

## Continuité de service sur Ethernet Industriel

Sylvain Kubler, Eric Rondeau, Jean-Philippe Georges

► **To cite this version:**

Sylvain Kubler, Eric Rondeau, Jean-Philippe Georges. Continuité de service sur Ethernet Industriel. Sixième Conférence Internationale Francophone d'Automatique, CIFA 2010, Jun 2010, Nancy, France. pp.CDROM, 2010. <hal-00489938>

**HAL Id: hal-00489938**

**<https://hal.archives-ouvertes.fr/hal-00489938>**

Submitted on 7 Jun 2010

**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.

# Continuité de service sur Ethernet Industriel

Sylvain KUBLER, Éric RONDEAU, Jean-Philippe GEORGES

Centre de Recherche en Automatique de Nancy,  
UMR 7039 – Nancy-Université, CNRS  
Faculté des Sciences BP 239, 54506 Vandœuvre lès Nancy, France.  
*Prénom.Nom@cran.uhp-nancy.fr*

**Résumé**— Les réseaux Ethernet utilisent le protocole de Spanning-tree (STP) pour reconfigurer leur topologie en cas de rupture d'équipements. Le temps mis par STP pour rétablir une topologie fonctionnelle peut s'avérer inadapté dans le contexte industriel. Ce papier présente une méthode basée sur la duplication de paquets. Cela permet d'augmenter la probabilité de succès concernant l'acheminement d'informations entre différents équipements réseau. Pour cela, les transmissions de trames sont dupliquées via une paire de chemins optimale. La sélection de cette paire est réalisée à l'aide d'une méthode basée sur les algorithmes génétiques. Dans ce papier, des simulations permettent d'analyser et comparer les résultats théoriques.

**Mots-clés**— Systèmes temps-réel, Tolérance aux défaillances, Spanning-tree, Algorithme génétique.

## I. INTRODUCTION

Les architectures de contrôle-commande modernes utilisent des réseaux de communication pour surveiller et agir sur les applications distribuées. Dans le contexte des systèmes embarqués ou industriels, les réseaux à déployer doivent répondre le plus souvent à des contraintes temporelles strictes, mais aussi garantir une continuité de service. Dans les années 1980, beaucoup de réseaux appelés réseaux de terrain ont été développés par des constructeurs d'automates afin de répondre à ces exigences, mais rendant les solutions propriétaires. Le consortium IAONA (Industrial Automation Open Networking Alliance) fondé en 1999, avait comme objectif de faire la promotion d'Ethernet en tant que standard de communication dans les industries. Les intérêts sont qu'Ethernet est indépendant des constructeurs, qu'il offre beaucoup plus de bande passante que les réseaux de terrain et enfin qu'il est largement diffusé dans le monde informatique assurant ainsi sa pérennité dans le temps. La contrepartie est qu'Ethernet utilise un protocole d'accès au médium non déterministe (CSMA/CD). Cependant, l'utilisation d'architectures commutées (IEEE 802.1d) permet d'inhiber les collisions et d'assurer une redondance de chemins.

Dans le domaine des réseaux, la redondance se traduit par la mise en place d'un certain nombre de boucles, posant alors des problèmes de retransmissions infinies, qui congestionnent le réseau. Des protocoles existent et permettent de remédier à ces problèmes, ils consistent à mettre en place des arbres couvrants sur le réseau. Par exemple le protocole Spanning Tree (STP) a pour but d'analyser le câblage du réseau, puis d'inhiber un certain nombre de liens de façon à éliminer les boucles du réseau, tout en assurant l'interconnexion de tous les nœuds. La fonction du protocole ne se limite pas à cela, elle surveille en permanence l'état des liens et des équipements du réseau. Lorsqu'une anomalie se produit (défaillance d'un câble, port, etc.), le protocole STP

cherche à maintenir la connectivité de l'ensemble des nœuds du réseau en réactivant des liens qu'il avait préalablement inhibés. Le protocole RSTP (Rapid-STP, IEEE 802.1w), évolution de STP permet également de réaliser ces fonctions. Pour limiter les problèmes de congestion sur un chemin donné, il est possible en activant des protocoles tels que MSTP (Multiple STP, IEEE 802.1s), ou encore PVST (Per-VLAN STP, protocole propriétaire Cisco), de définir non pas un arbre sur le réseau Ethernet mais un arbre par VLAN (Virtual LAN, 802.1Q). Le but est de pouvoir répartir la charge en utilisant les liens redondants du réseau pour éliminer les goulots d'étranglement. En effet, basiquement les VLAN s'appuient sur l'arbre unique défini par le protocole STP pour acheminer leurs messages, c'est-à-dire que toutes les communications, quel que soit le VLAN, passent par les mêmes chemins pour aller d'un point vers un autre. Cette configuration peut avoir comme incidence d'amplifier ou de générer de la congestion sur certains chemins. La Fig. 1 montre la différence entre une gestion unique et multiple du STP. Sur le graphe Fig. 1(a), un seul arbre est défini pour acheminer tous les messages. Dans ce cas, lorsque le sommet S4 envoie des messages typés VLAN1 vers S1 et le sommet S5 envoie des messages typés VLAN2 vers S3, alors le lien entre S1 et S2 est sollicité à chaque fois. Sur les deux autres graphes Fig. 1(b), le protocole MSTP est activé et propose d'inhiber le lien entre S2 et S3 pour l'arbre de VLAN1 et d'inhiber le lien entre S1 et S3 pour le VLAN2. Cette configuration permet de répartir les charges puisque le lien entre S1 et S2 est utilisé pour transporter le message typé VLAN1, alors que le message typé VLAN2 passe directement via le lien entre S2 et S3.

