times p$, s'il existe une matrice réelle symétrique définie positive $P \in \mathbb{R}^{N\times N}$ telle que*

(i) le système d'interconnexion M_{11} est $P \otimes \{X, Y, Z\}-$dissipatif (ou strictement dissipatif) ;

(ii) le sous-système local T_s est strictement $\left\{-Z, -Y^T, -X\right\}-$dissipatif (ou dissipatif tout court).

Le système global (4.1) *est alors stable de façon interne.*

Démonstration. Premièrement, on remarque que la matrice de fonctions de transfert du système global présenté sur la Figure 4.1 a la forme suivante :

$$\widetilde{T} \star M = M_{22} + M_{21}\widetilde{T}\left(I - M_{11}\widetilde{T}\right)^{-1} M_{12} \tag{4.3}$$

Étant donné que par définition le système d'interconnexion M est stable, la stabilité du (4.3) est équivalente à la stabilité du système dont la matrice de fonctions de transfert est égale à $\widetilde{T}\left(I - M_{11}\widetilde{T}\right)^{-1}$, c'est-à-dire une simple interconnexion de système $\widetilde{T}$ avec le système LTI M_{11} (voir la Figure 4.2 sans les entrées w et les sorties z extérieures).

Grâce au fait que les sous-systèmes sont identiques, la propriété de dissipativité stricte *(ii)* de l'opérateur local T_s (voir la Définition 3.4 page 127) peut être reformulée comme une propriété de $I_N \otimes \{-Z, -Y^T, -X\}$-dissipativité stricte de l'opérateur global diagonal $\widetilde{T}$ c'est-à-dire pour $\varepsilon > 0$:

$$\begin{aligned} &\begin{bmatrix} I_p \\ T_s(j\omega) \end{bmatrix}^* \begin{bmatrix} -Z & -Y^T \\ -Y & -X \end{bmatrix} \begin{bmatrix} I_p \\ T_s(j\omega) \end{bmatrix} \leq -\varepsilon \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \\ \Leftrightarrow\ &\begin{bmatrix} I_N \otimes T_s(j\omega) \\ I_N \otimes I_p \end{bmatrix}^* \begin{bmatrix} I_N \otimes X & I_N \otimes Y \\ I_N \otimes Y^T & I_N \otimes Z \end{bmatrix} \begin{bmatrix} I_N \otimes T_s(j\omega) \\ I_N \otimes I_p \end{bmatrix} \geq \varepsilon \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \end{aligned} \tag{4.4}$$

pour presque tout ω, $\omega \in \mathbb{R}$.

A l'aide de post- et pré-multiplications de la condition (4.4) par une matrice inversible de rang plein $Q \otimes I_p$ et de sa transposée et de l'utilisation de la propriété du

produit de Kronecker (voir l'Annexe C du chapitre 3 page 134) :

$$\begin{aligned}(I_N \otimes T_s)(Q \otimes I_p) &= (Q \otimes I_p)(I_N \otimes T_s) \\ (Q \otimes I_p)^T (Q \otimes I_p) &= \left(Q^T \otimes I_p\right)(Q \otimes I_p) = \left(Q^T Q \otimes I_p\right)\end{aligned} \tag{4.5}$$

nous obtenons :

$$\begin{bmatrix} I_N \otimes T_s(j\omega) \\ I_N \otimes I_p \end{bmatrix}^* \begin{bmatrix} (I_N \otimes X)\left(Q^T Q \otimes I_p\right) & (I_N \otimes Y)\left(Q^T Q \otimes I_p\right) \\ \left(I_N \otimes Y^T\right)\left(Q^T Q \otimes I_p\right) & (I_N \otimes Z)\left(Q^T Q \otimes I_p\right) \end{bmatrix} \begin{bmatrix} I_N \otimes T_s(j\omega) \\ I_N \otimes I_p \end{bmatrix} \geq \varepsilon \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$

ce qui est équivalent, pour une matrice $P = Q^T Q > 0$, à :

$$\begin{bmatrix} \widetilde{T}(j\omega) \\ I_{p\times N} \end{bmatrix}^* \begin{bmatrix} P \otimes X & P \otimes Y \\ P \otimes Y^T & P \otimes Z \end{bmatrix} \begin{bmatrix} \widetilde{T}(j\omega) \\ I_{p\times N} \end{bmatrix} > \varepsilon \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{4.6}$$

pour presque $\forall \omega \in \mathbb{R}$.

Utilisant l'hypothèse *(i)* sur le système d'interconnexion M_{11} et le fait que (4.6) est en réalité la condition de stricte $P \otimes \{-Z, -Y^T, -X\}$-dissipativité de l'opérateur global $\widetilde{T}$, le théorème des séparation de graphes (Théorème 3.6, page 129) peut être appliqué pour démontrer la stabilité du système interconnecté $\widetilde{T}\left(I - M_{11}\widetilde{T}\right)^{-1}$ et donc du système global (4.1) représenté sur la Figure 4.1.

La même démarche est appliquée pour démontrer la stabilité du système global (4.1) dans le cas de la dissipativité stricte de la matrice d'interconnexion M_{11} et la dissipativité simple du sous-système T_s. □

Remarque 4.3. En regardant de plus près la démonstration du Théorème 4.2, le lecteur peut faire une objection concernant l'utilité de la matrice P. En effet, le Théorème 4.2 peut être démontré sans l'utilisation de la matrice symétrique définie positive P : le théorème de séparation de graphes peut être appliqué directement avec la condition de $I_N \otimes \{-Z, -Y^T, -X\}$-dissipativité (4.4) de l'opérateur global $\widetilde{T}$. Cependant, les conditions du Théorème 4.2 sont seulement suffisantes étant donné que le principe de séparation des graphes (voir le Théorème 3.6 page 129) utilisé dans sa démonstration, fournit des conditions seulement suffisantes et non nécessaires. L'utilisation de la matrice symétrique définie positive P dans le cas de sous-systèmes identiques T_s réduit le conservatisme potentiel du Théorème 4.2. En effet, la matrice P est une matrice de paramétrisation qui ajoute certains degrés de liberté sur le choix de la caractérisation de la dissipativité. Dans le cas de sous-systèmes identiques satisfaisant la condition de dissipativité *(ii)*, l'opérateur global bloc-diagonal $\widetilde{T}$ satisfait la condition de dissipativité globale non seulement pour la matrice identité I_N comme dans (4.4) mais aussi pour toutes autres matrices symétriques définies positives possibles P. Dans l'optique qui nous intéresse, il suffit seulement de trouver une telle matrice qui n'est pas forcement égale à la matrice d'identité. De manière plus formelle, l'opérateur global des sous-systèmes peut être représenté comme $\widetilde{T} = I_N \otimes T_s$ seulement pour le cas de sous-systèmes identiques. Par conséquent, et seulement dans ce cas particulier, la propriété de permutation du produit de Kronecker (4.5) peut être appliquée afin d'obtenir la condition de

$P \otimes \{-Z, -Y^T, -X\}$-dissipativité (4.6) avec une matrice symétrique définie positive P. Par application du Théorème de séparation des graphes avec ces conditions étendues, la stabilité globale est donc testée pour le cas d'une structure *bloc diagonale* avec des blocs identiques de l'opérateur $\widetilde{T}$ et non pas pour le cas d'une structure quelconque. L'information sur la structure de l'opérateur $\widetilde{T}$ est alors exploitée avec une caractérisation plus fine de dissipativité ce que implique la réduction du conservatisme.

Remarque 4.4. La condition de dissipativité du système LTI d'interconnexion M_{11} (condition *(i)* du Théorème 4.6) avec les matrices $X = X^T \leq 0$, Y, $Z = Z^T$ données peut être transformée en une condition LMI de la variable de décision P :

1. Dans le cas de matrice constante réelle $M_{11} \in \mathbb{R}^{pN \times pN}$ (système d'interconnexion statique), cela est direct :

$$\exists P \in \mathbb{R}^{N \times N} : P = P^T > 0, \begin{bmatrix} I_{pN} \\ M_{11} \end{bmatrix}^T \begin{bmatrix} P \otimes X & P \otimes Y \\ P \otimes Y^T & P \otimes Z \end{bmatrix} \begin{bmatrix} I_{pN} \\ M_{11} \end{bmatrix} \leq 0 \quad (4.7)$$

2. Dans le cas d'un système d'interconnexion LTI dynamique M_{11} tel que la fonction de transfert

$$\begin{bmatrix} I_{pN} & M_{11}(s) \end{bmatrix}^T = C(sI - A)^{-1} B + D$$

avec les matrices de représentation d'états $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times pN}$, $C \in \mathbb{R}^{2pN \times n}$, $D \in \mathbb{R}^{2pN \times pN}$, la condition LMI peut être obtenue après une application du Kalman-Yakubovich-Popov (KYP) Lemme [Ran96] :

$$\begin{matrix} \exists P \in \mathbb{R}^{N \times N} : P = P^T > 0 \\ \exists R \in \mathbb{R}^{n \times n} : R = R^T > 0 \end{matrix}, \begin{bmatrix} A & B \\ I & 0 \\ C & D \end{bmatrix}^T \begin{bmatrix} 0 & R & & 0 & \\ R & 0 & & & \\ & & & P \otimes X & P \otimes Y \\ 0 & & & P \otimes Y^T & P \otimes Z \end{bmatrix} \begin{bmatrix} A & B \\ I & 0 \\ C & D \end{bmatrix} \leq 0 \quad (4.8)$$

Remarque 4.5. Pratiquement dans tous les cas d'application, les conditions de dissipativité *(i)* et *(ii)* du Théorème 4.2 sont appliquées avec la contrainte supplémentaire sur la matrice de dissipativité Z :

$$Z = Z^T \geq 0 \quad (4.9)$$

Cela implique que :
– Le problème de synthèse, c'est-à-dire le problème de recherche de la loi de commande K (voir la Figure 4.2, page 146) telle que le sous-système T_s est stable et satisfait la contrainte *(ii)* peut être formulé sous la forme d'un problème d'optimisation convexe (voir le cas B dans [SGC97]) ;
– le sous-système « zéro » ($T_s = 0$), lui aussi, satisfait la contrainte quadratique *(ii)* du Théorème 4.2 et si une loi de commande est trouvée telle que T_s satisfait

la contrainte *(ii)*, cela est automatiquement le cas pour θT_s, $\forall\theta$ avec $\theta \in [0,1]$. C'est une propriété de la stabilité robuste correspondant à la version robuste de séparation des graphes détaillée dans [MR97].

Fort de cette remarque, nous allons transformer le Théorème 4.2 en une autre version, plus adaptée à la synthèse. Pour faire cela, nous appliquons la procédure de « loop shifthing » définie par :

$$\hat{T} = (-X)^{\frac{1}{2}} \left(T_s + X^{-1}Y\right) \left(Z - Y^T X^{-1} Y\right)^{-\frac{1}{2}}. \tag{4.10}$$

La transformation (4.10) est forcement bien définie étant donné les définitions des matrices de dissipativité $X \leq 0$ et $Z \geq 0$ (voir (4.9)). La contrainte de dissipativité *(ii)* du Théorème 4.2 et ainsi transformée en une contrainte sous la forme H_∞-standard :

$$\left\|\hat{T}\right\|_\infty < 1. \tag{4.11}$$

Nous allons voir par la suite de ce document, que, grâce à cette transformation, une application de la synthèse H_∞-standard est possible (utilisée déjà dans la section 2.5 du chapitre 2). Ceci est très intéressant d'un point de vue de la compatibilité de la méthode de synthèse que nous allons proposer avec des méthodes et des outils de conception traditionnels de PLL (voir la discussion page 60 du chapitre 2). Ainsi, on reformule le Théorème 4.2 dans une version plus adaptée à la synthèse :

Théorème 4.6. *Soit le système bien posé et causal décrit par l'équation* (4.1)*. Pour les matrices réelles données $X = X^T \leq 0$, Y, $Z = Z^T \geq 0$ de dimensions $p \times p$, s'il existe une matrice réelle symétrique définie positive $P \in \mathbb{R}^{N\times N}$ telle que*

(i) le système d'interconnexion M_{11} est $P \otimes \{X, Y, Z\}-$dissipatif;

(ii) le sous-système local T_s est tel que :

$$\left\|\widehat{T}\right\|_\infty < 1 \tag{4.12}$$

avec $\hat{T} = (-X)^{\frac{1}{2}} \left(T_s + X^{-1}Y\right) \left(Z - Y^T X^{-1} Y\right)^{-\frac{1}{2}}$.

Le système global (4.1) *est alors stable de façon interne.*

Démonstration. Notons que la contrainte sur la norme H_∞ (4.12) est équivalente à la contrainte quadratique dans le domaine fréquentiel :

$$\begin{bmatrix} \widehat{T}(j\omega) \\ I_p \end{bmatrix}^* \begin{bmatrix} -I_p & 0 \\ 0 & I_p \end{bmatrix} \begin{bmatrix} \widehat{T}(j\omega) \\ I_p \end{bmatrix} \geq \varepsilon_1 \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{4.13}$$

pour presque tout ω et ε_1 tels que $\omega \in \mathbb{R}$ et $\varepsilon_1 \in \mathbb{R} : \varepsilon_1 > 0$.

En appliquant la procédure de « loop shifting » à l'aide de post- et pré-multiplications des deux parties de l'équation (4.13) par la matrice $\left(Z - Y^T X^{-1} Y\right)^{\frac{1}{2}}$ et sa transposée, et en tenant compte qu'elle est forcement définie positive selon la définition

des matrices X et Z, on remplace $\widehat{T}$ par son expression pour transformer l'inégalité (4.13) en :

$$\begin{bmatrix} T_s(j\omega) \\ I_p \end{bmatrix}^* \begin{bmatrix} I_p & 0 \\ Y^T X^{-T} & I_p \end{bmatrix} \begin{bmatrix} X & 0 \\ 0 & Z - Y^T X^{-1} Y \end{bmatrix} \begin{bmatrix} I_p & X^{-1}Y \\ 0 & I_p \end{bmatrix} \begin{bmatrix} T_s(j\omega) \\ I_p \end{bmatrix} \geq \varepsilon_2 \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}, \tag{4.14}$$

pour presque tout ω, $\omega \in \mathbb{R}$ et avec $0 < \varepsilon_2 \leq \varepsilon_1 \underline{\sigma}\left(Z - Y^T X^{-1} Y\right)$, où $\underline{\sigma}(A)$ désigne la valeur singulière (voir la Définition 2.7 page 75) minimale de la matrice A.

La dernière inégalité est équivalente à :

$$\begin{bmatrix} I_p \\ T_s(j\omega) \end{bmatrix}^* \begin{bmatrix} -Z & -Y^T \\ -Y & -X \end{bmatrix} \begin{bmatrix} I_p \\ T_s(j\omega) \end{bmatrix} \leq - \varepsilon_2 \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{4.15}$$

pour presque tout ω, $\omega \in \mathbb{R}$.

La condition (4.15) est une condition de stricte $\left\{-Z, -Y^T, -X\right\}$ −dissipativité du sous-système T_s (voir la Définition 3.4, page 127). La stabilité est donc démontrée par l'application du Théorème 4.2. □

Les Théorèmes 4.2 et 4.6 permettent de transformer la condition de stabilité globale du système (4.1) en une condition sur le système d'interconnexion M_{11} (4.7) ou (4.8) et une condition sur la dynamique locale du sous-système T_s (4.12). Ce résultat peut donc être combiné avec des résultats classiques de synthèse de la loi de commande (comme cela est par exemple présenté dans le chapitre 2) afin d'obtenir la loi de commande assurant *a priori* la stabilité globale. Le résultat assurant en outre la performance globale est obtenu de manière similaire comme présenté par la suite.

Théorème 4.7. *Soit le système bien posé et causal décrit par l'équation* (4.1). *Pour les matrices réelles données $X = X^T \leq 0$, Y, $Z = Z^T \geq 0$ de dimensions $p \times p$ et une borne donnée $\eta > 0$, s'il existe une matrice réelle symétrique définie positive $P \in \mathbb{R}^{N \times N}$ telle que*

(i) le système d'interconnexion M est $\left\{\widetilde{X}, \widetilde{Y}, \widetilde{Z}\right\}$ −dissipatif avec des matrices $\widetilde{X} = diag(P \otimes X, -\eta^2 I)$, $\widetilde{Y} = diag(P \otimes Y, 0)$, $\widetilde{Z} = diag(P \otimes Z, I)$ c'est-à-dire :

$$\begin{bmatrix} I \\ M(j\omega) \end{bmatrix}^* \begin{bmatrix} P \otimes X & 0 & P \otimes Y & 0 \\ 0 & -\eta^2 I & 0 & 0 \\ P \otimes Y^T & 0 & P \otimes Z & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} I \\ M(j\omega) \end{bmatrix} \leq 0 \tag{4.16}$$

pour presque tout ω, $\omega \in \mathbb{R}$;

(ii) le sous-système local T_s satisfait la condition (4.12) *c'est-à-dire :*

$$\left\|\widehat{T}\right\|_\infty < 1 \tag{4.17}$$

avec $\hat{T} = (-X)^{\frac{1}{2}} \left(T_s + X^{-1}Y\right) \left(Z - Y^T X^{-1} Y\right)^{-\frac{1}{2}}$

Le système global (4.1) *est stable de façon interne et la norme* H_∞ *de la fonction de transfert* $T_{w\to z} = M \star \widetilde{T}$ *est inférieure ou égale à* η *c'est-à-dire :*

$$\|T_{w\to z}\|_\infty < \eta. \tag{4.18}$$

Démonstration. Tout d'abord nous démontrons la stabilité et après la borne sur la norme H_∞ de la fonction de transfert globale $T_{w\to z}$.

La stabilité est impliquée par le Théorème 4.6. En effet, la condition *(ii)* du Théorème 4.6 est identique à la condition *(ii)* du présent Théorème et donc est satisfaite par (4.17).

D'un autre côté, la condition (4.16) est exprimée sous forme de :

$$\begin{bmatrix} I & 0 \\ M_{11}(j\omega) & M_{12}(j\omega) \\ M_{21}(j\omega) & M_{22}(j\omega) \\ 0 & I \end{bmatrix}^* \begin{bmatrix} P\otimes X & P\otimes Y & 0 & 0 \\ P\otimes Y^T & P\otimes Z & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & -\eta^2 I \end{bmatrix} \begin{bmatrix} I & 0 \\ M_{11}(j\omega) & M_{12}(j\omega) \\ M_{21}(j\omega) & M_{22}(j\omega) \\ 0 & I \end{bmatrix} \leq 0 \tag{4.19}$$

pour presque tout ω, $\omega \in \mathbb{R}$, ce qui implique

$$\begin{bmatrix} I \\ M_{11}(j\omega) \end{bmatrix}^* \begin{bmatrix} P\otimes X & P\otimes Y \\ P\otimes Y^T & P\otimes Z \end{bmatrix} \begin{bmatrix} I \\ M_{11}(j\omega) \end{bmatrix} \leq -M_{21}^T M_{21} \leq 0. \tag{4.20}$$

pour presque tout ω, $\omega \in \mathbb{R}$.

La dernière condition implique que le système d'interconnexion M_{11} est $P\otimes\{X,\, Y,\, Z\}$ - dissipatif c'est-à-dire la condition *(i)* du Théorème 4.6 est aussi satisfaite. La stabilité du système global (4.1) représenté sur la Figure 4.1 est alors assurée par l'application du Théorème 4.6.

Prouvons maintenant la condition sur la borne H_∞. La condition (4.16), en prenant en compte la description du système global (4.1), implique la condition quadratique suivante dans le domaine temporel :

$$\begin{aligned} &\int_0^\tau \begin{bmatrix} z(t) \\ w(t) \end{bmatrix}^T \begin{bmatrix} I & 0 \\ 0 & -\eta^2 I \end{bmatrix} \begin{bmatrix} z(t) \\ w(t) \end{bmatrix} dt \\ \leq\ &-\int_0^\tau \begin{bmatrix} \varphi(t) \\ r(t) \end{bmatrix}^T \begin{bmatrix} P\otimes X & P\otimes Y \\ P\otimes Y^T & P\otimes Z \end{bmatrix} \begin{bmatrix} \varphi(t) \\ r(t) \end{bmatrix} dt \end{aligned} \tag{4.21}$$

pour $\forall\tau > 0$, $\forall w \in \mathcal{U}_e^{n_w}$ et les signaux r, φ, z définis par (4.1).

La partie gauche de (4.21) exprime la relation entre les signaux externes d'entrée w et de sortie z du système global (4.1). La partie droite à son tour est définie en termes de signaux d'entrée r et de sortie φ de l'opérateur bloc diagonal $\widetilde{T}$ du système global ($\varphi = \widetilde{T}r$ voir la Figure 4.1).

Rappelons que la condition (4.17) définit la propriété de stricte $\{-Z,\ -Y^T,\ -X\}$ –dissipativité de l'opérateur T_s. Comme cela a été démontré dans le Théorème 4.2, la dernière propriété est équivalente à la propriété de strict $P\otimes\{-Z,\ -Y^T,\ -X\}$ –dissipati-

vité de l'opérateur global diagonal $\widetilde{T}$ et donc $\forall P = P^T > 0$ avec $\varphi = \widetilde{T}r$ la condition suivante est respectée :

$$\int_0^\tau \begin{bmatrix} \varphi(t) \\ r(t) \end{bmatrix}^T \begin{bmatrix} -P \otimes X & -P \otimes Y \\ -P \otimes Y^T & -P \otimes Z \end{bmatrix} \begin{bmatrix} \varphi(t) \\ r(t) \end{bmatrix} dt \leq -\varepsilon \int_0^\tau \begin{bmatrix} \varphi(t) \\ r(t) \end{bmatrix}^T \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \varphi(t) \\ r(t) \end{bmatrix} dt \tag{4.22}$$

pour un $\varepsilon > 0$.

En additionnant ensemble les deux conditions (4.22) et (4.21) nous obtenons :

$$\int_0^\tau \begin{bmatrix} z(t) \\ w(t) \end{bmatrix}^T \begin{bmatrix} I & 0 \\ 0 & -\eta^2 I \end{bmatrix} \begin{bmatrix} z(t) \\ w(t) \end{bmatrix} dt \leq -\varepsilon \int_0^\tau \begin{bmatrix} \varphi(t) \\ r(t) \end{bmatrix}^T \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \varphi(t) \\ r(t) \end{bmatrix} dt \tag{4.23}$$

La dernière condition implique que la norme induite $\mathcal{L}_2$ de l'opérateur global $T_{w\to z}$ est inférieure à $\eta > 0$ (voir le cas particulier de définition de l'opérateur $\mathcal{L}_2$-gain borné sur la page 128 du chapitre 3) qui correspond à la norme H_∞ dans le cas du système LTI. □

Remarque 4.8. La condition de dissipativité (4.16) sur le système d'interconnexion LTI M peut être transformée en une condition LMI des variables de décision P et R pour des matrices données $X = X^T \leq 0$, Y, $Z = Z^T \geq 0$ comme proposé dans la Remarque 4.4 avec nouveau système d'interconnexion M et sa nouvelle caractérisation de $\left\{\widetilde{X}, \widetilde{Y}, \widetilde{Z}\right\}$ −dissipativité définie dans la condition *(i)* du Théorème 4.7.

Remarque 4.9. La dernière partie du Théorème 4.7 a un lien fort avec le Lemme KYP. En effet, en replaçant la dynamique du sous-système par une dynamique d'intégrateur pur $T_s = 1/s$ et le système d'interconnexion par une matrice réelle $\widetilde{M} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ où A, B, C, D sont les matrices d'une représentation d'états de la matrice de fonctions de transfert $T_{w\to z}$, le Lemme KYP [Ran96] est obtenu où le principe de séparation des graphes est appliqué à l'interconnexion de sous-systèmes passifs c'est-à-dire $X = 0$, $Y = -I$, $Z = 0$. Cependant, il est bien connu que le résultat du Lemme KYP est un résultat suffisant et nécessaire [Ran96], alors que les conditions du Théorème 4.7 sont seulement suffisantes et non pas nécessaires. La nécessite est perdue à cause de la généralisation de passivité vers une condition de $\{X, Y, Z\}$-dissipativité.

Les résultats de cette section donnent des conditions locales nécessaires à satisfaire pour assurer la stabilité et la performance globales. La section suivante a pour but d'illustrer la recherche de la loi de commande décentralisée satisfaisant ces conditions, c'est-à-dire comment réaliser une synthèse de la loi de commande résolvant le Problème 4.1.

4.3 Synthèse de la loi de commande

Afin de proposer une synthèse de la loi de commande efficace, nous considérons tout d'abord que les matrices de la caractérisation de dissipativité, c'est-à-dire les matrices $X = X^T \leq 0$, Y, $Z = Z^T \geq 0$ sont données. Ensuite, dans la prochaine section, nous proposerons une méthodologie pour choisir convenablement ces matrices.

Pour commencer étudions le cas de la stabilité globale. Considérant les conditions du Théorème 4.6, nous remarquons que le test de la stabilité du système global est transformé en un test sur deux conditions. La première condition est une condition de dissipativité du système d'interconnexion M_{11} (4.7) ou (4.8), la deuxième condition est une contrainte sur la norme H_∞ de la fonction de transfert $\widehat{T}$ (4.12) incluant uniquement la dynamique du sous-système local T_s, c'est-à-dire une condition locale.

En se basant sur le Théorème 4.6, le problème de synthèse de la loi de commande assurant la stabilité locale et globale du système, c'est-à-dire la première partie du Problème 4.1, peut être reformulé comme suit :

Pour des matrices données $X = X^T \leq 0$, Y, $Z = Z^T \geq 0$ *trouver :*

1. *une matrice symétrique définie positive* P *telle que la condition* (4.7) ou (4.8) soit satisfaite ;
2. *un correcteur* K *assurant la contrainte de la norme* H_∞ (4.12).

La solution de la première partie du problème décrit est la solution d'un problème de faisabilité sous contraintes LMIs (voir le Remarque 4.4). Comme cela a été évoqué avant (voir l'Annexe A sur la page 102 du chapitre 2), on peut facilement tester la faisabilité des conditions sous contraintes LMIs en appliquant des algorithmes efficace d'optimisation convexe [BEFB94, BTN01]. Pour ces raisons, on restreint notre attention sur la deuxième partie du problème de synthèse.

4.3.1 Synthèse H_∞ de la loi de commande

La contrainte (4.12) du Théorème 4.6 est en réalité une contrainte sur la dynamique locale du sous-système T_s. Pour les matrices X, Y, Z données, elle peut être facilement imposée par une synthèse H_∞-standard. En effet, la Figure 4.3 illustre la mise en forme du problème de synthèse selon la synthèse H_∞-standard (voir la Définition 2.7 page 75 où $\widetilde{H_P} = H_s$, $\widetilde{w} = r_s$ et $\widetilde{z} = \varphi_s$). En résolvant le problème de synthèse H_∞-standard (voir l'Annexe B sur la page 106 du chapitre 2), un correcteur K qui

(i) stabilise le système sur la Figure 4.3 et donc le sous-système local T_s ;

(ii) minimise la norme H_∞ de l'opérateur LTI $\widehat{T} = H_s(s) \star K(s)$ entre les signaux externes d'entrée r_s et de sortie φ_s,

peut être trouvé. En outre, si cette norme H_∞ est inférieure ou égale à $\gamma < 1$ la condition (4.12) est donc satisfaite.

La synthèse de la loi de commande avec le problème de faisabilité d'optimisation LMI (4.7) ou (4.8) résout la première partie du Problème 4.1. Cette synthèse de la loi de

commande H_∞ appliquée uniquement à un seul sous-système du réseau et l'optimisation convexe sous contrainte LMI pour le système d'interconnexion admettent une réduction significative de la complexité du problème de synthèse assurant la stabilité globale. L'approche proposée devrait être comparée avec la commande possiblement complexe du réseau complet où l'approche de la stabilisation simultanée initialement suggérée par le Théorème 3 de [FM04] et utilisée dans [WKSA08] (page *1545*), [Wie10] (page *35*) et [LJDY09] (page *4474*).

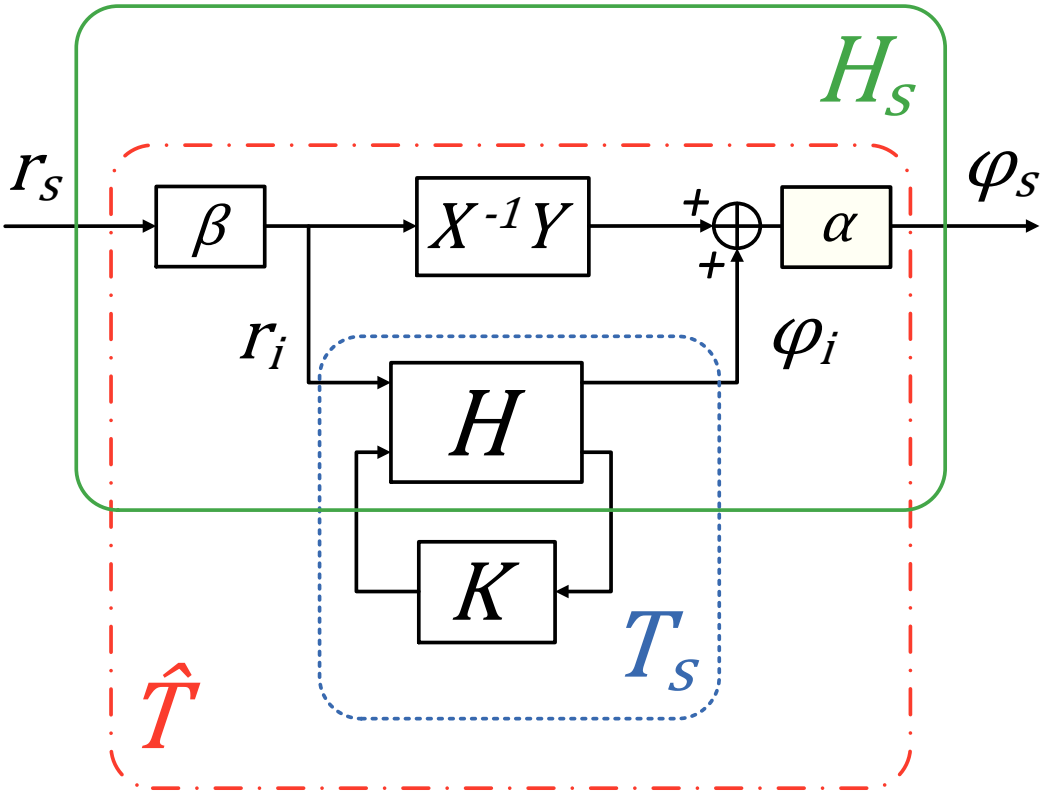


Figure 4.3: Modèle transformé pour une synthèse H_∞-standard assurant la stabilité globale. H_s représente le procédé augmenté à commander, $\alpha = (-X)^{\frac{1}{2}}$ et $\beta = \left(Z - Y^T X^{-1} Y\right)^{-\frac{1}{2}}$

Toutefois, nous n'allons pas appliquer directement ce résultat de synthèse, nous allons préférer exploiter la commande H_∞ comme une commande de systèmes MIMOs [SP05] en rajoutant les entrée et les sorties de performance afin d'assurer les spécifications de performance locale désirée en plus de la stabilité globale, c'est-à-dire la deuxième partie du Problème 4.1.

4.3.2 Performance Locale

Comme dans le cas de la stabilité globale, dans cette section, nous donnons des conditions locales sur la dynamique des sous-systèmes qui avec les conditions sur le système d'interconnexion assurent un niveau de performance locale et la stabilité globale.

Comme cela a été mentionné avant, nous introduisons dans le sous-système local des entrées et des sorties externes supplémentaires ainsi que les fonctions de pondération d'entrée W_i et de sortie W_o correspondantes (voir la Figure 4.4).

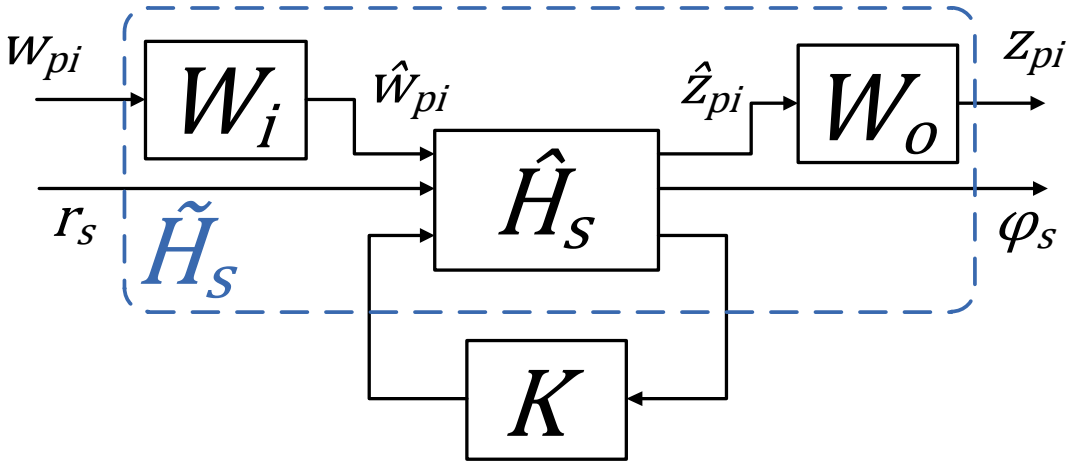


Figure 4.4: Sous-système augmenté pour les spécifications de performance locale

Le système $\widetilde{H}_s$ représenté sur la Figure 4.4 correspond au système $\widetilde{H}$ (décrit par (4.2) et représenté sur la Figure 4.2) augmenté par les entrées r_s et les sorties φ_s de stabilité nécessaires pour construire la matrice de fonctions de transfert $\widehat{T}$ (voir la Figure 4.3). Le sous-système $\widehat{H}_s$ est une version du système $\widetilde{H}_s$ sans les fonctions de pondération W_i et W_o.

Le sous-système représenté sur la Figure 4.4 est similaire à celui présenté dans la sous-section 2.5.3, page 74 du chapitre 2. En appliquant donc la synthèse H_∞-standard pour un sous-système local augmenté $\widetilde{H}_s$ (voir l'Algorithme 2.1, page 76), un correcteur K assurant la stabilité de la boucle fermée ainsi que la contrainte sur la norme H_∞ de la fonction de transfert $T_{i\to o} = \widetilde{H}_s \star K$ entre les signaux externes (4.24) peut être trouvé.

$$\left\|T_{i\to o}(s)\right\|_\infty = \left\| \begin{array}{cc} T_{w_{pi}\to z_{pi}}(s) & T_{r_S\to z_{pi}}(s) \\ T_{w_{pi}\to \varphi_S}(s) & T_{r_S\to \varphi_S}(s) \end{array} \right\|_\infty \leq \gamma \tag{4.24}$$

Si la loi de commande assurant (4.24) avec $\gamma < 1$ est trouvée, la stabilité locale est alors garantie. Ainsi par la propriété de la norme H_∞ (voir (2.24) sur la page 68) nous obtenons :

$$\begin{array}{lrcl} 1) & \left\|T_{r_S\to\varphi_S}(s)\right\|_\infty & = & \left\|\widehat{T}(s)\right\|_\infty < 1 \\ 2) & \left\|T_{w_{pi}\to z_{pi}}(s)\right\|_\infty & = & \left\|W_o(s)\, T_{\widehat{w}_{pi}\to\widehat{z}_{pi}}(s)\, W_i(s)\right\|_\infty < 1 \end{array} \tag{4.25}$$

La condition 1) de (4.25) assure la stabilité globale (voir le Théorème 4.6) tandis que les contraintes spécifiant la performance locale par un choix approprié des fonctions de pondération W_i et W_o (dont un exemple a été présenté dans les sous-sections 2.5.1

et 2.5.2 page 63 du chapitre 2) sont assurées par la condition 2) de (4.25). Les deux parties du Problème 4.1 sont alors résolues par une synthèse de la loi de commande proposée.

4.3.3 Performance Globale

La deuxième contrainte de (4.25) assure seulement la performance locale étant donné qu'un seul sous-système local a été considéré dans la synthèse de la loi de commande H_∞ proposée. Cependant, en fonction des applications, comme pour la synchronisation d'un réseau d'ADPLLs, les spécifications de performance doivent être assurées pour un réseau entier en prenant en compte tous les sous-systèmes du réseau ainsi que la dynamique d'interconnexion. Plus précisément, quelle que soit le système d'interconnexion donné, les correcteurs locaux doivent être synthétisés de manière à ce que la matrice de fonctions de transfert globale $T_{w\to z}$ satisfasse la contrainte *2)* de (4.25), c'est-à-dire la dernière partie du Problème 4.1 pour ce système d'interconnexion donné.

Selon le cahier des charges d'application considérée, il peut y avoir plusieurs contraintes fréquentielles sur plusieurs fonctions de transfert de performance. Nous appellerons la matrice de fonctions de transfert *directes de performance*, la matrice de fonctions de transfert dont les entrées et les sorties de performance sont appliquées au même sous-système. Une matrice de fonctions de transfert directes de performance pour le premier sous-système est représentée sur la Figure 4.5. La matrice de fonctions de transfert *croisées de performance* est une matrice de fonctions de transfert ayant les entrées et les sorties appliquées à différents sous-systèmes du réseau. La situation où les entrées sont appliquées au premier sous-système et les sorties sont appliquées au i-ème sous-système est illustrée sur la Figure 4.6. Pour englober un cas général, une matrice de fonctions de transfert de performance générale $T_{w\to z}$ présentée sur la Figure 4.1 est considérée dans la suite de cette étude. Elle définit par ailleurs une matrice d'interconnexion M plus générale qui n'est pas forcement égale à la matrice Laplacienne ou Adjacence (voir l'Annexe C page 134 du chapitre 3) souvent utilisées dans la littérature des systèmes Multi-Agents. Elle peut, en outre, être une matrice dynamique décrivant un système LTI d'interconnexion.

La synthèse de la loi de commande assurant la dernière partie du Problème 4.1 est inspirée par l'approche classique de la conception d'un réseau d'ADPLLs. Il s'agit là de la deuxième idée importante retenue à la fin du chapitre 1. Une fois que la synthèse d'une telle loi de commande est réalisée, la stabilité globale est testée et la différence entre le niveau de la performance locale et globale est analysée. Dans le même esprit, une extension logique de la méthode existante à la synthèse de la loi de commande assurant une borne sur cette « différence locale-globale » est proposée. Nous allons voir par la suite, qu'il est possible de minimiser cette borne par une synthèse de la loi de commande et un choix approprié de caractérisation de la dissipativité. Dans ce qui suit, nous appellerons cette approche, l'approche de la *performance globale relative* et la matrice de fonctions de transfert associée, exprimant la « différence entre les fonctions de transfert globales et locales », la matrice de *fonctions de transfert relatives pondérée*s.

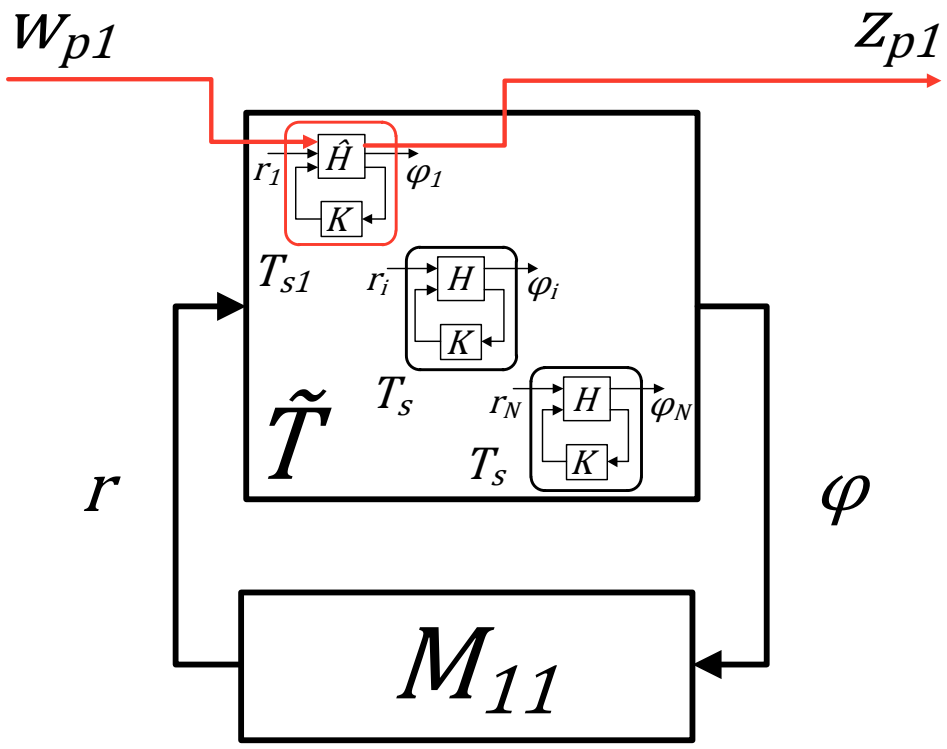


Figure 4.5: Fonction de transfert directe de performance

La fonction de transfert relative pondérée T_g est une fonction de transfert globale qui peut être facilement choisie afin de « comparer » différentes fonctions de transfert de performance (directes ou croisées) avec les fonctions de transfert de performance locales correspondantes : $T_g = f\left(T_{\widehat{w}_{pi}\to\widehat{z}_{pi}}, T_{w\to z}\right)$. Néanmoins, une condition doit être satisfaite : la fonction de transfert relative pondérée doit pouvoir être exprimée sous la forme d'une *UFT* (ou *LFT*, voir la Définition 2.6 page 74) *pour les sous-systèmes* T_s. Ce qui signifie que le bloc supérieur (respectivement inférieur) est un bloc-diagonal composé des sous-systèmes T_s comme présenté sur la Figure 4.7. Nous verrons à la fin de ce chapitre et dans le chapitre 5, en appliquant la méthode pour la synthèse de la loi de commande du réseau d'ADPLLs que cette hypothèse est pertinente et qu'elle permet de couvrir un grand nombre de fonctions de transfert de performance.

Sur la Figure 4.7, le bloc $\breve{T}$ est décomposé en deux parties : *(i)* N sous-systèmes T_s réels du réseau et k sous-systèmes T_s « virtuels » nécessaires à la construction de la fonction de transfert relative pondérée de performance globale T_g. Le système augmenté $\widetilde{M}$ décrit la topologie d'interconnexion entre ces sous-systèmes, ce qui est

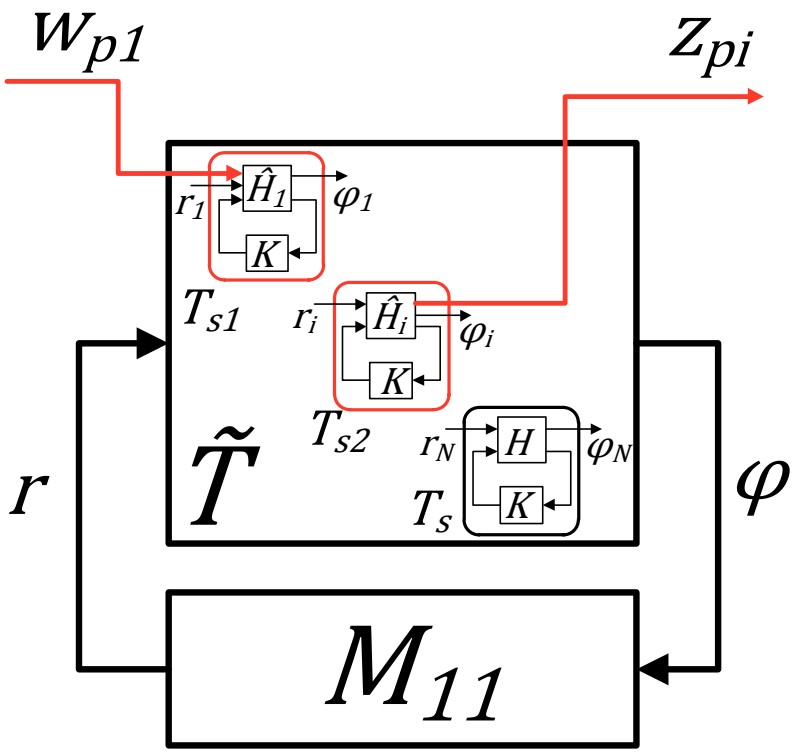


Figure 4.6: Fonction de transfert croisée de performance

résumé par l'équation suivante, similaire à celle de (4.1) :

$$\begin{aligned} \widetilde{\varphi} &= \overbrace{(I_{N+k} \otimes T_s)}^{\breve{T}} \widetilde{r} \\ \begin{bmatrix} \widetilde{r} \\ \widetilde{z} \end{bmatrix} &= \underbrace{\begin{bmatrix} \widetilde{M}_{11} & \widetilde{M}_{12} \\ \widetilde{M}_{21} & \widetilde{M}_{22} \end{bmatrix}}_{\widetilde{M}} \begin{bmatrix} \widetilde{\varphi} \\ \widetilde{w} \end{bmatrix} \end{aligned} \tag{4.26}$$

avec $T_s = \underbrace{\begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}}_{H} \star K$, $\widetilde{r}(t), \widetilde{\varphi}(t) \in \mathbb{R}^{p(N+k)}$, $\widetilde{w}(t) \in \mathbb{R}^{\widetilde{n}_w}$, $\widetilde{z}(t) \in \mathbb{R}^{\widetilde{n}_z}$.

La synthèse de la loi de commande résolvant le Problème 4.1 est donnée par le théorème suivant.

Théorème 4.10. *Soit le système bien posé et causal décrit par l'équation* (4.26). *Pour les matrices réelles données* $X = X^T \leq 0$, Y, $Z = Z^T \geq 0$ *de dimensions* $p \times p$ *et une borne donnée* $\eta > 0$ *s'il existe une matrice réelle symétrique définie positive*

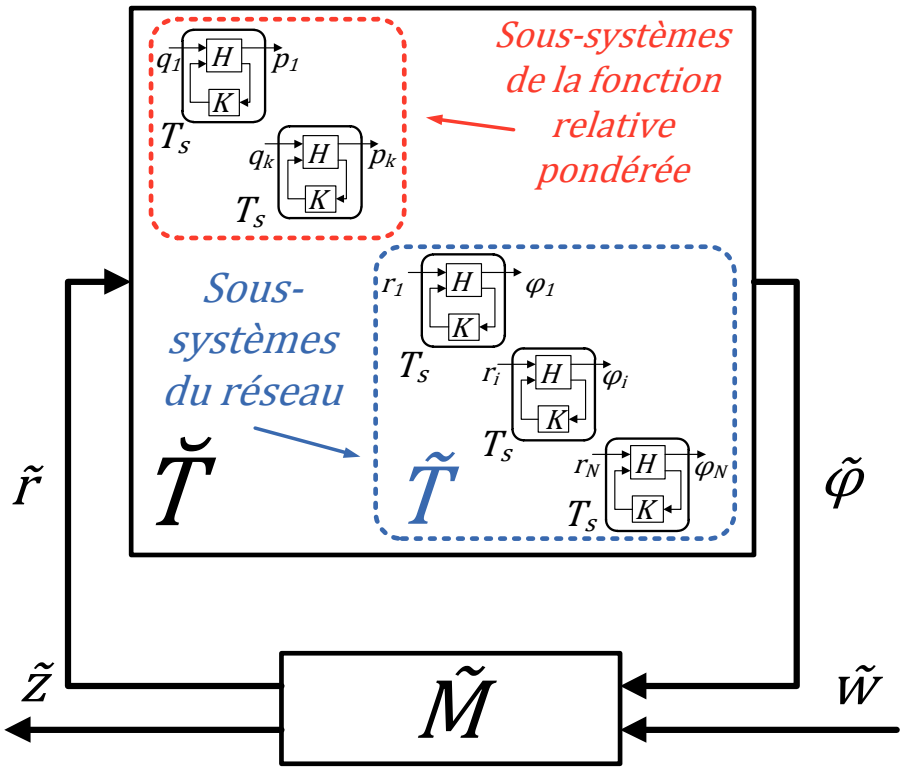


Figure 4.7: Fonction de transfert relative pondérée de la performance globale T_g sous forme d'une *UFT* en T_s

$\tilde{P} \in \mathbb{R}^{(N+k)\times(N+k)}$ *telle que*

$$\begin{bmatrix} I \\ \widetilde{M}(j\omega) \end{bmatrix}^* \begin{bmatrix} \tilde{P}\otimes X & 0 & \tilde{P}\otimes Y & 0 \\ 0 & -\eta^2 I & 0 & 0 \\ \tilde{P}\otimes Y^T & 0 & \tilde{P}\otimes Z & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} I \\ \widetilde{M}(j\omega) \end{bmatrix} \leq 0 \tag{4.27}$$

pour presque tout ω, $\omega \in \mathbb{R}$;

la loi de commande décentralisée associée au correcteur K *et assurant*

$$\|T_{i\to o}\|_\infty < 1 \tag{4.28}$$

résout le Problème 4.1 c'est-à-dire qu'elle :

(i) Stabilise chaque sous-système local (4.2) *représenté sur la Figure 4.2 ainsi que le réseau global* (4.26) *représenté sur la Figure 4.7 ;*

(ii) Assure les spécifications de la performance locale fixées par des fonction de pondérations locales W_i *et* W_o *selon la minimisation de la norme* H_∞ *de la fonction de transfert* $T_{w_{pi}\to z_{pi}}(s)$ *définie dans* (4.25) *entre les entrées* w_{pi} *et les sorties* z_{pi} *de performance c'est-à-dire*

$$\left\|T_{w_{pi}\to z_{pi}}\right\|_\infty < 1; \tag{4.29}$$

(iii) Assure les spécifications de la performance globale par une minimisation de la norme H_∞ de la fonction de transfert relative pondérée $T_g(s)$ entre les entrées $\widetilde{w}$ et les sorties $\widetilde{z}$ de performance globale définie dans (4.26) *et représentée sur la Figure 4.7 c'est-à-dire*

$$\|T_g\|_\infty \leq \eta. \tag{4.30}$$

Démonstration. Nous allons démontrer que le correcteur K résout le Problème 4.1 selon l'ordre successif.

(i) La stabilité locale découle de la condition (4.28), étant donné qu'un système instable ne peut pas avoir une norme H_∞ bornée. La stabilité globale à son tour est assurée par l'application du Théorème 4.6 avec le nouveau système d'interconnexion $\widetilde{M}$ et une nouvelle matrice symétrique définie positive $\widetilde{P}$. En effet, rappelons que nous avons démontré dans la démonstration du Théorème 4.7 que la condition (4.27) implique la $\widetilde{P}\otimes\{X,Y,Z\}$ –dissipativité du système M_{11} c'est-à-dire la première condition du Théorème 4.6, tandis que la condition (4.28) implique la condition (4.25) et donc la deuxième condition du Théorème 4.6. Par conséquent, le correcteur K résout la partie *(i)* du Problème 4.1.

(ii) La deuxième partie du Problème 4.1 est satisfaite par la condition (4.28) et la deuxième inégalité de (4.25).

(iii) La condition (4.30) est assurée par l'application directe du Théorème 4.7 avec un nouveau système d'interconnexion $\widetilde{M}$ et une nouvelle matrice symétrique définie positive $\widetilde{P}$. Le correcteur K résout la partie *(iii)* du Problème 4.1, ce qui conclut la démonstration. □

Remarque 4.11. La condition (4.27) peut être transformée en une condition LMI des variables de décision $\widetilde{P}$ et $\widetilde{R}$ pour des matrices données $X = X^T \leq 0$, Y, $Z = Z^T \geq 0$ selon la Remarque 4.4.

Le Théorème 4.10 permet de synthétiser la loi de commande décentralisée assurant la stabilité locale et globale ainsi qu'un niveau de performance locale et que la différence entre ce niveau et le niveau de performance globale définie par la fonction de transfert relative pondérée T_g n'est pas supérieure à η. Pour des matrices de caractérisation de dissipativité données $X = X^T \leq 0$, Y, $Z = Z^T \geq 0$, la condition (4.27) peut être transformée en une condition LMI de la variable de décision $\widetilde{P}$ (voir la Remarque 4.11). Une optimisation convexe (voir l'Annexe A 102 du chapitre 2) peut donc être appliquée pour minimiser la borne η et donc de minimiser cette différence.

Néanmoins, en fonction du problème considéré, le problème du choix de matrices X, Y, Z telles que les conditions de Théorème 4.10 sont satisfaites avec un niveau de performance locale et globale demandé (défini par (4.28) et η respectivement), peut être difficile. Pour ces raisons, la section suivante illustre un choix possible et efficace des matrices X, Y, Z en appliquant une optimisation quasi-convexe.

4.4 Choix de caractérisation de dissipativité

Premièrement, nous allons étudier le choix des matrices X, Y, Z pour les conditions du Théorème 4.6 c'est-à-dire le cas uniquement de la stabilité globale. Le choix approprié de la caractérisation de dissipativité pour les conditions de la performance globale et de la synthèse de la loi de commande, c'est-à-dire celles du Théorème 4.7 et du Théorème 4.10 est étudié après. Pour des raisons de clarté, l'étude se focalise sur le cas d'une interconnexion statique représentée par une matrice réelle M et des sous-systèmes SISO c'est-à-dire $p = 1$, x, y, z sont scalaires et le sous-système T_s a une entrée et une sortie scalaires. Le cas de l'interconnexion dynamique LTI peut être étudié de la même manière.

4.4.1 Conditions du Théorème 4.6

Étudions dans de plus amples détails les conditions de dissipativité du Théorème 4.6 (4.15) ou (4.12) ainsi que (4.7). La contrainte de dissipativité (4.12) du Théorème 4.6 est en réalité une contrainte sur la fonction de transfert du sous-système local T_s (ou de l'agent local). Elle peut être transformée sous la forme suivante pour $\varepsilon > 0$

$$\begin{bmatrix} T_s(j\omega) \\ 1 \end{bmatrix}^* \begin{bmatrix} x & y \\ y & z \end{bmatrix} \begin{bmatrix} T_s(j\omega) \\ 1 \end{bmatrix} > \varepsilon \tag{4.31}$$

$$\iff (T_s(j\omega) - c)^*(T_s(j\omega) - c) < r^2 + \varepsilon \tag{4.32}$$

et peut être interprétée comme « le lieu de Nyquist de la fonction de transfert $T_s(j\omega)$ est situé à l'intérieur d'un cercle avec de centre $c = -\frac{y}{x}$ et de rayon $r = \sqrt{\frac{y^2}{x^2} - \frac{z}{x}}$ ». La Figure 4.8 présente un lieu de Nyquist typique de $T_s(j\omega)$ dans le cas d'une ADPLL conçue par la méthode décrite dans le chapitre 2. Le cercle associé est représenté par une ligne pleine rouge. En observant la Figure 4.8 nous pouvons remarquer qu'une augmentation du rayon du cercle r pour un centre du cercle c fixé relaxe la contrainte (4.32) (et donc les conditions (4.15) et (4.12) du Théorème 4.6). Un unique moyen d'augmenter le rayon r est une augmentation de la valeur $-\frac{z}{x}$ qui est définie positive étant donnée la définition de $x \leq 0$ et $z \geq 0$.

D'un autre côté, la deuxième condition (4.7) du Théorème 4.6 pour $P = P^T > 0$ peut être transformée selon :

$$\left(M_{11} + \frac{y}{z}I\right)^T P \left(M_{11} + \frac{y}{z}I\right) \leq \left(\frac{y^2}{z^2} - \frac{x}{z}\right) P \tag{4.33}$$

Nous pouvons noter qu'une augmentation de la valeur $-\frac{z}{x}$ et donc du rayon du cercle (voir la Figure 4.8) contraint automatiquement la condition (4.33). Malheureusement, nous ne pouvons pas relaxer les deux contraintes à la fois et un problème d'optimisation doit être résolu qui peut être formulé de la manière suivante :

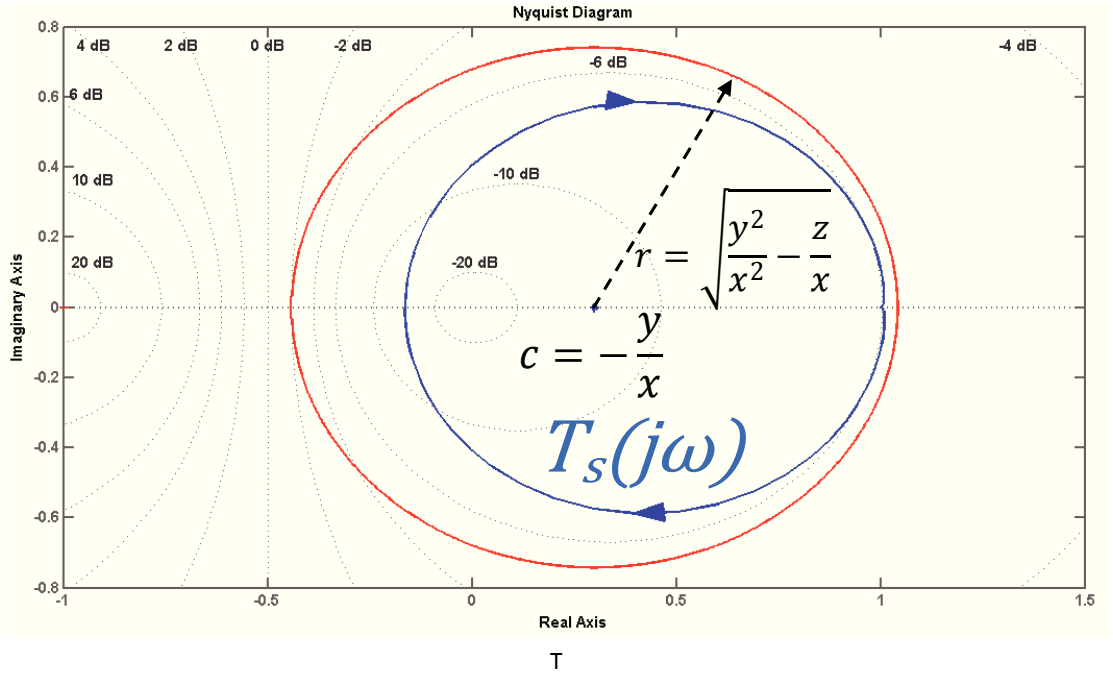


Figure 4.8: Lieu de Nyquist typique de la fonction de transfert T_s pour le cas d'une ADPLL obtenue lors de la synthèse décrite dans le chapitre (2)

Problème 4.12. *Trouver des valeurs x, y, z telles que la condition* (4.33) *soit satisfaite et que le rayon $r = \sqrt{\frac{y^2}{x^2} - \frac{z}{x}}$ soit maximal avec un centre du cercle c fixé pour la fonction de transfert locale T_s (voir* (4.32) *et la Figure 4.8).*

Pour résoudre le Problème 4.12, sans perte de généralité, une variable redondante peut être supprimée en posant par exemple $z = 1$. Cela est équivalent à la division de deux inégalités (4.7) et (4.31) par $z > 0$ et un changement de variables : $\tilde{x} = \frac{x}{z} \leq 0$, $\tilde{y} = \frac{y}{z}$. Par la suite, nous réécrivons la condition (4.33) selon les variables r et c et une matrice P symétrique définie positive pour formuler le problème d'optimisation suivant :

$$\min_{\chi, P} \quad \chi = \frac{1}{r^2} \tag{4.34}$$

$$\text{tel que} \quad \begin{aligned} P &> 0 \\ M_{11}^T P M_{11} &\leq \underbrace{\frac{1}{r^2}}_{\chi} (cM_{11} - I_N)^T P (cM_{11} - I_N) \end{aligned}$$

Pour une valeur du centre du cercle c donnée, le problème de minimisation de χ (et donc de maximisation du rayon r) pour les variables de décision $P \in \mathbb{R}^{N \times N}$ et $\chi \in \mathbb{R}^+$ est un problème de minimisation de la valeur propre généralisée maximale (voir [BEFB94] et l'Annexe A page 102 du chapitre 2). Étant un problème d'optimisation *quasi-convexe*, il peut être résolu de manière efficace en utilisant `Matlab©` par exemple. En résolvant le problème, le rayon maximal r pour la contrainte du

cercle sur la fonction de transfert T_s (4.32) pour lequel la condition de dissipativité sur la matrice d'interconnexion M_{11} (4.7) est satisfaite, peut être trouvé. Ensuite, on peut facilement démontrer que pour assurer la faisabilité de la condition (4.7) le centre du cercle c doit être contraint par :

$$|c| = \left|\frac{y}{x}\right| \leq \frac{1}{2\max_i(\left|Re\left(\lambda_i^{M_{11}}\right)\right|)} \tag{4.35}$$

où $\lambda_i^{M_{11}}$ est la i-ème valeur propre de la matrice M_{11}.

En variant la valeur du centre c dans la plage définie par (4.35), un meilleur couple c et r peut être trouvé afin de relaxer la contrainte du cercle (4.32) et par voie de conséquence la condition (4.12). Finalement, une simple transformation $\{r, c\} \Rightarrow \{x, y, z = 1\}$ est réalisée afin d'obtenir la caractérisation de $\{x, y, z\}$ −dissipativité recherchée.

Il faut noter aussi que la relaxation de la contrainte (4.32) (et donc des conditions (4.15) et (4.12)) donne plus de flexibilité pour satisfaire les contraintes de performance locale (voir (4.24) et (4.25)).

4.4.2 Conditions du Théorème 4.10

Le choix de la caractérisation de dissipativité pour les conditions du Théorème 4.7 et du Théorème 4.10 est réalisé de manière similaire. Ainsi, la condition (4.27) est exprimée de la même manière selon le centre c et le rayon r du cercle tels que définis dans la sous-section précédente (voir la Figure 4.8) :

$$\begin{array}{c} \widetilde{P} > 0 \\ \left[\begin{array}{c} I \\ \widetilde{M} \end{array}\right]^T \left[\begin{array}{cccc} -\widetilde{P} & 0 & c\widetilde{P} & 0 \\ 0 & -\eta^2 I & 0 & 0 \\ c\widetilde{P} & 0 & (r^2 - c^2)\,\widetilde{P} & 0 \\ 0 & 0 & 0 & I \end{array}\right] \left[\begin{array}{c} I \\ \widetilde{M} \end{array}\right] \leq 0. \end{array} \tag{4.36}$$

Cette condition est transformée grâce à une factorisation, en la condition :

$$\begin{array}{c} \widetilde{P} > 0 \\ \left[\begin{array}{c} I \\ \widetilde{M} \end{array}\right]^T \left[\begin{array}{cccc} I & 0 & -cI & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{array}\right]^T \left[\begin{array}{cccc} -\widetilde{P} & 0 & 0 & 0 \\ 0 & -\eta^2 I & 0 & 0 \\ 0 & 0 & r^2\widetilde{P} & 0 \\ 0 & 0 & 0 & I \end{array}\right] \left[\begin{array}{cccc} I & 0 & -cI & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{array}\right] \left[\begin{array}{c} I \\ \widetilde{M} \end{array}\right] \leq 0 \end{array} \tag{4.37}$$

En faisant le changement de variable $\widehat{P} = r^2\widetilde{P}$ et $\beta = \eta^2 r^2$, on obtient le problème

d'optimisation suivant :

$$\min_{\chi,P} \quad \chi = \tfrac{1}{r^2} \tag{4.38}$$

$$\widehat{P} \quad > \quad 0$$

$$\text{tel que} \quad \Phi^T \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \widehat{P} & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \Phi \quad \leq \quad \underbrace{\frac{1}{r^2}}_{\chi} \Phi^T \begin{bmatrix} \widehat{P} & 0 & 0 & 0 \\ 0 & \beta I & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \Phi$$

où $\Phi = \begin{bmatrix} I & 0 & -cI & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} I \\ \widetilde{M} \end{bmatrix}$.

Pour les valeurs de β et c fixées, le problème d'optimisation (4.38) avec les variables de décision $\widehat{P} \in \mathbb{R}^{(N+k)\times(N+k)}$ et $\chi \in \mathbb{R}^+$ est un problème de minimisation de la valeur propre généralisée maximale (voir [BEFB94] et l'Annexe A page 102 du chapitre 2). Ce problème peut être résolu de manière efficace en utilisant `Matlab©` par exemple. Finalement, une simple transformation $\{r, c\} \Rightarrow \{x, y, z\}$ est réalisée afin d'obtenir la caractérisation recherchée de $\{x, y, z\}$ −dissipativité.

Remarque 4.13. En minimisant la valeur de χ dans (4.38), nous maximisons à la fois le rayon du cercle r et donc nous relaxons la contrainte sur la fonction de transfert locale du sous-système T_s et nous minimisons la borne η sur la fonction de transfert relative de performance globale. En effet, étant donné que le rapport $\beta = \eta^2/\chi$ est une constante fixée, une minimisation de χ implique une minimisation de $\eta^2 = \chi\beta$. Le paramètre β est un paramètre de réglage de la condition (4.38). Si le problème d'optimisation (4.38) n'a pas de solution ou le rayon calculé $r = 1/\sqrt{\chi}$ contraint trop fortement la condition (4.12), la valeur de β doit être augmentée pour relaxer les conditions du problème d'optimisation (4.38).

Remarque 4.14. La résolution du problème d'optimisation (4.38) permet de trouver *(i)* la caractérisation de dissipativité x, y, z à assurer par le correcteur local pour garantir la stabilité globale et *(ii)* la borne supérieure sur la fonction de transfert relative pondérée $T_g(s)$, c'est-à-dire une distance maximale η entre les fonctions de transfert de performance locales et globales. Cette borne supérieure est déterminée avant la synthèse de la loi de commande fixant *a priori* la borne sur la « distance » entre les fonctions de transfert globales et locales. La minimisation de cette borne avec un choix approprié de contraintes de performance locales remplit les contraintes de performance globale une fois que les correcteurs sont synthétisés.

4.4.3 L'algorithme de la synthèse de la loi de commande décentralisée

Pour résumer, la synthèse de la loi de commande décentralisée résolvant le Problème 4.1 est transformée en une étape d'initialisation, c'est-à-dire un choix ap-

proprié de fonctions de transfert globale et locale pondérées, et en une étape de résolution *(i)* du problème d'optimisation (4.38) et *(ii)* du problème de synthèse H_∞ locale de la loi de commande assurant la contrainte (4.28) du Théorème 4.10. Cela peut être résumé par l'algorithme de synthèse de la loi de commande ci-après. Plus de détails sur différents points de l'algorithme ainsi que leurs illustrations sur un exemple simple sont reportés dans la section suivante de ce chapitre.

Algorithme 4.1 Synthèse de la loi de commande décentralisée

1. Résoudre le problème d'optimisation (4.34) pour plusieurs valeurs du centre $c = -\frac{y}{x}$ restreintes par (4.35) afin de trouver le meilleur couple centre c et rayon r du cercle au sens de relaxation de la contrainte (4.12), puis calculer les valeurs de la caractérisation de dissipativité x, y, z ;
2. Choisir les fonctions de pondération fixant, selon le cahier des charges, les spécifications de la performance locale définie par (4.29) ;
3. Appliquer la synthèse H_∞ de la loi de commande (voir l'Algorithme 2.1, page 76) sur le système étendu de la Figure 4.4 dont les entrées et les sorties définissent la contrainte de la performance locale (4.29) ainsi que la contrainte de la stabilité globale (4.12) exprimées sous la condition (4.28) ;
4. Si le correcteur K assurant $\gamma < 1$ est trouvé, c'est-à-dire la contrainte (4.28) du Théorème 4.10 est satisfaite, les deux premières parties du Problème 4.1 sont alors résolues, aller à l'étape 6 ;
5. Si le correcteur K n'est pas trouvé tel que $\gamma < 1$ c'est-à-dire la contrainte (4.28) du Théorème 4.10 n'est pas satisfaite, modifier les contraintes locales de performance et revenir à l'étape 3.
6. Choisir la fonction de transfert relative pondérée globale T_g qui peut être transformée sous forme d'une UFT (ou LFT) en sous-systèmes locaux T_s (voir la Figure 4.7) et qui évalue la « différence » entre les fonctions de transfert globales et locales de performance ;
7. Résoudre le problème de minimisation de η sous contrainte (4.27) pour les valeurs x, y, z trouvées à l'étape 1 (voir la Remarque 4.11)
8. Si le problème de minimisation de η sous contrainte (4.27) pour les valeurs x, y, z fixées avant est faisable, aller à l'étape 12, sinon calculer les explicitement en utilisant l'expression de la fonction de transfert T_g pour la loi de commande calculée à l'étape 3 et aller à l'étape 10 ;
9. Si la valeur β est définie, augmenter cette dernière et aller à l'étape 11, sinon continuer ;
10. Fixer le paramètre de réglage $\beta = \eta^2 r^2$ pour les valeurs η et r obtenues avant ;
11. Résoudre le problème d'optimisation (4.38) afin de minimiser la valeur χ et donc de minimiser η et maximiser le rayon r relaxant la contrainte de la fonction de transfert locale T_s (voir l'équation (4.28) du Théorème 4.10), calculer les nouveaux x, y, z ;
12. Adapter, si nécessaire, les fonctions de pondération fixant les spécifications de performance locale définie par (4.29) afin que, multipliées par η, elles fixent les contraintes de performance globale respectant le cahier de charge ;
13. Appliquer la synthèse H_∞ de la loi de commande (voir l'Algorithme 2.1, page 76) sur le système étendu de la Figure 4.4 ;
14. Si le correcteur K assurant $\gamma < 1$ est trouvé, le Problème 4.1 est alors résolu, c'est-à-dire que la stabilité locale et globale ainsi que la performance locale et globale (en terme de la fonction de transfert relative pondérée T_g) sont garanties, fin de l'algorithme ;
15. Si le correcteur K n'est pas trouvé ou la valeur de γ est trop élevée, telle que la contrainte (4.28) du Théorème 4.10 ne peut pas être satisfaite, soit modifier les contraintes locales de performance et revenir à l'étape 13 soit revenir à l'étape 9.

4.5 Exemple numérique

Dans cette section, nous allons appliquer à titre d'illustration les résultats obtenus dans ce chapitre sur un exemple numérique simple du réseau d'ADPLLs LTI présenté à la fin du chapitre 2. Il s'agit du réseau de modèles d'ADPLLs LTI (Modèle de synthèse) interconnectées selon la topologie (b) (voir la sous-section 2.8.1 page 96), c'est-à-dire le cas pour lequel la synthèse classique locale ne garantit pas la stabilité. Nous reprenons alors les valeurs numériques de la Configuration FPGA (voir le Tableau 2.1 page 55). On suppose également que pour respecter les spécifications de performance globale et donc le cahier des charges, il faut que la fonction de transfert globale $T_{w\to z} = S_{globale}$ (voir la formulation du Problème 4.1) entre le signal de référence $w = \varphi_{Ref}$ à l'entrée du réseau et l'erreur de son suivi par le premier nœud $z = \varepsilon_{Ref1} = \varphi_{Ref} - \varphi_1$, respecte la même contrainte fréquentielle que la fonction de sensibilité locale S_{locale} de chaque ADPLL (voir la Figure 4.9 et la première équation de (2.46), page 77).

4.5.1 Stabilité globale performance locale

Tout d'abord suivant les étapes 1-5 de l'Algorithme 4.1 proposé, illustrons le résultat de la stabilité globale du Théorème 4.6 en le combinant avec les contraintes de performance locale (4.24) et (4.25) c'est-à-dire le deux premières parties du Problème 4.1. Les contraintes de performance locale restent les mêmes comme dans l'exemple de synthèse de la loi de commande d'une seule ADPLL (voir les équations (2.46) et la Figure 2.34 page 80). Cela consiste en l'étape 2 de l'Algorithme 4.1, cependant on rajoute une contrainte supplémentaire sur la dynamique de l'ADPLL locale (4.12) assurant la stabilité globale. La solution du problème d'optimisation de l'étape 1 de l'Algorithme 4.1 (Problème 4.12 ou (4.34)) fournit les valeurs de caractérisation de dissipativité suivantes :

$$\begin{array}{rcl} r & = & 0.9548 \\ c & = & 0.1 \\ x & = & -1.1092 \\ y & = & 0.1109 \\ z & = & 1 \end{array} \tag{4.39}$$

En appliquant la synthèse H_∞ pour le sous-système augmenté représenté sur la Figure 4.4 (l'étape 3 de l'Algorithme 4.1), on obtient le correcteur d'ordre 1 simplifié [1] sous la forme

$$K(s) = K_{stab}(s) = \frac{0.6(s+190.5)}{s} \tag{4.40}$$

avec la valeur de γ correspondante : $\gamma = 1.27$. Le dépassement de valeur $\gamma > 1$ est expliqué par le fait que les contraintes de performance locales et la contrainte sup-

1. Après avoir approximé le pôle en basses fréquences issu du presque-intégrateur par un pôle en zéro pur.

plémentaire sur la stabilité globale (4.12) deviennent trop excessives et l'algorithme de synthèse n'arrive pas à trouver la solution satisfaisant toutes les contraintes. En effet, ceci est illustré sur la Figure 4.9 où les fonctions de transfert en boucle fermée obtenues (la ligne rouge pleine) sont comparées aux mêmes fonctions de transfert en boucle fermée lors de la synthèse locale à la fin du chapitre 2 (la ligne bleue en trait-pointillé). Bien que la valeur de $\gamma > 1$ signifie que certaines contraintes ne sont pas satisfaites, la contrainte de stabilité globale $\left\|\widehat{T}\right\|_\infty < 1$ (voir la condition (4.12) du Théorème 4.6) est tout de même satisfaite. En effet, en calculant cette valeur pour le correcteur synthétisé, on obtient :

$$\left\|\widehat{T}\right\|_\infty = 0.99.$$

Selon le Théorème 4.6, le correcteur (4.40) n'assurant pas forcément les contraintes de performance locale assure quand même la stabilité globale du réseau d'ADPLLs identiques interconnectées selon la topologie (b) (voir la Figure 2.53 page 98).

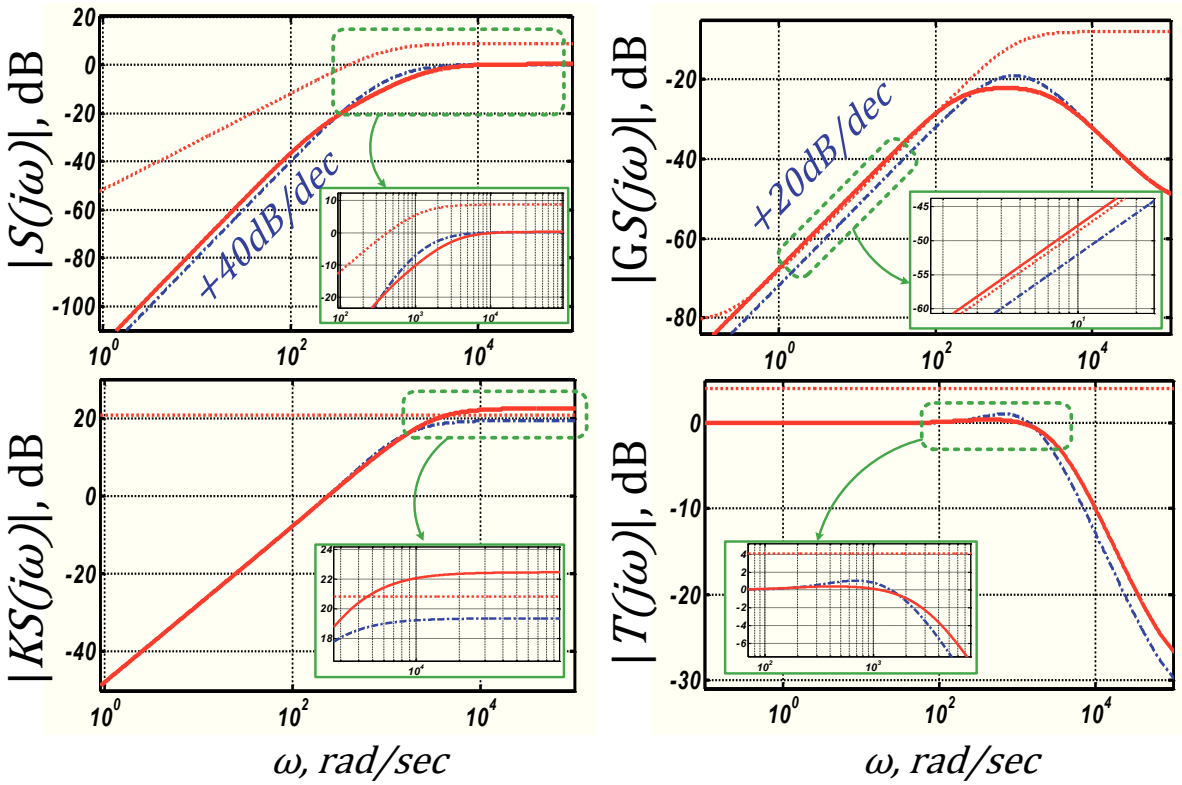


Figure 4.9: Fonctions de sensibilité obtenues après la synthèse H_∞ *(i)* locale (la ligne bleue en trait-pointillée), *(ii)* assurant la stabilité globale (la ligne rouge pleine), les contraintes fréquentielles pour la performance locale (la ligne rouge pointillée)

Le calcul de la valeur propre avec la partie réelle maximale de la représentation d'états du système global $\lambda_{max} = -83.89$ ainsi que la Figure 4.10 (le lieu de Nyquist généralisé d'après le Théorème 3.2 page 118) et la Figure 4.11 (les simulations temporelles du Modèle de synthèse d'ADPLLs interconnectées selon la topologie (b))

confirment ce résultat.

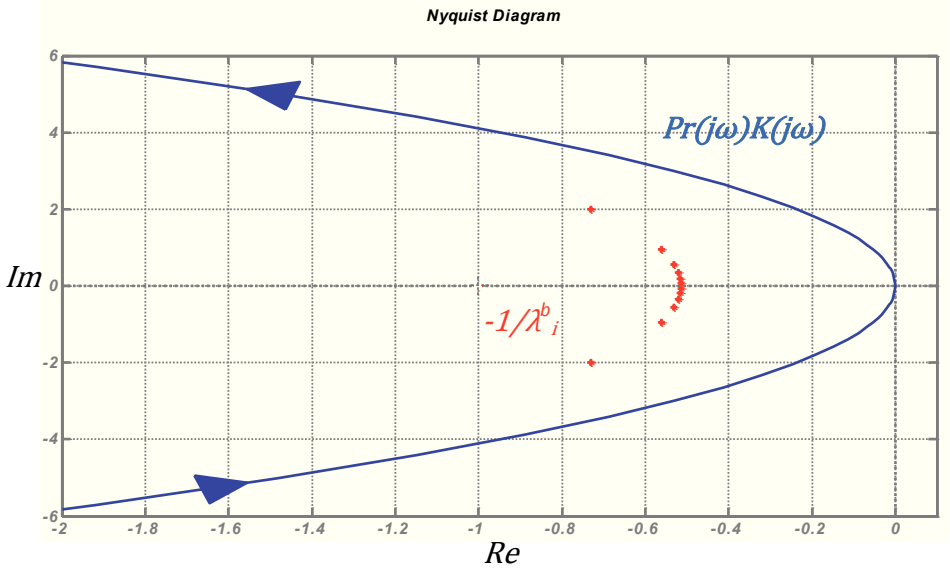


Figure 4.10: Lieux de Nyquist généralisé du réseau obtenu d'ADPLLs interconnectées selon la topologie (b) de la Figure 2.53

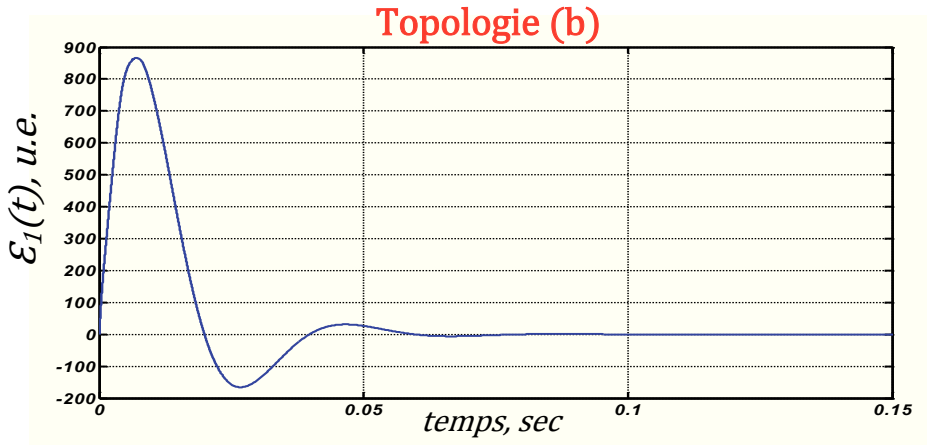


Figure 4.11: Simulation temporelle de la topologie (b) du réseau décrit sur la Figure 2.53 : évolution de l'erreur de suivi $\varepsilon_{Ref1} = \varphi_{Ref} - \varphi_1$ pour le premier nœud du réseau

Il est possible ensuite, en faisant quelques itérations des étapes 2-5 de l'Algorithme 4.1, d'adapter les spécifications de performance locale (en adaptant certaines contraintes) et la loi de commande afin d'assurer la condition $\gamma < 1$ c'est-à-dire la condition (4.28) du Théorème 4.10. Comme cette procédure est une procédure de synthèse H_∞ standard (illustrée dans la section 2.5 page 60 du chapitre 2), nous ne rentrerons pas dans les détails de cette démarche, nous continuons plutôt les étapes 6-15 de l'Algorithme 4.1 assurant la performance globale.

4.5.2 Performance globale

Étudions maintenant le cas de la performance globale en illustrant le résultat du Théorème 4.10. Commençons par l'étape 6 de l'Algorithme 4.1. Comme discuté avant, le cahier des charges, pour la fonction de transfert globale $T_{w\to z} = S_{globale}$ (voir la formulation du Problème 4.1) entre le signal de référence $w = \varphi_{Ref}$ à l'entrée du réseau et l'erreur de son suivi par le premier nœud $z = \varepsilon_{Ref1} = \varphi_{Ref} - \varphi_1$, fixe la même contrainte fréquentielle que pour la fonction de sensibilité locale S_{locale} de chaque ADPLL comme dans le cas d'une seule ADPLL (voir la Figure 4.9 et la première équation de (2.46), page 77). Dans ce cas-là, la fonction de transfert locale correspond à la fonction de sensibilité $S_{locale} = S = (1 + KG)^{-1}$ et la fonction de transfert globale $T_{w\to z} = S_{globale}$ est décrite par (4.1) avec [2]

$$
\begin{aligned}
M_{11}^{ij} &= \begin{cases} \dfrac{1}{m_i} & \text{, si l'ADPLL } i \text{ reçoit information de l'ADPLL } j \\ 0 & \text{, sinon} \end{cases} \\
M_{12} &= \left[\begin{array}{cc} \dfrac{1}{m_1} & \underbrace{0 \quad \cdots \quad 0}_{N-1} \end{array} \right]^T \\
M_{21} &= \left[\begin{array}{cc} -1 & \overbrace{0 \quad \cdots \quad 0}^{N-1} \end{array} \right] \\
M_{22} &= 1
\end{aligned}
\tag{4.41}
$$

où $N = 13$ représente le nombre d'ADPLLs en réseau et m_i le nombre d'entrées de la i-ème ADPLL.

Il est important de noter que dans notre exemple numérique, les fonctions de transfert globale et locale sont les fonctions de transfert SISO. En traçant ces fonctions de transfert pour le cas de la loi de commande obtenue dans la sous-section précédente (4.40) avec la contrainte fréquentielle correspondante (voir la Figure 4.12), on remarque clairement que les spécifications de performance globale ne sont pas satisfaites. En effet, il existe une certaine « différence » entre la fonction de transfert locale et globale apportée par la dynamique du réseau. Cette différence étant importante, elle provoque le dépassement de la contrainte fréquentielle défini par le cahier des charges préalable.

Appliquons la synthèse de la loi de commande proposée dans le Théorème 4.10 et l'Algorithme 4.1 pour respecter cette contrainte globale. Étant donné que l'on peut assurer les contraintes fréquentielles sur les fonctions de transfert locales (c'est-à-dire un niveau de la performance locale) et une « distance » maximale entre ces fonctions de transfert et les fonctions de transfert de performance globale associées, nous allons utiliser l'approche de la performance globale relative. Selon la démarche expliquée dans l'Algorithme 4.1 (l'étape 6), il faut tout d'abord trouver une fonction

2. La référence est prise en compte comme une entrée supplémentaire de la première PLL, c'est-à-dire $m_1 = 2$.

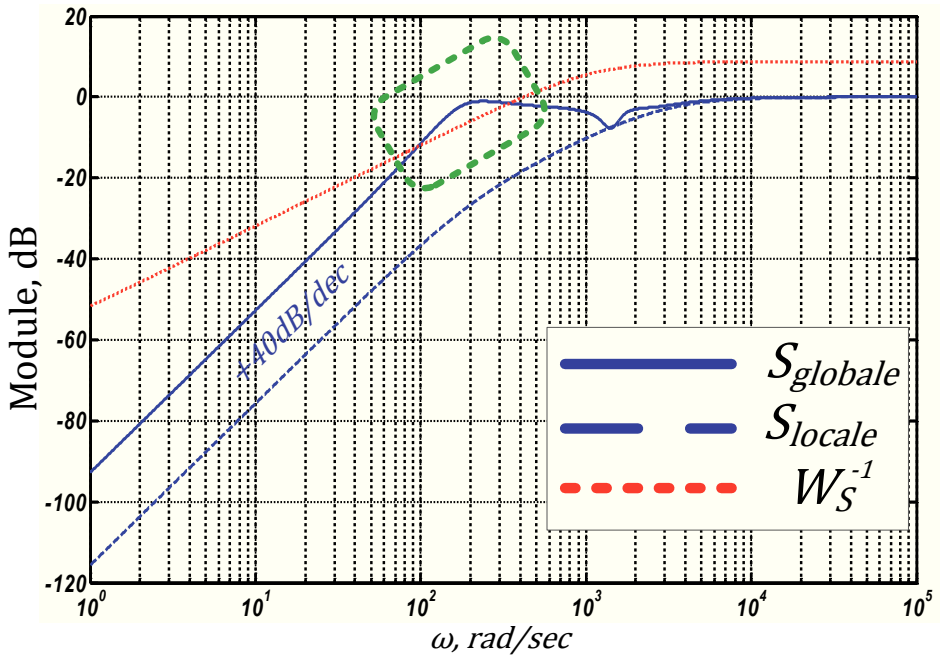


Figure 4.12: Fonction de transfert globale $S_{globale}$ (la ligne bleue pleine), fonction de transfert locale S_{locale} (la ligne bleue en trait long) et la contrainte fréquentielle correspondante (la ligne rouge en trait court)

de transfert relative pondérée T_g nécessaire pour évaluer la « différence » entre la performance locale et la performance globale et ensuite la transformer sous forme d'une UFT (ou LFT) en T_s.

Étant donné que notre objectif de performance est exprimé en échelle logarithmique, la « distance » entre les deux fonctions de transfert (globale et locale) va être évaluée en terme de leur rapport. Choisissons la fonction de transfert relative pondérée $T_g(s)$ sous la forme suivante :

$$T_g(s) = \frac{S_{globale}(s)}{S_{locale}(s) + \alpha} \tag{4.42}$$

où $\alpha \in \mathbb{R}^+$.

L'utilisation du paramètre α dans l'expression de la fonction de transfert relative pondérée (4.42) est motivée par le risque de problèmes numériques induit par l'évaluation de l'expression de la fonction de transfert relative (4.42). En effet, pour $\alpha = 0$ et étant donné que la fonction de sensibilité S tend vers zéro en basses fréquences (comportement intégrateur pur de la boucle ouverte $K(s)G(s)$), il en résulte une division par zéro, c'est-à-dire la fonction de transfert relative pondérée (4.42) est mal posée. Par conséquence, une valeur non nulle de α doit être imposée pour assurer la validité de l'expression (4.42) dans le domaine de pulsation voulu.

La performance globale est évaluée par le calcul de la borne supérieure η de la norme H_∞ de $T_g(s)$:

$$\|T_g(s)\|_\infty \leq \eta \Leftrightarrow |T_g(j\omega)| \leq \eta, \forall \omega \in \mathbb{R} \tag{4.43}$$

Dans l'échelle logarithmique et pour le cas SISO, (4.43) est équivalent à :

$$\begin{array}{rcl} 20\log_{10}\left(|S_{globale}\left(j\omega\right)|\right) - 20\log_{10}\left(|S_{locale}\left(j\omega\right)+\alpha|\right) & \leq & 20\log_{10}\left(\eta\right)=\mu \iff \\ |S_{globale}\left(j\omega\right)|_{dB} - |S_{locale}\left(j\omega\right)+\alpha|_{dB} & \leq & \mu \end{array}, \quad (4.44)$$

pour $\forall\omega \in \mathbb{R}$.

Dans le domaine de fréquences où $\alpha \ll |S_{locale}\left(j\omega\right)|$, le paramètre α peut être négligé. La fonction de transfert relative pondérée est alors exprimée comme $T_g\left(s\right) \approx \frac{S_{globale}(s)}{S_{locale}(s)}$ et la condition (4.44) est équivalente à :

$$|S_{globale}\left(j\omega\right)|_{dB} - |S_{locale}\left(j\omega\right)|_{dB} \leq \mu, \forall\omega \in \mathbb{R} \quad (4.45)$$

Autrement dit, dans l'échelle logarithmique, le module de la fonction de transfert de performance globale $S_{globale}$ est borné par le module de la fonction de transfert de performance locale S_{locale} plus une valeur de $\mu\, dB$. Cette situation est illustrée sur la Figure 4.13. Si la fonction de transfert locale (la ligne bleue pleine sur la Figure 4.13) plus $\mu\, dB$ fixe une borne (la ligne bleue en trait court sur la Figure 4.13) inférieure à la contrainte fréquentielle de performance à satisfaire selon le cahier des charges (la ligne rouge en trait long sur la Figure 4.13), alors la fonction de transfert globale (la ligne verte en trait interrompu mixte sur la Figure 4.13) satisfera cette contrainte.

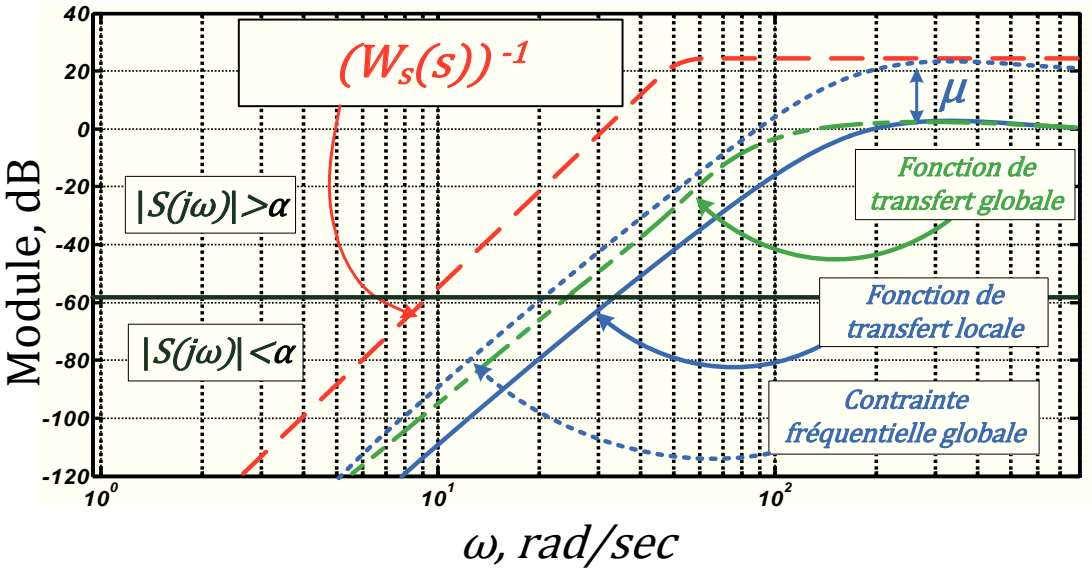


Figure 4.13: Contrainte fréquentielle sur la fonction de transfert de performance globale.