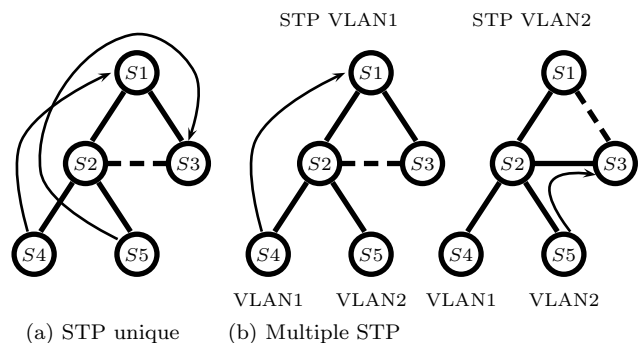


Fig. 1. Configuration STP unique vs Multiple STP

L'objectif de ce papier est de proposer une méthode permettant de minimiser la probabilité qu'une déconnection intervienne entre le(s) contrôleur(s), les capteurs et les actionneurs. La contrainte est d'utiliser uniquement des solu-

tions standardisées sans les modifier. Cet article s'organise de la façon suivante : la section 2 décrit les stratégies de reconfigurations d'arbres couvrants. La section 3 explique la problématique d'optimisation. La section 4 présente la méthode heuristique retenue. Enfin, la section 5 détaille quelques résultats obtenus par simulation via l'outil OP-NET, dans le contexte des Système Contrôlé en Réseau.

## II. ARBRES COUVRANTS

### A. Problème de reconfiguration

Le problème des protocoles STP, ainsi que de son évolution RSTP, concerne les temps de reconfiguration pour proposer de nouveaux arbres. En effet, STP nécessite une trentaine de secondes et RSTP en moyenne cinq secondes, des durées non adaptées aux environnements temps-réel. On constate alors que l'utilisation d'un seul arbre se révèle être une configuration critique, puisque la présence d'un dysfonctionnement d'équipement au sein du réseau a pour conséquence d'interrompre les communications entre les équipements du Système Contrôlé en Réseau (SCR). La société MOXA a mis en place un nouveau mécanisme nommé Turbo-ring afin de résoudre ce problème. Ce dernier assure une reconfiguration aux alentours des 20 ms, mais cette solution n'est pas normalisée, utilise une architecture particulière dite en anneau et peut se révéler trop longue. Ce qui nous pousse à écarter cette solution.

### B. Positionnement

Plusieurs stratégies peuvent être proposées afin d'éviter ou minimiser cette situation. Dans [5], un nouvel algorithme de routage multi-chemins disjoints est défini. Ces travaux se basent sur la théorie des arbres colorés et ont pour but de prendre en compte la défaillance des nœuds, dans le cadre d'un maintien de la communication entre un nœud et un puits. La spécificité de ces travaux concerne la minimisation des échanges sur le réseau. C'est-à-dire qu'il s'agit d'une approche en ligne permettant l'amélioration des performances de reconfigurations. Une autre approche consiste à anticiper de manière hors-ligne la défaillance d'un chemin. Ainsi [7] propose de dupliquer le médium afin de mettre en place une redondance concernant l'acheminement des trames. Cette solution est basée sur une architecture non commutée et présente une limitation en débit. La solution qui est étudiée dans ce papier utilise le protocole MSTP afin de définir plusieurs parcours possibles pour relier différents points du réseau, nous utiliserons alors la notion de chemin lors de notre étude. L'intérêt de cette approche est de définir un niveau de redondance adapté aux contraintes de l'application. On propose que le nombre d'équipements redondants soit fonction d'un niveau de SIL (Safety Integrity Level, IEC 61508) prédéterminé. Aussi, la stratégie vise à mettre en place une procédure passive en envoyant autant de messages que d'arbres définis. Le but est que si un chemin est defectueux, le destinataire reçoit les informations par au moins un des autres chemins. De cette manière, les communications entre équipements réseau seront interrompues uniquement si les chemins sur la totalité des arbres subissent une défaillance simultanée.