Prenons notre exemple d'ADPLLs interconnectées selon la topologie (b). Le domaine des pulsations nous intéressant est le domaine au-dessus de $100\, rad/sec$ (voir la Figure 4.12) car, à partir de cette pulsation, la contrainte fréquentielle n'est plus satisfaite. La loi de commande synthétisée doit alors imposer, si possible, la contrainte fréquentielle désirée dans ce domaine de pulsation [3]. On choisit alors pour notre

3. Dans tous les cas, si nous choisissons la valeur α telle que $20\log_{10}\left(\alpha\right) < -30dB$, le domaine

exemple la valeur de $\alpha = 0.01$ correspondante à $20\log_{10}(\alpha) = -40dB$. En conclusion, la borne supérieure μ sur le module de la fonction de transfert relative pondérée (4.42) implique que le module de la fonction de transfert globale $S_{globale}$ est borné par le module de la fonction de transfert S_{locale} plus $\mu\, dB$ dans le domaine de pulsation où $\alpha \ll |S_{locale}(j\omega)|$.

L'étape suivante est la transformation de la fonction de transfert relative pondérée (4.42) en une UFT en T_s. Étant donné que par définition de la fonction de sensibilité $S = 1 - T$ et que $S_{locale} = S$ et $T_s = T$ dans l'application considérée, on obtient sur la Figure 4.14 la représentation graphique de (4.42).

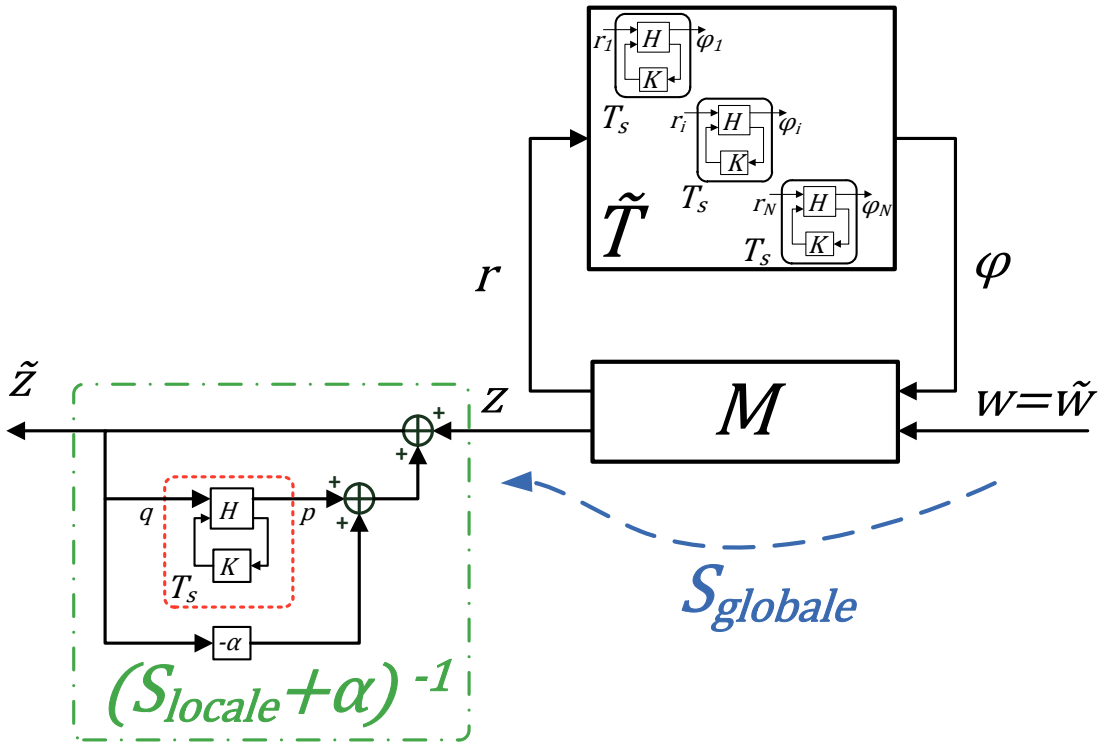


Figure 4.14: Fonction de transfert relative pondérée $T_g(s)$ en forme d'une UFT en T_s

Le schéma représenté sur la Figure 4.14 est facilement transformé sous la forme

de fréquence où la valeur de α ne peut pas être négligée n'est pas pertinent. En effet, même si la contrainte était dépassée dans le domaine de fréquence où $20\log_{10}(\alpha) \geq -30dB$, l'erreur de suivi serait petite et pourrait être négligée.

décrite sur le schéma de la Figure 4.7, page 162 avec $k = 1$ et :

$$\begin{aligned} \breve{T} &= I_{N+1} \otimes T_s \\ \widetilde{M} &= \begin{bmatrix} \frac{1}{1+\alpha} & \frac{1}{1+\alpha} M_{21} & \frac{1}{1+\alpha} M_{22} \\ 0 & M_{11} & M_{12} \\ \frac{1}{1+\alpha} & \frac{1}{1+\alpha} M_{21} & \frac{1}{1+\alpha} M_{22} \end{bmatrix} \\ \widetilde{r} = \begin{bmatrix} q & r_1 & \cdots & r_N \end{bmatrix}^T, \widetilde{\varphi} &= \begin{bmatrix} p & \varphi_1 & \cdots & \varphi_N \end{bmatrix}^T, \widetilde{w} = w, \widetilde{z} = q \end{aligned} \tag{4.46}$$

Ayant une description détaillée de la fonction de transfert relative pondérée (4.26) évaluant la performance globale, on peut appliquer le Théorème 4.10 pour la synthèse de la loi de commande.

A l'aide d'optimisation convexe LMI, une premier étape naturelle serait de trouver et de minimiser la borne η de la condition (4.27) pour la caractérisation de dissipativité trouvée précédemment dans le cas de la stabilité globale (c'est-à-dire x, y, z de (4.39)). Cela correspond à l'étape 7 de l'Algorithme 4.1, 169. En réalité, comme cela a été mentionné à la fin de la section 4.3, l'optimisation convexe avec les mêmes valeurs de x, y, z que pour le cas de la stabilité peut ne pas donner des résultats satisfaisants. C'est effectivement le cas de l'exemple numérique considéré ici et l'algorithme d'optimisation pour les x, y, z de (4.39) ne trouve pas de valeur de η^2 et $\widetilde{P}$ telles que la condition (4.27) soit satisfaite. On est alors amené à choisir la caractérisation de dissipativité en appliquant l'optimisation quasi-convexe discutée dans la section 4.4, c'est-à-dire les étapes 10 et 11 de l'Algorithme 4.1.

Tout d'abord, une valeur initiale du paramètre de réglage β doit être fixée. Dans la sous-section 4.4.2, ce paramètre a été introduit pour transformer le problème d'optimisation non-convexe en un problème quasi-convexe dont la solution existe et est efficace en général. Nous rappelons que $\beta = \eta^2 r^2$. Nous avons déjà à notre disposition l'ordre de grandeur du paramètre β après avoir effectué la synthèse de la loi de commande garantissant la stabilité globale (voir la seconde partie de l'étape 8 de l'Algorithme 4.1). En effet, les paramètres $r = 0.9548$ de (4.39) et $\eta \approx 40$ ce qui correspond à $\mu = 20\log_{10}(\eta) \approx 32\,dB$ sont alors choisis en se basant sur (4.42) et les fonctions de transfert $S_{globale}$ et S_{locale} obtenues précédemment (voir la Figure 4.12).

L'exécution de quelques itérations de l'algorithme d'optimisation quasi-convexe (4.38)

(c'est-à-dire l'étape 11 de l'Algorithme 4.1) donne les nouveaux résultats suivants :

$$\begin{aligned} r &= 0.9047 \\ c &= 0.1 \\ x &= -1.237 \\ y &= 0.124 \\ z &= 1 \\ \eta &= 14.57 \\ \mu &= 23.27\,dB \end{aligned} \tag{4.47}$$

Notons que la borne sur la « différence » entre la fonction de transfert globale et la fonction de transfert locale a diminué de $\approx 9\,dB$, ce qui suffit pour respecter la contrainte de performance globale. Cependant, nous notons également que le rayon du cercle r a diminué en même temps que la borne. Cela implique une contrainte plus forte sur la dynamique de l'ADPLL locale (c'est-à-dire la fonction de transfert $\widehat{T}$, voir les conditions (4.28) et (4.25)), ce qui semble logique car il faut plus contraindre la dynamique de chaque ADPLL en réseau pour « gagner » $\approx 9\,dB$ sur la borne supérieure de la fonction de transfert globale. Ce résultat est confirmé par la synthèse de la loi de commande finale.

D'après la discussion ci-dessus, afin de garantir que $\gamma < 1$, il est nécessaire d'adapter la contrainte fréquentielle sur la fonction de transfert en boucle fermée locale KS (voir la Figure 4.9), c'est-à-dire au lieu de $20.8dB$, nous fixons une contrainte de $45.2dB$ en HF. Les autres contraintes pour les autres fonctions de transfert locales en boucle fermée restent les mêmes. Cela fait appel au point 12 de l'Algorithme 4.1.

La synthèse H_∞ (le point 13 de l'Algorithme 4.1) fournit le correcteur suivant après la même approximation de son modèle :

$$K(s) = K^1_{perf}(s) = \tfrac{4.7(s+27.4)}{s} \tag{4.48}$$

avec la valeur de $\gamma = 0.999$. La norme H_∞ de la fonction de transfert $\widehat{T}$ est égale à $\left\|\widehat{T}\right\|_\infty = 0.995$, l'Algorithme 4.1 débouche donc sur la stabilité du réseau selon le Théorème 4.6. Les fonctions de transfert locales en boucle fermée correspondantes sont représentées sur la Figure 4.15.

La loi de commande obtenue (4.48) assure aussi la performance globale demandée (voir la Figure 4.16) : la contrainte fréquentielle (voir la ligne verte en trait interrompu mixte sur la Figure 4.16) pour la fonction de transfert globale $S_{globale}$ (voir la ligne bleue pleine sur la Figure 4.16) fixée par la fonction de transfert locale S_{locale} (voir la ligne bleue en trait long sur la Figure 4.16) plus $\mu \approx 23\,dB$ est respectée. Étant donné que cette contrainte est inférieure [4] à la contrainte du cahier des charges

4. On ne parle pas de domaine de HF où la contrainte fréquentielle pour la fonction de transfert globale (voir la ligne verte en trait interrompu mixte sur la Figure 4.16) n'est pas inférieure à la contrainte fréquentielle d'après le cahier des charges (voir la ligne rouge en trait court sur la Figure 4.16). On peut démontrer, néanmoins, [TM09, TM10] qu'en HF les deux fonctions de transfert $S_{globale}$ et S_{locale} coïncident et sont égales à 1.

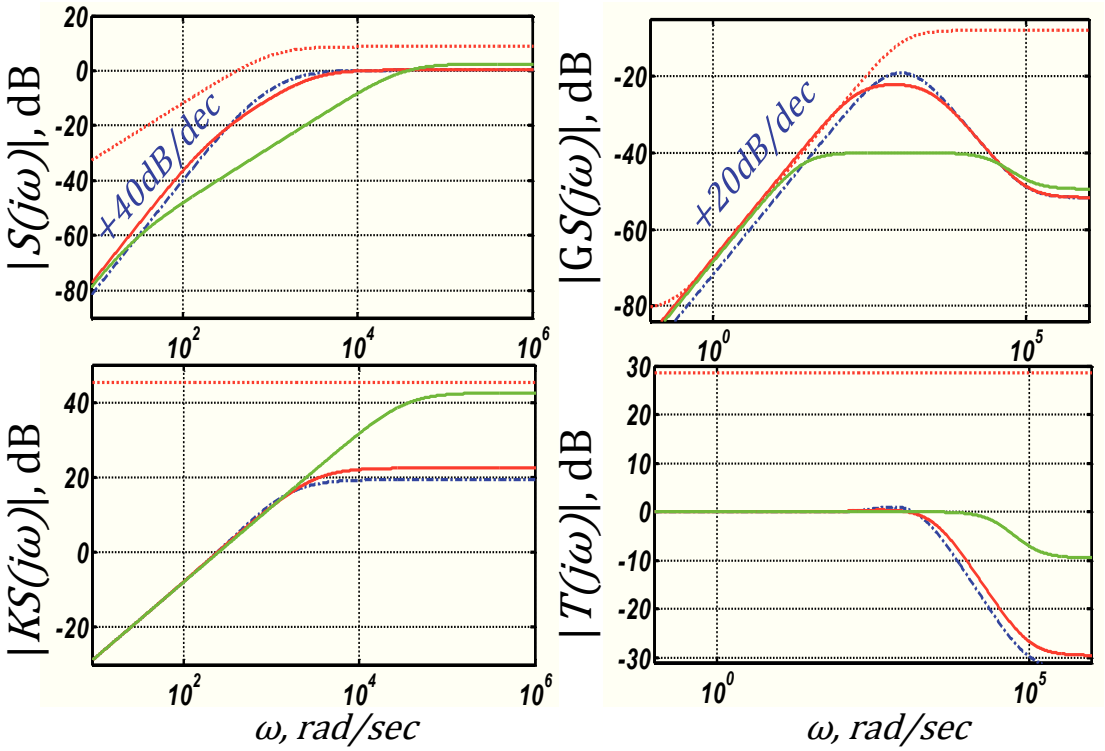


Figure 4.15: Fonctions de sensibilité obtenues après la synthèse H_∞ *(i)* locale (la ligne bleue en trait-pointillé), *(ii)* assurant la stabilité globale (la ligne rouge pleine), *(iii)* assurant la performance globale (la ligne verte pleine), les contraintes fréquentielles pour la performance locale (la ligne rouge pointillée)

(voir la ligne rouge en trait court sur la Figure 4.16), les spécifications de performance globale sont alors aussi respectées. La Figure 4.17 illustre l'amélioration de la performance de la loi de commande assurant la performance globale (4.48) par rapport au cas de la loi de commande assurant seulement la stabilité globale (4.40). Le calcul explicite de la norme H_∞ de la fonction de transfert relative pondérée T_g de (4.42) obtenue donne la valeur $\mu^{réel} = 21.6\,dB$. Ce résultat est très important pour illustrer le niveau faible de conservatisme de cette méthode de synthèse. En effet, étant donné que la méthode se base sur des conditions suffisantes, la borne estimée maximale μ peut être conservative. La borne réelle $\mu^{réel}$ proche de la borne estimée montre un conservatisme faible de la méthode.

Il est également possible, pour les valeurs de la caractérisation de dissipativité fixées, d'augmenter la rapidité du réseau d'ADPLLs en augmentant la contrainte sur la bande passante de la fonction de transfert $S_{globale}$. Le correcteur correspondant est alors obtenu :

$$K(s) = K_{perf}^2(s) = \frac{7.1(s+178.2)}{s} \tag{4.49}$$

Les fonctions de transfert globales $S_{globale}$ pour les trois lois de commande (4.40), (4.48) et (4.49) correspondant aux lois de commande assurant uniquement la stabilité, la performance et la performance « rapide » globales respectivement sont

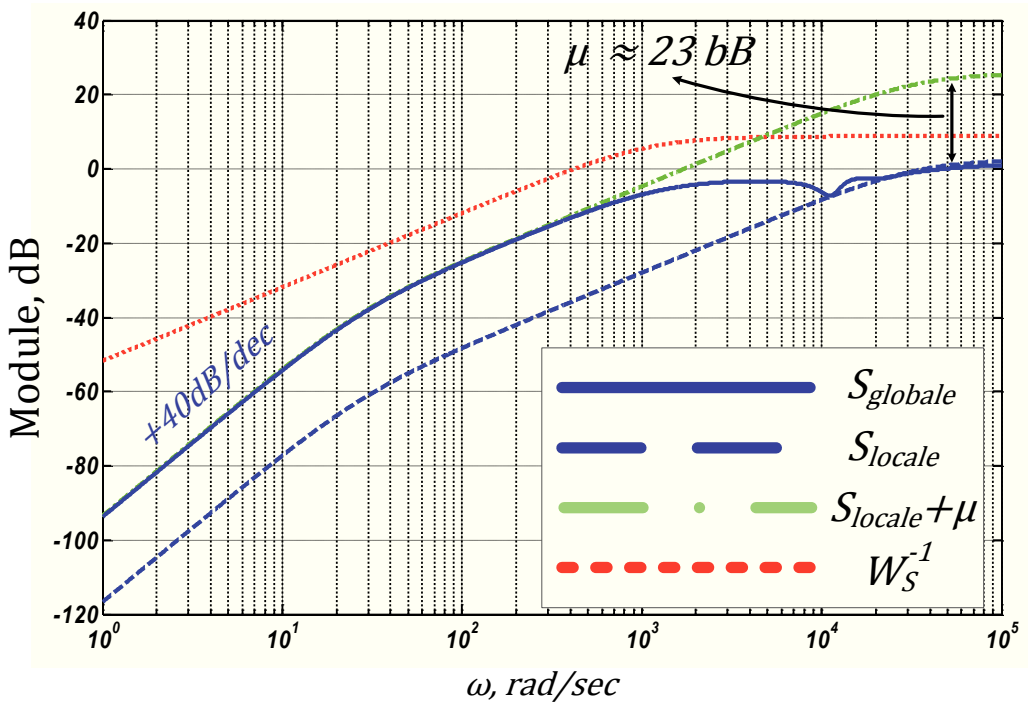


Figure 4.16: Fonction de transfert globale $S_{globale}$ (la ligne bleue pleine), fonction de transfert locale S_{locale} (la ligne bleue en trait long) et fonction de transfert locale $+ \mu\, dB$ (la ligne verte en trait interrompu mixte) et la contrainte fréquentielle correspondante (la ligne rouge en trait court) obtenues après la synthèse de la loi de commande assurant la performance globale

représentées sur la Figure 4.18. Les résultats de simulation temporelle des réseaux obtenus des Modèles de synthèse d'ADPLLs sous forme d'évolution dans le temps d'erreurs de suivi de la référence par la première ADPLL $\varepsilon_1 = \varphi_{Ref} - \varphi_1$ avec les mêmes conditions initiales et le même signal en entrée sont représentés sur la Figure 4.19. On remarque que le temps de réponse le plus court d'après les simulations du Modèle de synthèse correspond à la fonction de transfert globale avec la bande passante la plus large (la ligne rouge pleine).

Conclusion

Dans ce chapitre, nous avons développé une méthode originale de synthèse de la loi de commande décentralisée assurant la stabilité et un niveau de performance du système de grande dimension composé de sous-systèmes identiques. L'information importante sur l'identité des sous-systèmes a été prise en compte lors du développement de la méthode, ce qui, en combinaison avec l'approche proposée de la performance relative, a permit d'obtenir une méthode à la fois efficace et peu conservative de synthèse de la loi de commande d'un système complexe de grande dimension.

A la fin du chapitre nous avons illustré l'application de la méthode de synthèse de la loi de commande décentralisée proposée sur un exemple simple d'ADPLLs

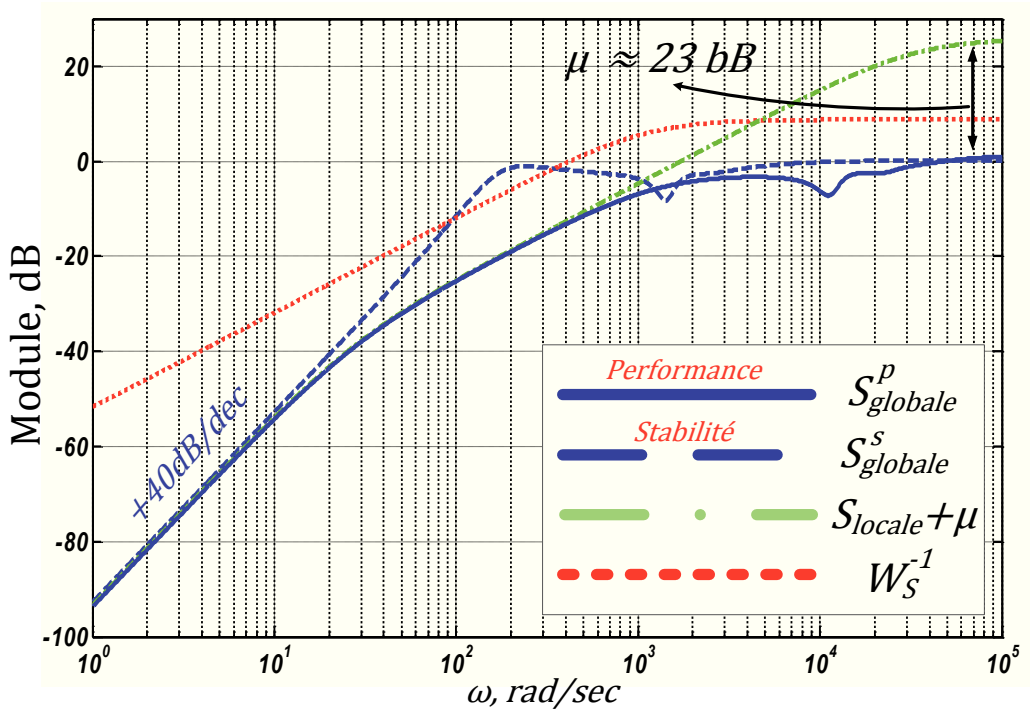


Figure 4.17: Fonction de transfert globale $S^p_{globale}$ (la ligne bleue pleine) issue du correcteur assurant la performance globale, fonction de transfert globale $S^s_{globale}$ (la ligne bleue en trait long) issue du correcteur assurant seulement la stabilité globale et fonction de transfert locale + $\mu\, dB$ (la ligne verte en trait interrompu mixte) et la contrainte fréquentielle correspondante (la ligne rouge en trait court)

interconnectée selon la topologie (b) décrit sur la Figure 2.53, page 98. La méthode est efficace et permet d'assurer la stabilité et la performance globales du réseau pour lequel la synthèse classique locale n'a pas donné de résultats satisfaisants. Dans le chapitre suivant, pour valoriser la méthode présentée dans ce chapitre, nous allons étudier sa mise en œuvre pratique pour la synthèse de la loi de commande de réseaux d'ADPLLs avec différentes topologies.

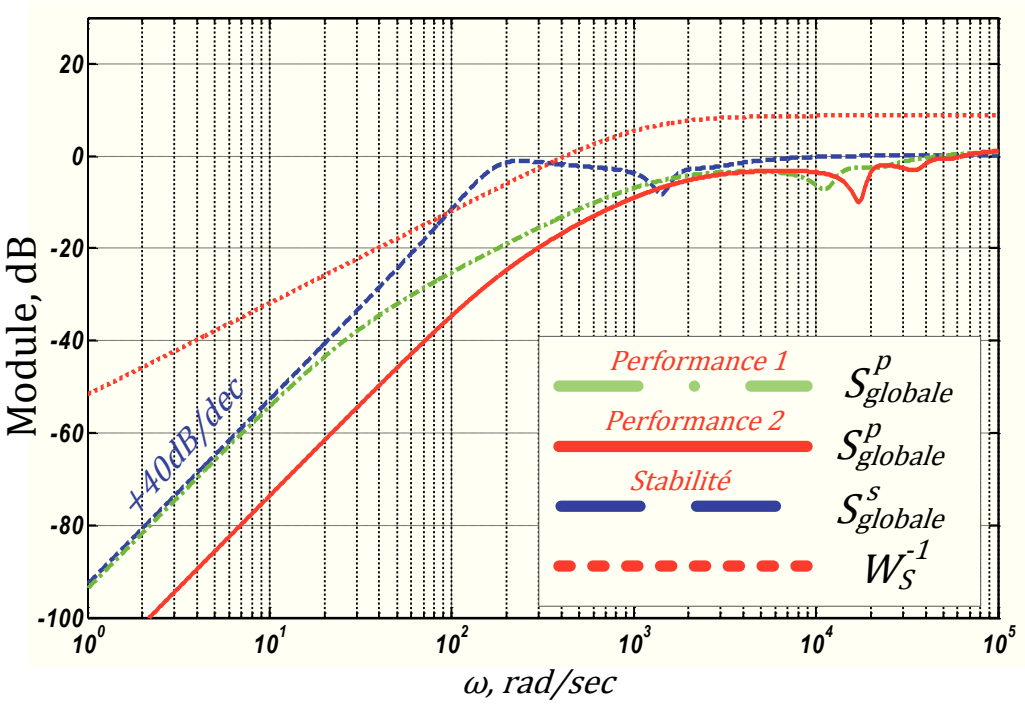


Figure 4.18: Fonction de transfert globale $S^s_{globale}$ (la ligne bleue en trait long) correspondante à la loi de commande (4.40) assurant seulement la stabilité globale, fonction de transfert globale $S^p_{globale}$ (la ligne verte en trait interrompu mixte) correspondante à la loi de commande (4.48) assurant la performance et fonction de transfert globale $S^p_{globale}$ (la ligne rouge pleine) correspondante à la loi de commande (4.49) assurant la performance « rapide » ainsi que la contrainte fréquentielle correspondante (la ligne rouge en trait court)

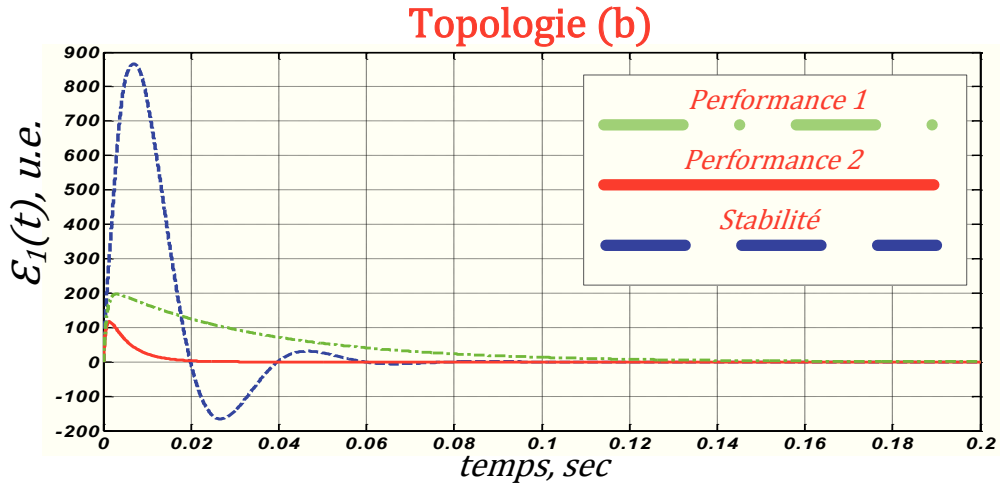


Figure 4.19: Simulations temporelles de la topologie (b) de réseau décrit sur la Figure 2.53 pour trois lois de commande (4.40), (4.48) et (4.49), l'évolution de l'erreur de suivi $\varepsilon_{Ref1} = \varphi_{Ref} - \varphi_1$ pour le premier nœud du réseau

5 Conception automatique du réseau d'ADPLLs

Introduction

Dans le chapitre précédent, nous avons développé une méthode de synthèse de la loi de commande décentralisée pour un système de grande dimension composé de sous-systèmes identiques. A titre d'illustration, la méthode a été appliquée au réseau de modèles interconnectés de PLLs simples LTIs (Modèles de synthèse) selon la topologie (b) de la Figure 2.53, page 98. Dans ce chapitre, nous allons déployer la méthode de synthèse de la loi de commande proposée pour la problématique considérée dans cette thèse, à savoir une conception des réseaux actifs de génération et de distribution du signal d'horloge.

Tout d'abord, nous présenterons les différentes topologies considérées du réseau dans cette étude. Par la suite, dans la section suivante, nous décrirons deux modèles linéaires du réseau de génération et de distribution du signal d'horloge, à savoir : le Modèle de synthèse et le Modèle d'analyse. Les subtilités non-linéaires du système liées au réseau et aux interconnexions telles que les points d'équilibre parasites appelés aussi « Mode-Locking », non abordés lors de l'étude d'une seule ADPLL, seront examinées avec plus de détails dans la section suivante lors d'une description du Modèle de simulation du réseau. Ensuite, une synthèse de la loi de commande assurant la stabilité globale et la performance globale du système LTI (Modèle de synthèse) sera appliquée selon la méthode proposée dans le chapitre 4 pour différentes topologies d'interconnexion du réseau. Par la suite, une méthode de choix d'emplacement du signal de référence dans le réseau sera décrite.

La méthode de synthèse de la loi de commande proposée est basée sur le fait que les sous-systèmes du réseau sont identiques. Puisqu'en pratique, à cause des dispersions technologiques de fabrication, ceci n'est jamais le cas, dans la section suivante, la méthode de μ-analyse du réseau d'ADPLLs complet obtenu est proposée afin d'assurer que la stabilité globale et/ou la performance globale soient conservées aussi pour le réseau d'ADPLLs incertaines (Modèle d'analyse).

A la fin du chapitre, pour tester l'efficacité de la loi de commande synthétisée et vérifier la validité des hypothèses prises durant la modélisation, les résultats des simulations sur le modèle complexe non-linéaire (Modèle de simulation) seront présentés.

5.1 Topologies d'interconnexion considérées

Dans ce chapitre, nous considérons trois types de topologie du réseau d'ADPLLs interconnectées dont les graphes associés sont représentés sur la Figure 5.1. La synthèse de la loi de commande ainsi que l'analyse de l'influence d'incertitudes pour la Topologie A (Figure 5.1a) seront discutées en détails. Concernant les deux autres topologies, seule une brève discussion ainsi que les résultats principaux de la synthèse seront mis en évidence afin de confirmer la généralité de la méthode proposée.

Une première étape logique de l'étude d'un réseau complexe est l'étude d'un réseau ayant une certaine régularité et/ou symétrie dans sa topologie d'interconnexion. C'est pour ces raisons que la première topologie (Topologie A) du réseau d'ADPLLs a été choisie régulière et correspondante au graphe non-orienté (voir l'annexe C du chapitre 3, page 134 pour plus de détails sur ce sujet).

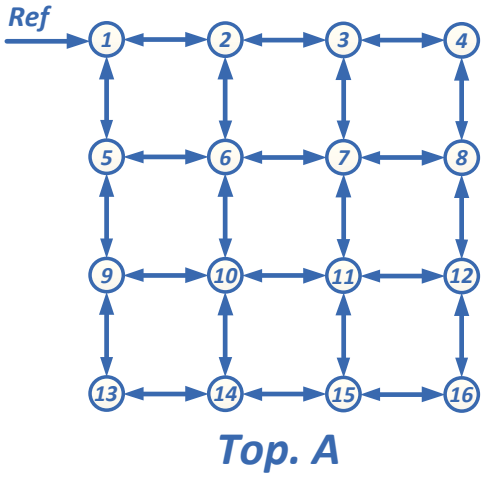
La Topologie B du réseau d'ADPLLs possède aussi une certaine régularité, cependant elle correspond au graphe orienté formant un anneau extérieur connecté à la référence. Cela permet de forcer les ADPLLs situées à l'intérieur du réseau à la suivre plus rapidement, ce qui est intéressant du point de vue de l'application, et d'illustrer la méthode de synthèse proposée dans le cas de réseaux asymétriques.

La Topologie C est à son tour irrégulière, asymétrique et a été choisie de manière aléatoire tout en garantissant la connexité du graphe orienté correspondant par rapport à la référence. Cela garantit qu'il existe au moins un chemin entre la référence et chaque sous-système du réseau, ce qui implique la possibilité de synchroniser toutes les ADPLLs avec la référence. Cette topologie permettra de tester l'efficacité de la méthode de synthèse développée dans le cas d'un réseau complexe, irrégulier, asymétrique d'ADPLLs interconnectées.

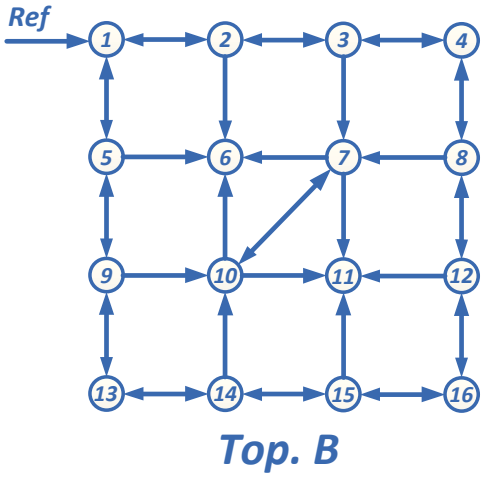
Comme dans le cas d'une seule ADPLL (voir le chapitre 2) nous allons considérer les trois modèles du réseau d'ADPLLs interconnectées. Les Modèles de synthèse et d'analyse seront présentés ensemble dans la section suivante, alors que le Modèle de simulation ainsi que l'aspect pratique le plus important de sa réalisation, lié au cyclisme de la phase, sera présenté dans la section 5.3.

5.2 Modèle de synthèse et d'analyse du réseau

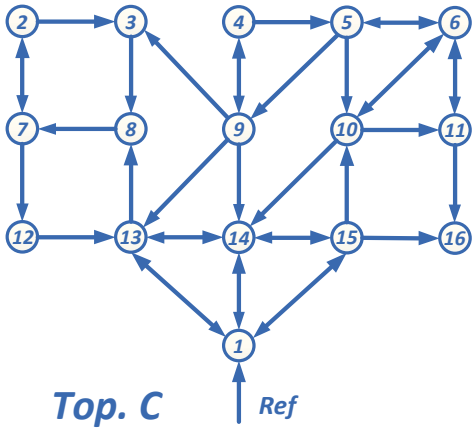
Les réseaux d'ADPLLs pour toutes les topologies représentées sur la Figure 5.1 sont composés d'ADPLLs identiques dont le schéma est représenté sur la Figure 5.2. Identique signifie ici que chaque ADPLLs possède les mêmes sous-systèmes élémentaires tels que le *PD*s, le filtre *F*, le *DCO* et le diviseur de fréquence que dans le cas d'une seule ADPLL. Seul le nombre d'entrées m_i, et donc de comparateurs de phases et les coefficients des sommes pondérées $\frac{1}{m_i}$, changent en fonction de la position de l'ADPLL au sein du réseau. Il est aussi important de préciser qu'afin de réduire le nombre de comparateurs de phase utilisés et l'influence des retards de propagation du signal d'horloge entre les différents oscillateurs, les *PD*s sont partagés par les ADPLLs adjacentes et sont placés à mi-distance de chaque oscillateur (voir la Figure 5.2). Pour plus de détails sur la réalisation pratique de la structure du



(a) Topologie A



(b) Topologie B



(c) Topologie C

Figure 5.1: Quatre topologies du réseau d'interconnexion d'ADPLLs considérées

réseau d'ADPLLs interconnectées ainsi que son implémentation, le lecteur est invité à consulter [JZA+11, CZRC11].

Les valeurs de tous les paramètres de toutes les ADPLLs de réseaux pour toutes les topologies (représentées sur la Figure 5.1a,b,c) sont les mêmes et sont résumées dans le Tableau 2.1, page 55.

En réalisant la même démarche de modélisation pour chaque sous-système de chaque ADPLL et la transformation discret-continue comme cela a été effectuée dans la section 2.2 du chapitre 2, on obtient deux modèles de la $i^{ième}$ ADPLL au sein du réseau : le Modèle de synthèse (Figure 5.3 qui est identique à la Figure 2.51, page 97) et le Modèle d'analyse (Figure 5.4).

Sur ces figures, le signal φ_i représente la phase de la $i^{ième}$ ADPLL, les signaux φ_j, φ_k, φ_l, *etc.* représentent les phases des ADPLLs voisines à la $i^{ième}$ ADPLL. Les indices j, k, l *etc.* ainsi que leur nombre m_i dépend de la topologie du réseau d'interconnexion et de la position de la $i^{ième}$ ADPLL au sein de ce réseau. Le nombre de *PD*s de la

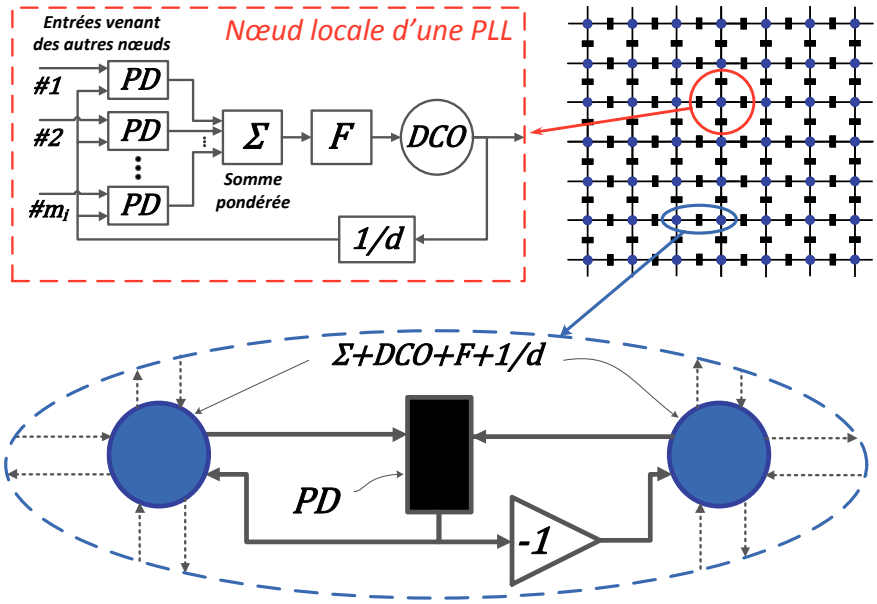


Figure 5.2: ADPLLs du réseau actif de génération et de distribution du signal d'horloge

$i^{ième}$ ADPLL ainsi que le coefficient de division de la somme pondérée (représentée sur la Figure 1.14, page 26) sont alors égaux au nombre d'entrées m_i de la $i^{ième}$ ADPLL.

Étant donné que la $i^{ième}$ ADPLL a m_i comparateurs de phase, elle a le même nombre de blocs d'incertitude Δ_{PD}^{ji} et de signaux de perturbation $q_{PDj,i}$ correspondant aux comparateurs de phase. Tous les autres signaux ainsi que les blocs d'incertitudes sont identiques à ceux de l'ADPLL seule et ont déjà été présentés dans la section 2.2 du chapitre 2, page 43.

Modélisation des incertitudes du réseau. Il est nécessaire, néanmoins, d'apporter quelques précisions concernant la modélisation des incertitudes. Étant donné que le circuit est composé de plusieurs ADPLLs identiques et non pas d'une seule, il peut y avoir lors de la fabrication deux types d'incertitudes sur leurs paramètres [Nas01] :

(i) les incertitudes d'un circuit à l'autre (*Die-to-Die* variations en anglais) c'est-à-dire les incertitudes provoquées par le fait que différents réseaux d'ADPLLs ont été fabriqués sur différentes plaquettes de silicium sous différentes conditions de fabrication ;

(ii) les incertitudes au sein du même circuit (*Within-Die* variations en anglais) c'est-à-dire les incertitudes provoquées par le fait que la gravure du circuit s'est déroulée sous différentes conditions pour différents endroits du circuit et/ou les incertitudes provoquées par les variations de dimensions physiques de transistors au sein du même circuit.

L'influence de la variation des dimensions physiques des transistors sur l'incertitude peut être minimisée par des techniques bien connues en conception microélectronique

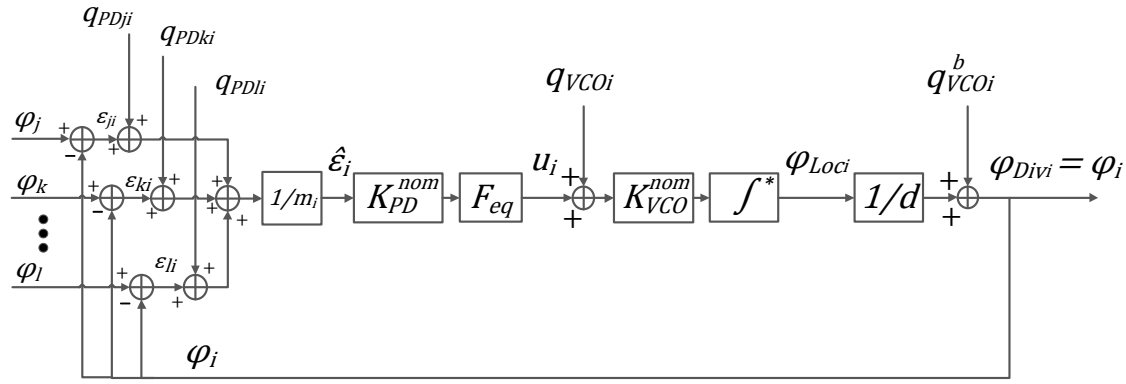


Figure 5.3: Modèle de synthèse de la $i^{ième}$ ADPLLs au sein du réseau

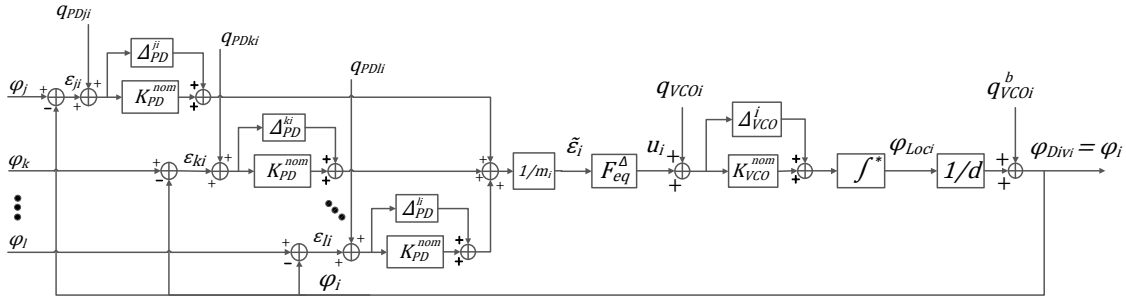


Figure 5.4: Modèle d'analyse de la $i^{ième}$ ADPLLs au sein du réseau

[Hof04, GHLM09]. C'est pour cette raison que pour un circuit bien conçu de petite taille le deuxième type d'incertitude est généralement négligé. Un circuit réalisant une seule ADPLL est un circuit de relativement petite taille et en conséquence durant la modélisation d'une seule ADPLL (voir la section 2.2 du chapitre 2, page 43) l'incertitude de deuxième type est négligée ou plutôt incorporée dans l'incertitude plus importante du premier type. Un circuit réalisant un réseau d'ADPLLs interconnectées est un circuit de grande dimension avec plusieurs sous-systèmes identiques et nous devons, lors de sa modélisation, prendre en compte l'existence de deux types d'incertitudes. Ceci est encore plus d'actualité pour des technologies plus récentes (au-dessous de $130\,nm$ pour la longueur de transistor utilisé).

Une des possibilités pour prendre en compte durant la modélisation ces deux types d'incertitude pour chaque paramètre est leur combinaison en une seule incertitude différente pour chaque ADPLL. Ceci résulte en une modélisation plus simple du réseau. Cependant, de part la description plus fine du modèle, ce qui potentiellement permettra de réduire le conservatisme du problème d'analyse considéré par la suite, nous allons distinguer ces deux types d'incertitude pour chaque paramètre incertain du modèle du réseau. En effet, la méthode de μ-analyse a été développée notamment pour ces raisons-là, elle prend en compte l'information sur la structure et le type des incertitudes afin d'avoir le résultat d'analyse le plus proche de la réalité. Nous allons ainsi modéliser les incertitudes de chaque ADPLL du réseau par une partie commune pour toutes les ADPLLs et une partie variable d'une ADPLL à l'autre.

Les incertitudes de filtres F sont une exception. Les filtres dans notre architecture sont numériques et la seule source importante d'incertitudes est le nombre réduit de bits pour codage des coefficients (voir la section 2.2 du chapitre 2, page 43) qui ne varie pas d'une ADPLL à l'autre. La modélisation suivante est alors proposée pour les incertitudes paramétriques de la $i^{ième}$ ADPLL du réseau :

$$\begin{aligned}
&\Delta_{PD}^{ji} = \Delta_{PD} + \widetilde{\Delta}_{PD}^{ji} = K_{PD}^{gab} \cdot \delta_{PD} + \widetilde{K}_{PD}^{gab} \cdot \delta_{PD}^{ji};\\
&\Delta_{VCO}^{i} = \Delta_{VCO} + \widetilde{\Delta}_{VCO}^{i} = K_{VCO}^{gab} \cdot \delta_{VCO} + \widetilde{K}_{VCO}^{gab} \cdot \delta_{VCO}^{i};\\
&\Delta_{P}^{i} = \Delta_{P} = K_{P}^{gab} \cdot \delta_{P};\\
&\Delta_{I}^{i} = \Delta_{I} = K_{I}^{gab} \cdot \delta_{I},
\end{aligned} \tag{5.1}$$

où Δ_{PD}, Δ_{VCO}, Δ_P, Δ_I, K_{PD}^{gab}, K_{VCO}^{gab}, K_P^{gab}, K_I^{gab} sont respectivement les incertitudes communes et leurs gabarits (dont les valeurs sont les mêmes comme dans le Tableau 2.1 page 55), δ_{PD}, δ_{VCO}, δ_P, δ_I sont les incertitudes communes réelles normalisées : $|\delta_{PD}| < 1$,$|\delta_{VCO}| < 1$,$|\delta_P| < 1$,$|\delta_I| < 1$, δ_{PD}^{ji}, δ_{VCO}^{i}, δ_P^i, δ_I^i sont les incertitudes réelles normalisées variables : $\left|\delta_{PD}^{ji}\right| < 1$,$|\delta_{VCO}^{i}| < 1$,$|\delta_P^i| < 1$,$|\delta_I^i| < 1$ pour $\forall i$ et $\forall j$, ainsi que $\widetilde{K}_{PD}^{gab}$, $\widetilde{K}_{VCO}^{gab}$, $\widetilde{K}_P^{gab}$, $\widetilde{K}_I^{gab}$ sont les gabarits d'incertitudes variables dont les valeurs numériques sont présentées dans le Tableau 5.1.

Paramètre	**Configuration FPGA**
Le gabarit d'incertitude variable du gain du PD	$\widetilde{K}_{PD}^{gab} = 0.05K_{PD}^{nom}(5\%)$
Le gabarit d'incertitude variable du gain du DCO	$\widetilde{K}_{VCO}^{gab} = 0.05K_{VCO}^{nom}$ (5%)

Table 5.1: Configuration des paramètres pour les incertitudes variables en réseau

L'expression des paramètres incertains de la $i^{ième}$ ADPLL prend alors la forme :

$$\begin{aligned}
K_{VCO}^{i} &= K_{VCO}^{nom} + \overbrace{K_{VCO}^{gab} \cdot \delta_{VCO}}^{\Delta_{VCO}} + \overbrace{\widetilde{K}_{VCO}^{gab} \cdot \delta_{VCO}^{i}}^{\widetilde{\Delta}_{VCO}^{i}};\\
K_{PD}^{ji} &= K_{PD}^{nom} + \overbrace{K_{PD}^{gab} \cdot \delta_{PD}}^{\Delta_{PD}} + \overbrace{\widetilde{K}_{PD}^{gab} \cdot \delta_{PD}^{ji}}^{\widetilde{\Delta}_{PD}^{ji}};\\
K_{P}^{i} &= K_{P}^{nom} + \overbrace{K_{P}^{gab}\delta_{P}}^{\Delta_{P}} = K_{P};\\
K_{I}^{i} &= K_{I}^{nom} + \overbrace{K_{I}^{gab}\delta_{I}}^{\Delta_{I}} = K_{I}.
\end{aligned} \tag{5.2}$$

D'après la Figure 5.4, le signal $m_i\widetilde{\varepsilon}_i$ a l'expression suivante :

$$\begin{aligned} m_i\widetilde{\varepsilon}_i &= \sum_{q\in\mathcal{N}_i}^{m_i} \overbrace{(\varepsilon_{qi}+q_{PDqi})}^{\widehat{\varepsilon}_{qi}} \left(K_{PD}^{nom}+K_{PD}^{gab}\cdot\delta_{PD}+\widetilde{K}_{PD}^{gab}\cdot\delta_{PD}^{qi}\right) = \cdots \\ &\cdots \left(K_{PD}^{nom}+K_{PD}^{gab}\cdot\delta_{PD}\right)\left(\sum_{q\in\mathcal{N}_i}^{m_i}\widehat{\varepsilon}_{qi}\right) + \cdots \\ &\cdots \widetilde{K}_{PD}^{gab}\left(\widehat{\varepsilon}_{ji}\delta_{PD}^{ji}+\widehat{\varepsilon}_{ki}\delta_{PD}^{ki}+\cdots+\widehat{\varepsilon}_{li}\delta_{PD}^{li}\right) \end{aligned} \tag{5.3}$$

où $\mathcal{N}_i=\{j,k,\cdots,l\}$ désigne l'ensemble des ADPLLs voisines de la $i^{ième}$ ADPLL du réseau.

Notons que la première partie de (5.3) concerne l'incertitude de *PD* commune à toutes les ADPLLs, et la deuxième partie peut être modélisée de manière suivante :

$$\widetilde{K}_{PD}^{gab}\left(\widehat{\varepsilon}_{ji}\delta_{PD}^{ji}+\widehat{\varepsilon}_{ki}\delta_{PD}^{ki}+\cdots+\widehat{\varepsilon}_{li}\delta_{PD}^{li}\right)=\widetilde{K}_{PD}^{gab}\delta_{PD}^{i}\sum_{q\in\mathcal{N}_i}^{m_i}\widehat{\varepsilon}_{qi} \tag{5.4}$$

avec $|\delta_{PD}^{i}|<1$ et $\widetilde{\Delta}_{PD}^{i}=\widetilde{K}_{PD}^{gab}\delta_{PD}^{i}$.

En effet, cela est possible du fait de la relation :

$$-\sum_{q\in\mathcal{N}_i}^{m_i}|\widehat{\varepsilon}_{qi}|\leq\left|\widehat{\varepsilon}_{ji}\delta_{PD}^{ji}+\widehat{\varepsilon}_{ki}\delta_{PD}^{ki}+\cdots+\widehat{\varepsilon}_{li}\delta_{PD}^{li}\right|\leq\sum_{q\in\mathcal{N}_i}^{m_i}|\widehat{\varepsilon}_{qi}|$$

Grâce aux transformations (5.3) et (5.4), il est possible de transformer les deux schémas des Figures 5.3 et 5.4 avec $\sum_{q\in\mathcal{N}_i}^{m_i}\widehat{\varepsilon}_{qi}=\sum_{q\in\mathcal{N}_i}^{m_i}\varepsilon_{qi}+\sum_{q\in\mathcal{N}_i}^{m_i}q_{PDqi}=\varepsilon_i+q_{PDi}$ et $r_i=\sum_{q\in\mathcal{N}_i}^{m_i}\varphi_q$ en deux schémas équivalents simplifiés des Figures 5.5 et 5.6, avec :

$$\begin{aligned} \Delta_{VCO}^{i} &= \Delta_{VCO}+\widetilde{\Delta}_{VCO}^{i} = K_{VCO}^{gab}\cdot\delta_{VCO}+\widetilde{K}_{VCO}^{gab}\cdot\delta_{VCO}^{i} \\ \Delta_{PD}^{i} &= \Delta_{PD}+\widetilde{\Delta}_{PD}^{i} = K_{PD}^{gab}\cdot\delta_{PD}+\widetilde{K}_{PD}^{gab}\delta_{PD}^{i} \end{aligned}$$

En notant la fonction de transfert entre l'entrée r_i du nœud i de l'ADPLL et sa sortie φ_i : $T_{r_i\rightarrow\varphi_i}=T_s$ on obtient le Modèle global de synthèse décrit par l'équation (4.1) et représenté sur la Figure 4.1, page 147.

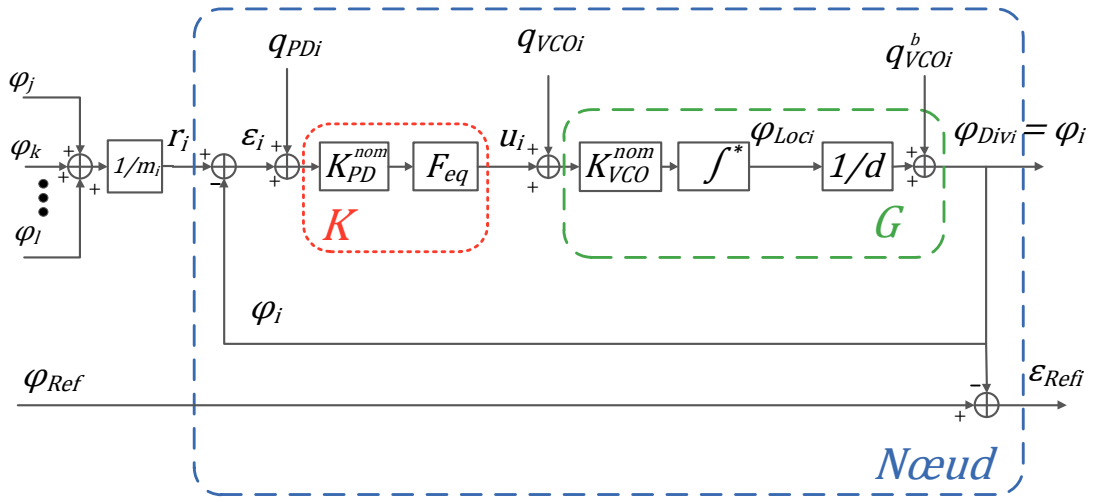


Figure 5.5: Modèle de synthèse de la $i^{ième}$ ADPLLs au sein du réseau équivalent transformé du modèle de la Figure 5.3

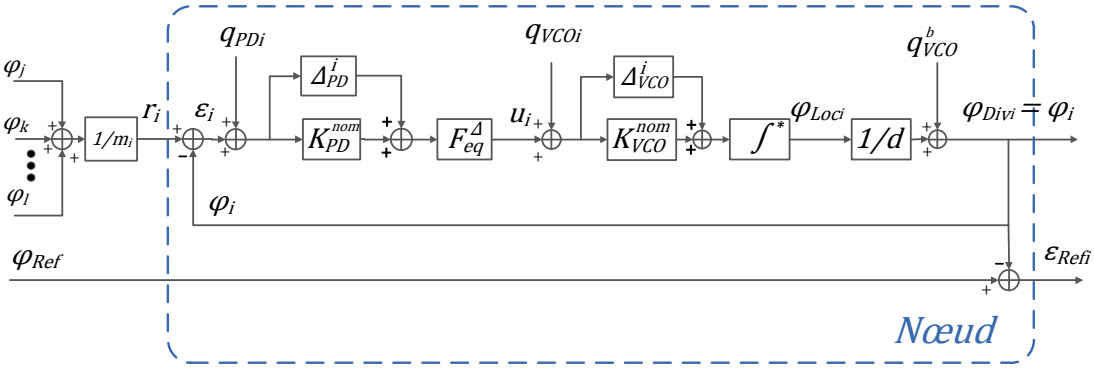


Figure 5.6: Modèle d'analyse de la $i^{ième}$ ADPLLs au sein du réseau équivalent transformé du modèle de la Figure 5.4

5.3 Modèle de simulation du réseau et Mode-Locking

5.3.1 Description

Dans la section précédente, nous avons présenté deux modèles LTIs du réseau complet d'ADPLLs interconnectées considérés dans cette étude : le Modèle nominal de synthèse et le Modèle incertain d'analyse (voir page 33 pour plus de détails). Ils décrivent le comportement du réseau d'ADPLLs réel proche du point de fonctionnement, le point où toutes les phases de tous les signaux générés par les ADPLLs sont alignées.

Dans cette section, nous présentons le Modèle de simulation le plus complet, prenant en compte, en plus des non-linéarités, des bruits et des effets de quantifications, la nature cyclique de la phase. Dans la mesure où les premiers points ont déjà été présentés dans la section 2.2, page 43 du chapitre 2, nous présentons en détails dans cette section l'influence particulièrement importante de la nature cyclique de

la phase sur la dynamique du réseau entier.

Si la loi de commande est proprement synthétisée garantissant la stabilité et le niveau de performance demandé, toutes les erreurs totales de chaque ADPLLs après une certaine période transitoire tendent vers zéro : $\varepsilon_i \to 0$. Pour une structure du réseau qui est modélisée par un graphe connexe, cela implique que les modèles d'ADPLLs linéaires (Modèle de synthèse et Modèle d'analyse) sont synchronisés, c'est-à-dire qu'ils possèdent un seul point d'équilibre : le point pour lequel toutes les erreurs de phases relatives entre les ADPLLs interconnectées sont égales à zéro.

Cependant, du fait de la nature cyclique de la phase ($\varphi = 0 \mod 2\pi$), il existe en réalité d'autres points d'équilibre dont certains peuvent être stables. Ces points d'équilibre stables seront appelés par la suite *Mode-Locking* [PN95, GC00]. Ces modes sont caractérisés par une fréquence égale à chaque nœud avec des déphasages constants. Dans ce cas-là, même si la loi de commande a été proprement synthétisée pour assurer la stabilité et la performance globale du réseau et que le graphe est connexe, même en dépit du fait que toutes les erreurs totales tendent vers zéro : $\varepsilon_i \to 0$, le réseau global de l'ADPLLs n'est pas synchronisé.

Pour illustrer ce phénomène, prenons un exemple de 4 ADPLLs interconnectées selon la topologie présentée sur la Figure 5.7.

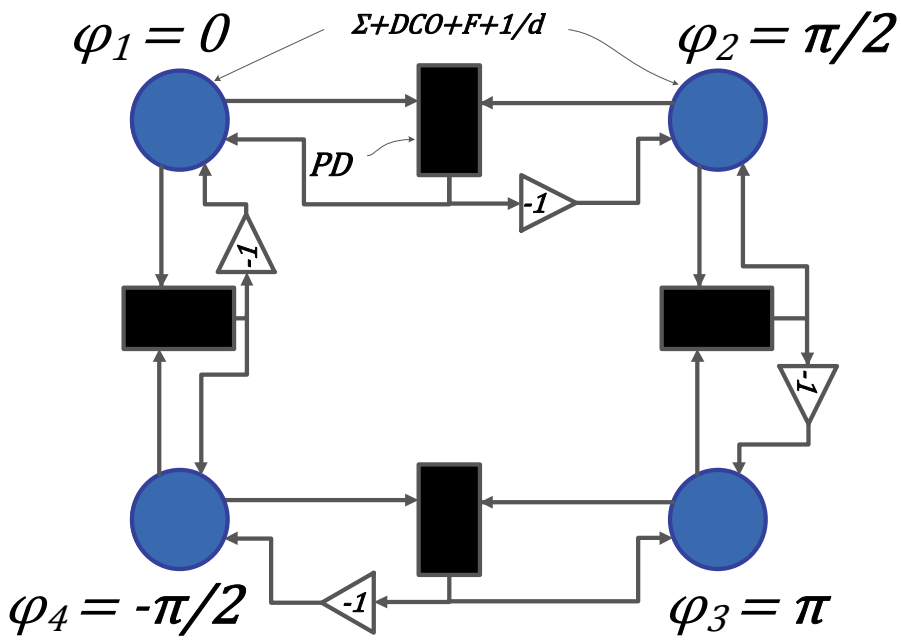


Figure 5.7: L'état d'un réseau dans le régime du Mode-Locking

Dans cette configuration particulière, en supposant que les fréquences sont identiques pour les 4 nœuds et que les phases ont des valeurs comme indiquées sur la Figure 5.7, l'erreur de phase d'un voisin est exactement annulée par l'erreur de phase de l'autre. L'erreur totale de phase pour les 4 nœuds est nulle malgré l'inégalité des phases entre chaque nœud du réseau. Le réseau est donc maintenu dans un état stable parasite appelé Mode-Locking.

5.3.2 Élimination de Mode-Locking

Une solution pour éliminer le Mode-Locking repose sur un échange d'information unidirectionnel sans boucle de retour [PN95] comme par exemple montré sur la Figure 5.8. L'avantage d'une telle topologie de propagation d'information est sa simplicité car la stabilité du réseau global est impliquée par la stabilité de chaque ADPLL séparée [KCSB09] (stabilité locale). L'inconvénient d'une telle topologie est lié au fait que l'on perd une partie de l'information sur la phase (non-retour à l'origine). Ceci provoque un temps de réponse à une perturbation plus élevé en comparaison à la topologie bidirectionnelle de certaines ADPLLs, et une sensibilité plus élevée aux défaillances possibles au sein du réseau.

Une autre solution d'élimination de Mode-Lockings est proposée dans [PN92, PN95] et réalisée dans [GC00]. Les auteurs résolvent le problème en utilisant un détecteur de phases particulier. Ils démontrent que si la topologie du réseau est Cartésienne de deux dimensions (comme représenté sur la Figure 5.1a) et qu'elle est en régime du Mode-Locking, il existe au moins un détecteur de phase mesurant un déphasage $\Delta\varphi \geq \pi/2$. Aussi, en réalisant un détecteur de phase avec une réponse statique représentée sur la Figure 5.9 (avec une pente négative pour $\Delta\varphi \geq \pi/2$) on déstabilise les points stationnaires des modes parasites (rendant le système instable pour les erreurs au-delà de l'angle $\pi/2$). Les auteurs de [PN92, PN95] démontrent la possibilité de supprimer des Mode-Locking avec une telle configuration en se basant sur l'approche énergétique d'étude de stabilité de points stationnaires.

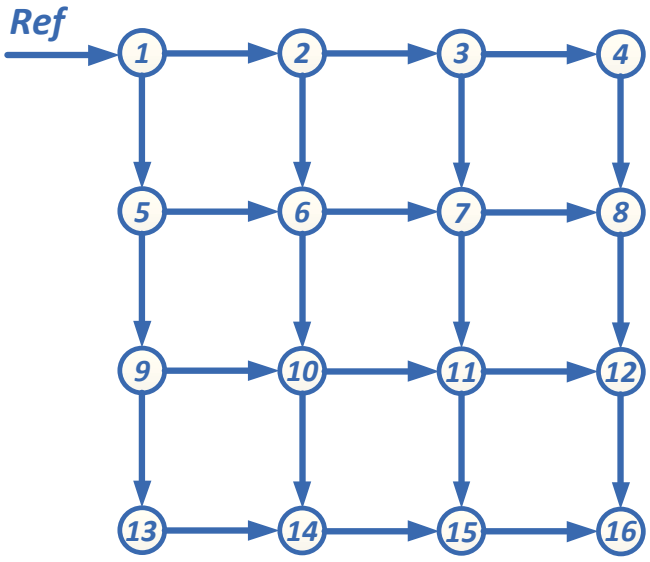


Figure 5.8: Topologie unidirectionnelle de la propagation de l'information de phase

L'implémentation pratique de cette solution, très délicate, est la cause de problèmes pour assurer avec certitude la stabilité du système de distribution d'horloge sous différentes conditions de fonctionnement (tension, température, bruit). En outre, la méthode ne s'applique pas pour les topologies autres que les topologies Cartésiennes de deux dimensions.

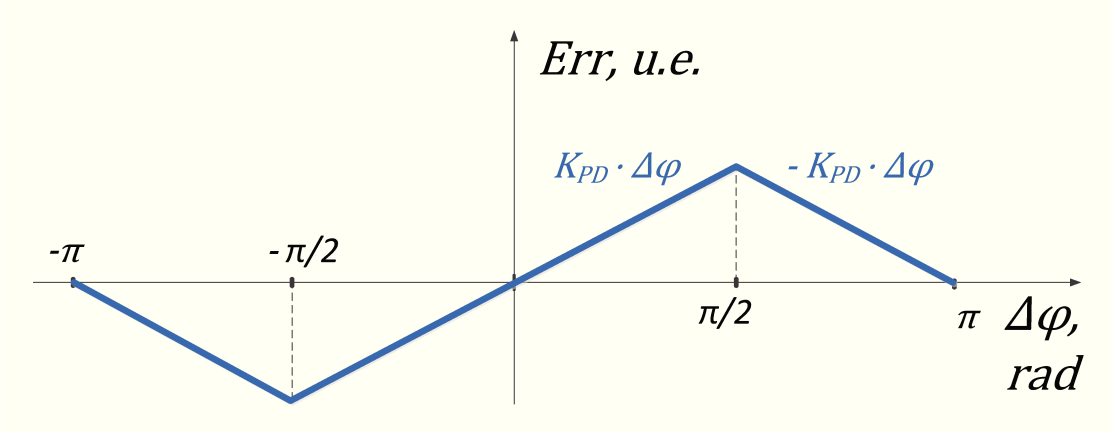


Figure 5.9: Caractéristique du modèle du *PDs* utilisés dans [PN92, PN95, GC00]

Étant donné que les deux solutions existant dans la littérature d'élimination de Mode-Locking souffrent d'inconvénients importants nous proposons deux autres solutions inspirées par le régime unidirectionnel de propagation de l'information.

Élimination de Mode-Locking, solution 1

Premièrement, on illustre la méthode d'élimination de Mode-Locking pour un réseau d'ADPLLs avec des comparateurs de phase bang-bang de un bit c'est-à-dire les mêmes comparateurs de phase que ceux décrits dans la section 2.1.1 du chapitre 2 sans convertisseurs temps numérique (*TDC*s). Ensuite on adapte la méthode pour l'appliquer au réseau d'ADPLLs avec des comparateurs de phase complets.

La méthode d'élimination utilise le fait que le signal de sortie des comparateurs de phase est borné (« 1 » ou « 0 » correspondant à l'erreur de phase « +1 » ou « -1 » *u.e.*). Étant limité, ce signal peut être facilement pondéré avec des coefficients (gains) appropriés pour réaliser un changement automatique de régime de fonctionnement de l'ADPLL entre bidirectionnel (voir la Figure 5.1a par exemple) et unidirectionnel (voir la Figure 5.8 par exemple) afin d'éviter le Mode-Locking.

Ce principe est illustré par le réseau de 4 nœuds interconnectés en Mode-Locking de la Figure 5.7. Dans ce cas présent, chaque nœud a deux détecteurs de phase avec deux entrées #1, #2 (issues des nœuds voisins) considérées comme des références (voir la Figure 5.10). Toutes les fréquences initiales des nœuds sont égales.

Sur la Figure 5.10 (droite) le comportement du nœud d'ADPLL avec les phases initialisées de cette manière est illustré. On peut en déduire 2 étapes : 1ère étape – il n'y a pas de mode-locking, chaque erreur est négative ce qui donne une erreur totale égale à -2. Cela provoque la diminution de la fréquence locale (courbe rouge). La 2ème étape commence dès que la phase locale (Div) entre dans l'espace borné par les deux phases de références (Ref_1 et Ref_2). Durant la deuxième étape l'erreur totale est nulle (un détecteur de phase donne « +1 », l'autre « -1 ») donc la phase ne change plus, elle continue de croître avec la fréquence nominale de l'oscillateur f_0 (courbe rouge avec une pente constante). Si cette fréquence n'est que très légèrement décalée des autres, on observe un Mode-Locking pouvant durer longtemps.

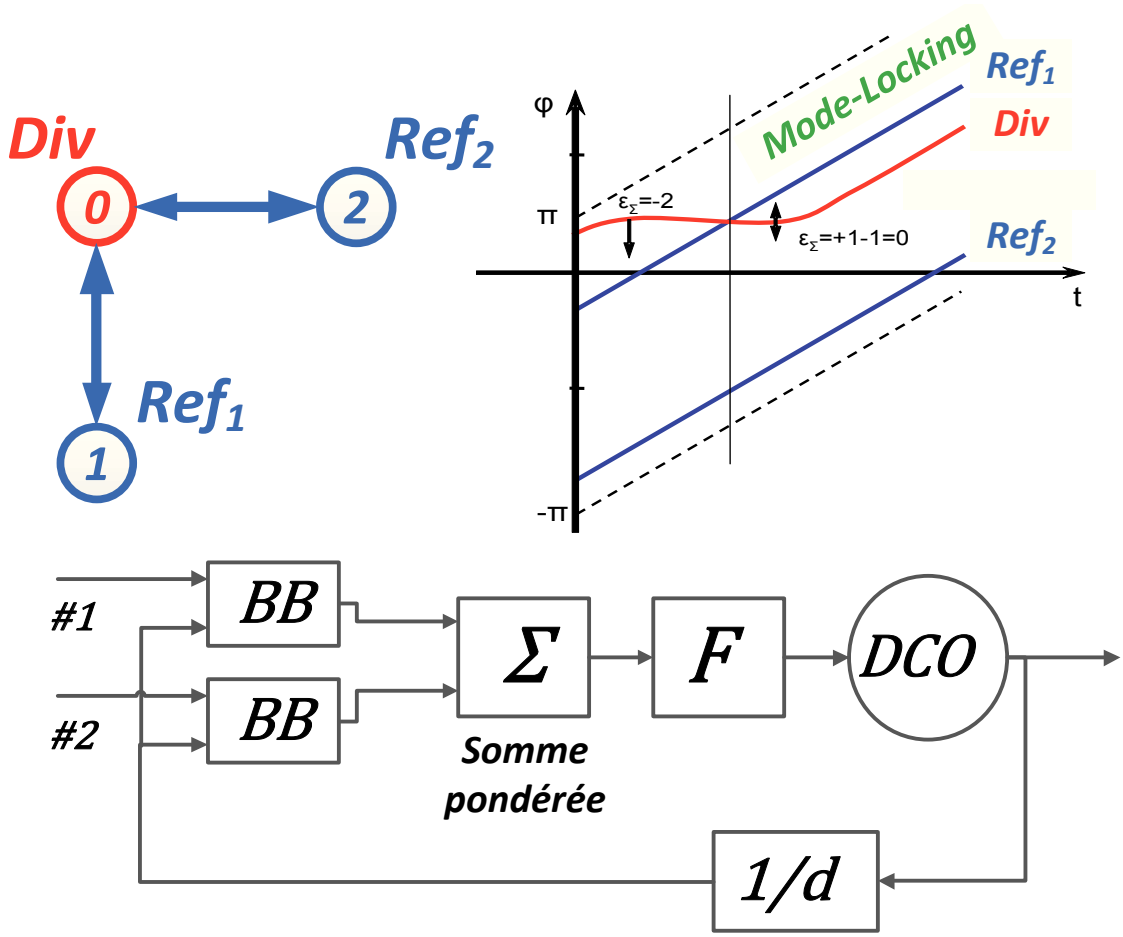


Figure 5.10: Le comportement d'un nœud de ADPLL avec 2 entrées provoquant le Mode-Locking

La solution de ce problème de Mode-Locking est la suivante. On utilise le même schéma d'ADPLL que précédemment mais cette fois-ci, on privilégie une des deux erreurs à l'aide d'un coefficient de pondération $w_1 = 2$ que nous allons appeler par la suite coefficient *majeur*. L'autre coefficient reste égale à $w_2 = 1$, que nous allons appeler par la suite le coefficient *mineur*. Nous appellerons ce mode de fonctionnement de l'ADPLL, le mode asymétrique de comparaison, dans la mesure où dans ce cas-là, on renforce un des liens par rapport aux autres (voir la Figure 5.11).

En 1ère étape, chaque erreur est négative ce qui donne une erreur totale de -3. Une fois que la phase locale est entre les deux phases de référence on risque d'avoir un Mode-Locking et l'ADPLL « change » de régime de fonctionnement. Le coefficient de pondération de la première référence (Ref_1) est supérieur à celui de la Ref_2, donc l'erreur négative venant de Ref_2 est totalement compensée par l'erreur positive venant de Ref_1, l'erreur totale est égale alors à « $+1$ ». Dans ce cas-là, on ne prend en compte que la première référence venant du nœud 1. Le régime de propagation de l'information de phase devient unidirectionnel (sans aucune boucle fermée) et donc on évite le mode-locking. Ce changement de régime se produit de manière automatique et simple pour chaque nœud du système.

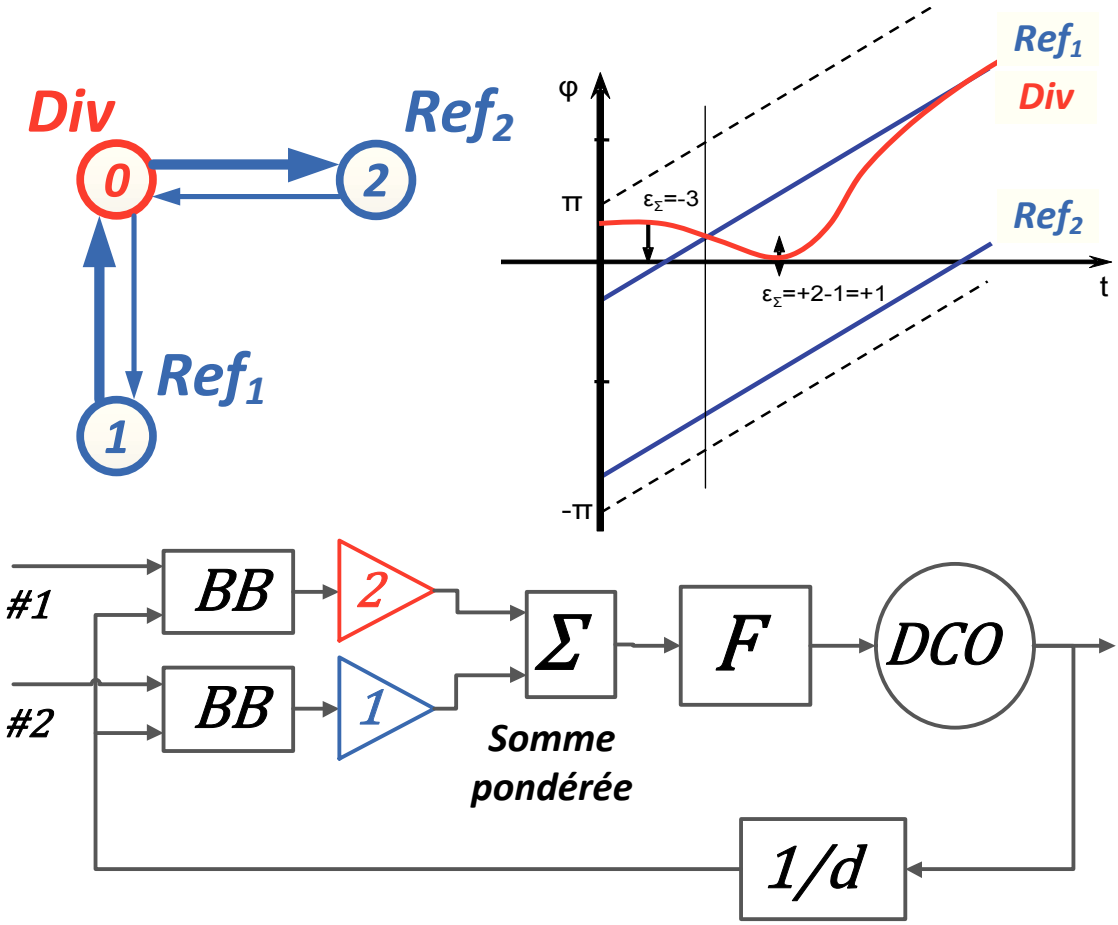


Figure 5.11: Le comportement d'un nœud d'ADPLL avec une entrée privilégiée illustrant l'élimination du Mode-Locking

Ainsi, en choisissant les coefficients de pondérations majeurs sur les sorties des comparateurs de phase du réseau d'ADPLLs de manière à ce que les liens correspondant ne forment pas de boucle fermée en échange d'information de phase, le Mode-Locking est évité quelque soit la topologie d'interconnexion donnée. Une seule exigence est que les coefficients majeurs doivent satisfaire la condition suivante :

$$\sum_{j \in Maj} w_j > \sum_{i \in Min, i \neq j} w_i \tag{5.5}$$

où Maj et Min sont respectivement les ensembles des coefficients majeurs et des coefficients mineurs de chaque ADPLL.

Plus de détails sur la réalisation de cette méthode d'élimination du Mode-Locking ainsi que les résultats de simulations peuvent être trouvés dans [KCS+10].

Le résultat est facilement généralisé pour les comparateur de phase de k bit (avec les TDCs, voir la section 2.1.1 du chapitre 2) avec une petite modification dans la

condition (5.5) :

$$\sum_{j \in Maj} w_j > E_{sat} \sum_{i \in Min, i \neq j} w_i \ . \tag{5.6}$$

Le lecteur peut trouver de plus amples détails dans [CGK11].

Élimination de Mode-Locking, solution 2

La solution précédente d'élimination du Mode-Locking a un inconvénient important à savoir la pénalisation de certaines fonctions de transfert de performance. En effet, en mettant des coefficients de pondération majeurs pour certains liens au sein du réseau, certaines ADPLLs sont pénalisées par rapport aux autres en vue de la réjection des perturbations rentrant sur le réseau. Ceci signifie que certaines ADPLLs rejettent la perturbation et se synchronisent avec le signal de référence plus rapidement que les autres. Étant donné que notre objectif est une synchronisation et une réjection de la perturbation du réseau entier cette pénalisation sous forme d'asymétrie implique en global la dégradation de la performance du réseau.

Pour ces raisons-là, la deuxième solution consiste alors en une détection du régime de Mode-Locking et en un changement du mode de fonctionnement du réseau entier ou en une modification de certains liens entre ADPLLs en fonction de la présence de Mode-Locking. Pour la détection du Mode-Locking la caractérisation suivante est utilisée :

1. Régime Mode-Locking : les erreurs totales de phase ε_i pour certaines ADPLLs sont proches de zéro $\varepsilon_i \approx 0$ *u.e.* [1] alors que les erreurs relatives de phase de la même ADPLL avec ces voisines ont une valeur constante non nulle $|\varepsilon_{ji}| > 3$ *u.e.* ;
2. Régime synchronisé : les erreurs totales de phase ε_i ainsi que les erreurs de phase relative pour toutes les ADPLLs sont proches de zéro $\varepsilon_i \approx 0$ *u.e.*, $\varepsilon_i \approx 0$ *u.e.*.

Le changement de mode de fonctionnement peut être de deux types : *(i)* le changement entre « avec » et « sans » boucles en propagation de l'information de phase (bidirectionnel - unidirectionnel) ; *(ii)* le changement entre le mode symétrique et asymétrique de comparaison. Dans notre architecture, pour des raisons de simplicité de schéma nous utiliserons le premier type de changement de mode de fonctionnement. Dans les deux cas, le changement de mode de fonctionnement peut être considéré comme une perturbation à l'entrée de l'ADPLL et une loi de commande appropriée pour sa réjection peut être synthétisée comme présenté dans la section suivante.

1. Idéalement elles sont égales à zéro, mais en vue du comportement « bang-bang » du comparateur de phase, en pratique ce n'est jamais le cas et en réalité on a des oscillations autour $\varepsilon_i = \pm 2$ *u.e.*

5.4 Synthèse de la loi de commande décentralisée pour le réseau d'ADPLL

Dans cette section nous allons appliquer l'algorithme de synthèse de la loi de commande décentralisée (voir l'Algorithme 4.1, page 169) présenté dans le chapitre 4. Comme cela a déjà été dit au début de ce chapitre, seule la Topologie A (voir la Figure 5.1a) sera étudiée en détails, et seuls les résultats principaux seront discutés pour les autres topologies.

5.4.1 Cahier des charges pour le réseau d'ADPLL

Avant appliquer l'algorithme de synthèse il est nécessaire de définir le cahier des charges pour le réseau d'ADPLL. Nous allons reprendre un cahier des charges similaire à celui d'une seule ADPLL, à l'exception de quelques modifications nécessaires pour le cas du réseau d'ADPLLs.

Stabilité

Comme dans le cas d'une seule ADPLL, nous commençons par la stabilité mais cette fois-ci celle du réseau complet d'ADPLLs. Ainsi une première exigence du cahier des charges est la stabilité des modèles de synthèse ainsi que de modèles d'analyse au sens entrée-sortie de façon interne (voir les Définition 2.2 et 2.3, page 61).

Performance

Comme la stabilité seule ne garantit pas forcément un bon comportement du réseau d'ADPLLs, on fixe d'autres propriétés désirées appelées des exigences en performance qui garantiront un bon fonctionnement du système complet. Le Tableau 5.2 résume les propriétés de performance désirées. Les propriétés de performance du Tableau 5.2 sont semblables à celles pour le cas d'une seule ADPLLs présentées dans le Tableau 2.2, page 62 à l'exception du fait qu'elles concernent tous les signaux et les fonctions de transfert correspondantes de toutes les ADPLLs. C'est pour ces raisons que nous n'allons pas discuter en détails de ces propriétés comme dans le chapitre 2, nous allons plutôt examiner les questions de leur formulation sous la forme d'un critère mathématique.

5.4.2 Cahier des charges sous forme d'un critère mathématique

Chaque ADPLL au sein du réseau possède plusieurs signaux d'entrée et de sortie la reliant à l'environnement extérieur et aux autres ADPLLs via l'interconnexion. Selon la modélisation proposée (voir la Figure 5.5 et la Figure 5.6), pour $\forall i \in \{1, \cdots, N\}$, on retrouve les signaux d'entrée suivants :
– la consigne $\varphi_{_{Ref}}$;
– les perturbations et bruits $q_{_{VCOi}}$, $q^{b}_{_{VCOi}}$ et $q_{_{PDi}}$

Spécifications	Commentaires
Synchronisation	Suivi de référence (rampe), toutes les erreurs statiques nulles $\varepsilon^i_{stat} = 0$
Rapidité	Fréquence de coupure ω^c_{Si} de toutes $T_{\varphi_{Ref}\to\varepsilon_{Refi}}$: $\omega^c_{Si}>\omega_{gab}$
Filtrage du bruit	Filtrer les HF de q_{PDi}, les BF de q_{VCOi} et q^b_{VCOi}
Rejet de perturbations	Perturbations en entrée du *VCO* q_{VCOi}
Limitation de la commande	Commande modérée

Table 5.2: Propriétés de performance désirées du réseau d'ADPLLs

ainsi que les signaux de sortie :

– la sortie de l'ADPLL φ_i ;
– l'erreur de suivi de référence $\varepsilon_{Refi} = \varphi_{Ref} - \varphi_i$;
– la commande u_i délivrée par le correcteur K de la $i^{ième}$ ADPLL.

Comme cela a déjà été illustré dans le chapitre 4, étant donné que dans un réseau actif de distribution d'horloge, il y a plusieurs ADPLLs et donc plusieurs signaux d'entrée et de sortie, le système possède alors plusieurs fonctions de transfert évaluant la performance globale du système. Pour assurer un niveau de performance globale, il est donc nécessaire de fixer et de garantir que toutes ces fonctions de transfert ou au moins les plus importantes parmi elles respectent certaines contraintes fréquentielles de performance semblables à celles proposées dans la sous-section 2.5.2, page 63. En se basant sur la Figure 5.5 et la Figure 5.1, les relations entre les signaux du réseau dans le domaine Laplace peuvent être écrites de manière similaire au cas d'une seule PLL (voir (2.16)) :

$$\begin{aligned}
\varepsilon_{Refi}(s) &= T_{\varphi_{Ref}\to\varepsilon_{Refi}}(s)\,\varphi_{Ref}(s) + \sum_{j=1}^{N} T_{q_{PDj}\to\varepsilon_{Refi}}(s)\,q_{PDj}(s) + \cdots \\
&\cdots \sum_{j=1}^{N} T_{q_{VCOj}\to\varepsilon_{Refi}}(s)\,q_{VCOj}(s) + \sum_{j=1}^{N} T_{q^b_{VCOj}\to\varepsilon_{Refi}}(s)\,q^b_{VCOj}(s)\,; \\
\varphi_i(s) &= T_{\varphi_{Ref}\to\varphi_i}(s)\,\varphi_{Ref}(s) + \sum_{j=1}^{N} T_{q_{PDj}\to\varphi_i}(s)\,q_{PDj}(s) + \cdots \\
&\cdots \sum_{j=1}^{N} T_{q_{VCOj}\to\varphi_i}(s)\,q_{VCOj}(s) + \sum_{j=1}^{N} T_{q^b_{VCOj}\to\varphi_i}(s)\,q^b_{VCOj}(s)\,;
\end{aligned} \tag{5.7}$$

$$\begin{aligned}
\varepsilon_i(s) &= T_{\varphi_{Ref}\to\varepsilon_i}(s)\,\varphi_{Ref}(s) + \sum_{j=1}^{N} T_{q_{PDj}\to\varepsilon_i}(s)\,q_{PDj}(s) + \cdots \\
&\cdots \sum_{j=1}^{N} T_{q_{VCOj}\to\varepsilon_i}(s)\,q_{VCOj}(s) + \sum_{j=1}^{N} T_{q^b_{VCOj}\to\varepsilon_i}(s)\,q^b_{VCOj}(s)\,; \\
u_i(s) &= T_{\varphi_{Ref}\to u_i}(s)\,\varphi_{Ref}(s) + \sum_{j=1}^{N} T_{q_{PDj}\to u_i}(s)\,q_{PDj}(s) + \cdots \\
&\cdots \sum_{j=1}^{N} T_{q_{VCOj}\to u_i}(s)\,q_{VCOj}(s) + \sum_{j=1}^{N} T_{q^b_{VCOj}\to u_i}(s)\,q^b_{VCOj}(s)\,,
\end{aligned} \tag{5.8}$$

pour $\forall i \in \{1, \cdots, N\}$.

Parmi toutes les fonctions de transfert de performance globale, nous distinguons deux types les plus importants : les *fonctions de transfert de synchronisation* $S_{ref\to i} = T_{\varphi_{Ref}\to\varepsilon_{Refi}}$ et les *fonctions de transfert du bruit* $S^b_{j\to i} = T_{q^b_{VCOj}\to\varphi_i}$. En effet, pour assurer la synchronisation du réseau d'ADPLLs avec un signal de référence à son entrée φ_{Ref}, il faut assurer une contrainte fréquentielle (semblable à celle fixée pour la fonction de sensibilité S dans le cas d'une seule ADPLL, voir la sous-section 2.5.2, page 63) pour chaque fonction de transfert reliant ce signal de référence avec l'erreur de son suivi $\varepsilon_{Refi} = \varphi_{Ref} - \varphi_i$ pour chaque ADPLL. Ceci correspond à la fonction de transfert de synchronisation $S_{ref\to i} = T_{\varphi_{Ref}\to\varepsilon_{Refi}}$ où i désigne le numéro de l'ADPLL concernée. Un autre type de fonction de transfert qui est également important, ce sont les fonctions de transfert du bruit $S^b_{j\to i} = T_{q^b_{VCOj}\to\varphi_i}$ reliant le bruit à la sortie de chaque oscillateur q^b_{VCOj} au signal de la phase généré par la $i^{ième}$ ADPLL φ_i. Ici, comme précédemment, pour assurer la réjection de ce bruit, il faut que la fonction de transfert correspondante respecte une contrainte fréquentielle semblable à celle fixée pour la fonction de sensibilité S dans le cas d'une seule ADPLL (voir la sous-section 2.5.2, page 63).

De manière équivalente nous définissons toutes les autres fonctions de transfert en boucle fermé d'une seule ADPLL (T, GS, KS) suivante :

$$\begin{aligned}
T_{\varphi_{Ref}\to\varphi_i}(s) &= 1 - S_{ref\to i}(s) &&= T_{ref\to i}(s)\,; \\
T_{\varphi_{Ref}\to\varepsilon_i}(s) &= (K(s)\cdot G(s))^{-1}(1 - S_{ref\to i}(s)) &&= (K(s)\cdot G(s))^{-1} T_{ref\to i}(s)\,; \\
T_{\varphi_{Ref}\to u_i}(s) &= G^{-1}(s)(1 - S_{ref\to i}(s)) &&= G^{-1}(s)\,T_{ref\to i}(s)\,; \\
T_{q_{PDj}\to\varphi_i}(s) &= K(s)\cdot G(s)\cdot S^b_{j\to i}(s) &&= T^b_{j\to i}(s)\,; \\
T_{q_{PDj}\to\varepsilon_i}(s) &= -K(s)\cdot G(s)\cdot S^b_{j\to i}(s) &&= -T^b_{j\to i}(s)\,; \\
T_{q_{VCOj}\to u_i}(s) &= -K(s)\cdot G(s)\cdot S^b_{j\to i}(s) &&= -T^b_{j\to i}(s)\,; \\
T_{q_{PDj}\to\varepsilon_{Refi}}(s) &= -T_{q_{PDj}\to\varphi_i}(s) &&= -T^b_{j\to i}(s)\,;
\end{aligned}$$

$$
\begin{aligned}
T_{q_{VCOj}\to\varepsilon_{Refi}}(s) &= -G(s)\cdot S^b_{j\to i}(s) &&= -GS^b_{j\to i}(s)\,;\\
T_{q_{VCOj}\to\varphi_i}(s) &= G(s)\cdot S^b_{j\to i}(s) &&= GS^b_{j\to i}(s)\,;\\
T_{q_{VCOj}\to\varepsilon_i}(s) &= -G(s)\cdot S^b_{j\to i}(s) &&= -GS^b_{j\to i}(s)\,;\\
T_{q_{PDj}\to u_i}(s) &= K(s)\cdot S^b_{j\to i}(s) &&= KS^b_{j\to i}(s)\,;\\
T_{q^b_{VCOj}\to u_i}(s) &= -K(s)\cdot S^b_{j\to i}(s) &&= -KS^b_{j\to i}(s)\,;\\
T_{q^b_{VCOj}\to\varepsilon_i}(s) &= -T_{q^b_{VCOj}\to\varphi_i}(s) &&= -S^b_{j\to i}(s)\\
T_{q^b_{VCOj}\to\varepsilon_{Refi}}(s) &= -T_{q^b_{VCOj}\to\varphi_i}(s) &&= -S^b_{j\to i}(s)
\end{aligned}
$$

pour $\forall i \in \{1,\cdots,N\}$, $\forall j \in \{1,\cdots,N\}$.

On peut noter que toutes ces fonctions de transfert sont définies en terme soit de fonctions de transfert de synchronisation $S_{ref\to i}$ soit de fonctions de transfert du bruit $S^b_{j\to i}$. Cela confirme l'affirmation que les fonctions de transfert de synchronisation $S_{ref\to i}$ et du bruit $S^b_{j\to i}$ jouent un rôle important en fixant le comportement dynamique du réseau entier. En conséquence, durant la procédure de synthèse de la loi de commande, nous allons assurer que les fonctions de transfert de synchronisation $S_{ref\to i}$ et du bruit $S^b_{j\to i}$ respectent la contrainte fréquentielle définie par la fonction de pondération $W_s^{globale}$ égale à W_s de (2.46), page 77, c'est-à-dire pour $\forall\omega\in\mathbb{R}$ et $\forall i \in \{1,\cdots,N\}$, $\forall j \in \{1,\cdots,N\}$:

$$
\begin{aligned}
|S_{ref\to i}(j\omega)| &< \left|\left(W_s^{globale}(j\omega)\right)^{-1}\right|\\
\left|S^b_{j\to i}(j\omega)\right| &< \left|\left(W_s^{globale}(j\omega)\right)^{-1}\right|
\end{aligned}
\tag{5.9}
$$

ce qui est résumé dans le Tableau 5.3.