## III. FORMALISATION

### A. Introduction

Le but revient à définir une expression formalisant le problème de la redondance de chemins entre les équipements d'un réseau. Le premier critère sera de définir des chemins minimisant le nombre d'équipements réseau. En effet, si on considère que les taux de panne et de réparation sont identiques pour chaque équipement réseau, alors un chemin plus court, c'est-à-dire constitué de moins d'équipements, a une probabilité de défaillance moindre que pour un chemin plus long. A noter que ce critère peut aussi avoir un impact sur les performances temporelles d'acheminement des messages. Le deuxième critère est de pouvoir proposer des chemins redondants contenant un nombre minimum d'équipements réseau en commun. Ce critère est important puisque si deux chemins utilisent un même équipement et que celui-ci tombe en panne, alors les deux chemins ne permettent plus d'aller vers le destinataire. Finalement, la solution globale à retenir est un compromis entre le premier et le second critère.

### B. Analyse de fiabilité pour des chemins indépendants

La fonction objectif de notre étude est définie comme la probabilité que le système (le réseau) soit défaillant. Nous allons analyser la fiabilité du réseau en considérant dans un premier temps des chemins indépendants. Pour cela, définissons  $\alpha_i$  comme étant le nombre de branches composant le chemin,  $\lambda$  le taux de défaillance pour tout équipement réseau et  $\mu = 1 - \lambda$  la probabilité de succès. La probabilité de défaillance concernant un chemin est donnée par (1). La probabilité de défaillance du système, c'est-à-dire que tous les chemins soient défaillants est donnée par (2), avec  $j$  représentant le nombre de chemins ( $j \geq 1$ ).

$$P_{D_1} = 1 - (1 - \lambda)^{\alpha_1} = 1 - \mu^{\alpha_1} \quad (1)$$

$$P_{D_j} = \prod_{i=1}^j 1 - \mu^{\alpha_i} = \prod_{i=1}^j P_{D_i} \quad (2)$$

La probabilité de défaillance est sélectionnée à partir des spécifications SIL de la norme IEC 61508 selon un fonctionnement en mode continu. Le contexte de ce travail est de fiabiliser une architecture Ethernet industriel. Le niveau de SIL dépend des contraintes applicatives. On utilise généralement pour un SCR des niveaux de SIL d'ordres 3 ou 4 (aviation, plate-forme nucléaire, etc.). Nous nous placerons dans le cas où les équipements réseau seront SIL 4 (avec  $\lambda = 10^{-8}$ ). Le but est alors d'évaluer l'architecture globale du réseau en fonction du niveau de SIL retenu. La Fig. 2 présente les probabilités de défaillances estimées par (2) en fonction du nombre d'équipements composant les chemins et ce, pour 3 niveaux de redondance. A noter que pour l'étude, nous avons pris une symétrie parfaite concernant la longueur des différents chemins. La première observation qui peut être faite, est que la probabilité de fonctionnement du réseau se dégrade rapidement avec l'utilisation d'un chemin unique. La seconde remarque est qu'il n'est pas nécessaire de définir plus d'un chemin redondant (soit deux au total), puisque nous respectons les conditions définies par le niveau SIL 4 et ce, jusqu'à atteindre

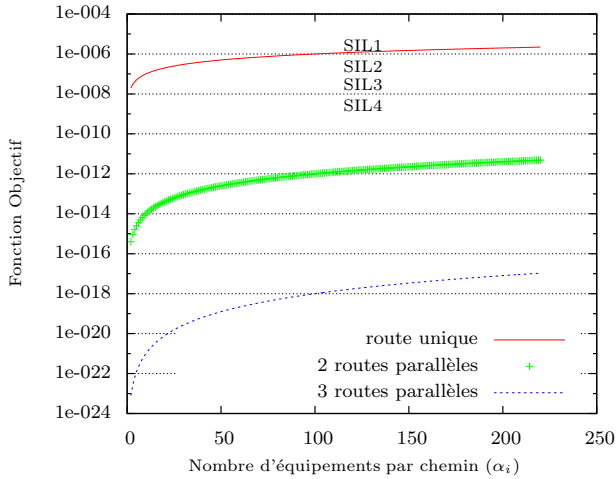


Fig. 2. Analyse de fiabilité pour des chemins indépendants

10001 équipements sur les chemins respectifs. Ce nombre semble largement suffisant pour l'interconnexion de SCR. De plus, multiplier les routes a un impact sur le coût de la solution mise en œuvre (bande-passante, liens, etc.). La conclusion est telle qu'à partir de deux chemins il existe des solutions compatibles avec les exigences de niveau SIL 4, ainsi une architecture réseau implémentant deux chemins parallèles est un compromis acceptable en terme : fiabilité/coût.