Fonctions de transfert globales	Contraintes concernées réalisées par $W_s^{globale}$
$S_{ref\to i}$	$+40dB/dec$ en BF, la pulsation de coupure minimale $\omega_{min} = 420\,rad/sec$, le temps de réponse maximal correspondant $\tau_m^{max} \approx 7\cdot 10^{-3}\,sec$, gain maximal de $8\,dB$
$S^b_{j\to i}$	Filtrage du bruit de phase en BF, la limitation du gain en MF, gain maximal de $8\,dB$

Table 5.3: Contraintes fréquentielles sur les fonctions de transfert globales du modèle du réseau d'ADPLLs

5.4.3 Exemple numérique de la synthèse d'une loi de commande pour le réseau d'ADPLL

Topologie A

Nous allons appliquer les différents points de l'Algorithme tout en donnant quelques commentaires importants pour son implémentation.

1) Trouvons tout d'abord la caractéristique de dissipativité assurant la stabilité du système global. Le système global est alors défini par l'équation (4.1) et la Figure 4.1, page 147, sans les entrées et sorties externes de performance ; de même le sous-système local T_s est défini par (4.2) et la Figure 4.2, page 148 correspondant à la Figure 5.5 sans les entrées et sorties externes de performance à l'exception de l'entrée r_i et de la sortie φ_i nécessaires pour réaliser l'interconnexion en réseau. La matrice d'interconnexion associée M_{11} est définie de manière suivante :

$$M_{11}^{ij} = \begin{cases} \dfrac{1}{m_i} & \text{, si la ADPLL } i \text{ reçoit information de l'ADPLL } j \\ \\ 0 & \text{, autrement} \end{cases} \tag{5.10}$$

La solution du problème d'optimisation de l'étape 1 de l'Algorithme 4.1 (Problème 4.12, page 165 ou (4.34), page 165) fournit les valeurs de caractérisation de dissipativité suivantes :

$$\begin{array}{rcl} r & = & 0.9119; \\ c & = & 0.1; \\ x & = & -1.2172; \\ y & = & 0.1217; \\ z & = & 1. \end{array} \tag{5.11}$$

2) Les fonctions de pondération fixant la performance locale définie par (4.29), page 162, sont les mêmes que dans le cas d'une seule ADPLL (voir (2.46), page 77) à l'exception des contraintes sur les fonctions de transfert KS et T qui ont été augmentées d'environ de $20\,dB$ à $30\,dB$ et de $8\,dB$ à $13.5\,dB$ respectivement :

$$\begin{array}{rcl} W_{S}(s) = W_{\varepsilon}(s)\,W_{Ref}(s) & = & \dfrac{0.36\,(s+1085)}{s+0.24}; \\ \\ W_{GS}(s) = W_{\varepsilon}(s)\,W_{VCO}(s) & = & \dfrac{2.5\,(s+1085)}{s+0.24}; \\ \\ W_{KS}(s) = W_{u}(s)\,W_{Ref}(s) & = & 0.0309; \\ \\ W_{T}(s) = W_{u}(s)\,W_{VCO}(s) & = & 0.2125. \end{array} \tag{5.12}$$

3), 4), 5) En appliquant la synthèse H_∞ pour le sous-système augmenté représenté sur la Figure 4.4, page 158 on obtient le correcteur d'ordre 1 simplifié [2] sous la forme

$$K(s) = K_{stab}(s) = \frac{1.34(s+93.7)}{s} \tag{5.13}$$

avec la valeur de γ correspondante : $\gamma = 1.05$. Le dépassement de valeur $\gamma > 1$ est expliqué de la même manière qu'à la fin du chapitre 4, page 170, c'est-à-dire par le fait que les mêmes contraintes de performances locales auxquelles s'ajoute la contrainte sur la stabilité globale ((4.12), page 152), deviennent trop fortes et l'algorithme de synthèse n'arrive pas à trouver la solution satisfaisant toutes les contraintes. En effet, ceci est illustré sur la Figure 5.12 où les fonctions de transfert locales en boucle fermée obtenues (la ligne rouge plein) sont comparées aux mêmes fonctions de transfert en boucle fermée lors de la synthèse locale à la fin du Chapitre 2 (la ligne bleue en trait-pointillée). Bien que la valeur de $\gamma > 1$ signifiant que certaines contraintes ne sont pas satisfaites, la contrainte de stabilité globale $\left\|\widehat{T}\right\|_\infty < 1$ (voir la condition (4.12) du Théorème 4.6) est quand même satisfaite. En effet, en calculant cette valeur pour le correcteur synthétisé on obtient :

$$\left\|\widehat{T}\right\|_\infty = 0.99.$$

Selon le Théorème 4.6, page 152 le correcteur (5.13) n'assure pas forcement les contraintes de performance locale mais il assure quand même la stabilité globale du réseau d'ADPLLs identiques interconnectées selon la Topologie A (voir la Figure 5.1a, page 185).

Le calcul de la valeur propre avec la partie réelle maximale de la représentation d'état du système global $\lambda_{max} = -42$ confirme ce résultat.

6) Selon l'approche de la performance globale relative décrite dans la section 4.3 du chapitre 4, page 156, nous construisons une ou plusieurs fonctions de transfert relatives pondérées T_g évaluant la « différence » entre les fonctions de transfert globales de (5.7) et les fonctions de transfert locales de (2.16), page 65. Par la suite, un choix de la caractérisation de dissipativité x, y, z et une synthèse de la loi de commande décentralisée sont effectués afin de minimiser et d'assurer une borne η sur cette fonction de transfert et donc sur la « différence » entre les fonctions de transfert globales et locales.

Il est évident que, pour des raisons de simplicité et d'efficacité du calcul numérique, il n'est pas souhaitable de prendre en compte toutes les fonctions de transfert à la fois. Ceci est d'autant plus vrai que leur nombre augmente de manière exponentielle en fonction du nombre des ADPLLs au sein du réseau. Une autre raison motivant la restriction du nombre de fonctions de transfert de performance globale prises en compte est le conservatisme potentiel du problème obtenu. En effet, pour minimiser la norme H_∞ de plusieurs fonctions de transfert à la fois [3], il est nécessaire d'utiliser la propriété de la norme (2.24), page 68, comme cela a déjà été fait lors de

2. Après avoir approximé le pôle du correcteur obtenu en BFs par un pôle pur en zéro.

3. En résolvant le problème de minimisation de η sous contrainte (4.27), page 162 ou le problème d'optimisation (4.38), page 167 par exemple.

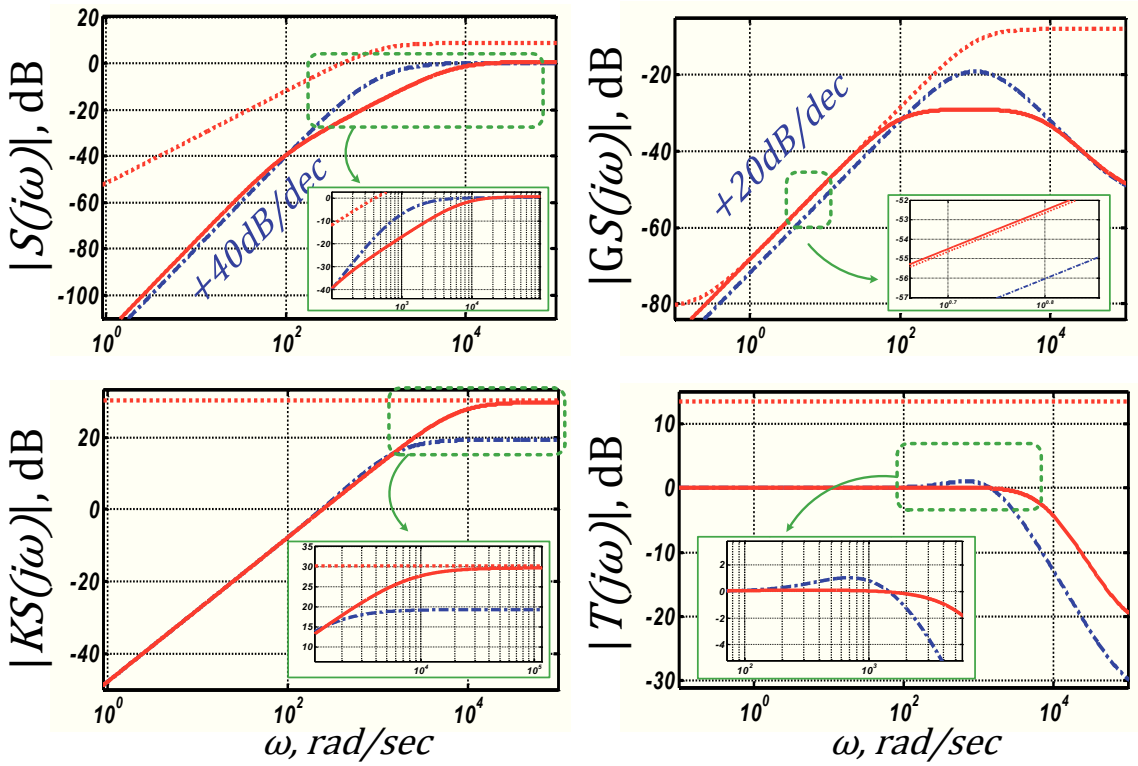


Figure 5.12: Fonctions de sensibilité obtenues après la synthèse H_∞ *(i)* locale (la ligne bleue en trait-pointillée), *(ii)* assurant la stabilité globale (la ligne rouge pleine), les contraintes fréquentielles pour la performance locale (la ligne rouge pointillée)

la synthèse de la loi de commande pour une seule ADPLL dans le chapitre 2 (voir les équations (2.36)-(2.38), page 73). La propriété de la norme (2.24), exprimant une inégalité et non pas une égalité, apporte un certain conservatisme au résultat final. Cependant, contrairement au cas de la synthèse de la loi de commande locale où les normes des fonctions de transfert sont différentes pour différentes plages de fréquences [SP05], dans le cas de fonctions de transfert globales, où les normes correspondantes sont les mêmes pour les différentes fonctions de transfert sur les mêmes plages de fréquence, le résultat peut subir un niveau de conservatisme beaucoup plus important.

Pour ces raisons nous proposons de procéder en deux étapes :

(i) Dans le cas où la caractérisation de dissipativité x, y, z est donnée, afin d'obtenir les contraintes fréquentielles qui seront satisfaites après la synthèse par les fonctions de transfert globales (selon la section 4.5, page 170), on résout le problème de minimisation de η_i sous contrainte (4.27), page 162 pour chaque fonction de transfert relative pondérée T_g^i séparément, correspondant aux fonctions de transfert globales les plus importantes ;

(ii) Dans le cas où la caractérisation de dissipativité x, y, z n'est pas donnée ou ne donne pas de résultat satisfaisant, afin de choisir les valeurs x, y, z acceptables, on résout le problème d'optimisation (4.38), page 167 pour une fonction de transfert

relative pondéré $\widehat{T}_g$ correspondant au pire-cas.

La fonction de transfert correspondant à la fonction de transfert de pire-cas peut être choisie selon la structure d'interconnexion donnée. Un exemple pour le cas de fonction de transfert de synchronisation peut être la fonction de transfert qui relie le signal de référence en entrée du réseau avec le signal de phase généré par l'ADPLL la plus « éloignée » du réseau. Il est vrai que selon la complexité de la topologie, il peut être difficile de choisir l'ADPLL la plus « éloignée » du réseau. La solution est qu'il est toujours possible de calculer les fonctions de transfert correspondant à la loi de commande assurant seulement la stabilité globale (voir l'étape 1-5 de l'Algorithme 4.1, page 169) et de choisir parmi elles la fonction dont la différence avec le cas locale S_{locale} est la plus importante [4].

Ainsi pour la Topologie A (voir la Figure 5.1a) nous choisissons la fonction de transfert de synchronisation $\widehat{S}_{Ref\to 16}$ reliant le signal de référence $w = \varphi_{Ref}$ avec l'erreur de son suivi par l'ADPLL 16 : $z = \varepsilon_{16} = \varphi_{Ref} - \varphi_{16}$ comme la fonction de transfert pire-cas. Son expression est donnée par (4.1) avec M_{11} défini dans (5.10) et :

$$\begin{aligned} M_{12} &= \left[\begin{array}{cc} \dfrac{1}{m_1} & \underbrace{0 \quad \cdots \quad 0}_{N-1} \end{array}\right]^T; \\ M_{21} &= \left[\begin{array}{c} \overbrace{0 \quad \cdots \quad 0}^{N-1} \quad -1 \end{array}\right]; \\ M_{22} &= 1. \end{aligned} \tag{5.14}$$

Pour appliquer l'approche de la performance globale relative décrit dans la section 4.3 du chapitre 4, on choisit la fonction de transfert relative pondérée sous la forme de (4.42), page 174 avec $S_{globale} = S_{Ref\to 16}$ et $\alpha = 0.05$ qui est à son tour facilement décrite sous forme d'une UFT en T_s par (4.26) et la Figure 4.7, page 162 avec (4.46), page 177.

7) Le problème de minimisation de η sous contrainte (4.27) pour les valeurs x, y, z trouvées à l'étape 1 de l'Algorithme ne donne pas de résultat satisfaisant.

8-9) Le calcul explicite de la borne η donne le résultat suivant $\eta \approx 165.6$ ce qui correspond à $\mu = 20\log_{10}(\eta) \approx 44.5\,dB$. Les fonctions de transfert de synchronisation $S_{Ref\to i}$ ainsi que les fonctions de transfert du bruit $S^b_{j\to i}$ correspondantes sont représentées sur la Figure 5.13 et Figure 5.14 respectivement. Les figures illustrent que la contrainte fréquentielle n'est pas satisfaite par toutes les fonctions de transfert de synchronisation. En revanche elle est satisfaite par toutes les fonctions de transfert

4. Il est possible également de choisir une fonction de transfert qui semble correspondre au pire-cas. Dans tous les cas, une erreur dans ce choix n'est pas très pénalisant, car les valeurs de caractérisation de dissipativité x, y, z trouvées en se basant sur cette fonction de transfert, conviendront probablement pour les autres fonctions de transfert. S'il se trouve que ce n'est pas le cas pour une des fonctions de transfert, elle peut être « nommée » comme une fonction de transfert pire-cas et l'itération est répétée, si nécessaire, encore une fois.

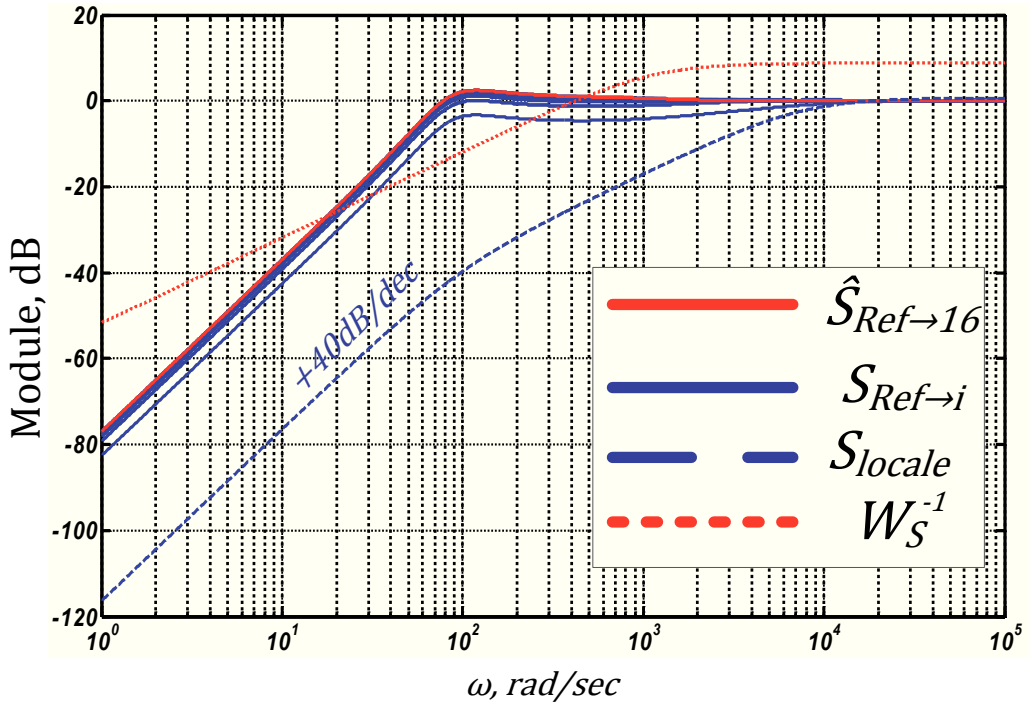


Figure 5.13: Fonctions de transfert de synchronisation $S_{Ref\to i}$ (les lignes bleues pleines), fonctions de transfert de synchronisation pire-cas $\widehat{S}_{Ref\to 16}$ (la ligne rouge pleine), fonction de transfert locale S_{locale} (la ligne bleue en trait long) et la contrainte fréquentielle correspondante (la ligne rouge en trait court)

du bruit ce qui confirme également que la fonction de transfert $\widehat{S}_{Ref\to 16}$ correspond bien au pire-cas. Appliquons la synthèse de la loi de commande afin de garantir la contrainte fréquentielle pour toutes ces fonctions de transfert.

10) Le paramètre β est fixé comme $\beta = \eta^2 r^2$ avec les valeurs de $\eta = 165.6$ et $r = 0.9119$ trouvées précédemment à l'étape 8) et 1) de l'Algorithme (voir (5.11)).

11) La résolution du problème d'optimisation (4.38) fournit les valeurs suivantes :

$$\begin{aligned} r &= 0.905; \\ c &= 0.1; \\ x &= -1.237; \\ y &= 0.124; \\ z &= 1; \\ \eta &= 148; \\ \mu &= 43.4\,dB. \end{aligned} \tag{5.15}$$

Pour une loi de commande K respectant la condition (4.17), page 153, la norme H_∞ de la fonction de transfert relative pondérée $\widehat{T}_g$ sera bornée par la nouvelle valeur de μ trouvée (5.15). Cette valeur est inférieure à celle obtenue à l'étape 8-9) de

l'algorithme lors de calcul explicite pour la loi de commande assurant seulement la stabilité globale. On note également que la caractérisation de $\{x, y, z\}$-dissipativité a aussi changé en fixant une contrainte sur la dynamique locale plus forte (voir l'équation (4.12), page 152) car la valeur du rayon r a diminué (voir la section 4.4 du chapitre 4 pour plus des détails). Cela explique la diminution de la valeur η par rapport au cas de stabilité globale (voir l'étape 8-9 de l'algorithme).

La nouvelle borne η et donc μ impliquent (voir 4.42, page 174 avec la discussion sur le sujet) que la fonction de transfert globale $\widehat{S}_{Ref\to 16}$ respectera la contrainte fréquentielle définie par le module de la fonction de transfert locale $S_{locale} = S$ plus $\mu\, dB$ dans la plage de fréquence où le paramètre α sera petit par rapport au module de la fonction de sensibilité locale S. Étant donné que la fonction de transfert $\widehat{S}_{Ref\to 16}$ correspond au pire-cas, toutes les autres fonctions de transfert de synchronisation $S_{Ref\to i}$ elles-aussi respecterons la même contrainte fréquentielle sur la même plage de fréquences.

Cependant, avant d'appliquer la synthèse de la loi de commande, on propose d'évaluer la différence entre les fonctions de transfert globales et locales pour d'autres types de fonctions de transfert : $S^b_{j\to i}$, $T^b_{j\to i}$ et $T_{Ref\to i}$. On calcule alors les bornes μ_{Sb}, μ_{Tb} et μ_{TRef} correspondant aux cas des fonctions de transfert en résolvant le problème de minimisation de η_i sous contrainte (4.27), page 162 pour x, y, z trouvés dans (5.15).

Ainsi les fonctions de transfert globales $S^b_{j\to i}$, $T^b_{j\to i}$ et $T_{Ref\to i}$ sont mises en forme (4.1) (voir la Figure 4.1, page 147) avec M_{11} définie comme dans le Tableau 5.4 où $M_{11}\,(i,:)$

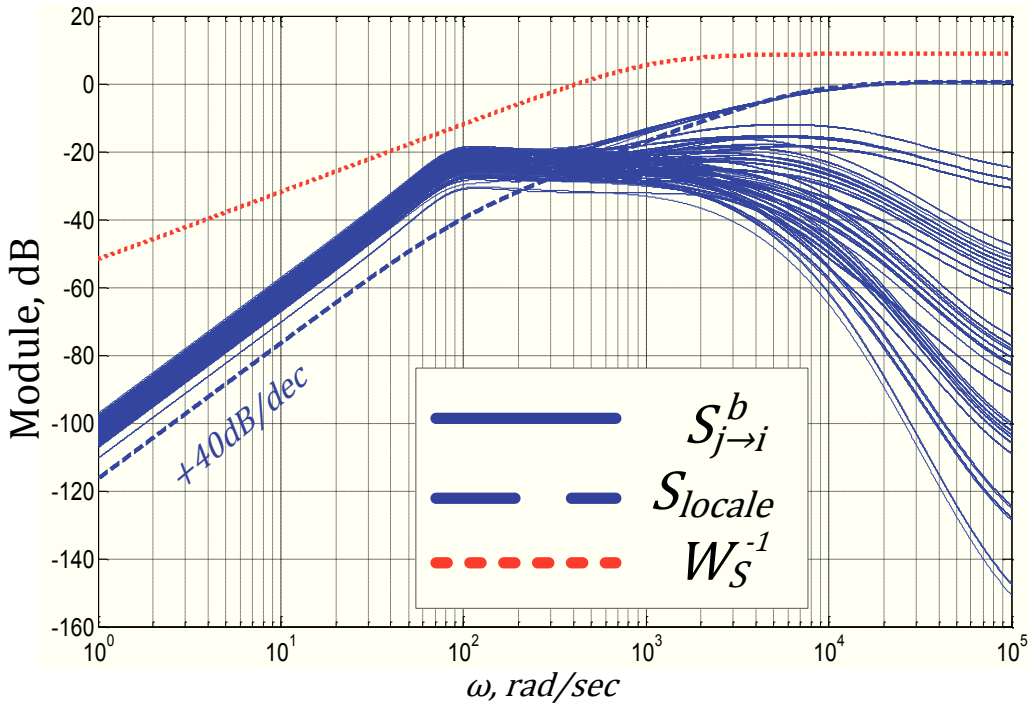


Figure 5.14: Fonctions de transfert du bruit $S^b_{j\to i}(s)$ (les lignes bleues pleines), fonction de transfert locale S_{locale} (la ligne bleue en trait long) et la contrainte fréquentielle correspondante (la ligne rouge en trait court)

désigne i-ème ligne de la matrice M_{11} et $Z_j = \left[\underbrace{0 \ \cdots \ 0}_{j-1} \ \overset{j}{1} \ \underbrace{0 \ \cdots \ 0}_{N-j} \right]^T$.

Fonction de transfert	$S^b_{j\to i}$	$T^b_{j\to i}$	$T_{Ref\to i}$
M_{12}	Z_j	Z_j	$\left[\begin{array}{c}\frac{1}{m_1}\\ \hline 0\\ \vdots\\ 0\end{array}\right] \begin{array}{l}\leftarrow 1\\ \left.\begin{array}{l}\\ \\ \\ \end{array}\right\} N-1\end{array}$
M_{21}	$M_{11}(i,:) - Z_i^T$	Z_i^T	Z_i^T
M_{22}	$\begin{cases} 1 & , \text{si } i = j \\ 0 & , \text{si } i \neq j \end{cases}$	0	0
w	q^b_{VCOj}	q_{PDj}	φ_{Ref}
z	φ_i	φ_i	φ_i

Table 5.4: Fonctions de transfert globales de performance

On définit par la suite les nouvelles fonctions de transfert relatives pondérées :

$$\begin{aligned} T^{Sb}_{g_{j\to i}}(s) &= \frac{S^b_{j\to i}(s)}{S(s)+\alpha}; \\ T^{Tb}_{g_{j\to i}}(s) &= \frac{T^b_{j\to i}(s)}{T_s(s)}; \\ T_{g_{Ref\to i}}(s) &= \frac{T_{Ref\to i}(s)}{T_s(s)}, \end{aligned} \tag{5.16}$$

pour $\forall i \in \{1, \cdots, N\}, \forall j \in \{1, \cdots, N\}$ et avec $\alpha = 0.05$ le même que précédemment.

Il est important de noter que dans l'expression de fonctions relatives pondérées $T^{Tb}_{g_{j\to i}}$ et $T_{g_{Ref\to i}}$ nous n'avons pas besoin de paramètre α, étant donné que le module de la fonction de transfert T_s correspondant à la dynamique locale est égal à 1 en BFs et la division par zéro lors d'évaluation de l'expression de la fonction de transfert relative pondérée n'a plus lieu. En conséquence, les bornes trouvées μ_{Tb} et μ_{TRef} sur ces fonctions de transfert seront valables pour toute une plage de fréquences.

Par la suite, les fonctions de transfert relatives pondérées (5.16) sont mises en forme d'une *UFT* (4.26) (voir la Figure 4.7) pour les matrices M_{ij} définies dans le Tableau 5.4, ce qui est résumé dans le Tableau 5.5 pour

$$M_S = \begin{bmatrix} \frac{1}{1+\alpha} & \frac{1}{1+\alpha}M_{21} & \frac{1}{1+\alpha}M_{22} \\ 0 & M_{11} & M_{12} \\ \frac{1}{1+\alpha} & \frac{1}{1+\alpha}M_{21} & \frac{1}{1+\alpha}M_{22} \end{bmatrix} \quad M_T = \begin{bmatrix} M_{11} & M_{12} \\ M_{11}(j,:) & M_{12}(j) \end{bmatrix}$$

et $M_{11}(j,:)$ désigne j-ème ligne de la matrice M_{11}, $M_{12}(j)$ désigne j-ème l'élément de la matrice colonne M_{12}.

Fonction de transfert	$T^{Sb}_{g_{j\to i}}$	$T^{Tb}_{g_{j\to i}}$	$T_{g_{Ref\to i}}$
$\breve{T}$	$I_{N+1}\otimes T_s$	$I_N \otimes T_s$	$I_N \otimes T_s$
$\widetilde{M}$	M_S	M_T	M_T
$\widetilde{r}$	$\begin{bmatrix} q \\ r_1 \\ \vdots \\ r_N \end{bmatrix}$	r	r
$\widetilde{w}$	q^b_{VCOj}	q_{PDj}	φ_{Ref}
$\widetilde{z}$	q	r_i	r_i

Table 5.5: Fonctions de transfert relatives pondérées

En résolvant le problème de minimisation de η sous contrainte (4.27), page 162 pour les valeurs x, y, z trouvées précédemment (voir (5.15)) et pour chaque fonction de transfert relative pondérée (5.16) respectivement on trouve facilement les nouvelles bornes maximales sur la « différence » entre les fonctions de transfert locales et globales définies par (5.16) :

$$\begin{aligned} \mu_{Sb} &= 22.8\,dB; \\ \mu_{Tb} &= 23.38\,dB; \\ \mu_{TRef} &= 5.84\,dB. \end{aligned} \tag{5.17}$$

Les bornes trouvées (5.17) sont inférieures à la borne μ correspondant à la fonction de transfert $\widehat{S}_{Ref\to 16}$ (voir (5.15)) ce qui confirme que la fonction de pire-cas a été choisie correctement.

Il est important de noter que les bornes (5.17) ainsi que la borne correspondant à la fonction de transfert de synchronisation $S_{Ref\to i}$ (5.15) sont *a priori* définies avant la synthèse de la loi de commande assurant la performance globale. En outre, la performance globale du réseau est « normalisée » c'est-à-dire est évaluée par rapport à la performance locale d'un seul sous-système de manière relative. Ainsi, indépendamment de la dynamique des sous-systèmes locaux (1.3), page 30, à conditions

que tous les sous-systèmes locaux soient identiques et respectent la contrainte de $\{x, y, z\}$-dissipativité (voir la condition (4.17) du Théorème 4.7, page 153), la « différence » entre les fonctions de transfert globales $S_{Ref\to i}$, $S^b_{j\to i}$, $T^b_{j\to i}$ et $T_{Ref\to i}$ et les fonctions de transfert locales $S_{locale} = S$ et $T_{locale} = T_s$ pour $\forall i \in \{1, \cdots, N\}$, $\forall j \in \{1, \cdots, N\}$ sera bornée par les valeurs de μ, μ_{Sb}, μ_{Tb} et μ_{TRef} respectivement.

12) En se basant sur la valeur de la borne μ calculée dans l'étape précédente (5.15), nous adaptons les contraintes de performance locale afin que *(i)* la contrainte de stabilité globale (voir l'équation (4.12), page 152) pour les nouvelles valeurs de x, y, z soit respectée et *(ii)* les fonctions de transfert globales respectent le cahier des charges défini par les Tableaux 5.2 et 5.3 :

$$\begin{aligned} W_{S}(s) &= W_{\varepsilon}(s)\, W_{Ref}(s) = \frac{0.36\,(s+11090)}{s+2.47}; \\ W_{GS}(s) &= W_{\varepsilon}(s)\, W_{VCO}(s) = \frac{2.5\,(s+11090)}{s+2.47}; \\ W_{KS}(s) &= W_{u}(s)\, W_{Ref}(s) = 9.09\cdot 10^{-4}; \\ W_{T}(s) &= W_{u}(s)\, W_{VCO}(s) = 6.25\cdot 10^{-3}. \end{aligned} \tag{5.18}$$

13-15) En appliquant la synthèse H_∞ de la loi de commande (voir l'Algorithme 2.1, page 76) sur le système étendu de la Figure 4.4, page 158, on obtient le correcteur d'ordre 1 simplifié [5] sous la forme :

$$K(s) = K_{perf}(s) = \tfrac{7.12(s+182.2)}{s} \tag{5.19}$$

avec la valeur de γ correspondante : $\gamma = 1.007$. Le dépassement de valeur $\gamma > 1$ est expliqué de la même manière que précédemment. En effet, étant donné que la fonction de transfert locale S est utilisée pour fixer la contrainte fréquentielle à respecter par la fonction de transfert globale de synchronisation $S_{Ref\to i}$, nous fixons volontairement la contrainte fréquentielle sur la fonction de sensibilité locale S plus forte que celle définie dans le cahier de charge d'une seule ADPLL décrit dans (2.46), page 77. En conséquence, les contraintes fréquentielles locales sont légèrement dépassées, ce qui est illustré sur la Figure 5.15. Dans cette figure, les fonctions de transfert locales en boucle fermée obtenues (la ligne rouge plein) sont comparées aux mêmes fonctions de transfert locales en boucle fermée lors de la synthèse à l'étape 1 de l'algorithme assurant seulement la stabilité globale (la ligne bleue en trait-pointillée). En ajoutant en outre la contrainte forte de la stabilité globale $\left\|\widehat{T}\right\|_\infty < 1$ (voir l'équation (4.12), page 152), un léger dépassement de valeur $\gamma > 1$ est logique.

Bien que les contraintes fréquentielles (5.18) soient légèrement dépassées, elles impliquent quand même que la contrainte fréquentielle est valide pour la fonction de transfert globale $S_{Ref\to i}$ (voir (5.9)). En outre, la contrainte de stabilité globale $\left\|\widehat{T}\right\|_\infty < 1$ (voir la condition (4.12) du Théorème 4.6) est également satisfaite. En

5. Après avoir approximé le pôle du correcteur obtenu en BFs par un pôle pur en zéro.

effet, en calculant cette valeur pour le correcteur synthétisé, on obtient :

$$\left\|\widehat{T}\right\|_{\infty} = 0.996$$

Selon le Théorème 4.10, page 161, le correcteur (5.19) assure la stabilité et un niveau de performance globale du réseau d'ADPLLs identiques interconnectées selon la topologie A (voir la Figure 5.1a, page 185).

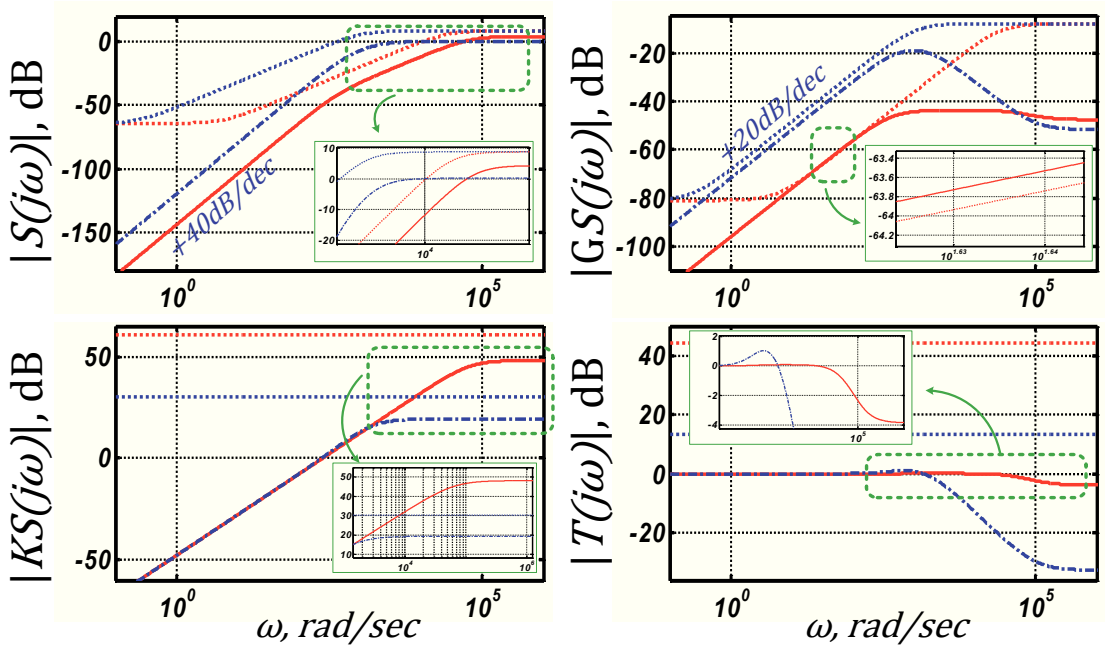


Figure 5.15: Fonctions de sensibilité obtenues après la synthèse H_∞ *(i)* garantissant seulement la stabilité globale (la ligne bleue en trait-pointillée), *(ii)* assurant la performance globale (la ligne rouge pleine), les contraintes fréquentielles de performance locale pour les deux cas (la ligne rouge pointillée et la ligne bleue pointillée respectivement)

Ceci est confirmé par la Figure 5.16 où les fonctions de transfert de synchronisation $S_{Ref\to i}$ pour tous les ADPLLS sont présentées. En effet, toutes les fonctions de transfert de synchronisation $S_{Ref\to i}$ respectent la contrainte fréquentielle (voir (5.9) et le Tableau 5.3) issue du cahier des charges. Par ailleurs, la même contrainte est respectée par toutes les fonctions de transfert du bruit $S^b_{j\to i}$ ce qui est présenté sur la Figure 5.17.

La Figure 5.18 et la Figure 5.19 confirment que les bornes maximales μ_{Tb} et μ_{TRef} trouvées à l'étape 11 de l'algorithme (voir (5.17)) sont aussi respectées par toutes les fonctions de transfert $T^b_{j\to i}$ et $T_{Ref\to i}$ respectivement.

Il est intéressant de comparer les fonctions de transfert de synchronisation pire-cas pour les deux lois de commande (5.13) assurant la stabilité globale et (5.19) assurant en plus la performance globale. La Figure 5.20 nous illustre la différence entre ces deux cas.

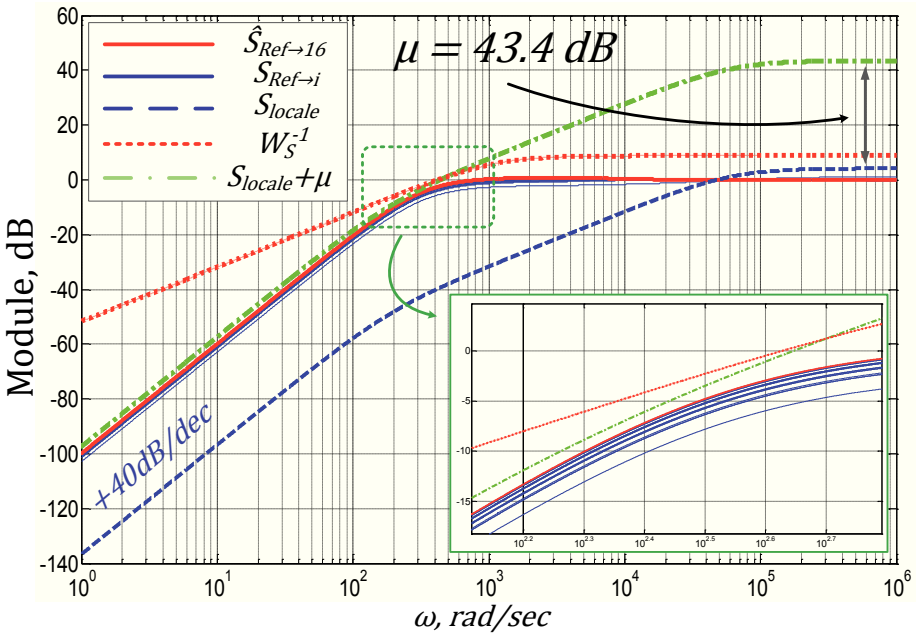


Figure 5.16: Fonctions de transfert de synchronisation $S_{Ref\to i}$ (les lignes bleues pleines), fonction de transfert de synchronisation pire-cas $\widehat{S}_{Ref\to 16}$ (la ligne rouge pleine), fonction de transfert locale S_{locale} (la ligne bleue en trait long), fonction de transfert locale + $\mu\, dB$ (la ligne verte en trait interrompu mixte) et la contrainte fréquentielle du cahier des charges pour le réseau global (la ligne rouge en trait court)

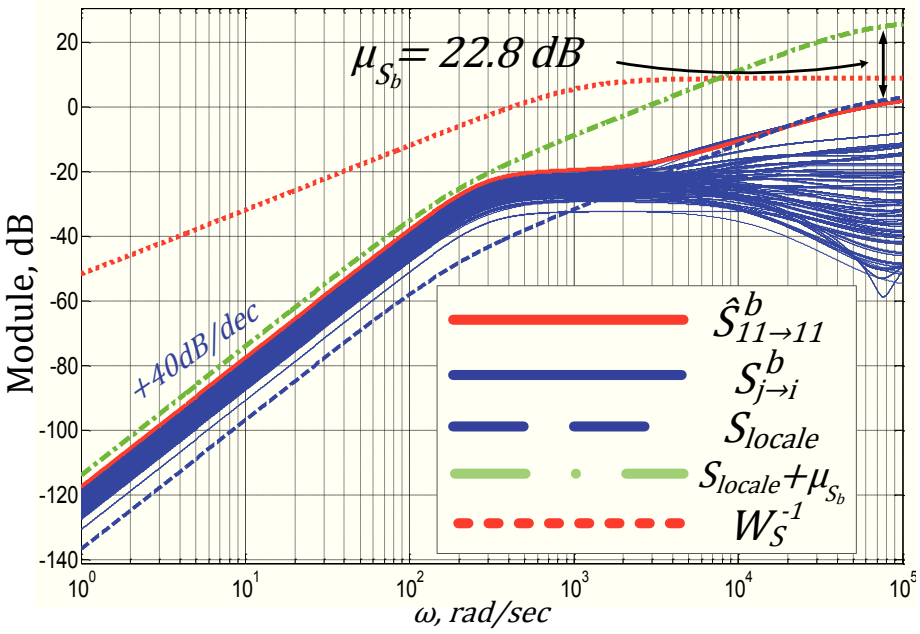


Figure 5.17: Fonctions de transfert du bruit $S^b_{j\to i}$ (les lignes bleues pleines), fonction de transfert du bruit pire-cas $\widehat{S}^b_{11\to 11}$ (la ligne rouge pleine), fonction de transfert locale S_{locale} (la ligne bleue en trait long), fonction de transfert locale + $\mu_{S_b}\, dB$ (la ligne verte en trait interrompu mixte) et la contrainte fréquentielle du cahier des charges pour le réseau global (la ligne rouge en trait court)

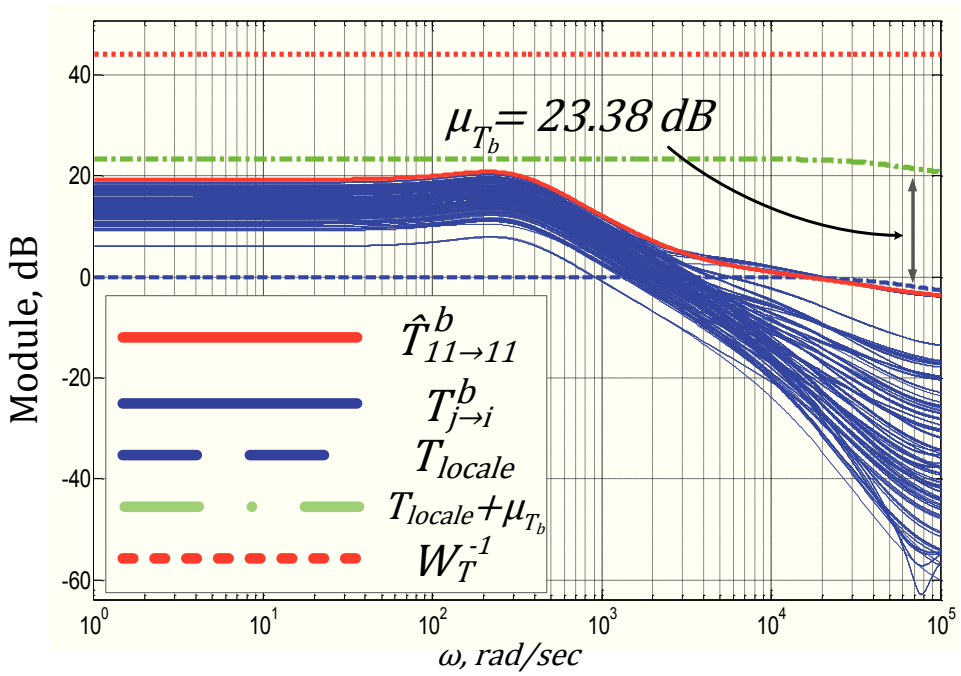


Figure 5.18: Fonctions de transfert $T^b_{j\to i}$ (les lignes bleues pleines), fonction de transfert de bruit pire-cas $\widehat{T}^b_{11\to 11}$ (la ligne rouge pleine), fonction de transfert locale T_{locale} (la ligne bleue en trait long), fonction de transfert locale + $\mu_{T_b}\,dB$ (la ligne verte en trait interrompu mixte) et la contrainte fréquentielle locale correspondante (la ligne rouge en trait court)

Topologie B et C

De la même manière que pour la Topologie A (voir la Figure 5.1a, page 185), l'Algorithme 4.1, page 169 de synthèse de la loi de commande décentralisée est appliqué pour les autres topologies représentées sur la Figure 5.1b et c respectivement, page 185.

Les résultats numériques sont résumés dans les Tableaux 5.6 et 5.7 et sont illustrés sur les Figures 5.21-5.24. On remarque que pour la Topologie C nous n'avons pas réussi à satisfaire la contrainte de la performance globale sur la fonction de synchronisation pire-cas $\widehat{S}_{Ref\to i}$ (voir la partie augmentée de la Figure 5.21). C'est possiblement lié au fait que le réseau de la Topologie C apporte une grande inertie à la dynamique du système global qui n'a pas pu être compensée par la synthèse de correcteurs locaux. En effet, une augmentation de la fréquence de coupure de la fonction de transfert locale S_{locale} pouvant apporter une augmentation de la fréquence de coupure des fonctions de transfert de synchronisation (voir l'exemple numérique pour le cas de la Topologie A et notamment les étapes 12-15 de l'Algorithme) n'est pas possible du fait de la contrainte de la stabilité globale $\left\|\widehat{T}\right\|_\infty < 1$ (voir la condition (4.17) du Théorème 4.7, page 153).

Comme dans le cas de synthèse H_∞ standard [SP05], la méthode de synthèse de la loi de commande décentralisée possède aussi certaines limitations sur le niveau de performance atteignable. Il n'est donc pas possible de satisfaire toutes les contraintes

à la fois pour certaines topologies d'interconnexion, ce qui semble logique.

Néanmoins, il est possible d'adapter la topologie afin de garantir ces contraintes. La section suivante illustre une amélioration possible de la topologie en se basant sur les outils développés dans le chapitre 4.