### C. Contiguïté entre chemins

Dans cette section, l'impact de la contiguïté sur le réseau est étudié. Pour cela, nous considérons un réseau composé de deux chemins non indépendants. Définissons  $\alpha_i$  comme étant le nombre d'équipements réseau composant le chemin et non communs à l'autre chemin, et  $\beta$  comme étant le nombre d'équipements communs aux deux chemins. La longueur totale d'un chemin sera ainsi équivalente à  $\alpha_i + \beta$ . En nous appuyant sur l'équation (2), il est possible de déterminer la probabilité de dysfonctionnement du système, représenté par (3), avec  $P_S$  correspondant à la probabilité de succès du système (le réseau).

$$P_D = 1 - P_S = 1 - (1 - P_{D_1} \cdot P_{D_2}) \mu^\beta \quad (3)$$

La Fig. 3 illustre les probabilités de défaillances estimées par notre fonction objectif (3) en fonction de la longueur des chemins et de la proportion d'équipements en commun entre chemins ( $\beta/\mu$ ). L'observation qui peut-être faite ici est qu'il suffit d'un équipement en commun sur les deux chemins, pour que la probabilité de défaillance soit d'ordre SIL 3 environ. Ceci montre que la contiguïté entre chemins amène un coût qui devra être évalué par la suite eu égard à la probabilité de défaillance désirée.

### D. Critère d'optimisation

Le problème à résoudre consiste à trouver la paire d'arbre permettant de minimiser la fonction objectif (3), c'est-à-dire de minimiser la probabilité de défaillance de la paire de chemins. En fait, il s'agit ici de rechercher deux arbres  $A_1$  et  $A_2$  définissant chacun un sous-arbre  $a_i$  interconnectant trois points du réseau. Sachant que les algorithmes

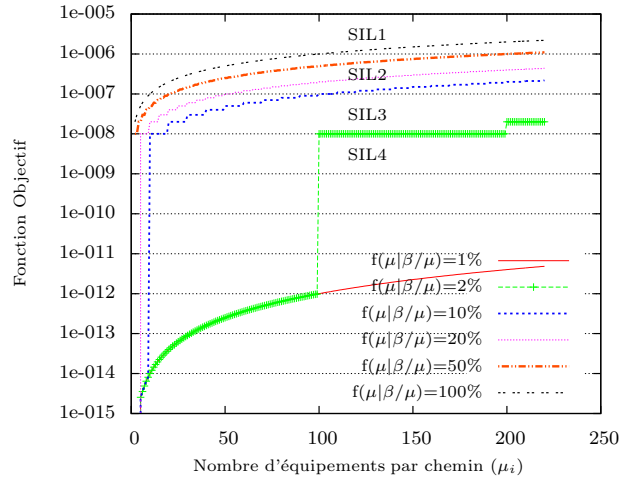


Fig. 3. Analyse de fiabilité pour deux chemins (*non indépendants*)

développés concernant la simple recherche d'arbres dis-joints [5] présentent dans le pire cas une complexité d'ordre  $O(|L||N|)$ , avec  $|L|$  le nombre de liens et  $|N|$  le nombre de noeuds, il est proposé l'utilisation d'heuristiques pour la recherche d'une paire d'arbres  $A_i/a_i$ .

## IV. ALGORITHME D'OPTIMISATION

### A. Choix de l'heuristique

On distingue trois classes principales, à savoir, les méthodes constructives (algorithmes glouton, méthode Pilote), les méthodes à recherche locale (recuit simulé, recherche Tabou) et les méthodes considérées comme évolutionnaires (algorithmes génétiques, systèmes de fourmis). L'évaluation de méthodes heuristiques repose essentiellement sur des analyses statistiques sur les résultats obtenus après avoir appliqué plusieurs fois de suite ces méthodes. L'auteur de [1] établit une comparaison entre les méthodes du recuit simulé, la recherche de type tabou, les colonies de fourmis et les algorithmes évolutionnaires (AG). Cette comparaison repose sur des diagrammes donnant la probabilité qu'une méthode soit meilleure que l'autre. On constate dans ces travaux que pour un même problème donné, les AG donnent pour ce type d'étude les meilleurs résultats.

### B. Codages des éléments du réseau

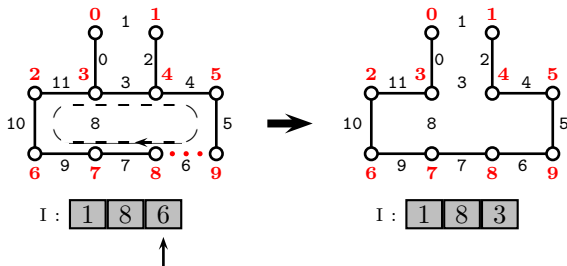
La technique de codage concernant les graphes représente la manière de modéliser les connexions entre les noeuds d'un graphe (orientées ou non). Le plus souvent, ces connexions sont exprimées par des matrices. Un graphe non orienté, où les arcs correspondent aux liens est défini à partir des probabilités de défaillance sur les liens réseau. Nous créons ainsi une matrice dite "matrice d'incidence sommets-arêtes" de dimension  $m * n$  où  $m$  correspond au nombre de noeuds et  $n$  au nombre d'arêtes. Lorsque l'arête  $n_i$  est composée des noeuds d'extrémités  $m_j$  et  $m_k$ , alors on place à la ligne  $m_j$ , ainsi qu'à la ligne  $m_k$  de la colonne  $n_i$  la valeur 1, puis le reste de la colonne est complétée par la valeur 0. A partir de cette matrice, il est possible de construire avec les noeuds extrémités de chaque arête, deux vecteurs de taille égale au nombre d'arêtes du graphe.

Il existe plusieurs méthodes de représentation d'arbres

couvants d'un graphe [3] [9] [12] [10] [11] [8]. Nous avons choisi la méthode "Codage numérique des branches ouvertes" [10], ce qui va permettre de réaliser des échanges entre les branches de deux arbres couvants. Mais nous reviendrons sur les avantages de travailler avec cette méthode, après avoir défini l'algorithme d'optimisation sur lequel nous allons nous appuyer.

### C. Algorithme génétique (AG)

Dans notre étude l'individu sera représentatif d'un arbre couvrant, dont les gènes composant le chromosome correspondront aux branches ouvertes. On remarque qu'en fonction du type de codage cités préalablement, les opérateurs de croisement et mutation pourront générer (avec un facteur plus ou moins élevé) des individus non valides par rapport aux contraintes topologiques. [2] explique qu'en prenant la méthode de codage basée sur la représentation en co-arbre (codage numérique des branches ouvertes), cela évite la détermination des boucles fondamentales permettant de tester la validité des topologies résultantes des processus de croisement/mutation. En effet, une stratégie de croisement et de mutation est développée à partir des approches de la théorie des graphes et des matroïdes. La technique de mutation consiste à tirer au hasard un gène d'un chromosome (représentatif d'une branche ouverte), puis à déterminer la boucle formée suite à la fermeture de cette branche et enfin, de choisir aléatoirement au sein de cette boucle une branche à ouvrir. Ce principe est illustré Fig. 4.



**Step 1 :** Choix aléatoire d'un gène (branche) à muter.

**Step 2 :** Choix aléatoire d'une branche issue de la boucle formée suite à la fermeture de la branche 6. Boucle : [3, 4, 5, 6, 7, 9, 10, 11]

Fig. 4. Technique de mutation

La technique de croisement illustrée à la Fig. 5, consiste quant à elle à tirer au hasard un point de référence dans le chromosome de l'individu I1, puis déterminer la boucle formée à la suite de la fermeture de cette branche et enfin, regarder quelles branches seraient susceptibles d'être échangées entre les deux co-arbres. Pour cela, il faut analyser le chromosome de l'individu I2. S'il y a des gènes appartenant à la boucle, il faut alors effectuer le croisement avec l'un d'entre eux. Il faut réitérer cette démarche jusqu'à atteindre le dernier gène.

### D. Ajustement des paramètres de l'AG

Goldberg [4] définit des paramètres conventionnels pour les AG tels que la taille de la population  $t_{pop}$ , la probabilité de croisement  $p_c$  et de mutation  $p_m$ . Ces paramètres influent beaucoup sur les performances de l'algorithme, c'est pourquoi l'étude de l'influence de chacun d'eux afin de

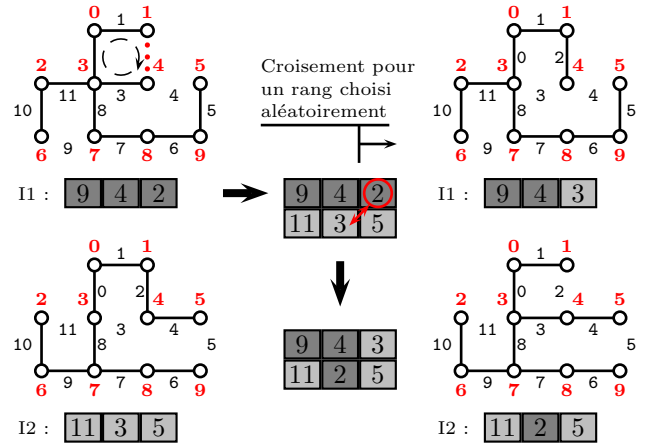


Fig. 5. Technique de croisement

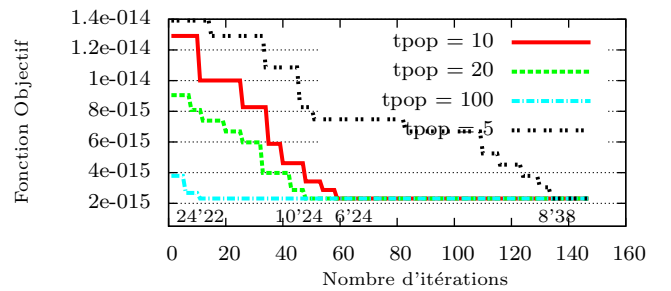


Fig. 6. Ajustement du paramètre  $t_{pop}$

déterminer leur valeur donnant les performances optimales, se révèle importante. Krommenacker [6] met en place une démarche permettant d'effectuer ces réglages, que nous avons repris ici. A titre indicatif, nous avons représenté Fig. 6 l'influence de  $t_{pop}$  sur la qualité de la solution et le temps de calcul pour un réseau commuté full duplex point à point composé de 100 nœuds, 240 branches. Ces réseaux sont du même type que celui présenté Fig. 9 où le graphe source est généré aléatoirement.