Topologie	Top.A	Top.B	Top.C
r	0.9119	0.92	0.9079
c	0.1	0.1	0.1
x	-1.2172	-1.1957	-1.2281
y	0.1217	0.1196	0.1228
z	1	1	1
$K_{stab}(s)$	$\frac{1.34(s+93.7)}{s}$	$\frac{2.21(s+58.23)}{s}$	$\frac{7.2(s+40.8)}{s}$

Table 5.6: Résultats numériques de la synthèse de la loi de commande assurant la stabilité pour les topologies A, B et C représentées sur la Figure 5.1a, b et c respectivement

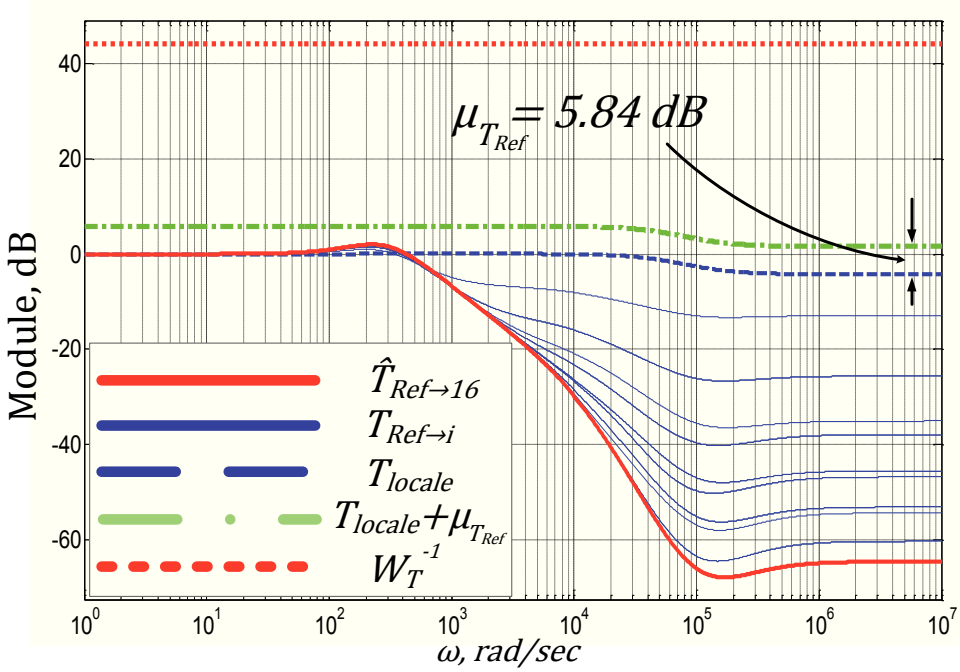


Figure 5.19: Fonctions de transfert $T_{Ref\to i}$ (les lignes bleues pleines), fonction de transfert de bruit pire-cas $\widehat{T}_{Ref\to 16}$ (la ligne rouge pleine), fonction de transfert locale T_{locale} (la ligne bleue en trait long), fonction de transfert locale + $\mu_{T_{Ref}}\,dB$ (la ligne verte en trait interrompu mixte) et la contrainte fréquentielle locale correspondante (la ligne rouge en trait court)

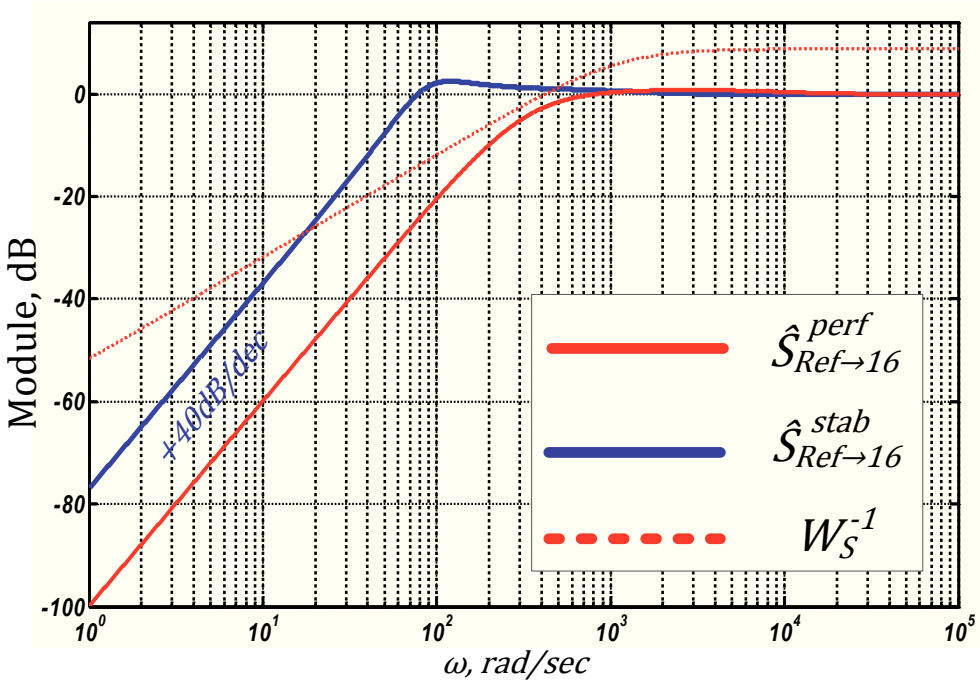


Figure 5.20: Comparaison de la fonction de transfert de synchronisation pire-cas correspondant à la loi de commande garantissant la stabilité globale $\widehat{S}^{stab}_{Ref\to 16}$ (la ligne bleue pleine) et la fonction de transfert de synchronisation pire-cas correspondant à la loi de commande garantissant la performance globale $\widehat{S}^{perf}_{Ref\to 16}$ (la ligne rouge pleine), la contrainte fréquentielle du cahier des charges pour le réseau global (la ligne rouge en trait court) est donnée à titre illustratif

Topologie	Top.A	Top.B	Top.C
r	0.905	0.903	0.901
c	0.1	0.1	0.1
x	-1.236	-1.2421	-1.247
y	0.124	0.1242	0.125
z	1	1	1
μ	$44.2\,dB$	$39.43\,dB$	$45.25\,dB$
μ_{Sb}	$23.58\,dB$	$21.55\,dB$	$31.9\,dB$
μ_{Tb}	$23.57\,dB$	$19.18\,dB$	$30.85\,dB$
μ_{TRef}	$3.57\,dB$	$1.84\,dB$	$1.63\,dB$
$K_{perf}(s)$	$\frac{7.32(s+419.5)}{s}$	$\frac{7.16(s+239.6)}{s}$	$\frac{7.22(s+40.16)}{s}$
$\left\|\widehat{T}\right\|_\infty$	0.996	0.9983	0.99996
α	0.01	0.003	0.002

Table 5.7: Résultats numériques de la synthèse de la loi de commande assurant la performance pour les topologies A, B et C représentées sur la Figure 5.1a, b et c respectivement

5.5 Choix d'emplacement de la référence en réseau

Dans cette section, nous étudions la problématique du choix de l'emplacement du signal de référence en réseau, c'est-à-dire comment connecter la référence au réseau afin d'avoir la plus grande rapidité de synchronisation possible. En effet, si l'application le permet, le choix de l'emplacement joue assez souvent un rôle important sur la dynamique du réseau global. Ainsi, par exemple pour la Topologie A du réseau d'ADPLLs de la Figure 5.1a, il est plus intéressant de connecter la référence à une ADPLL placée au milieu du réseau (l'ADPLL 6, 7, 10 ou 11). Dans ce cas-là, la « distance » entre le signal de référence et l'ADPLL la plus éloignée par rapport au cas représenté sur la Figure 5.1a est plus petite. En conséquence, la « différence » μ entre la fonction de transfert de synchronisation pire-cas $\widehat{S}_{Ref\to j}$ et la fonction de transfert locale S sera donc plus petite. Il est néanmoins nécessaire que l'ADPLL à laquelle le signal de référence est connecté soit une ADPLL ayant au moins un chemin de connexion à toutes les autres ADPLLs, autrement la synchronisation ne peut pas être garantie (voir la page 121 pour plus de détails sur ce sujet).

Cependant, selon la topologie considérée, il peut être difficile de choisir l'emplacement optimal du point de vue de la synchronisation la plus rapide. Un exemple d'une telle topologie est la topologie représentée sur la Figure 5.1c. Nous allons voir par la suite, qu'en utilisant la méthode développée dans le chapitre 4 nous pouvons résoudre un tel problème. Formalisons-le.

Soit M_0 le système LTI décrivant l'interconnexion de sous-systèmes en réseau sans le signal de référence φ_{Ref} :

$$\begin{array}{rcl} \varphi & = & \overbrace{(I_N \otimes T_s)}^{\widetilde{T}} \quad r \\ r & = & M_0 \qquad \varphi \end{array} \tag{5.20}$$

avec $T_s = \underbrace{\begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}}_{H} \star K$, $r(t)$, $\varphi(t) \in \mathbb{R}^{pN}$.

Soit M_i le système LTI décrivant l'interconnexion de sous-systèmes en réseau avec le signal de référence φ_{Ref} connecté au i-ème sous-système :

$$\begin{array}{rcl} \varphi & = & \overbrace{(I_N \otimes T_s)}^{\widetilde{T}} \qquad r \\ \begin{bmatrix} r \\ \varepsilon_{Ref} \end{bmatrix} & = & \underbrace{\begin{bmatrix} M_{11}^i & M_{12}^i \\ M_{21}^i & M_{22}^i \end{bmatrix}}_{M_i} \begin{bmatrix} \varphi \\ \varphi_{Ref} \end{bmatrix} \end{array} \tag{5.21}$$

avec $T_s = \underbrace{\begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}}_{H} \star K$, $r(t)$, $\varphi(t) \in \mathbb{R}^{pN}$, $\varphi_{Ref}(t) \in \mathbb{R}^{p}$, $\varepsilon_{Ref} = \mathbf{1}_N \otimes \varphi_{Ref} - \varphi$,

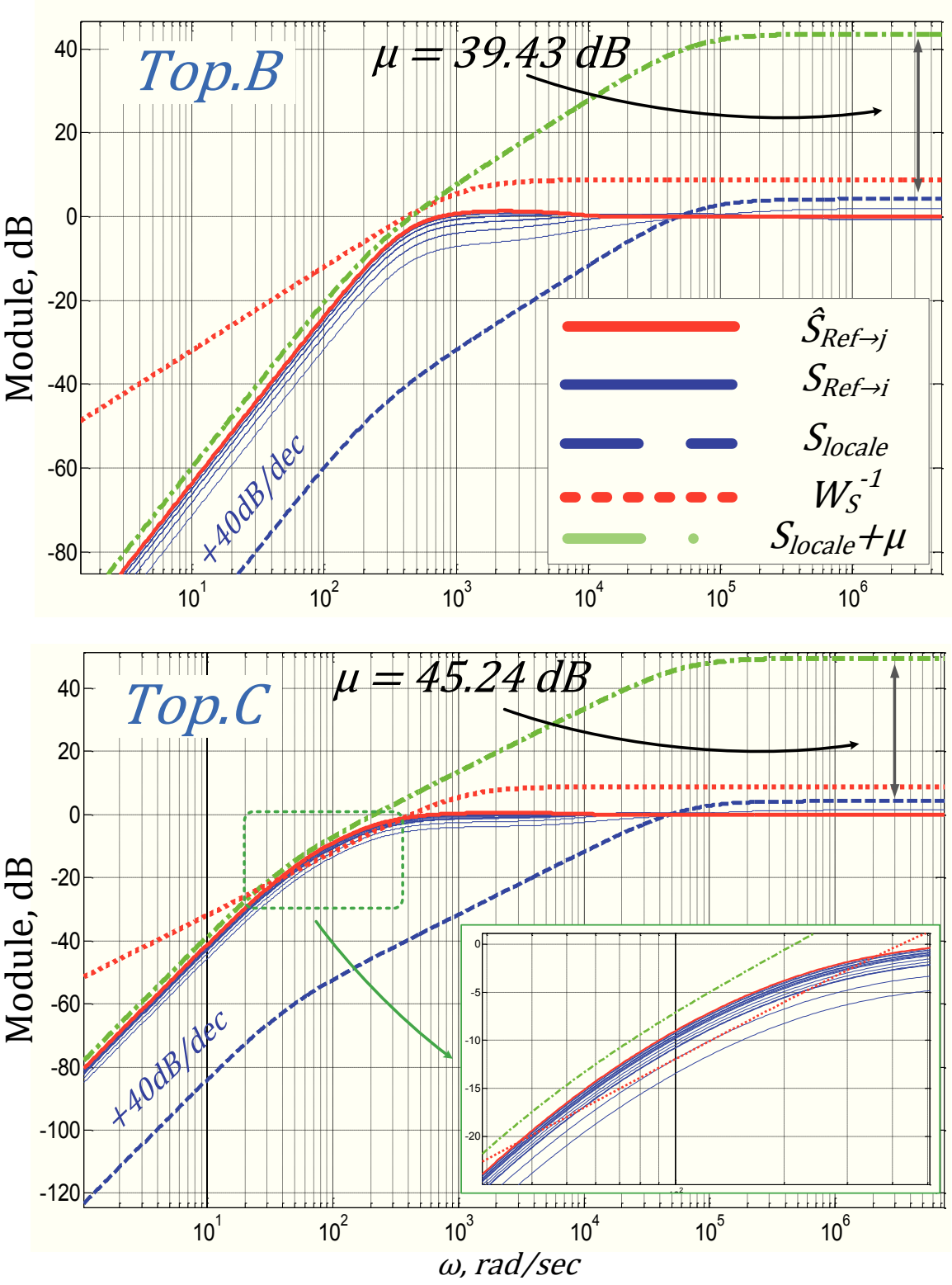
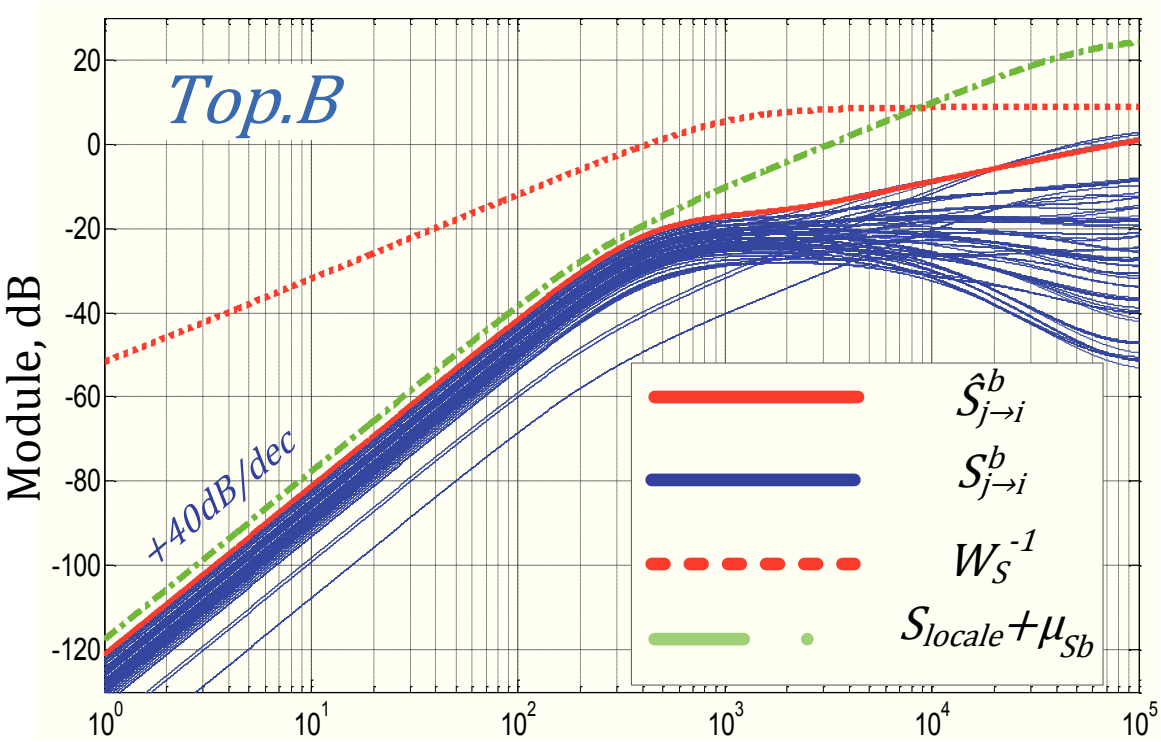


Figure 5.21: Fonctions de transfert de synchronisation $S_{Ref\to i}$ (les lignes bleues pleines), fonction de transfert de synchronisation pire-cas $\hat{S}_{Ref\to j}$ (la ligne rouge pleine), fonction de transfert locale S_{locale} (la ligne bleue en trait long), fonction de transfert locale + $\mu\,dB$ (la ligne verte en trait interrompu mixte) et la contrainte fréquentielle du cahier des charges pour le réseau global (la ligne rouge en trait court) pour les deux topologies : Top.B et Top.C de la Figure 5.1b et c respectivement



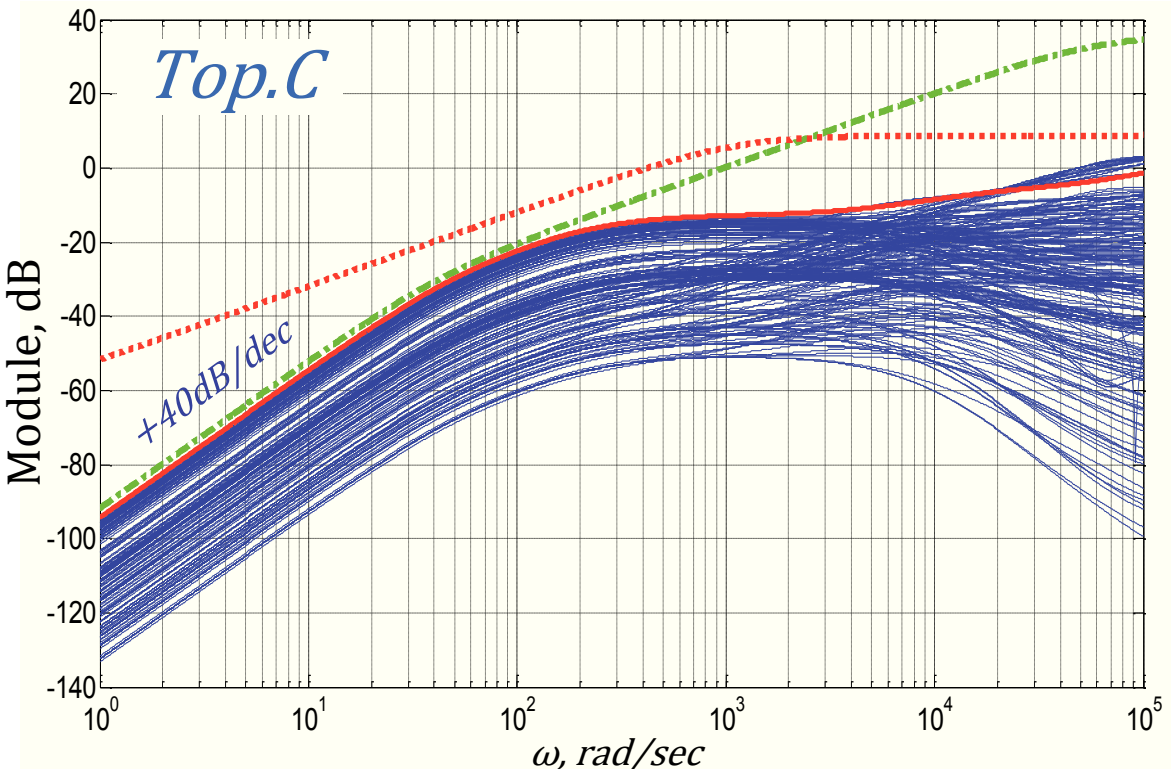


Figure 5.22: Fonctions de transfert du bruit $S^b_{j\to i}$ (les lignes bleues pleines), fonction de transfert de synchronisation pire-cas $\widehat{S}^b_{j\to i}$ (la ligne rouge pleine), fonction de transfert locale $S_{locale} + \mu\, dB$ (la ligne verte en trait interrompu mixte) et la contrainte fréquentielle du cahier des charges pour le réseau global (la ligne rouge en trait court) pour les deux topologies : Top.B et Top.C de la Figure 5.1b et c respectivement

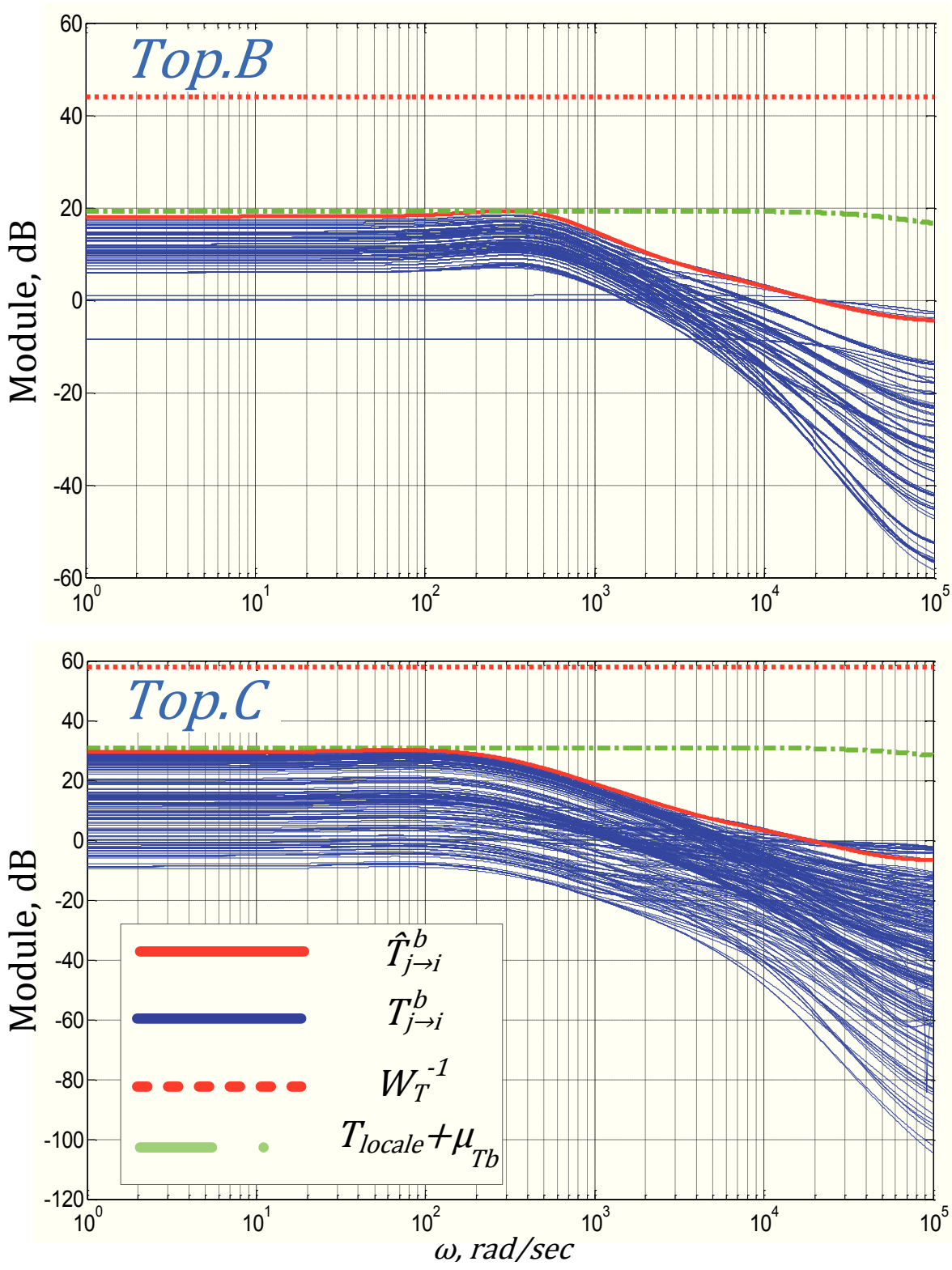


Figure 5.23: Fonctions de transfert $T^b_{j\to i}$ (les lignes bleues pleines), fonction de transfert de bruit pire-cas $\widehat{T}^b_{j\to i}$ (la ligne rouge pleine), fonction de transfert locale $T_{locale} + \mu\, dB$ (la ligne verte en trait interrompu mixte) et la contrainte fréquentielle locale correspondante (la ligne rouge en trait court) pour les deux topologies : Top.B et Top.C de la Figure 5.1b et c respectivement

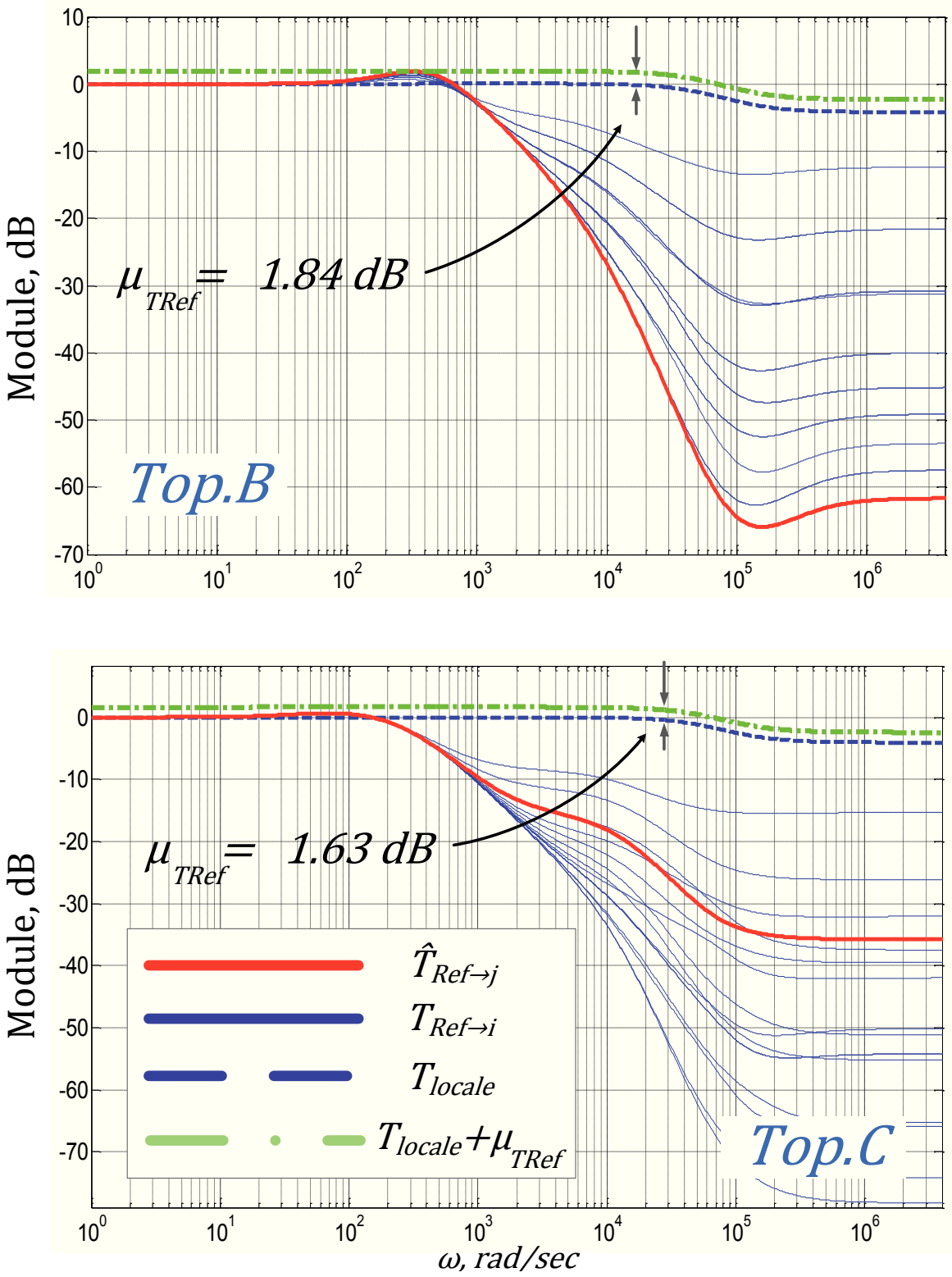


Figure 5.24: Fonctions de transfert $T_{Ref\to j}$ (les lignes bleues pleines), fonction de transfert de bruit pire-cas $\widehat{T}_{Ref\to i}$ (la ligne rouge pleine), fonction de transfert locale T_{locale} (la ligne bleue en trait long), fonction de transfert locale + $\mu\, dB$ (la ligne verte en trait interrompu mixte) pour les deux topologies : Top.B et Top.C de la Figure 5.1b et c respectivement

$\varepsilon_{Ref} \in \mathbb{R}^{pN}$ et où $\mathbf{1}_N$ est une vecteur colonne composé de 1.

Le système décrit par (5.21) est un système avec une entrée φ_{Ref} et N sorties ε_{Refj} correspondant à l'erreur de suivi du signal de référence par chaque sous-système $\varepsilon_{Refj} = \varphi_{Ref} - \varphi_j$. Il est important de mentionner que pour notre application $M_0 \neq M_{11}^i$ étant donné que la connexion du signal de référence au réseau implique un ajout au sous-système concerné d'un bloc d'addition (un comparateur de phase) et un changement de son coefficient de sommation m_i.

La fonction de transfert de synchronisation $S_{Ref\to j}^i = T_{\varphi_{Ref}\to\varepsilon_{Refj}}^i$ définie la rapidité de synchronisation du réseau entre le signal de référence φ_{Ref} et de l'erreur de son suivi par le j-ème sous-système pour la topologie d'interconnexion dans laquelle le signal de référence est connecté au i-ème sous-système.

Définissons la fonction de transfert relative pondérée $T_g^i(s)$ de la même manière comme dans (4.42), page 174 :

$$T_g^{i\to j}(s) = \frac{S_{Ref\to j}^i(s)}{S(s) + \alpha}, \tag{5.22}$$

où $\alpha = 0.05$.

La norme H_∞ de la fonction de transfert relative pondérée (5.22) évalue la « différence » entre la performance globale et la performance locale définie par la fonction de sensibilité locale S (voir la section 4.3 du chapitre 4 pour plus de détails). Elle est facilement décrite sous forme d'une UFT en dynamique locale de sous-systèmes T_s par (4.26) et la Figure 4.7, page 162 avec

$$\begin{aligned} \breve{T} &= I_{N+1} \otimes T_s \\ \widetilde{M}^{i\to j} &= \begin{bmatrix} \frac{1}{1+\alpha} & \frac{1}{1+\alpha} M_{21}^i(j,:) & \frac{1}{1+\alpha} M_{22}^i(j) \\ 0 & M_{11}^i & M_{12}^i \\ \frac{1}{1+\alpha} & \frac{1}{1+\alpha} M_{21}^i(j,:) & \frac{1}{1+\alpha} M_{22}^i(j) \end{bmatrix} \end{aligned} \tag{5.23}$$

$$\widetilde{r} = \begin{bmatrix} q & r_1 & \cdots & r_N \end{bmatrix}^T, \widetilde{\varphi} = \begin{bmatrix} p & \varphi_1 & \cdots & \varphi_N \end{bmatrix}^T, \widetilde{w} = \varphi_{Ref}, \widetilde{z} = q$$

où $M_{21}(j,:)$ désigne la j-ème ligne de la matrice M_{21}, $M_{22}(j)$ désigne le j-ème élément de la matrice colonne M_{22}.

Le problème du choix de l'emplacement de connexion du signal de référence afin d'assurer la synchronisation la plus rapide de toutes les ADPLLs est formalisé comme suit :

Problème 5.1. Pour un système d'interconnexion donné (5.20), trouver un indice optimal i de connexion du signal de référence au réseau au sens que la norme H_∞ maximale de toutes les fonctions de transfert $T_g^{i\to j}(s)$ définies par (5.23) et (5.21)

soit la plus petite ou autrement dit :

$$\min_i \quad \mu_{\max} = \max_j \left(20\log_{10} \left\| T_g^{i\to j}(s) \right\|_\infty\right) \tag{5.24}$$

Le Problème 5.1 en général n'est pas un problème d'optimisation convexe et donc on peut se poser la question de l'existence de méthodes de résolution efficaces. Cependant en utilisant le Théorème 4.7, page 153 et le fait qu'il existe un nombre fini N de valeurs possibles d'indice i, on peut résoudre le problème de manière directe. En effet, nous allons nous baser sur les trois observations suivantes.

1. L'utilisation de l'approche de performance globale relative permet d'évaluer la « différence » entre la fonction de transfert de synchronisation $S^i_{Ref\to j}$ et la fonction de sensibilité locale S indépendamment de la dynamique des sous-systèmes T_s à condition qu'ils soient tous identiques et $\{x, y, z\}$-dissipatifs. Étant donné que tous les sous-systèmes sont identiques, l'approche de performance globale relative permet de réaliser une sorte de « normalisation » de l'évaluation de la performance globale par la dynamique locale des sous-systèmes. Cette évaluation de performance dépend donc seulement de l'interconnexion M_i.
2. Le Théorème 4.7, page 153 permet d'estimer *a priori* le niveau de performance globale relative η, avant le calcul explicite de la loi de commande K ou bien pour toutes les lois de commande K satisfaisant la condition (4.17) c'est-à-dire la condition de $\{x, y, z\}$-dissipative de chaque sous-système T_s.
3. Pour les valeurs de caractérisation de dissipativité x, y, z fixées (en résolvant par exemple le problème d'optimisation (4.38) pour des indices i, j fixés c'est-à-dire un système donné (5.21) avec une sortie j donnée), le problème de minimisation de η sous contrainte (4.16) avec $M = \widetilde{M}^{i\to j}$ est un problème d'optimisation convexe LMI. La résolution de ce problème pour toutes les $l = N^2$ combinaison possible peut être réalisée de manière efficace même pour un grande nombre N de sous-systèmes.

Le Problème 5.1 est ainsi facilement résolu pour chaque topologie d'interconnexion M_i par la résolution du problème de minimisation de η sous contrainte (4.27) pour les valeurs x, y, z fixées et avec $M = \widetilde{M}^{i\to j}$, $\forall j \in \{1, 2, \cdots, N\}$.

En appliquant la méthode décrite de la résolution du Problème 5.1 pour le cas de Topologie C (voir la Figure 5.1c), on obtient le résultat représenté sur la Figure 5.25. A côté de chaque sous-système sur la Figure 5.25 nous avons indiqué la valeur maximale de la norme H_∞ μ_i^j en dB de la fonction de transfert $\widehat{T}_g^{i\to j}(s)$, c'est-à-dire la fonction de transfert correspondante au pire-cas. L'indice inférieur i indique le numéro de sous-système auquel le signal de référence a été connecté tandis que l'indice supérieur j à son tour indique le numéro de sous-système le plus éloigné du sous-système i, c'est-à-dire dont la fonction de transfert de synchronisation $S^i_{Ref\to j}$ correspond au pire-cas. Le sous-système $i = 16$ est un sous-système n'ayant pas de chemins le reliant à tous les autres sous-systèmes, c'est pour ces raisons que nous n'avons pas appliqué l'optimisation pour le système d'interconnexion M_{16}.

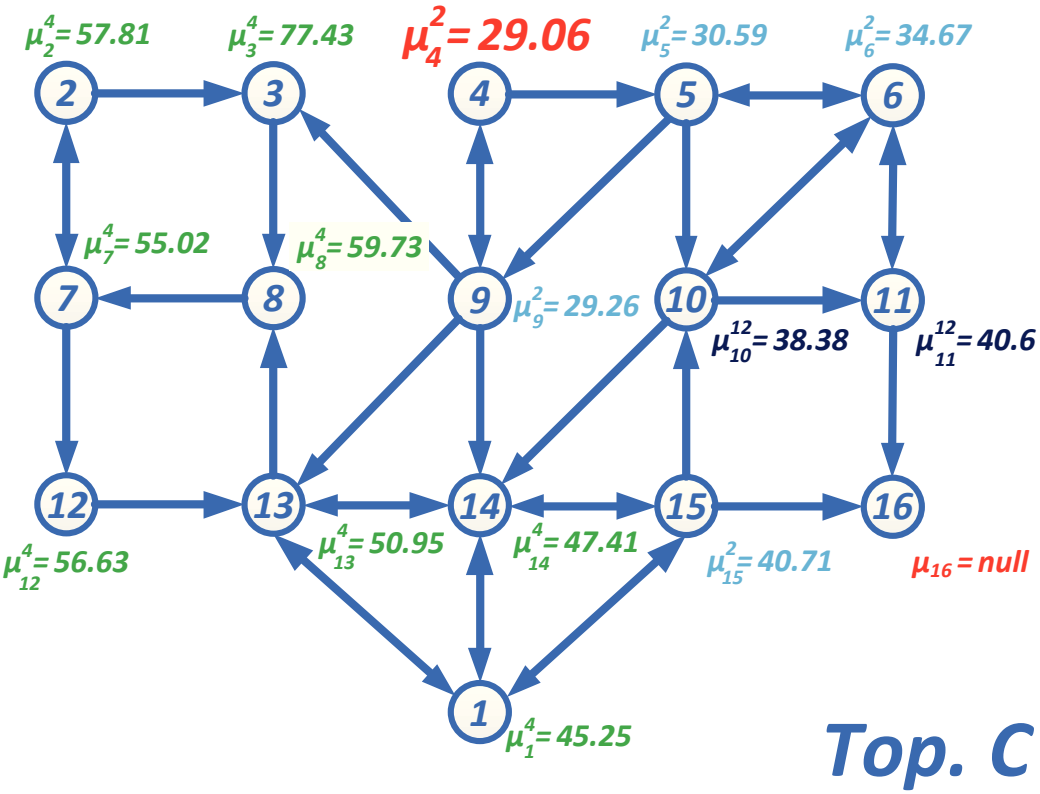


Figure 5.25: Résultat du choix d'emplacement du signal de référence pour la Topologie C

Analysant le résultat obtenu, nous arrivons à la conclusion que le meilleur endroit pour l'emplacement du signal de référence est le sous-système $i = 4$ (représenté en rouge sur la Figure 5.25). Ce résultat est confirmé par la Figure 5.26 où nous avons présenté toutes les fonctions de transfert de synchronisation pour la même loi de commande K_{perf} synthétisée dans la section précédente (voir le Tableau 5.6, page 213) $S^4_{Ref\to j}$ pour $\forall j \in \{1, 2, \cdots, N\}$ (les lignes bleues pleines) avec la contrainte de la performance du cahier des charges pour le réseau global (la ligne rouge en trait court) ainsi que la fonction des transfert de synchronisation pire-cas $\widehat{S}^4_{Ref\to 2}$ (la ligne rouge pleine). Une comparaison des fonctions de transfert de synchronisation pire-cas pour la Topologie C avec la référence connectée au premier sous-système $\widehat{S}^1_{Ref\to 16}$ (la ligne bleue en trait long) et avec la référence connectée au quatrième sous-système $\widehat{S}^4_{Ref\to 2}$ (la ligne rouge pleine) est présentée sur la Figure 5.27.

Nous constatons que la fonction de synchronisation $\widehat{S}^4_{Ref\to 2}$, contrairement au premier cas (voir aussi la Figure 5.21) étant à $\mu_1^{16} - \mu_4^2\, dB$ plus « proche » de la fonction de transfert locale S_{locale} (la ligne bleue en trait long sur la Figure 5.27), respecte la contrainte fréquentielle de performance globale $\left(W_s^{globale}\right)^{-1}$ (la ligne rouge en trait court sur la Figure 5.27).

La même démarche peut être appliquée pour la Topologie A du réseau. Cependant à cause des exigences de réalisation pratique sur la carte FPGA d'un tel réseau d'ADPLL, nous continuons à réaliser la μ-analyse sur le Modèle d'analyse ainsi que

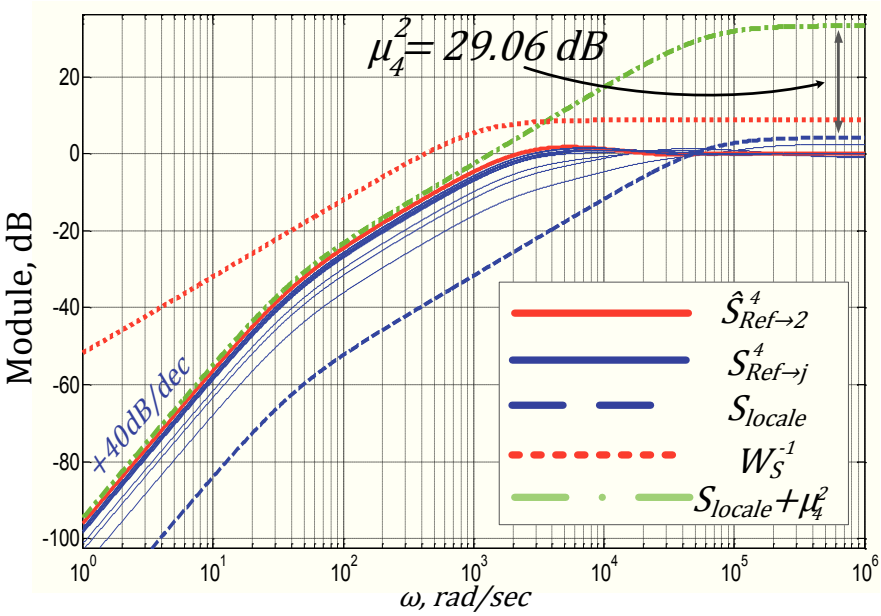


Figure 5.26: Fonctions de transfert de synchronisation $S^4_{Ref\to j}$ (les lignes bleues pleines), fonction de transfert de synchronisation pire-cas $\widehat{S}^4_{Ref\to 2}$ (la ligne rouge pleine), fonction de transfert locale S_{locale} (la ligne bleue en trait long), fonction de transfert locale + $\mu\, dB$ (la ligne verte en trait interrompu mixte) et la contrainte fréquentielle du cahier des charges pour le réseau global (la ligne rouge en trait court)

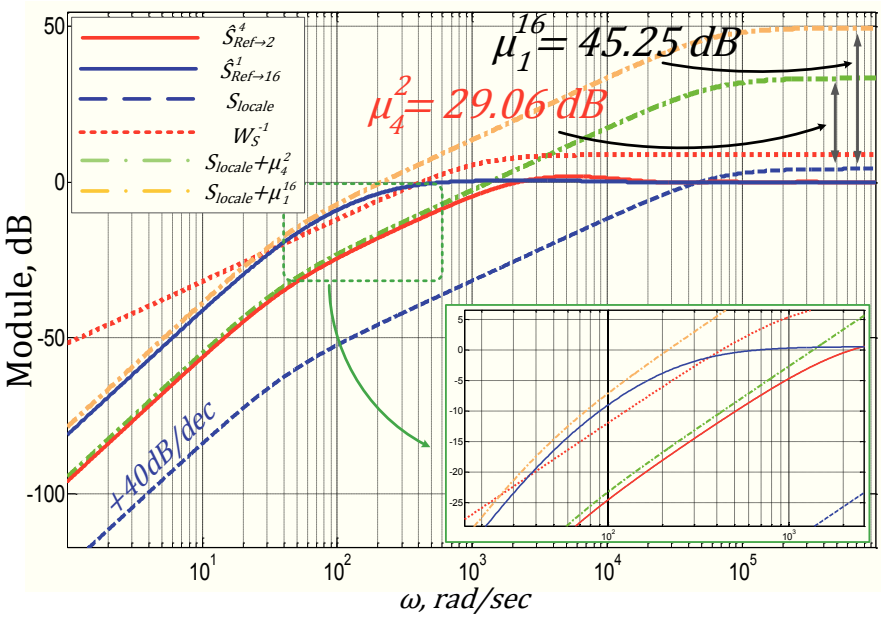


Figure 5.27: Comparaison de fonctions de transfert de synchronisation pire-cas $\widehat{S}^4_{Ref\to 2}$(la ligne rouge pleine) et $\widehat{S}^1_{Ref\to 16}$ (la ligne bleue pleine), ainsi que de fonction de transfert locale S_{locale} (la ligne bleue en trait long) + $\mu_4^2\, dB$ (la ligne verte en trait interrompu mixte) et $\mu_1^{16}\, dB$ (la ligne orange en trait interrompu mixte), la contrainte fréquentielle du cahier des charges pour le réseau global (la ligne rouge en trait court)

les simulations temporelles sur le Modèle de simulation dans les sections qui suivent pour le cas de la Topologie A comme elle est présentée sur la Figure 5.1a c'est-à-dire avec le signal de référence connecté au premier sous-système.

5.6 Robustesse du réseau vis-à-vis d'incertitudes paramétriques

La méthode de synthèse de la loi de commande décrite dans le chapitre 4 et appliquée dans la section 5.4 de ce chapitre se base sur le modèle nominal du réseau (voir la Figure 5.5 la Figure 4.2, page 148, la Figure 4.1, page 147) et l'identité de sous-systèmes locaux (voir les équations (4.1), page 146 et (4.2), page 146). Cependant en pratique, à cause de dispersions technologiques, les paramètres des sous-systèmes ne correspondent jamais aux valeurs nominales ; par conséquent les sous-systèmes ne sont jamais identiques. Dans cette section, comme cela a été discuté à la fin du chapitre 1, page 26, on propose d'appliquer une méthode d'analyse complémentaire à la méthode d'analyse Monte-Carlo (voir le début de section 2.6 du chapitre 2, page 79 pour plus de détails). Il s'agit de la méthode de μ-analyse du comportement du système global qui a été discuté en détails dans la section 2.6 du chapitre 2, page 79, et a été appliquée pour l'analyse du comportement du Modèle d'analyse d'une seule ADPLL dans la sous-section 2.6.3, page 88. Dans cette section, nous allons étudier comment la méthode de μ-analyse peut être appliquée au réseau d'ADPLLs.

Le correcteur réalisant la loi de commande décentralisée conçu dans la section 5.4 est un correcteur PI (voir 5.19, page 209). Ainsi le Modèle d'analyse de la $i^{ième}$ ADPLLs au sein du réseau (voir la Figure 5.6, page 190) devient comme présenté sur la Figure 5.28. Sur la Figure 5.28, pour la clarté de la présentation, nous n'avons pas présenté les signaux de perturbation q_{PDi}, q_{VCOi} et q^b_{VCOi}. Les paramètres K_P^{nom}et K_I^{nom} sont les paramètres nominaux du correcteur continu $K_{perf}(s)$, Δ_P et Δ_I sont leurs incertitudes paramétriques (voir (5.2), page 188).

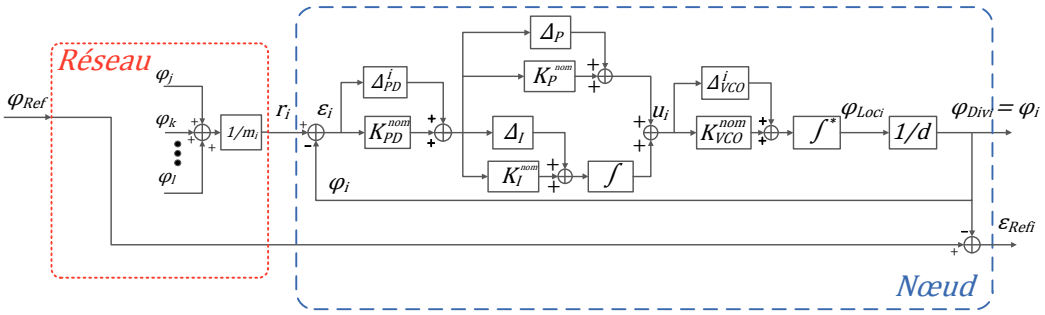


Figure 5.28: Modèle d'analyse de la $i^{ième}$ ADPLLs au sein du réseau équivalent avec le correcteur PI

Afin de réduire le nombre d'incertitudes à prendre en compte, nous faisons un changement de variables semblable à celui effectué lors d'analyse du comportement d'une seule ADPLL (voir (2.53), page 88). Cependant, cette fois-ci, nous regroupons séparément des incertitudes communes (Δ_P, Δ_I, Δ_{VCO} et Δ_{PD}) et des incertitudes

variables ($\widetilde{\Delta}^i_{VCO}$,$\widetilde{\Delta}^i_{PD}$) de la manière suivante :

$$\begin{aligned} K &= \underbrace{K^{nom}_{PD} \cdot K^{nom}_{VCO} \cdot \frac{1}{d}}_{K_0} \left(1 + \overbrace{K_d \delta_K}^{\Delta_K} + \overbrace{\widetilde{K}_d \delta^i_K}^{\widetilde{\Delta}_{K_i}} \right); \\ K_P &= K^{nom}_P \left(1 + \overbrace{K_{dP} \delta_P}^{\Delta_P} \right); \\ K_I &= K^{nom}_I \left(1 + \overbrace{K_{dI} \delta_I}^{\Delta_I} \right), \end{aligned} \tag{5.25}$$

avec

$$\begin{aligned} K_d &= \left(1 + \frac{K^{gab}_{PD}}{K^{nom}_{PD}}\right)\left(1 + \frac{K^{gab}_{VCO}}{K^{nom}_{VCO}}\right) - 1; \\ \widetilde{K}_d &= \left(1 + \frac{K^{gab}_{PD}}{K^{nom}_{PD}}\right)\frac{\widetilde{K}^{gab}_{VCO}}{K^{nom}_{VCO}} + \left(1 + \frac{K^{gab}_{VCO}}{K^{nom}_{VCO}}\right)\frac{\widetilde{K}^{gab}_{PD}}{K^{nom}_{PD}} + \frac{\widetilde{K}^{gab}_{VCO}}{K^{nom}_{VCO}}\frac{\widetilde{K}^{gab}_{PD}}{K^{nom}_{PD}}; \\ K_{dP} &= \frac{K^{gab}_P}{K^{nom}_P} \qquad K_{dI} = \frac{c}{K^{nom}_I}, \end{aligned} \tag{5.26}$$

et δ_K, δ^i_K, δ_P, $\delta_I \in \mathbb{R} : |\delta_K| < 1$, $|\delta^i_K| < 1$, $|\delta_P| < 1$, $|\delta_I| < 1$.

La Figure 5.29 représente une version du Modèle d'analyse de chaque ADPLL sous forme standard de μ-analyse.

Dans la suite, nous regroupons ensembles les entrées w_{ji} et les sorties z_{ji} de chaque ADPLL de sorte à former le réseau global d'ADPLLs interconnectées représenté sur la Figure 5.30 avec $\widetilde{\Delta} \in \underline{\widetilde{\Delta}}$ où

$$\underline{\widetilde{\Delta}} = \left\{ \widetilde{\Delta} \in \mathbb{R}^{4N \times 4N} / \widetilde{\Delta} = \begin{bmatrix} \delta^1_K & \cdots & 0 & & & & & & & & & \\ \vdots & \ddots & \vdots & & \cdots & & & \cdots & & & 0 & \\ 0 & \cdots & \delta^N_K & & & & & & & & & \\ & & & \delta_K & \cdots & 0 & & & & & & \\ & \vdots & & \vdots & \ddots & \vdots & & \vdots & & & \vdots & \\ & & & 0 & \cdots & \delta_K & & & & & & \\ & & & & & & \delta_P & \cdots & 0 & & & \\ & \vdots & & & \cdots & & \vdots & \ddots & \vdots & & \vdots & \\ & & & & & & 0 & \cdots & \delta_P & & & \\ & & & & & & & & & \delta_I & \cdots & 0 \\ & 0 & & & \cdots & & & \cdots & & \vdots & \ddots & \vdots \\ & & & & & & & & & 0 & \cdots & \delta_I \end{bmatrix} \right\}, \tag{5.27}$$

la structure des incertitudes du réseau complet.

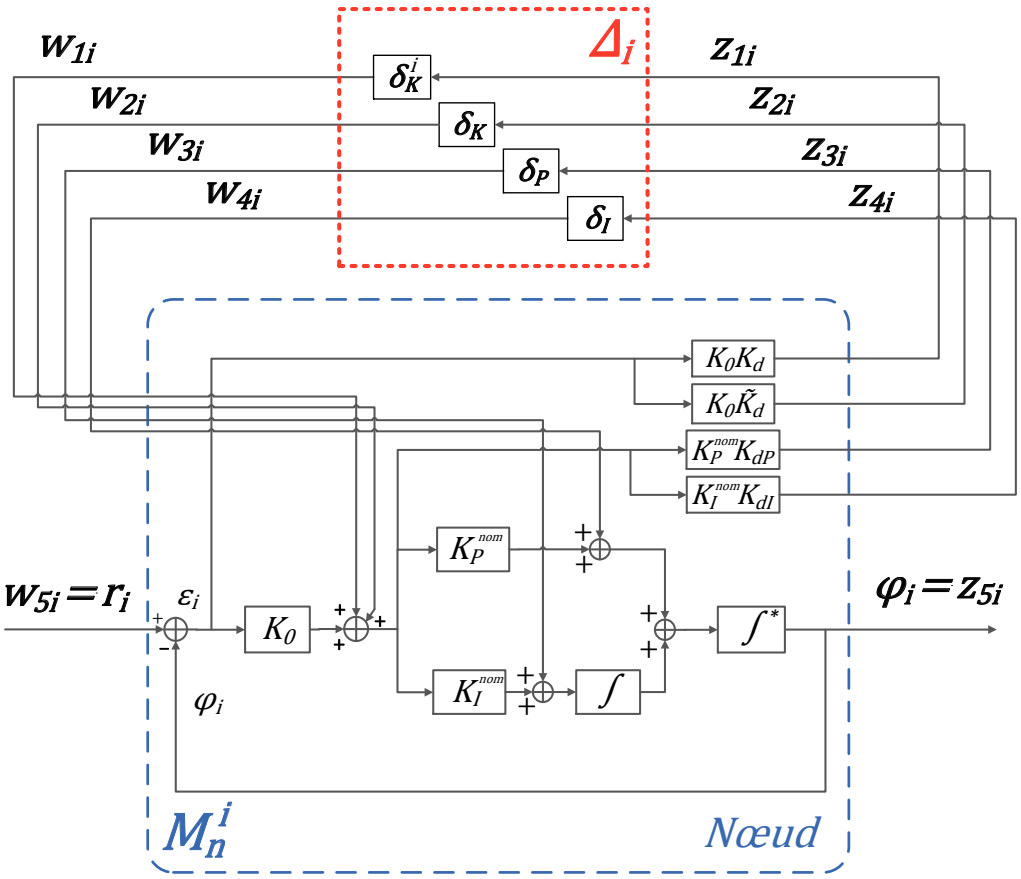


Figure 5.29: Schéma $\Delta_i \star M_{ni}$ standard de la μ-analyse de la i-ème ADPLL du réseau

5.6.1 Stabilité robuste du réseau

Pour tester la stabilité robuste du réseau complet nous appliquons la méthode de μ-analyse décrite dans la section 2.6 du chapitre 2 avec les valeurs d'incertitudes initiales du Tableau 2.1, page 55, Tableau 5.1, page 188 ainsi que les valeurs d'incertitudes paramétriques du correcteur trouvées lors de la μ-analyse de la performance robuste d'une seule ADPLL (voir (2.56), page 91).

Les résultats de calcul des bornes supérieure et inférieure de la valeur singulière structurée fréquence par fréquence sont représentés sur la Figure 5.31. Ils confirment que le réseau complet est stable avec des incertitudes décrites par le Tableau 2.1 et le Tableau 5.1 ainsi que les valeurs d'incertitude paramétriques du correcteur trouvées lors de la μ-analyse de la performance robuste d'une seule ADPLL (voir (2.56), page 91).

5.6.2 Performance globale robuste du réseau

Pour analyser la performance globale robuste du réseau d'ADPLLs, il est nécessaire tout d'abord de choisir une ou plusieurs fonctions de transfert globale de perfor-

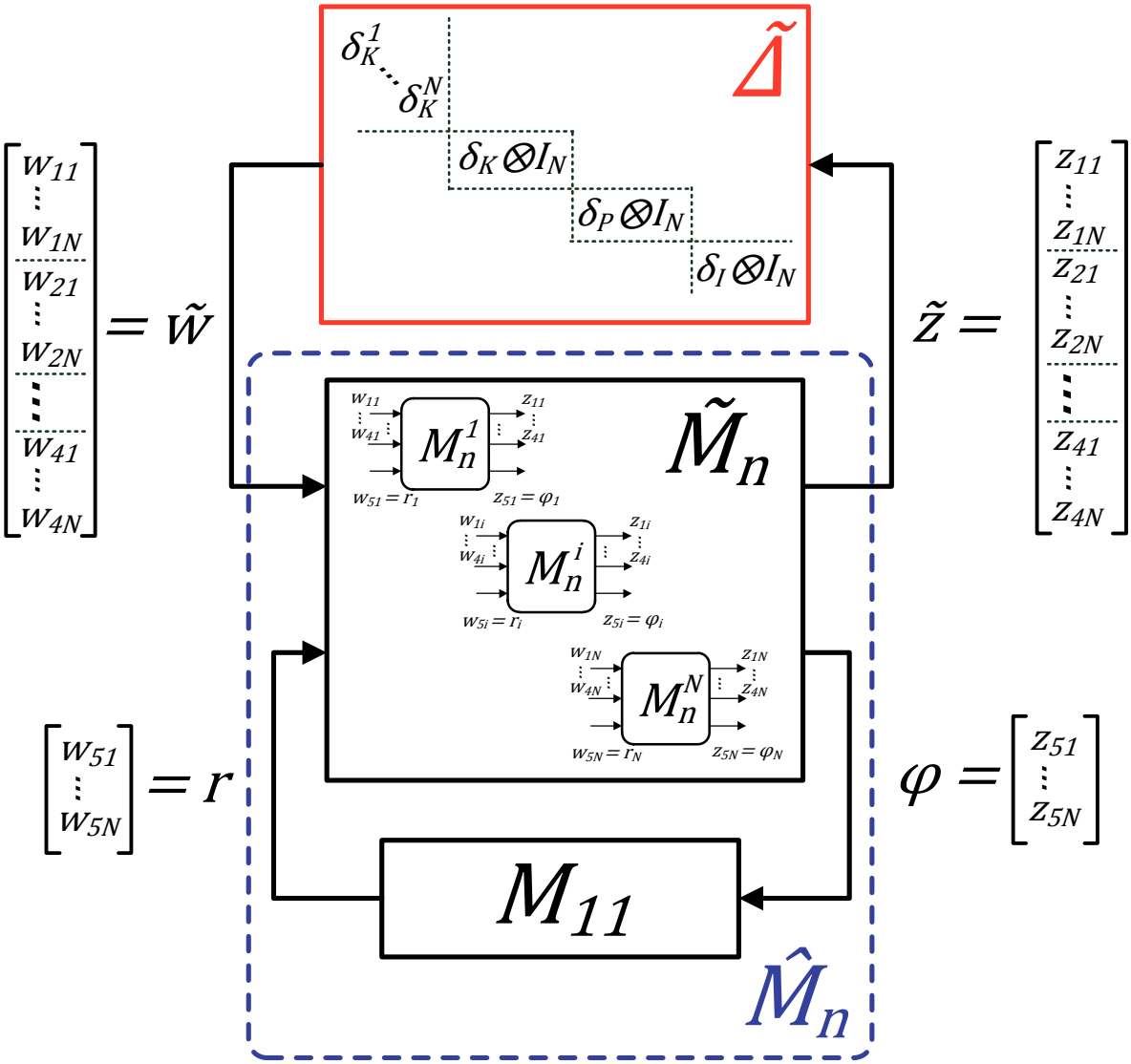


Figure 5.30: Schéma $\tilde{\Delta} \star \widehat{M}_n$ standard de la μ-analyse de la stabilité robuste du réseau d'ADPLLs

mance à analyser et ensuite, selon la méthode décrite dans la sous-section 2.6.2 du chapitre 2, d'introduire dans le schéma global de la Figure 5.30, des entrées et sorties correspondant à ces fonctions de transfert ainsi que les fonctions de pondération fixant les contraintes fréquentielles de performance.

Ainsi, à titre d'illustration pour la Topologie A (voir la Figure 5.1a, page 185), nous choisissons la fonction de transfert globale de synchronisation de pire-cas $\widehat{S}_{Ref \to 16}$ et la contrainte de performance globale fixée par la fonction de pondération $W_s^{globale}$ de (5.9), page 200. Le schéma $\widetilde{\Delta} \star \widehat{M}_n$ standard de la μ-analyse de la performance robuste du réseau d'ADPLLs correspondant est représenté sur la Figure 5.32 avec blocs M_{ij} définis dans (5.14), page 204.

Le calcul de la borne inférieure et la borne supérieure d'indicateur μ pour les valeurs des incertitudes initiales du Tableau 2.1, page 55, Tableau 5.1 ainsi que les

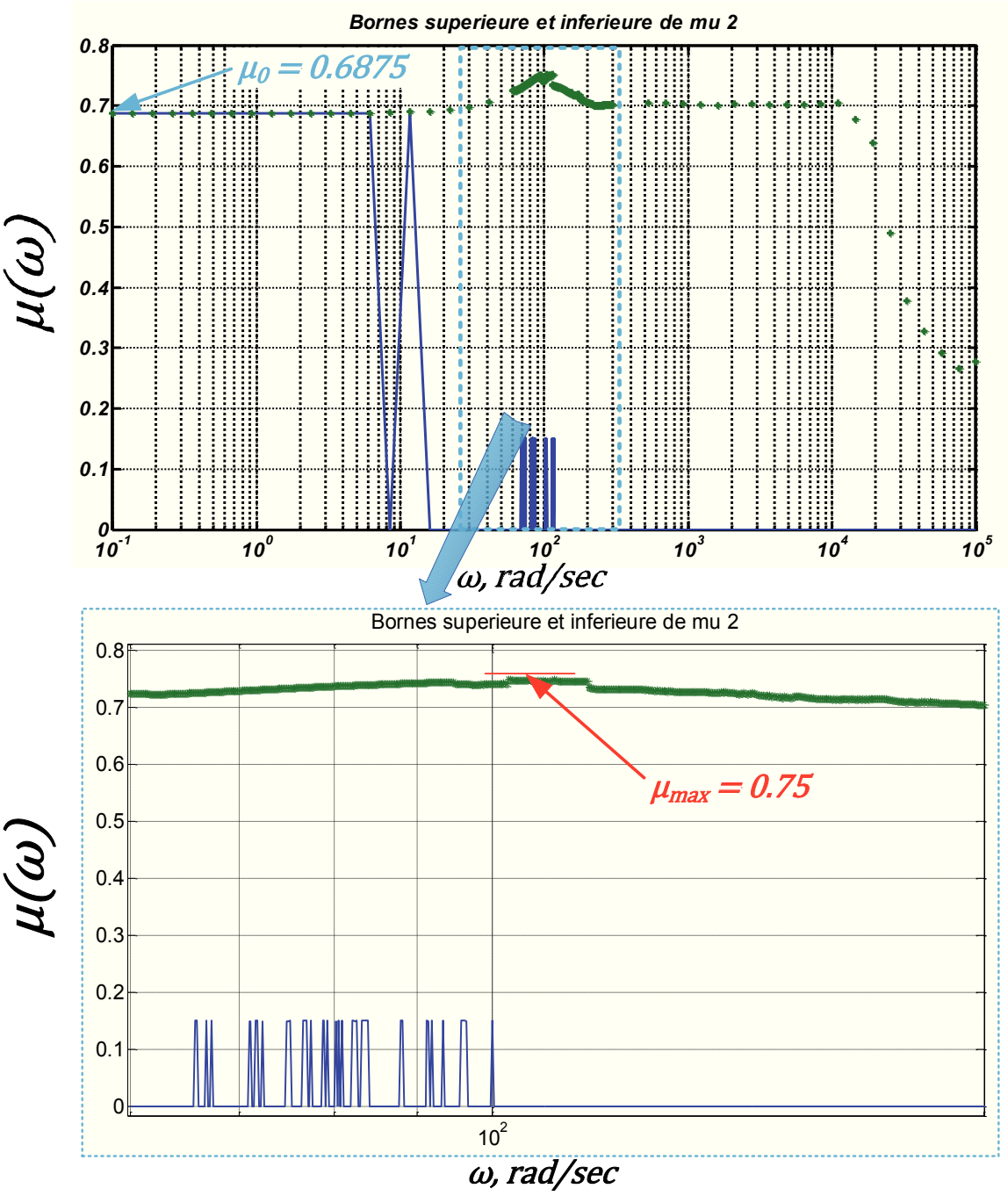


Figure 5.31: Les bornes inférieure (ligne bleue) et supérieure (points verts) de l'indicateur μ de l'exemple numérique du réseau d'ADPLL considéré pour la stabilité robuste

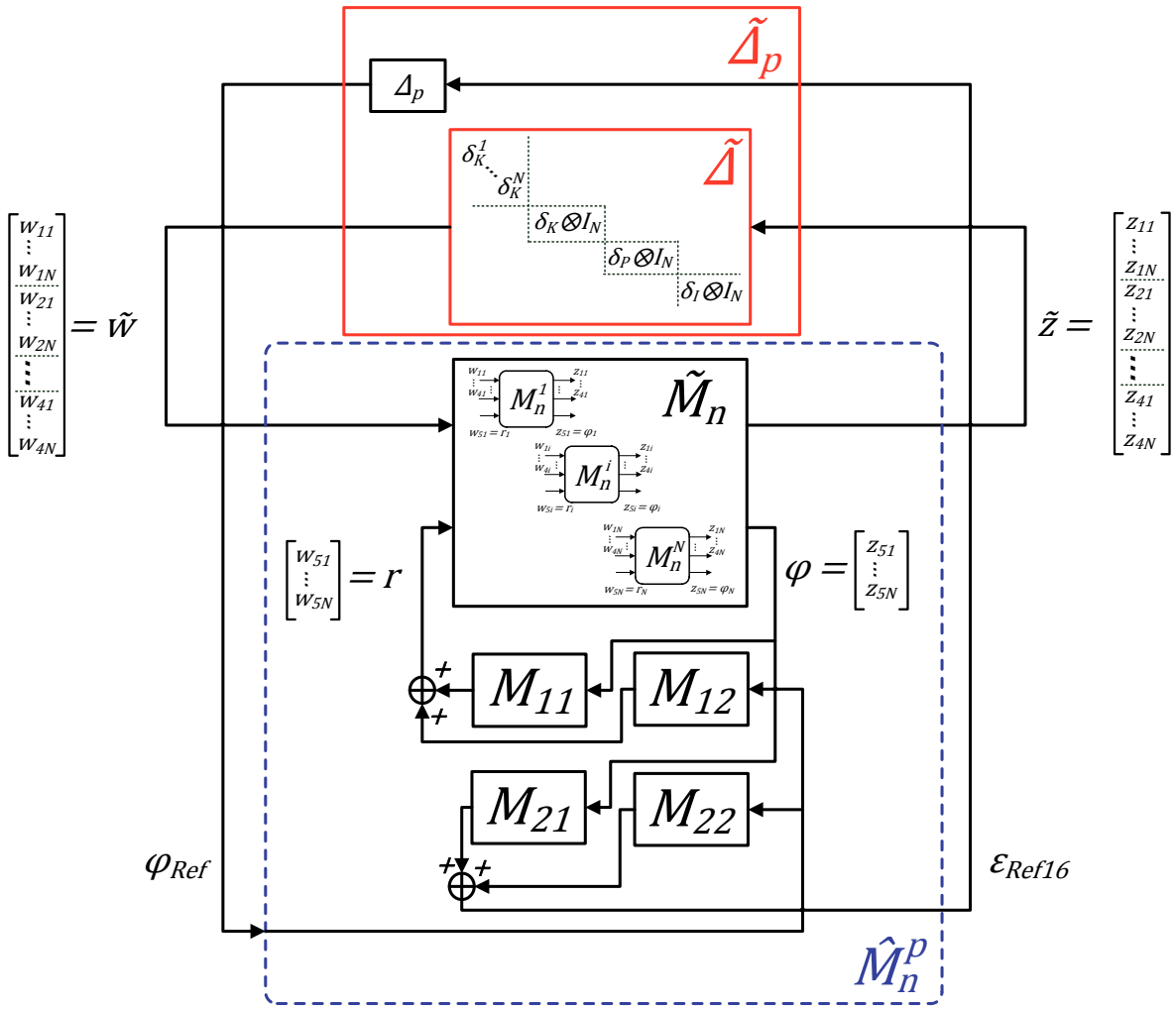


Figure 5.32: Schéma $\widetilde{\Delta} \star \widehat{M}_n$ standard de la μ-analyse de la performance robuste du réseau d'ADPLLs

valeurs d'incertitudes de paramètres du correcteur trouvées lors de la μ-analyse de la performance robuste d'une seule ADPLL (2.56), page 91 donne une valeur de $\mu_{max} > 1$. Étant donné que la stabilité robuste a été testée pour les mêmes valeurs d'incertitudes dans la sous-section précédente, cela signifie que la performance robuste exprimée par la contrainte fréquentielle (voir (5.9), page 200 et la Figure 5.16, page 211) n'est pas assurée.

Cela semble logique dans la mesure où la contrainte fréquentielle est saturée déjà dans le cas nominal (voir la Figure 5.16, page 211). Ainsi, une incertitude très faible sur des paramètres des ADPLLs provoque un dépassement de la contrainte. Pour ces raisons, nous proposons de changer la stratégie de μ-analyse adaptée dans le cas d'une seule ADPLL. Aussi, pour les incertitudes fixées sur les paramètres de *PD*s et de *VCO*s (K_{PD}^{gab} et K_{VCO}^{gab} voir le Tableau 2.1, page 55 ainsi que $\widetilde{K}_{PD}^{gab}$ et $\widetilde{K}_{VCO}^{gab}$ voir le Tableau 5.1), au lieu de considérer la problématique de recherche des incertitudes maximales sur les paramètres du correcteur K_P^{gab} et K_{PD}^{gab} pour garantir la performance robuste, nous allons considérer la problématique de recherche du niveau maximal de performance pour différents jeux de valeurs des incertitudes paramétriques du correcteur K_P^{gab} et K_I^{gab}. Étant donné que la fonction de transfert de synchronisation $\widehat{S}_{Ref\to 16}$ nominale est la plus proche de la contrainte fréquentielle en moyennes fréquences (voir la Figure 5.16, page 211) le niveau de performance sera évalué par la valeur de la fréquence de coupure ω_c^i de la fonction de pondération.

Les résultats de cette démarche pour les différentes valeurs d'incertitudes sur les paramètres du correcteur K_P^{gab} et K_I^{gab} résumées dans le Tableau 5.8 sont représentés sur la Figure 5.33. L'alternative 1 du Tableau 5.8 correspond aux valeurs idéales des paramètres du correcteur c'est-à-dire seules les incertitudes sur les *PD*s et les *VCO*s sont considérées. Ce choix des incertitudes du correcteur irréaliste permet d'évaluer un niveau de performance maximale atteignable dans le cas de correcteurs idéaux. Pour notre cas, il correspond à la valeur de la fréquence de coupure $\omega_c^1 = 110\,rad/sec$. L'alternative 2 du Tableau 5.8 correspond aux valeurs des incertitudes du correcteur retenues lors de la μ-analyse de la performance robuste d'une seule ADPLL (voir (2.56), page 91).

En analysant les résultats obtenus, nous arrivons à la conclusion que l'incertitude sur le gain de la branche proportionnel du correcteur K_P^{gab} est très importante pour la performance robuste. Plus ce paramètre est important, plus la fréquence de coupure de la fonction de pondération fixant le niveau maximal de performance atteint est basse. Ceci est logique car c'est le coefficient statique du correcteur qui fixe la bande passante de la fonction de transfert de synchronisation. Cette tendance se manifeste moins pour le gain de la branche intégrale du correcteur K_I^{gab}.

Nous retenons alors les valeurs d'incertitude paramétrique du filtre de l'alternative 6 assurant la fréquence de coupure de $\omega_c^6 = 96\,rad/sec$ (voir le Tableau 5.8) et correspondant aux incertitudes sur les paramètres du correcteur numérique résumés ci-dessus :

$$\begin{aligned} K_P^{*gab} &= \pm 0.7123 \ \text{ soit } \ \pm 10\%; \\ K_I^{*gab} &= \pm 7.8 \cdot 10^{-3} \ \text{ soit } \ \pm 30\%. \end{aligned} \tag{5.28}$$

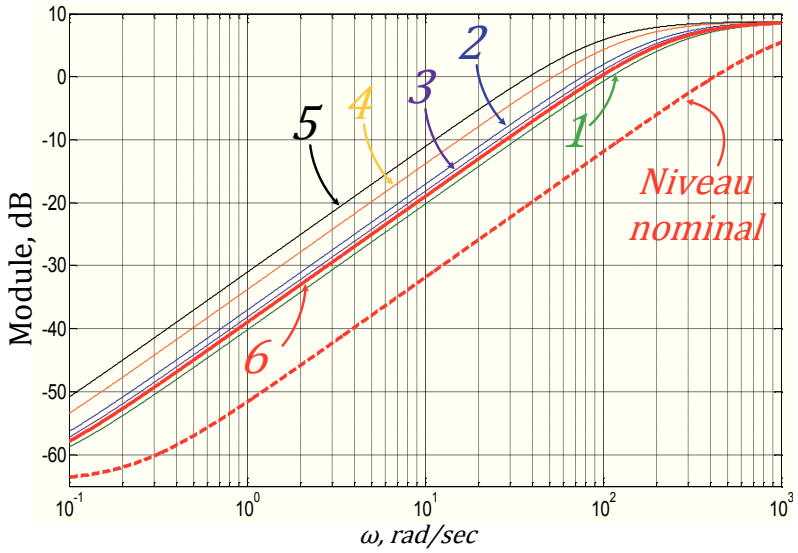


Figure 5.33: Niveaux de performance robuste garantie pour différentes valeurs d'incertitudes sur les paramètres du correcteur

Alternative	K_P^{gab}	K_I^{gab}	ω_c
1	0	0	$110\,rad/sec$
2	0.15	0.73	$76\,rad/sec$
3	0.15	0.3	$86\,rad/sec$
4	0.3	0.15	$52\,rad/sec$
5	0.3	0.3	$38\,rad/sec$
6	0.1	0.3	$96\,rad/sec$

Table 5.8: Jeux de valeurs d'incertitudes paramétriques du correcteur testées

La fonction de pondération associée a la forme suivante :

$$W_s^{globale}(s) \;=\; W_\varepsilon(s)\,W_{Ref}(s) \;=\; \frac{0.36\,(s+246.5)}{s+0.06}. \tag{5.29}$$

Les résultats du calcul de l'indicateur μ pour cette alternative sont représentés sur la Figure 5.34. Ils confirment que la performance robuste est respectée. La Figure 5.35 représente les tracés du module de la fonction de transfert de synchronisation pire-cas $\widehat{S}_{Ref\to 16}$ pour les différentes valeurs des incertitudes. Les valeurs d'incertitudes communes (K_P^{gab} , K_I^{gab}, K_{PD}^{gab} et K_{VCO}^{gab}) ont été choisies selon une grille de 10 valeurs équidistantes, alors que les valeurs d'incertitudes variables ($\widetilde{K}_{PD}^{gab}$ et $\widetilde{K}_{VCO}^{gab}$) ont été choisies dans leurs plages de variations de manière aléatoire pour chaque sous-système séparément et chaque combinaison de valeurs d'incertitudes communes. La Figure 5.35 confirme les résultats de μ-analyse de la performance robuste. En effet, la

famille de fonctions de transfert de synchronisation définie par les incertitudes paramétriques du Tableau 2.1, page 55, du Tableau 5.1 et de (5.28) respecte la contrainte fréquentielle fixée par la fonction de pondération $\left(W_s^{globale}\right)^{-1}$ définie par (5.29).

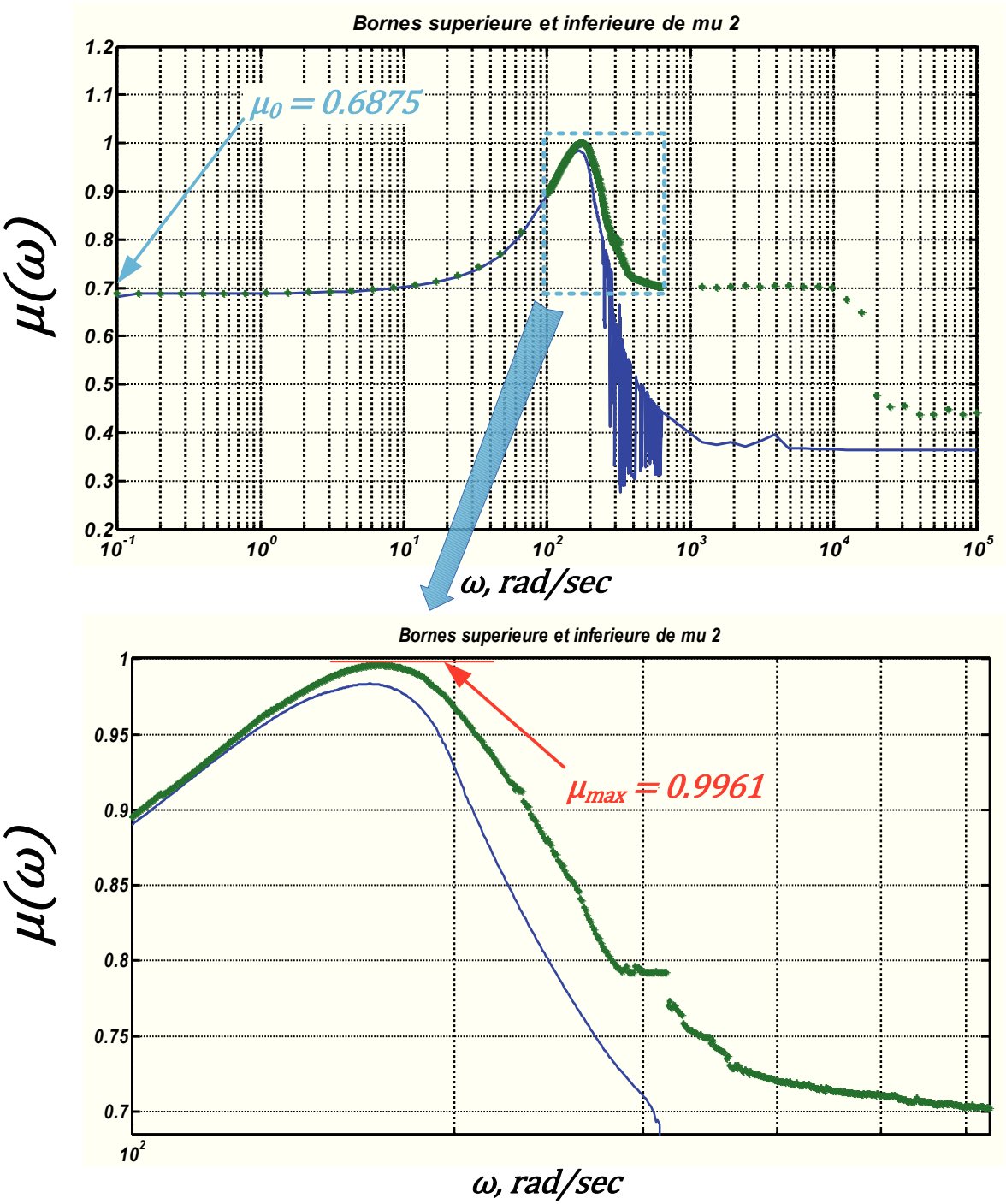


Figure 5.34: Les bornes inférieure (ligne bleue) et supérieure (points verts) de l'indicateur μ de l'exemple numérique du réseau d'ADPLL considéré pour la performance robuste correspondant à l'alternative 6 du Tableau 5.8

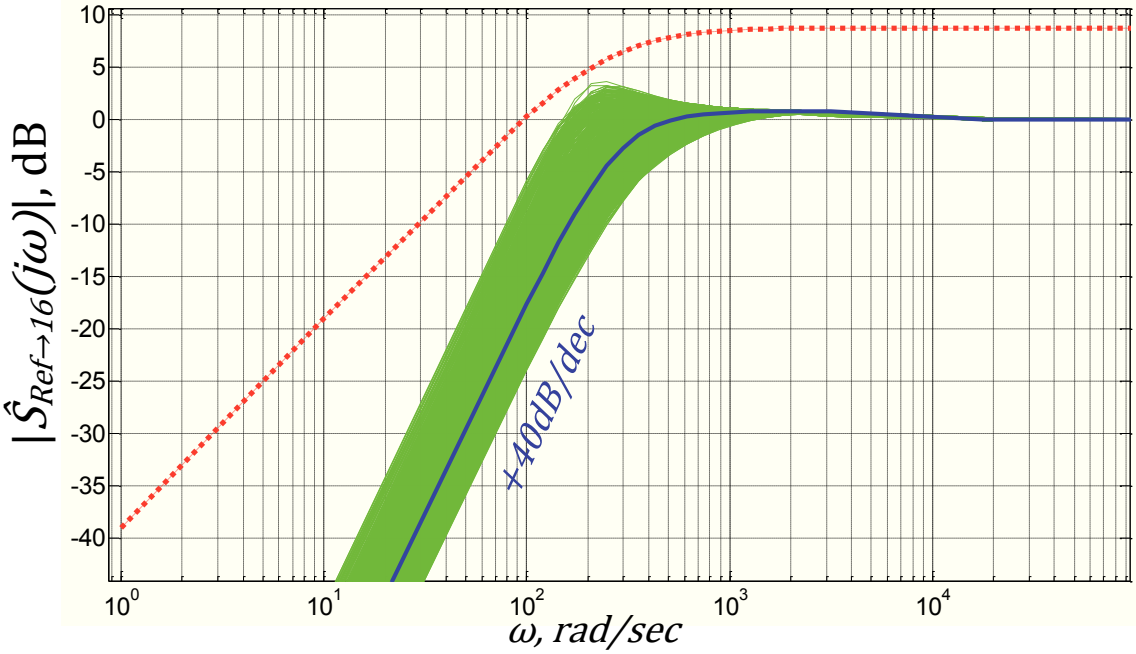


Figure 5.35: La fonction de transfert de synchronisation pire-cas $\widehat{S}_{Ref\to 16}$ incertaine pour différentes valeurs d'incertitudes (en vert) ainsi que la fonction de transfert de synchronisation pire-cas $\widehat{S}_{Ref\to 16}$ nominale (en bleue) et la contrainte fréquentielle (en rouge pointillé)

Par la suite, on réalise des simulations du réseau complet pour les Modèles de synthèse nominaux, les Modèles d'analyse incertains ainsi que les Modèles de simulation non-linéaires.

5.7 Simulation du réseau complet de modèles d'ADPLLs

Pour confirmer les résultats obtenus lors de la synthèse et de l'analyse proposées, des simulations temporelles ont été effectuées sur les différents modèles du réseau d'ADPLLs. La Figure 5.36 montre l'évolution des signaux d'erreur de suivi ε_{Refi} et de commande u_i de toutes les ADPLLs au cours du temps. Ces signaux sont issus de simulations du réseau des Modèles de synthèse (voir la Figure 5.3, page 187) sans bruits extérieurs c'est-à-dire des modèles simple LTIs de l'ADPLL (voir la page 33 pour plus de détails) sans la contribution de signaux du bruit incorporés dans les signaux q_{PDij}, q_{VCOi}, q^b_{VCOi}.

La simulation a été partagée en trois parties :
– durant la partie I, seul le signal échelon unitaire est appliqué à l'entrée du réseau d'ADPLLs (φ_{Ref}), ce qui permet d'évaluer le temps de montée et donc le temps de réponse [6] ;

6. Le temps de montée est, par ailleurs, inférieur au temps de montée indicatif de la fonction de transfert canonique de deuxième ordre $T = \frac{\omega_0}{s^2+2\zeta\omega_0 s+\omega_0^2}$ $(7.1 \cdot 10^{-3}\ sec)$.

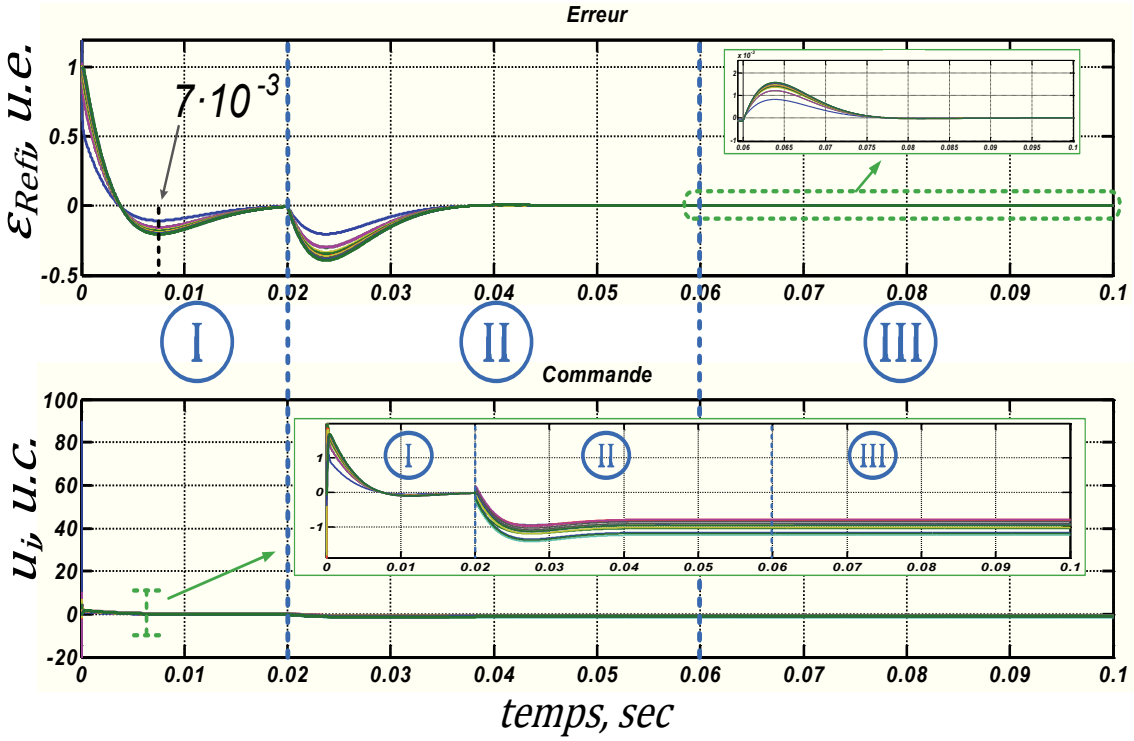


Figure 5.36: Simulations temporelles du réseau des Modèles de synthèse de l'ADPLL sans le bruit extérieur, l'évolution de l'erreur de suivi ε_{Refi} de toutes les ADPLLs (première sous-figure) et de commandes de leurs correcteurs u_i (deuxième sous-figure) au cours du temps.

– durant la partie II, les perturbations q_{VCOi} sous forme d'échelon avec la valeur d'amplitude choisie aléatoirement entre -1 et $+1$, sont appliquées à l'entrée du procédé de chaque ADPLL pour confirmer leur rejet ;
– durant la partie III, une rampe unitaire est appliquée à l'entrée du réseau d'ADPLLs (φ_{Ref}) pour confirmer la synchronisation du réseau avec une source périodique extérieure.

Les mêmes simulations sur le Modèle de synthèse mais en présence de bruit sont représentées sur la Figure 5.37.

Pour confirmer que les hypothèses prises durant la modélisation sont valables autour du point de fonctionnement et pour valider l'efficacité de la loi de commande obtenue sur le procédé réelle, des simulations sur le réseau de Modèles de simulation incluant les non-linéarités, l'effet de quantification, des incertitudes (voir la section 2.2 du chapitre 2, page 43) ont été réalisées. Les résultats de simulation sont représentés sur les Figures 5.38 et 5.39. La Figure 5.38 présente l'évolution de la fréquence instantanée f_i de toutes les ADPLLs et des signaux d'erreur à la sortie de quelques comparateurs de phase choisis Err_{ij} (courbes bleues) au cours du temps pendant deux étapes. La première étape illustre la synchronisation du réseau en mode unidirectionnel de propagation de l'information de phase (voir la Figure 5.8 de la section 5.3 de ce chapitre) dans le cas où les fréquences instantanées initiales des toutes les ADPLLs sont différentes, choisies de manière aléatoire parmi des valeurs définies dans le Tableau 2.1, page 55. La deuxième étape, après un changement

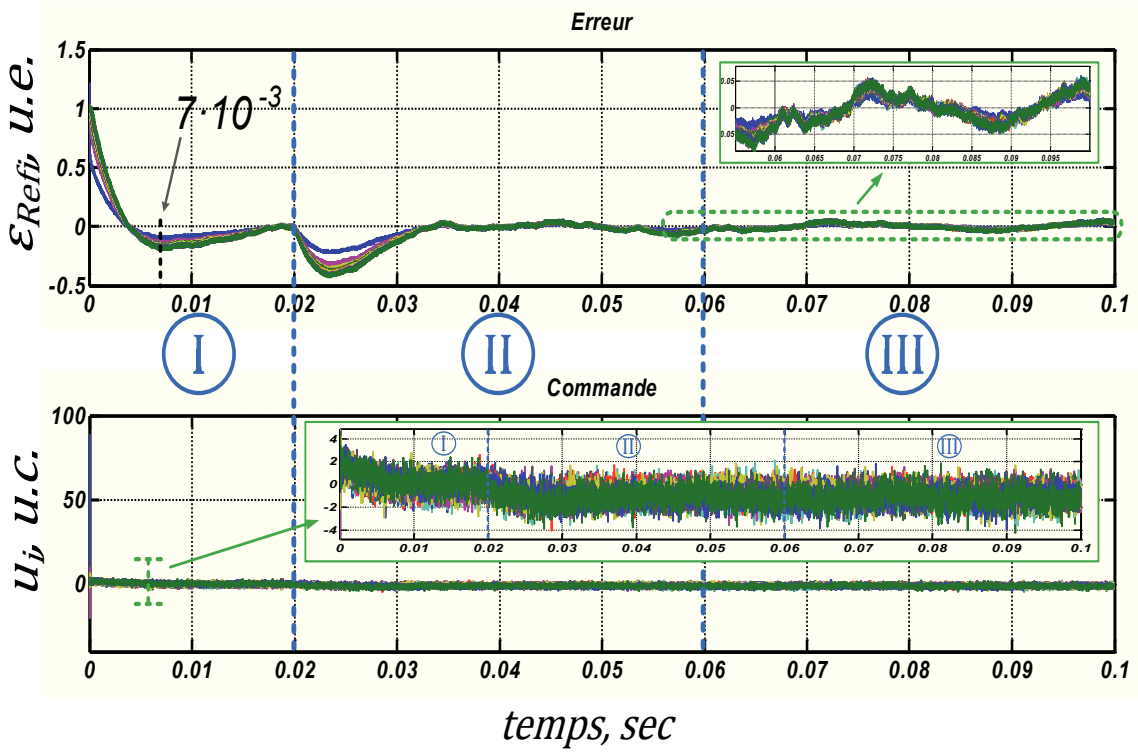


Figure 5.37: Simulations temporelles du réseau des Modèles de synthèse de l'ADPLL avec le bruit extérieur, l'évolution de l'erreur de suivi ε_{Refi} de toutes les ADPLLs (première sous-figure) et de commande de leurs correcteurs u_i (deuxième sous-figure) au cours du temps.

de mode de propagation de l'information de phase d'unidirectionnel à bidirectionnel (voir la Figure 5.1a, page 185) ce qui peut être considérée comme une perturbation du réseau, illustre le maintien de la synchronisation. Le premier signal d'erreur correspond à l'erreur entre la référence ref et le signal de sortie de fréquence divisée de la première ADPLL div_1. La courbe verte sur la première sous-figure correspond à la fréquence du signal de référence Ref appliqué à l'entrée du réseau PLL. Les fluctuations en régime permanent sont dûes à la nature « bang-bang » du comparateur de phase utilisé (voir la sous-section 2.1.1, page 36) et aux différentes sources de bruit (bruit de phase du VCO, les effets de quantification, *etc.*).

La Figure 5.39 présente l'évolution au cours du temps du signal d'horloge appliqué à l'entrée du réseau Ref et des signaux générés par les ADPLLs 1, 4, et 16 respectivement Div_i [7]. On remarque que les signaux sont synchronisés (avec une certaine erreur) en phase et en fréquence après une période transitoire. Les mêmes observations sont confirmées pour les signaux d'horloge générés par les autres ADPLLs et on peut donc conclure que la loi de commande est efficace aussi pour le réseau de Modèles de simulation le plus proche du procédé réel.