La figure indique le temps mis dans chaque cas pour atteindre le même optimum local. On constate que plus la taille de la population augmente, moins l'algorithme nécessite d'itérations pour atteindre l'optimum commun, mais cela au détriment du temps de recherche. Dans notre cas, le meilleur compromis semble avoir lieu pour une taille de population de 10. Afin de fixer la valeur  $p_c$  et  $p_m$ , nous avons étudié l'influence de ces probabilités pour une série de 100 réseaux de taille fixe (100 nœuds, 240 branches). Dans un premier temps, l'AG a été exécuté pour  $p_c$  variant de 10 à 90%, avec les autres paramètres fixés à :  $t_{pop} = 10$ ,  $p_m = 40\%$  et ce sur 150 itérations. Nous avons ainsi relevé pour chaque valeur du taux de croisement, la valeur optimale de la fonction objectif. La courbe "croisement" Fig. 7 illustre la courbe de tendance obtenue à partir des valeurs moyennes sur les 100 graphes. On constate alors qu'un taux  $p_c$  de 70% permet d'obtenir la meilleure solution. Dans un second temps, l'AG a été exécuté pour  $p_m$  dans les mêmes conditions, avec les autres paramètres fixés à :  $t_{pop} = 10$ ,  $p_c = 70\%$ . La courbe "mutation" Fig. 7 montre alors qu'un taux  $p_m$  de 40% permet d'obtenir la meilleure solution.

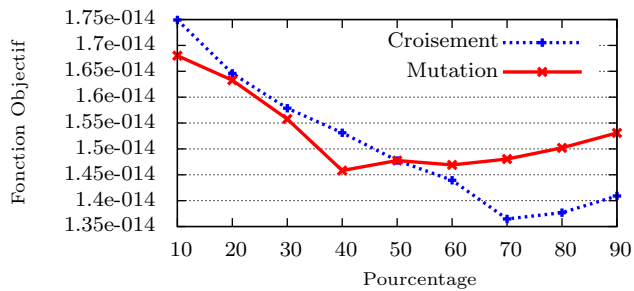


Fig. 7. Ajustement des paramètres  $p_m$  et  $p_c$

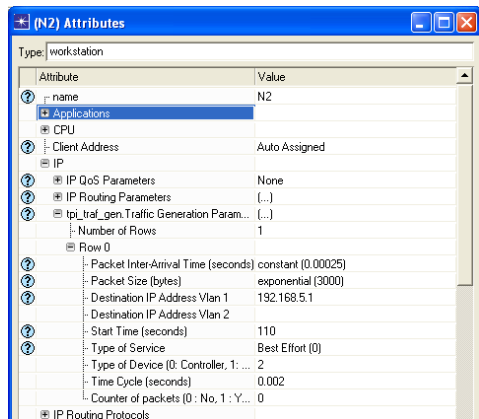


Fig. 8. Module OPNET de simulation d'un SCR

## V. SIMULATION

### A. Introduction

Le but de cette section est de montrer l'intérêt de nos travaux sur un Système Contrôlé en Réseau (SCR) simulé sous Opnet. Un module permettant de réaliser la partie contrôleur, actionneur, capteur a été développé sous Opnet. La Fig. 8 présente un aperçu de ce module, dans lequel on peut associer à un nœud le temps de cycle automate, le temps de cycle de communication.

Les équations du système implantées dans ce module sont données ci-dessous.

$$P(s) = \frac{2}{(s+5)(s+0,2)}, \quad C(s) = \frac{0.5508s + 0.4529}{s}$$

La plate-forme réseau étudiée est illustrée Fig. 9, elle est composée de 30 nœuds (commutateurs) et 56 branches (liens). Les équipements du SCR (contrôleur, capteur, actionneur) sont respectivement reliés aux commutateurs 20, 5 et 11. Le temps de cycle automate est de 2 ms, les équipements envoient des paquets de 64 octets toutes les 1 ms. L'étude se décompose en deux phases, la première sera dédiée à l'analyse de la plateforme via une solution classique (STP). Puis on procédera à la mise en place de la meilleure paire de chemins sur nos VLAN respectifs. Il est à noter que nous utiliserons le protocole RSTP lors de l'étude. Par nécessité de clarté concernant les arbres couvrant, nous représenterons uniquement les liens faisant partie des chemins reliant nos équipements du SCR.

### B. Solution classique (sans redondance de chemin)

L'arbre couvrant défini par défaut par STP utilise les liens 3, 2, 5, 8, 11, 1, 14 afin de relier le contrôleur, l'actionneur et le capteur. Nous procédons à  $t = 115s$  à une

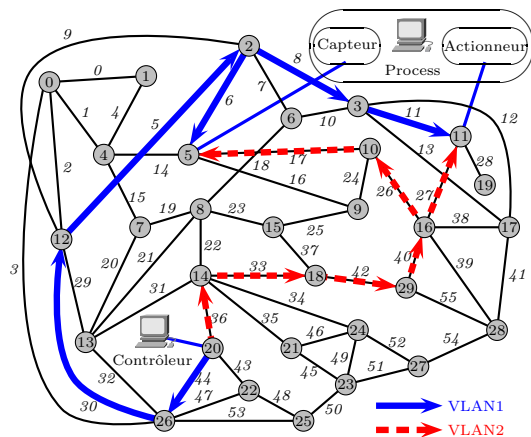


Fig. 9. Plate-forme de l'étude

défaillance du lien 8. La Fig. 10 représentative du SCR, montre que la réponse du système  $y(t)$  ne suit plus la consigne du fait des informations non reçues par l'actionneur. Le SCR redevient stable lorsque le réseau a terminé la reconfiguration d'un nouvel arbre, c'est-à-dire à  $t = 120s$ .