7. Il s'agit bien du signal à fréquence divisée Div_i issus du diviseur de fréquence, le signal généré par le VCO i du haute fréquence Loc_i est cependant synchronisé avec le signal à fréquence divisée Div_i.

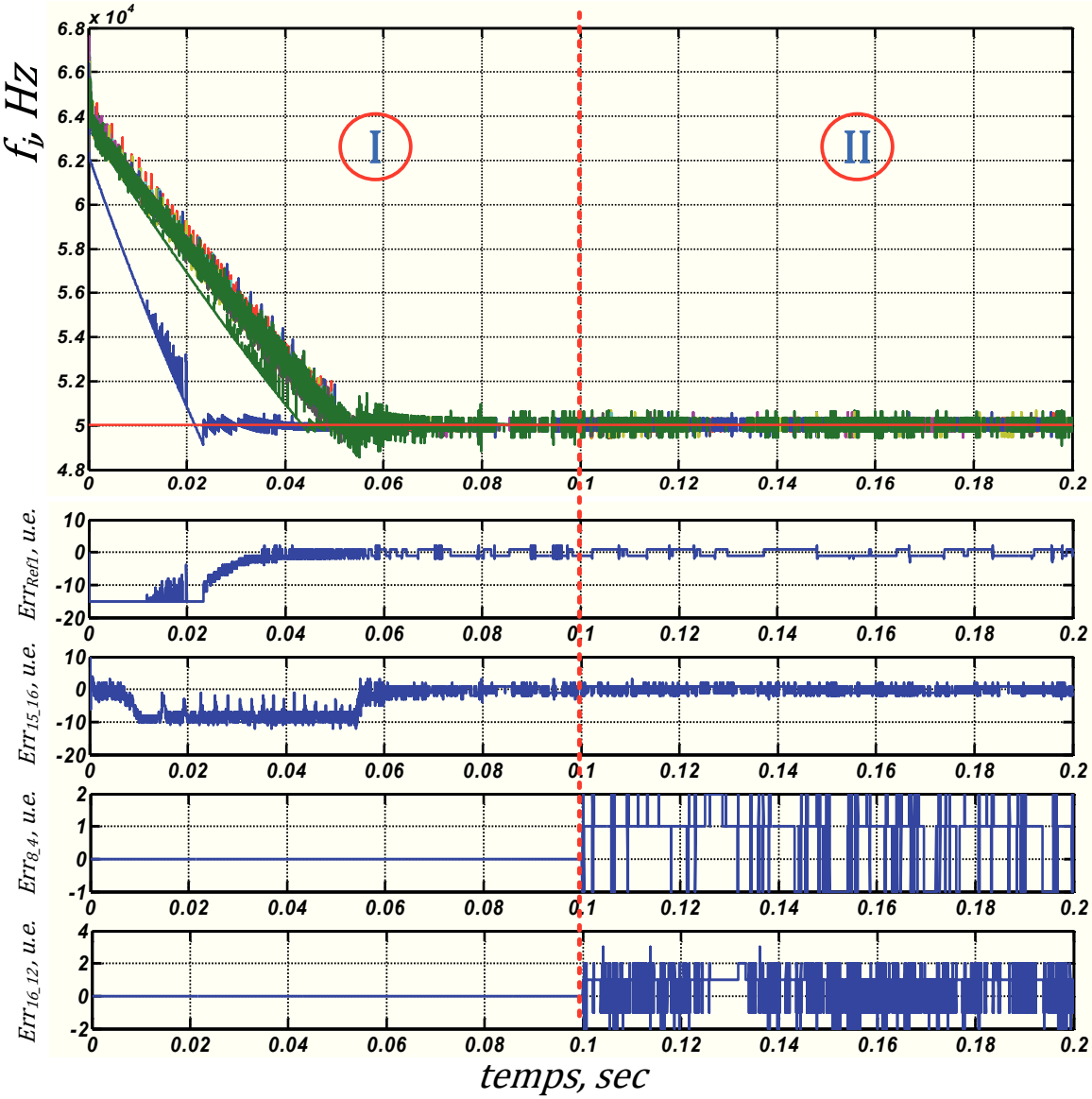


Figure 5.38: Simulations temporelles du réseau de Modèles de simulation d'ADPLLs, l'évolution de la fréquence instantanée de toutes les ADPLLs f_i et de l'erreur de suivi ε au cours du temps.

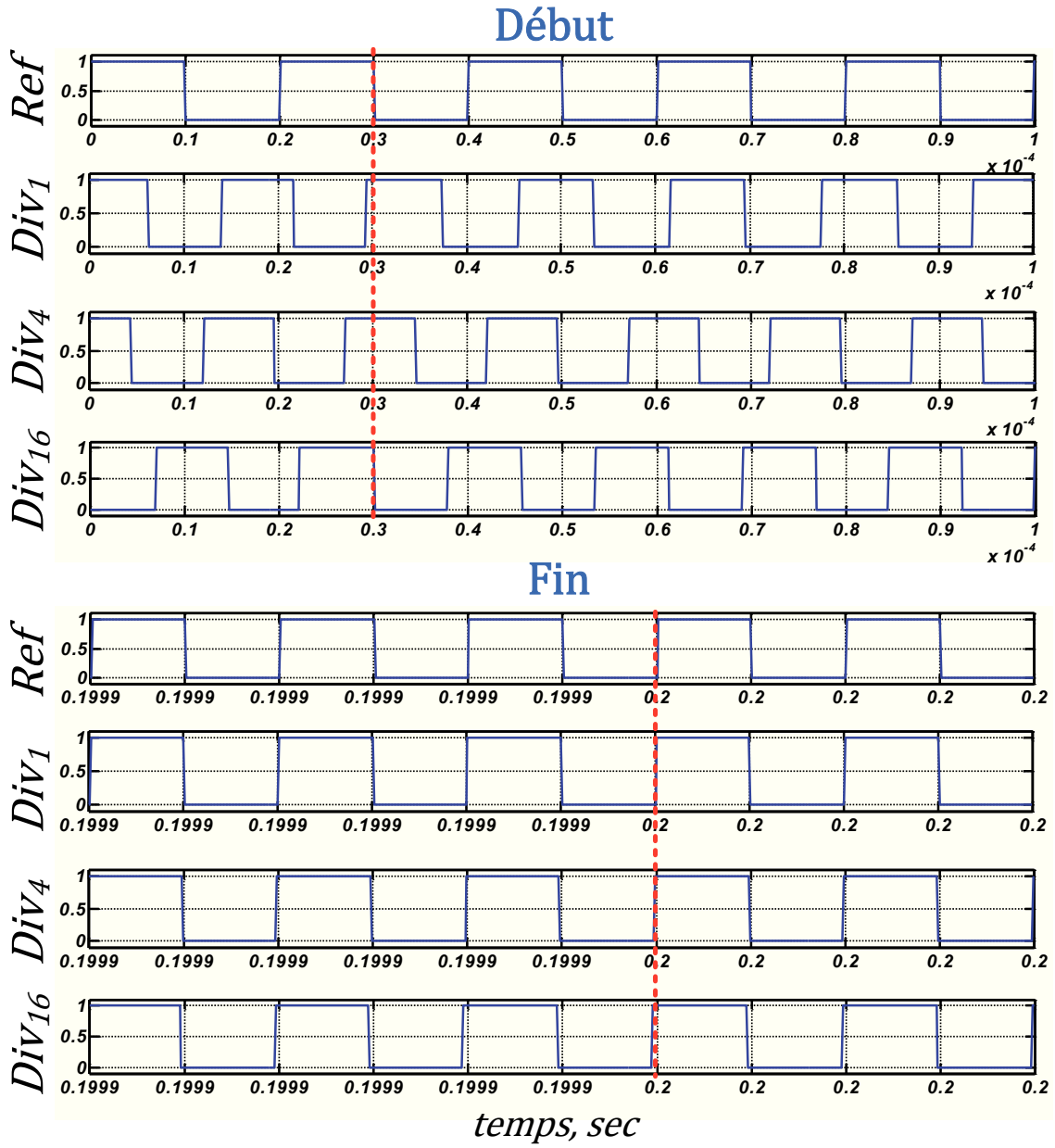


Figure 5.39: Simulations temporelles du réseau de Modèles de simulation de l'ADPLLs, l'évolution au cours du temps des signaux d'horloge appliqués à l'entrée du réseau d'ADPLLs Ref et les signaux générés par l'ADPLL 1, 4 et 16 (Div_1, Div_4, Div_{16}) respectivement

Conclusion

Dans ce chapitre, nous avons appliqué la méthode de synthèse de la loi de commande décentralisée développée dans le chapitre 4 aux Modèles de synthèse de réseaux de distribution du signal d'horloge présentant un réseau d'ADPLLs identiques interconnectées. Ces exemples numériques démontrent l'efficacité de la méthode garantissant a priori la stabilité du système complet ainsi que sa performance.

Deux autres problématiques pratiques d'implémentation : le phénomène de Mode-Locking et le choix de l'emplacement de connexion du signal de référence au réseau d'ADPLLs pour une meilleure performance ainsi que leurs solutions ont été également étudiées dans ce chapitre.

Étant donné que la méthode de synthèse de la loi de commande se base sur l'identité de sous-systèmes, ce qui n'est jamais le cas en pratique à cause de dispersions technologiques inévitables, une méthode d'analyse a été appliquée afin de tester si les propriétés assurées dans le cas nominal le sont également pour les Modèles d'analyse incertains et sinon qu'elle est la dégradation maximale. Cette méthode d'analyse est composée de deux étapes : *(i)* les simulations Monte-Carlo de sous-blocs élémentaires du système et *(ii)* la μ-analyse du système global décrit par des modèles d'incertitudes simples. La première étape permet de déduire les modèles simples d'incertitudes de sous-blocs élémentaires décrits par les modèles complexes de bas niveau, la deuxième étape à son tour avec la connaissance de ces modèles simples d'incertitudes permet d'effectuer une analyse du comportement du système global de grande dimension. L'application de la deuxième étape a été illustrée sur le réseau de distribution du signal d'horloge considéré dans cette étude.

A la fin de ce chapitre, les simulations ont démontré l'efficacité de la loi de commande pour le modèle du réseau d'ADPLLs le plus proche de la réalité : le Modèle de simulation.

Conclusion générale et perspectives

Au cours de cette thèse, nous nous sommes intéressés à la conception du réseau actif de génération et de distribution du signal d'horloge, un système alternatif aux arbres traditionnels de génération et de distribution du signal d'horloge, permettant d'élargir et de surmonter les limites technologiques de ces derniers. Cet exemple numérique motivant, provenant du domaine de la Microélectronique, lance plusieurs défis intéressants et complexes à la fois. En effet, étant un système complexe avec un comportement dynamique riche, il nécessite pour sa conception un développement et l'application de méthodes et d'approches pluridisciplinaires sortant du cadre traditionnel de la Microélectronique. Des méthodes et des approches provenant d'autres sciences comme les Mathématiques, la Théorie et la Modélisation de systèmes électroniques non-linéaires, analogiques et mixtes, et l'Automatique sont nécessaires pour développer une méthode de conception du système considéré dans ces études. L'objectif principal de cette thèse était la réalisation de cette démarche.

Du point de vue de l'Automatique, la conception du réseau de génération et de distribution du signal d'horloge consiste en la synthèse de la loi de commande garantissant les propriétés désirées du système, c'est-à-dire la stabilité et un niveau de performance acceptable, et en une analyse du comportement du système afin de confirmer que ces propriétés sont conservées en dépit de l'influence de facteurs qui n'ont pas été pris en compte durant la synthèse. Nous nous sommes fixés comme objectif de développer une méthode de synthèse et d'analyse d'un tel système qui remplisse les deux exigences principales : *(i)* la méthode doit être compatible avec des méthodes et des outils existant en Microélectronique afin de garantir la continuité de la démarche de la conception et *(ii)* la méthode doit être la plus générale possible afin qu'elle puisse être appliquée aux autres applications issues d'autres problématiques, d'autres domaines de sciences de l'Ingénieur.

A l'issue de ces trois ans d'étude, nous pouvons dire que l'objectif a été atteint et que les deux exigences ont été respectées. En effet, nous avons développé une méthode nouvelle et originale de la synthèse de la loi de commande décentralisée pour un système de grande dimension composé de sous-systèmes identiques répondant bien aux exigences considérées simultanément.

1) Cette méthode de synthèse est inspirée par la même démarche de conception « locale » utilisée jusqu'à maintenant en Microélectronique. Cependant, contrairement à l'approche « locale », la méthode qui a été proposée dans cette thèse grâce aux conditions supplémentaires sur les sous-systèmes locaux et sur l'interconnexion, permet de réaliser un lien « global - local » et d'assurer a priori lors de la conception que les propriétés désirées du système entier et non seulement des sous-systèmes locaux séparés seront garanties. Par la suite, des méthodes de synthèse de la loi de commande locale H_∞ et de l'optimisation convexe sous contraintes LMI ont été

proposées afin de satisfaire les premières et respectivement deuxièmes conditions et donc d'assurer un bon comportement du système global. La synthèse de la loi de commande proposée H_∞ est particulièrement bien adaptée aux approches et aux méthodes existantes de conception d'une ADPLL, un sous-système du réseau actif considéré, en Microélectronique. Premièrement, la méthode permet de « traduire » des exigences classiques de performance sous forme d'un critère mathématique dans le domaine fréquentiel, c'est-à-dire dans un « langage » commun avec des concepteurs d'équipements radio-fréquentiels, tel qu'une ADPLL, en Microélectronique. Deuxièmement, le problème de synthèse H_∞ peut être résolu de manière efficace grâce aux avancées récentes en optimisation convexe et dont l'algorithme de résolution peut être facilement programmé et intégré dans le flux de CAO existant en conception de circuits intégrés de grande densité. Le dernière point concerne également l'optimisation convexe sous contraintes LMI de la méthode proposée afin de garantir les conditions sur l'interconnexion.

2) D'un autre côté, la synthèse H_∞ assurant la performance locale dans le cas de plusieurs entrées et sorties et l'approche de la performance relative proposée réalisant un lien entre la performance locale et globale, permettent d'englober des exigences générales de performance nécessaires pour une généralisation importante du problème. En effet, un réseau actif de génération et de distribution du signal d'horloge peut être modélisé par un système qui est un cas particulier de système de grande dimension composé de sous-systèmes identiques comme considéré dans cette thèse. Aussi, tous les résultats du chapitre 4 ont été développés pour une description générale de sous-systèmes locaux ainsi que leur interconnexion avec des contraintes fréquentielles de la performance générale. Ceci permet une application facile de la méthode de synthèse proposée aux autres problèmes couvrant un large spectre d'applications des systèmes de grande dimension composés de sous-systèmes identiques.

Concernant l'analyse de la robustesse, une combinaison de la méthode d'analyse de la robustesse par les simulations Monte-Carlo avec la méthode de μ-analyse est proposée dans cette thèse. Ainsi, l'analyse du comportement du réseau de génération et de distribution du signal d'horloge vis-à-vis des dispersions technologiques est partagée en deux étapes. La première étape consiste en des simulations Monte-Carlo de sous-blocs élémentaires du système permettant de déduire les modèles simples d'incertitudes de sous-blocs élémentaires décrits par les modèles complexes de bas niveau. La deuxième étape consiste en la μ-analyse du système global décrit par des modèles d'incertitudes simples obtenus à la première étape, ce qui permet d'effectuer une analyse du comportement du système global de grande dimension sans consommation importante du temps et de calcul nécessaires.

Perspectives

Le travail effectué durant la thèse a donné lieu à plusieurs perspectives intéressantes.

1) La contribution principale de la thèse, c'est-à-dire une méthode de synthèse de loi de commande décentralisée est basée sur le fait que les sous-systèmes du système de grande dimension considéré sont identiques. Cela a permis non seulement de simplifier la synthèse en la transformant en une synthèse de la loi de commande locale

et en s'approchant de la méthode de conception « locale » existant en Microélectronique, mais aussi de réduire le conservatisme potentiel du résultat inévitable lors de l'utilisation des outils de la commande décentralisée. La μ-analyse a été proposée par la suite afin de confirmer *a posteriori* que la loi de commande synthétisée par la méthode proposée est efficace en condition de dispersions technologiques impliquant en pratique une non-identité des sous-systèmes. Une des perspectives est ainsi la généralisation du résultat aux sous-systèmes non-identiques dont la différence est plus importante que les dispersions technologiques. Cela va garantir, en outre, l'efficacité de la loi de commande pour ce types de systèmes a priori lors de sa synthèse.

Les auteurs de résultats issus de la Commande Décentralisée, qui ont fondé la base de la méthode de synthèse proposée dans cette thèse, appliquent la méthode de synthèse directement aux systèmes de grande dimension généraux composés de sous-systèmes différents. Cela implique une certaine complexité de calcul (plusieurs conditions à satisfaire) et un certain conservatisme du résultat obtenu.

Dans notre perspective, nous proposons de procéder autrement et de passer, si c'est possible, par une étape intermédiaire de sous-systèmes identiques. En effet, supposons que le système de grande dimension soit composé de sous-systèmes différents G_i interconnectés et contrôlés par des correcteur différents K_i qu'il faut déterminer afin de garantir un bon comportement du système global. Si l'on a des connaissances sur les modèles G_i, il est possible de trouver les pré-compensateurs W_i^e et W_i^s tels que leurs produits avec les modèles de sous-systèmes G_i soient identiques, c'est-à-dire $\widetilde{G} = W_i^s G_i W_i^e$ pour $\forall i$. Dans ce cas-là, nous obtenons un système de grande dimension transformé avec des sous-systèmes identiques $\widetilde{G}$. Il est donc possible d'appliquer la méthode de synthèse de la loi de commande décentralisée $\widetilde{K}$ identique à tous les sous-systèmes comme proposée dans cette thèse. Par la suite, les correcteurs différents finaux obtenus sont déterminés comme $K_i = W_i^e \widetilde{K} W_i^s$.

La première perspective est alors une étude plus détaillée dans cette direction afin de découvrir les conditions sous lesquelles cette démarche sera possible et ses résultats potentiels.

2) La deuxième perspective concerne le choix de la topologie de l'interconnexion. En effet, nous avons vu dans le chapitre 5 que le réseau d'interconnexion joue un rôle important en dynamique globale du système de grande dimension fixant parfois des limites contraignantes de performance atteignable. Il est donc intéressant, pour les applications qui le permettent, d'étudier comment choisir la topologie d'interconnexion afin de dépasser ces limites de performance. Un exemple d'une telle démarche a été déjà illustré dans cette thèse dans la section 5.5 du chapitre 5 lors du choix de l'emplacement de la connexion du signal de référence au réseau.

3) Une autre perspective concerne la résolution des problèmes d'optimisation décrits dans la section 4.4 du chapitre 4 du choix de la caractérisation de dissipativité des sous-systèmes. Ces problèmes sont en général non-convexes et dans cette thèse nous les avons transformés en des problèmes convexes ou quasi-convexes en fixant certaines variables de décision. Il est possible en perspective d'appliquer différentes méthodes d'optimisation non-convexes dont les points d'initialisation sont fixés par les valeurs obtenues lors de la méthode d'optimisation proposée dans cette thèse.

Il est également possible à travers cette extension de prendre en compte le cas de

sous-systèmes MIMO et donc des matrices de caractérisation de la dissipativité X, Y et Z.

4) La quatrième perspective consiste en l'analyse du comportement du modèle du système global prenant en compte les non-linéarités, un comportement hybride et/ou les délais existant dans les systèmes réels. Il est possible d'utiliser l'information sur l'identité de sous-systèmes afin de réduire la complexité et le conservatisme des méthodes d'analyse non-linéaires existantes. En outre, le résultat de séparation des graphes ainsi que la notion de norme H_∞ placés au cœur de la méthode proposée dans cette thèse, sont généralisables au cas des systèmes non-linéaires à travers la séparation des graphes d'opérateurs non-linéaires et de leurs normes $\mathcal{L}_2$.

Il est également intéressant, de combiner la méthode de μ-analyse avec des méthodes probabilistes d'analyse afin de rapprocher encore plus cette méthode de la méthode d'analyse par les simulations Monte-Carlo.

Liste des symboles

ADPLL	All Digital Phase Locked Loop ou boucle à verrouillage de phase entièrement numérique
BMI	Inégalités matricielles bilinéaires ou Bilinear Matrix Inequalities en anglais
CAN	Conversion Analogique Numérique
CAO	Conception Assistée par Ordinateur
CNA	Convertisseur Numérique Analogique
DCO	Digitally Controlled Oscillator en anglais ou oscillateur contrôlé par un mot numérique en français
DSP	Densité Spectrale de Puissance
FPGA	Field-Programmable Gate Array
GALS	Globalement Asynchrone et Localement Synchrone
KYP	Kalman-Yakubovich-Popov
LFT	Transformation fractionnelle (linéaire) inférieure ou Lower (linear) Fractional Transformation en anglais
LMI	Inégalité Matricielle Linéaire ou Linear Matrix Inequality en anglais
LTI	Linéaire Temps Invariant ou Linear Time Invariant en anglais. Il s'agit souvant d'un système dynamique.
MIMO	Multi-Input Multi-Output
PD	Phase Detector en anglais ou comparateur de phase en français
PFD	Phase Frequency Detector en anglais ou convertisseur temps numérique en frainçais
PLL	Phase Locked Loop ou boucle à verrouillage de phase
SISO	Système monovariable, Single-Input Single-Output en anglais
TDC	Time to Digital Converter en anglais ou convertisseur temps numérique en frainçais
UFT	Transformation fractionnelle (linéaire) supérieure ou Upper (linear) Fractional Transformation en anglais
VCO	Voltage controlled oscillator en anglais ou l'oscillateur contrôlé en tension

SOC	Systems On Chip en anglais ou système sur puce en français
$\|S\|_\infty$	Norme H_∞ d'une matrice de fonctions de transfert S
0	Matrice nulle de dimension définie par le contexte
0_n	Matrice nulle de dimension n
$\lambda(A)$	Valeur propre de la matrice A
$\mathbb{C}$	Ensemble des nombres complexes
$\mathbb{R}$	Ensemble des nombres réels
$\mathbf{1}$	Vecteur colonne composé de 1 et de dimension définie par le contexte
$\mathbf{1}_N$	Vecteur colonne de dimension N composé de 1
$\mu_{\underline{\Delta}}(A)$	Valeur singulière structurée de la matrice A par rapport à la structure $\underline{\Delta}$
$\star$	Produit Redheffer ou produit étoile (star product en anglais)
Im	Partie imaginaire d'un nombre complexe
Re	Partie réelle d'un nombre complexe
I	Matrice identité de dimension définie par le contexte
I_n	Matrice identité de dimension n
j	$\sqrt{-1}$
M^*	Transposée conjuguée de la matrice M
M^T	Transposée de la matrice M
s	Variable Laplace
$\overline{\sigma}(A)$	la valeur singulière maximale de la matrice complexe A
$\underline{\sigma}(A)$	la valeur singulière minimale de la matrice complexe A

Bibliographie

[AJGC10] J. M. Akré, J. Juillard, D. Galayko, and E. Colinet. Synchronized state in networks of digital phase-locked loops. In *8th IEEE International NEWCAS Conference*, pages 89–92, Montreal, Jun. 2010.

[AJGC11] J. M. Akré, J. Juillard, D. Galayko, and E. Colinet. Synchronization analysis of networks of self-sampled all-digital phase-locked loops. *IEEE Trans. on Circuits ans Systems II*, 2011. To be appear.

[AJJ+11] J. M. Akré, J. Juillard, M. Javidan, E. Zianbetov, D. Galayko, A. Korniienko, and E. Colinet. A design approach for networks of self-sampled all-digital phase-locked loops. In *European Conference on Circuit Theory and Design*, pages 725–728, Linköping, Aug. 2011.

[AJO+10] J. M. Akré, J. Juillard, S. Olaru, D. Galayko, and E. Colinet. Determination of the behaviour of self-sampled digital phase-locked loops. In *53rd IEEE International Midwest Symposium on Circuits and Systems*, pages 1089–1092, Seattle, Aug. 2010.

[Anc03] F. Anceau. Une technique de réduction de la puissance dissipée par l'horlogerie des circuits complexes rapides. In *4ème journées francophones d'étude de Faible Tension Faible Consommation (FTFC'2003)*, Paris, May 2003.

[APK+01] C.J. Anderson, J. Petrovick, J.M. Keaty, J. Warnock, G. Nussbaum, J.M. Tendier, C. Carter, S. Chu, J. Clabes, J. DiLullo, P. Dudley, P. Harvey, B. Krauter, J. LeBlanc, Pong-Fei Lu, B. McCredie, G. Plum, P.J. Restle, S. Runyon, M. Scheuermann, S. Schmidt, J. Wagoner, R. Weiss, S. Weitzel, and B. Zoric. Physical design of a fourth-generation power ghz microprocessor. In *Solid-State Circuits Conference, 2001. Digest of Technical Papers. ISSCC. 2001 IEEE International*, pages 232 –233, 451, San Francisco, 2001.

[Arc07] M. Arcak. Passivity as a design tool for group coordination. *Automatic Control, IEEE Transactions on*, 52(8) : 1380–1390, Aug. 2007.

[BAW08] H. Bai, M. Arcak, and J. T. Wen. Adaptive design for reference velocity recovery in motion coordination. *Systems and Control Letters*, 57(8) : 602–610, Aug. 2008.

[BB98] D.W. Bailey and B.J. Benschneider. Clocking design and analysis for a 600-mhz alpha microprocessor. *Solid-State Circuits, IEEE Journal of*, 33(11) : 1627 –1633, nov 1998.

[BEFB94] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. Studies in Applied Mathematics. SIAM, Philadelphia, PA, June 1994.

[Bel32] H. de Bellescize. La réception synchrone. *L'Onde électrique*, 11 : 230–240, 1932.

[BG93] V. Blondel and M. Gevers. Simultaneous stabilizability of three linear systems is rationally undecidable. *Mathematics of Control, Signals, and Systems (MCSS)*, 6 : 135–145, 1993.

[Big94] Norman Biggs. *Algebraic Graph Theory*. Cambridge mathematical library. Cambridge University Press, second edition, February 1994.

[Bre78] J.W. Brewer. Kronecker products and matrix calculus in system theory. *IEEE Transactions on Circuits and Systems*, CAS-25(9) : 772–780, 1978.

[BTN01] A. Ben-Tal and A. Nemirovski. *Lectures on Modern Convex Optimization : Analysis, Algorithms and Engineering Applications*. SIAM, 2001.

[Buc38] J. B. Buck. Synchronous rhythmic flashing of fireflies. *The Quaterly Review of Biology*, 13(3) : 301–314, 1938.

[Buc88] J. B. Buck. Synchronous rhythmic flashing of fireflies ii. *The Quaterly Review of Biology*, 63(3) : 265–289, 1988.

[Cad09] Cadence Design System, Inc. *EncounterTM User Guide*, product version 8.1.2 edition, June 2009.

[CGK11] E. Colinet, D. Galayko, and A. Korniienko. Device for generating clock signals for asymmetric comparison of phase, May 2011.

[CS05] N. Chopra and M. W. Spong. On synchronization of networked passive systems with time delays and application to bilateral teleoperation. In *Proceedings of the SICE Annual Conference*, pages 3424–3429, Okayama, Jan. 2005.

[CS06] N. Chopra and M. W. Spong. Output synchronization of nonlinear systems with time delay in communication. In *Proceedings of the 48th IEEE Conference on Decision and Control (CDC)*, pages 4986–4992, San Diego, Dec. 2006.

[CS07] N. Chopra and M. W. Spong. Delay-independent stability for interconnected nonlinear systems with finite l_2 gain. In *Proceedings of the 46th IEEE Conference on Decision and Control (CDC)*, pages 3847–3852, New Orleans, 2007.

[CZRC11] S. Chuan, E. Zianbetov, R. Romain, and E. Colinet. Fpga implementation of reconfigurable adpll network for distributed clock generation. In *International Conference on Field Programmable Technology*, New Delhi, Dec. 2011. To be appear.

[DF99] G. Duc and S. Font. *Commande H_∞ et μ-analyse : des outils pour la robustesse*. Hermès, 1999.

[DGKF89] J. Doyle, K. Glover, P. P. Khargonekar, and B. A. Francis. State-space solutions to standard $\mathbf{H}_2$ and $\mathbf{H}_\infty$ control problems. *IEEE Trans. Aut. Control*, 34(8) : 831–847, August 1989.

[Duc03] G. Duc. *Commande Robuste Multivariable*. polycopié 003317/..2003 Ecole Supérieure d'Electricité, 2003.

[Eul36] L. Euler. Solutio problematis ad geometriam situs pertinentis. *Commenrarii Academiae Scientiarum Imperialis Petropolitanae*, 8 : 128–140, 1736.

[FM04] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicle formations. *Automatic Control, IEEE Transactions on*, 49(9) : 1465 – 1476, 2004.

[Fon95] S. Font. *Méthodologie pour prendre en compte la robustesse des systèmes asservis : optimisation H_∞ et approche symbolique de la forme standard*. PhD thesis, Université Paris XI Orsay, France, 1995.

[FPE86] G. F. Franklin, J. D. Powell, and A. Emami-Naeni. *Feedback Control of Dynamic Systems*. Addison-Wesley, 1986.

[Fri01] Eby G. Friedman. Clock distribution networks in synchronous digital integrated circuits. In *Proc. IEEE*, volume 89, pages 665–692, May 2001.

[GA94] P. Gahinet and P. Apkarian. A linear matrix inequality approach to H_∞ control. *International Journal of Robust and Nonlinear Control*, 4 : 421–448, 1994.

[GC00] V. Gutnik and A.P. Chandrakasan. Active ghz clock network using distributed plls. *Solid-State Circuits, IEEE Journal of*, 35(11) : 1553 –1560, November 2000.

[GHLM09] P.R. Gray, P.J. Hurst, S.H. Lewis, and R.G. Meyer. *Analysis and Design of Analog Integrated Circuits*. John Wiley & Sons, Inc., 4 edition, 2009.

[GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability : A guide to the theory of NP-Completeness*. W. H. Freeman, 1979.

[GNLC95] P. Gahinet, A. Nemirovsky, A. L. Laub, and M. Chilali. *LMI Control Toolbox*. The Mathworks Inc., 1995.

[Han83] H. Hanson. Comparative studies of firefly pacemakers. *Federation Proc.*, 37 : 2158–2164, 1983.

[HAW07] B. He, M. Arcak, and J. T. Wen. Group coordination when the reference velocity is available only to the leader : An adaptive design. In *American Control Conference (ACC)*, pages 5400–5405, New York, Jul. 2007.

[HJ85] R. A. Horn and C. A. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[HLCF98] Hong-Yean Hsieh, Wentai Liu, M. Clements, and P. Franzon. Self-calibrating clock distribution with scheduled skews. In *Circuits and Systems, 1998. ISCAS '98. Proceedings of the 1998 IEEE International Symposium on*, volume 2, pages 470 –473 vol.2, June 1998.

[Hof04] K. Hoffman. *System Integration. From transistor design to large scale integrated circuits*. John Wiley & Sons, 2004.

[Hou07] M. Houdebine. *Contribution pour l'amélioration de la robustesse et du bruit de phase des synthétiseurs de fréquences*. PhD thesis, Institut national polytechnique de Grenoble, 2007.

[HS94] M. Hovd and S. Skogestad. Sequential design of decentralized controllers. *Automatica*, 30(10) : 1601–1607, 1994.

[IZF96] M. Ikeda, G. Zhai, and Y. Fujisaki. Decentralized h_∞ controller design for large-scale systems : a matrix inequality approach using a homotopy method. In *Decision and Control, 1996., Proceedings of the 35th IEEE*, volume 1, pages 1 –6, Kobe, dec 1996.

[JKF07] U. Jönsson, C. . Kao, and H. Fujioka. A popov criterion for networked systems. *Systems and Control Letters*, 56(9-10) : 603–610, Sep.-Oct. 2007.

[Jön01] U. Jönsson. Lecture notes on integral quardatic constraints. Royal Institut of Technology, May 2001.

[JZA+11] M. Javidan, E. Zianbetov, F. Anceau, D. Galayko, A. Korniienko, E. Colinet, G. Scorletti, J. M. Akre, and J. Juillard. All-digital pll array provides reliable distributed clock for socs. In *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on*, pages 2589 –2592, Rio de Janeiro, may 2011.

[KCS+10] A. Korniienko, E. Colinet, G. Scorletti, E. Blanco, D. Galayko, and J. Juillard. A clock network of distributed adplls using an asymmetric comparison strategy. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 3212–3215, Paris, May-Jun. 2010.

[KCSB09] A. Korniienko, E. Colinet, G. Scorletti, and E. Blanco. H_∞ loop shaping control for distributed pll network. In *Proceedings of 2009 IEEE Ph.D. Research in Microelectronics & Electronics (PRIME)*, pages 336–339, Cork, Jul. 2009.

[KCV08] C. Kharrat, E. Colinet, and A. Voda. H_∞ loop shaping control for pll-based mechanical resonance tracking in nems resonant mass sensors. In *Proceedings of 2008 IEEE Sensors*, pages 1135–1138, Lecce, Oct. 2008.

[Kha92] H.K. Khalil. *Nonlinear Systems*. Macmillan, New York, 1992.

[KJF09] C. . Kao, U. Jönsson, and H. Fujioka. Characterization of robust stability of a class of interconnected systems. *Automatica*, 45(1) : 217–224, Jan. 2009.

[Kra98] W. Krauth. Introduction to monte carlo algorithms. In J. Kertész and I. Kondor, editors, *Advances in Computer Simulation*, volume 501 of *Lecture Notes in Physics*, pages 1–35. Springer Berlin / Heidelberg, 1998.

[Kra06] W. Krauth. *Statistical Mechanics : Algorithms and Computations*. Oxford Master Series in Statistical, Computational, and Theoretical Physics. Oxford University Press, Nov. 2006.

[Kro03] V.F. Kroupa. *Phase lock loops and frequency synthesis*. John Wiley & Sons, June 2003.

[Lee66] D.B. Leeson. A simple model of feedback oscillator noise spectrum. *Proc. IEEE*, 54(2) : 329–330, Feb. 1966.

[LJ10] Y. Liu and Y. Jia. H_∞ consensus control of multi-agent systems with switching topology : a dynamic output feedback protocol. *International Journal of Control*, 83(3) : 527–537, 2010.

[LJDY09] Y. Liu, Y. Jia, J. Du, and S. Yuan. Dynamic output feedback control for consensus of multi-agent systems : An h_∞ approach. In *American Control Conference, 2009. ACC '09.*, pages 4470 –4475, St. Louis, june 2009.

[LKR04] Jri L., K.S. Kundert, and B. Razavi. Analysis and modeling of bang-bang clock and data recovery circuits. *JSSC*, 39(9) : 1571–1580, september 2004.

[LM76] E.L. Lasley and A.N. Michel. Input-output stability of interconnected systems. *IEEE Trans. Aut. Control*, pages 84–89, february 1976.

[LML+06] N. Lewis, G. Monnerie, L. Lewis, J. Sabatier, and P. Melchior. Automatic procedure generating noise models for discrete-time applications. In *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, pages 5720 – 5723, May 2006.

[LMS+04] N. Lewis, G. Monnerie, J. Sabatier, P. Melchior, M. Robbe, and H. Levi. Using fractional differentiation for the modeling of 1/f1 noise application to discrete-time noise sources in vhdl-ams. In *Industrial Electronics, 2004 IEEE International Symposium on*, volume 1, pages 205 – 210, May 2004.

[LR04] P.M. Levine and G.W. Roberts. A high-resolution flash time-to-digital converter and calibration scheme. In *Proc. Int. Test Conf. ITC 2004.*, pages 1148 – 1157, Oct. 2004.

[Lun91] J. Lunze. *Feedback Control of Large Scale Systems*. Prentice-Hall, 1991.

[Maq03] D. Maquin. Eléments de théorie des graphes et programmation linéaire. Cours, Institut National Polytechnique de Lorraine, May 2003.

[Mar04] N. I. Margaris. Introduction. In *Theory of the Non-linear Analog Phase Locked Loop*, volume 304 of *Lecture Notes in Control and Information Sciences*, pages 3–11. Springer Berlin / Heidelberg, 2004.

[MH78] P. J. Moylan and D. J. Hill. Stability criteria for large-scale systems. *IEEE Trans. Aut. Control*, AC-23(2) : 143–149, Apr. 1978.

[MI98] H. Mizuno and K. Ishibashi. A noise-immune ghz-clock distribution scheme using synchronous distributed oscillators. In *Solid-State Circuits Conference, 1998. Digest of Technical Papers. 1998 IEEE International*, pages 404 –405, 474, February 1998.

[Mor04] L. Moreau. Stability of continuous-time distributed consensus algorithms. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 4, pages 3998 – 4003, Atlantis, dec. 2004.

[MR97] A. Megretski and A. Rantzer. System analysis via integral quadratic constraints. *IEEE Trans. Aut. Control*, 42(6) : 819–830, June 1997.

[Mur07] R.M. Murray. Recent research in cooperative control of multi-vehicle systems. *Journal of Dynamic Systems, Measurement and Control*, Vol.129, No. 5 : 571–583, 2007.

[Nas01] S.R. Nassif. Modeling and analysis of manufacturing variations. In *Custom Integrated Circuits, 2001, IEEE Conference on.*, pages 223 – 228, San Diego, May 2001.

[NY83] A. Nemirovskii and D. Yudin. *Problem Complexity and Method Efficiency in Optimization*. John Wiley & Sons, 1983.

[OSFM07] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1) : 215 –233, 2007.

[PN92] G. A. Pratt and J. Nguyen. Synchronization of hardware oscillators in a mesh-connected parallel processor, Aug. 1992.

[PN95] Gill A. Pratt and John Nguyen. Distributed synchronous clocking. *IEEE Trans. Parallel Distrib. Syst.*, 6(3) : 314–328, March 1995.

[PW09] A. P. Popov and H. Werner. A robust control approach to formation control. In *European Contr. Conf. 2009*, Budpest, Aug. 2009.

[Ran96] A. Rantzer. On the Kalman-Yakubovich-Popov lemma. *Systems and Control Letters*, 27(5), January 1996.

[Raz96] B. Razavi. *Monolithic phase-locked loops and clock recovery circuits : theory and design*. IEEE Press, Avr. 1996.

[Raz98] B. Razavi. *RF Microelectronics*. Prentice Hall PTR, 1998.

[RBA05] Wei Ren, R.W. Beard, and E.M. Atkins. A survey of consensus problems in multi-agent coordination. In *American Control Conference, 2005. Proceedings of the 2005*, volume 3, pages 1859 – 1864, Portland, june 2005.

[RK08] R.Y. Rubinstein and D.P. Kroese. *Simulation and the Monte Carlo method*, volume 707 of *Wiley series in probability and mathematical statistics. Probability and mathematical statistics*. John Wiley & Sons, 2 edition, 2008.

[RMW+01] P.J. Restle, T.G. McNamara, D.A. Webber, P.J. Camporese, K.F. Eng, K.A. Jenkins, D.H. Allen, M.J. Rohn, M.P. Quaranta, D.W. Boerstler, C.J. Alpert, C.A. Carter, R.N. Bailey, J.G. Petrovic, B.L. Krauter, and B.D. McCredie. A clock distribution network for microprocessors. *IEEE Journal of Solid-State Circuits*, 36(5) : 792 – 799, May 2001.

[Roh97] U.L. Rohde. *Microwave and wireless synthesizers : theory and design*. A Wiley Interscience publication. John Wiley & Sons, December 1997.

[RTE+07] A. V. Rylyakov, J. A. Tierno, G. J. English, D. Friedman, and M. Meghelli. A wide power-supply range (0.5v-to-1.3v) wide tuning range (500 mhz-to-8 ghz) all-static cmos ad pll in 65nm soi. In *Proceedings of ISSCC*, Feb. 2007.

[Rub10] E. Rubiola. *Phase noise and frequency stability in oscillators*. The Cambridge RF and microwave engineering series. Cambridge University Press, 2010.

[SB06] R.B. Staszewski and P.T. Balsara. *All-Digital Frequency Synthesizer in Deep- Submicron CMOS*. Wiley-Interscience, 2006.

[Sco97] G. Scorletti. *Approche Unifiée de l'analyse et la commande des systèmes par formulation LMI*. PhD thesis, Université d'Orsay, Paris, France, 1997. In French.

[Sco06] G. Scortletti. Outils pour la robustesse : inégalités matricielles affines. Notes de cours, Master-2, Recherche Sciences de l'Automatique et du Traitement du Signal, Universitée d'Orsay SUPELEC, Feb. 2006.

[SD01] G. Scorletti and G. Duc. An LMI approach to decentralized H_∞ control. *Int. J. Control*, 74(3) : 211–224, 2001.

[SF08] G. Scorletti and V. Fromion. Automatique fréquentielle avancée. Cours 3ièeme année cycle ingénieur, Ecole Centrale de Lyon, Master GEGP filière GSA, Université de Lyon 1, INSA de Lyon, 2008.

[SFRY99] R. Senthinathan, S. Fischer, H. Rangchi, and H. Yazdanmehr. A 650-mhz, ia-32 microprocessor with enhanced data streaming for graphics and video. *Solid-State Circuits, IEEE Journal of*, 34(11) : 1454 –1465, November 1999.

[SGC97] C. Scherer, P. Gahinet, and M. Chilali. Multiobjective output-feedback control via LMI optimization. *IEEE Trans. Aut. Control*, 42(7) : 896–911, July 1997.

[SJK97] R. Sepulchre, M. Jankovic, and P. Kokotovic. *Constructive nonlinear control*. Springer Verlag. London, 1997.

[SP05] S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control, Analysis and Design*. John Wiley and Sons Chischester, 2005.

[SS09] L. Scardovi and R. Sepulchre. Synchronization in networks of identical linear systems. *Automatica*, 45(11) : 2557 – 2562, 2009.

[Sta11] R.B. Staszewski. State-of-the-art and future directions of high-performance all-digital frequency synthesis in nanometer cmos. *Circuits and Systems I : Regular Papers, IEEE Transactions on*, 58(7) : 1497 –1510, Jul. 2011.

[Str01] S.H. Strogatz. *Nonlinear Dynamics And Chaos : With applications to physics, biology, chemistry, and engineering*. Studies in Nonlinearity. Westview Press, Jan. 2001.

[SVAS78] N. R. Sandell, P. Varaiya, M. Athans, and M. G. Safonov. Survey of decentralized control methods for large scale systems. *IEEE Trans. Aut. Control*, 23(2) : 108–128, 1978.

[SY98] H. Sutoh and K. Yamakoshi. A clock distribution technique with an automatic skew compensation circuit. In *Proc. IEICE Tran. on Electronics*, volume E81-C, pages 277–283, Feb. 1998.

[TM09] S. Tonetti and R.M. Murray. Limits on the network sensitivity function for multi-agent systems on a graph. Technical report, California Institute of Technology, Pasadena, 2009.

[TM10] S. Tonetti and R.M. Murray. Limits on the network sensitivity function for homogeneous multi-agent systems on a graph. In *American Control Conference (ACC)*, pages 3217 –3222, Baltimore, Jul. 2010.

[WA10] P. Wieland and F. Allgöwer. On consensus among identical linear systems using input-decoupled functional observers. In *American Control Conference (ACC)*, pages 1641 –1646, Baltimore, Jul. 2010.

[Wie10] P. Wieland. *From Static to Dynamic Couplings in Consensus and Synchronization among Identical and Non-Identical Systems*. PhD thesis, Institut für Systemtheorie und Regelungstechnik Universität Stuttgart, Sep. 2010.

[Wil72] J. C. Willems. Dissipative dynamical systems, part i : General theory. *Archive for Rational Mechanics and Analysis*, 45 : 321–351, 1972.

[Wil76] J. C. Willems. Mechanisms for the stability and instability in feedback systems. *Proc. IEEE*, 64 : 24–35, 1976.

[Wil97] U. Wilensky. Netlogo fireflies model. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL., 1997.

[WKSA08] P. Wieland, J. . Kim, H. Scheu, and F. Allgöwer. On consensus in multi-agent systems with linear high-order agents. In *Proceedings of the 17th IFAC World Congress*, pages 1541 – 1546, Seoul, Jul. 2008.

[WLH09] J. Wang, Z. Liu, and X. Hu. Consensus of high order linear multi-agent systems using output error feedback. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pages 3685 –3690, Shanghai, dec. 2009.

[Woo01] M. Wooldridge. *An Introduction to MultiAgent Systems*. John Wiley & Sons, Ltd., 2nd edition, 2001.

[Zam81] G. Zames. Feedback and optimal sensitivity : Model reference transformations, multiplicative seminorms, and approximate inverses. *IEEE Trans. Aut. Control*, AC-26(2) : 301–320, April 1981.

[ZIF01] G. Zhai, M. Ikeda, and Y. Fujisaki. Decentralized h_∞ controller design : a matrix inequality approach using a homotopy method. *Automatica*, 37(4) : 565 – 572, 2001.

[ZOIK09a] G. Zhai, S. Okuno, J. Imae, and T. Kobayashi. A matrix inequality based design method for consensus problems in multi-agent systems. *Int. J. Appl. Math. Comput. Sci.*, 19 : 639–646, December 2009.

[ZOIK09b] G. Zhai, S. Okuno, J. Imae, and T. Kobayashi. A new consensus algorithm for multi-agent systems via dynamic output feedback control. In *Control Applications, (CCA) Intelligent Control, (ISIC), 2009 IEEE*, pages 890 – 895, Saint Petersburg, Jul. 2009.

[ZOIK10] G. Zhai, S. Okuno, J. Imae, and T. Kobayashi. Extended consensus algorithm for multi-agent systems [brief paper]. *Control Theory Applications, IET*, 4(10) : 2232 –2238, october 2010.