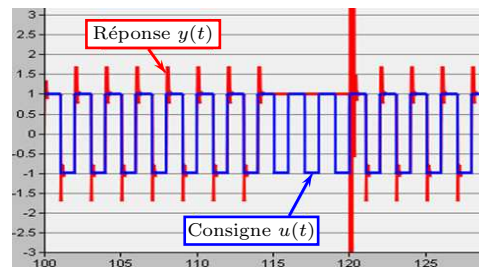


Fig. 10. Solution classique (Sans Multi-VLAN)

### C. Mise en place de la solution MSTP

Cette nouvelle étude se base sur la technique de duplication de paquets sur une paire de chemins. Cette paire est obtenue via notre outil JPES implantant l'algorithme défini précédemment, elle est présentée Fig. 9. Nous procédons dans un premier temps à une défaillance sur le lien 6 à  $t = 110s$ . Cette dernière empêche alors les messages envoyés par le capteur de parvenir jusqu'au contrôleur via le VLAN1. La Fig. 11 montre que cela ne perturbe nullement le système du fait que l'acheminement de ces mêmes messages peut se faire via le VLAN2. Nous remettons dans un second temps le lien 6 en fonctionnement, les arbres vont ainsi redevenir tels qu'ils étaient initialement. Nous procédons alors à  $t = 140s$  à une rupture du lien 33 qui lui, appartient au VLAN2, coupant ainsi toute communication entre contrôleur/actionneur et capteur/contrôleur de ce VLAN. On constate encore une fois que le SCR ne s'en trouve pas perturbé du fait que les messages sont acheminés via le VLAN1. Nous remettons ensuite le lien 33 en fonctionnement, les arbres redeviennent tels qu'ils étaient initialement. Nous procédons pour finir à une rupture des liens 6 et 33 à  $t = 170s$ , ce qui a pour conséquence de rendre un lien de chaque VLAN défaillant en même temps. La réponse du SCR  $y(t)$  devient alors instable du fait du non retour des informations du capteur au contrôleur, mais aussi du contrôleur à l'actionneur et ce, jusqu'à ce qu'un des



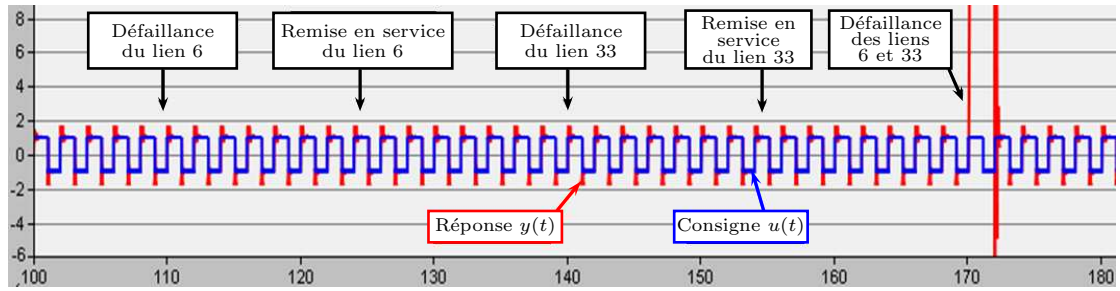


Fig. 11. Solution Multi-VLAN

arbres (STP) soit reconfiguré, ce qui se produit à  $t = 173s$ . Nous avons finalement constaté que pour rendre le SCR instable avec la solution de "duplication de paquets", il faut simultanément une rupture sur chaque VLAN, d'où l'intérêt d'avoir deux chemins les plus disjoints possibles.

#### D. Validité temporelle des messages

Nous allons maintenant montrer l'incidence des retards sur les VLAN. Rappelons le principe de base de la méthode passive, qui est de dupliquer un message envoyé par un équipement sur les différents VLAN (VLAN1 et 2). Nous avons mis en place un trafic sur le lien 42 à  $t = 130s$ , afin de générer de la congestion sur le VLAN2. Cette congestion génère des retards plus importants sur les messages empruntant le chemin VLAN2 par rapport aux messages du VLAN1. Des mêmes informations sont alors reçues et traitées à des instants différents, entraînant ainsi une instabilité du SCR comme le montre la Fig. 12. Il faut donc ajouter des dates ou numéros à chaque message de façon à éliminer les informations "périmées". Nous avons choisi dans cette étude de numéroter les paquets. La Fig. 13 montre qu'avec cette solution le SCR reste stable, du fait que les équipements du SCR ne prennent plus en compte des informations dites "périmées".

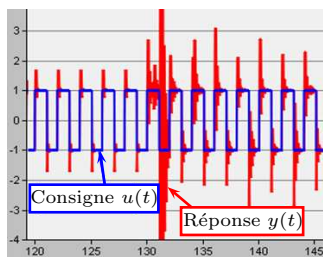


Fig. 12. Sans numérotation

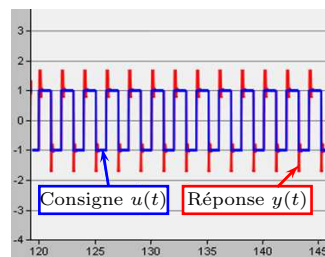


Fig. 13. Avec numérotation

## VI. CONCLUSION

Dans ce papier, une méthode permettant d'améliorer la continuité de service sur Ethernet a été développée, basée sur une méthode passive mettant en œuvre deux arbres couvrants sur le réseau. Ces arbres couvrants sont déterminés à l'aide d'une méthode basée sur les algorithmes génétiques, pour laquelle une fonction objectif a été définie. Cette fonction évalue des paires d'arbres couvrants, d'une part en calculant le nombre de sauts de chaque chemin et d'autre part, en évaluant le nombre d'équipements communs aux deux chemins. Cette solution permet de réduire la probabilité que les applications critiques subissent des coupures de communication entraînant de longues phases

de reconfiguration. La contrepartie de notre solution, c'est qu'il y a un coût en bande passante à évaluer. Dans la simulation OPNET réalisée précédemment, la solution MSTP proposée consomme  $500Kb/s$  de bande passante supplémentaire au niveau d'un nœud par rapport à une solution classique. Sachant que des liens à  $10Mb/s$  sont utilisés, cela signifie que l'on consomme  $1/20^{ème}$  de bande passante supplémentaire dans notre cas. Il est alors indispensable de faire en parallèle de cette étude, une étude temporelle sur les délais de transmissions. Une des perspectives de cette étude est d'étendre cette solution au cas des architectures sans fil et hybrides.

#### REMERCIEMENTS

Ce travail a été soutenu par OPNET Technologies, Inc. à travers le programme "Teaching with OPNET".

#### RÉFÉRENCES

- [1] Dreoj J. *Metaheuristics for hard optimization : methods and case studies*. Springer, Berlin, 2006.
- [2] Enacheanu B. Outils d'aide à la conduite pour les opérateurs des réseaux de distribution. Thèse de doctorat de l'Université INP Grenoble, 26 Octobre 2007.
- [3] Enacheanu B., Raison B., Ivanova D., Caire R., Aubry A. et Hadjsaid N. Flexible Electric Infrastructures for Advanced Distribution Automation. Third International Conference on Critical Infrastructures, CRIS 2006, Alexandria, États-Unis, 2006.
- [4] Goldberg D. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1989
- [5] Jayavelu G., Ramasubramanian S. et Younis O. Maintaining colored trees for disjoint multipath routing under node failures. *IEEE/ACM Trans. Netw.*, vol. 17, n° 1, pp. 346-359, 2009.
- [6] Krommenacker N. Heuristiques de conception de topologies réseaux : application aux réseaux locaux industriels. Thèse de doctorat de l'Université Henri Poincaré, Nancy, Décembre 2002.
- [7] Limal S., Potier S., Denis B. et Lesage J.-J. Formal verification of redundant media extension of Ethernet PowerLink. Proceedings of 12th IEEE Conference on Emerging Technologies and Factory Automation, Patras, Grèce, 2007.
- [8] Lin W.-M., Cheng F.-S. et Tsay M.-T. Distribution feeder reconfiguration with refined genetic algorithm. *IEEE Transactions on Power Systems Proceedings - Generation, Transmission and Distribution*, vol. 147, n° 6, pp. 349-354, 2000.
- [9] Nara K., Shiose A., Kitagawa M. et Ishihara T. Implementation of genetic algorithm for distribution systems loss minimum reconfiguration. *IEEE Transactions on Power Systems*, vol. 7, n° 3, pp. 1044-1051, 1992.
- [10] Radha B., Ah King R.T.F. et Rughooputh H.C.S. A modified genetic algorithm for optimal electrical distribution network reconfiguration. Proceedings of the 2003 Congress on Evolutionary Computation, CEC2003, Canberra, Australie, Décembre 2003.
- [11] Ramos E. R., Exposito A. G., Santos J. R. et Iborra F. L. Path-based distribution network modeling : application to reconfiguration for loss reduction. *Power Systems, IEEE Transactions on*, vol. 20, n° 3, pp. 556-564, 2005.
- [12] Zhu Y. et Tomsovic K. Adaptive Power Flow Method for Distribution Systems With Dispersed Generation. *Power Engineering Review, IEEE*, vol. 22, n° 5, pp. 72-72, 2002.