On the observational semantics of fair parallelism

Philippe Darondeau, Laurent Kott

IRIA

CENTRE DE RENNES

IRISA

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P.105
78153 Le Chesnay Cedex
France

Tél.:(3) 954 90 20

Rapports de Recherche

N° 262

ON THE OBSERVATIONAL SEMANTICS OF FAIR PARALLELISM

Philippe DARONDEAU
Laurent KOTT

Décembre 1983

IRISA

INSTITUT DE RECHERCHE EN INFORMATIQUE
ET SYSTÈMES ALÉATOIRES

Campus Universitaire de Beaulieu
Avenue du Général Leclerc
35042 - RENNES CÉDEX
FRANCE
Tél. : (99) 36.20.00
Télex : UNIRISA 95 0473 F

ON THE OBSERVATIONAL SEMANTICS OF FAIR PARALLELISM

by

Ph. DARONDEAU, L. KOTT

Publication Interne n° 211
80 pages
Octobre 1983

RESUME : Nous montrons qu'il existe une procédure de décision de la congruence observationnelle des programmes pour le sous-ensemble rationnel de CCS sous l'hypothèse d'équité forte. Ce rapport est la version complète d'un texte qui a été présenté à l'ICALP 83.

ABSTRACT :

We prove that there exists an effective procedure for deciding the observational congruence of programs in the rational subset of CCS under the assumption of strong fairness. This report is the full version of an extended abstract which was presented at ICALP 83.

CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE
(L.A. 227)
UNIVERSITÉ DE RENNES 1   I.N.S.A. DE RENNES

INSTITUT NATIONAL DE RECHERCHE
EN INFORMATIQUE ET EN AUTOMATIQUE
(LABORATOIRE DE RENNES)

TABLE OF CONTENTS

ON THE OBSERVATIONAL SEMANTICS OF FAIR PARALLELISM

Ph. Darondeau and L. Kott
IRISA
Campus de Beaulieu
F-35042 RENNES CEDEX

1. INTRODUCTION

The work reported below stems from several remarks upon Milner's calculus of communicating systems (CCS) [Mi80].

- Among conditions to be fulfilled by observationally equivalent systems S and S', it is required that for any sequence ρ of observable actions, and possible state σ of S after experiment ρ, there exists some equivalent state σ' of S' possibly reached after identical experiment. Systems may happen to be discriminated that way although no experiment makes any difference between them.
- The observational equivalence is not a congruence, which bears evidence of the weaknesses of principles assumed for observing systems.
- The unrestricted power of the programming language entails incompleteness of the calculus, so that it is impossible to decide wether its parallel composition is fair or unfair.
- Parallelism is reduced to sequential nondeterminism, which involves debts in the fairness issue.

In the case of finite behaviours, the first remark already led us to enhance an enlarged equivalence and an associated proof system [Da82]. In the vein of Hoare's theory of CSP [Ho81], our proposal excludes the sum operator (+) to the benefit of n-ary guarding operators $(\mu_1,\ldots,\mu_n)$ and makes the assumption of a sequential observer which presents ambiguous action demands such as $(\mu_1,\ldots,\mu_n)$. By the way, the observer is able to simulate any non-deterministic program context, whence the identity between the equivalence and the adjoined congruence : both relations are defined as the equality between the alternated demand and response languages which represent the observations of programs. In that framework, the observational semantics of programs cannot be expressed by the conditional rewriting systems of Plotkin [Pl 81] in the exact way they are used in [He 80, Mi 80, HeP 80], since actions must be combined with action demands in the labelling of the rewriting relations.

The present work aims to extend our previous approach to infinite behaviours and to achieve fairness of the parallel composition, which requirement has been evaded as yet in all studies inspiring from asynchronous CCS. The intended kind of fairness is one of possible derivatives of the original property described by Park, several interpretations of which make sense when applied to communicating systems [KuR 82]. Our intention is to validate the following statement

PARALLELISM = FAIRNESS $\neq$ SEQUENTIAL NON DETERMINISM

For example, we take it as granted that the machine H+T which repeats indefinitely the non deterministic choice between two actions "head" and "tail", equally looked for by an observer, has some extra behaviours which do not pertain to the parallel composition H|T of two machines H and T, each of which iterates the corresponding action "head" or "tail". A subtle distinction is then established between asynchronous parallelism and non-determinism, much more refined than the usual difference which lays in the occasional simultaneity of events [Wi80, Da80, CMF82] : although asynchronous systems with finite behaviours have always purely sequential equivalents, some infinite asynchronous systems have the prerogative not to support any sequential equivalent. Nor will be considered here the strong simultaneity of events, taken as a basic phenomenon in synchronized calculi such as [Mi82, AuB82]. The fact is that, although the history of interactions approach might still be used, strong simultaneity of events would require a much more refined expression of the condition of fairness, since a process could be waiting for several resources used one at a time by other concurrent processes.

By showing that for synchronization programs with bounded parallelism, languages of observations may be composed according to the structure of programs and still remain in the well known class of rational languages, this paper establishes the existence of a decision procedure for the observational congruence of fairly communicating processes. This result makes it reasonable to search for a corresponding formal proof system ; the main difficulty lies in the axiomatization of infinitary rational expressions.

The remaining sections are organized as follows. Section 2 states our principles of observation. Section 3 describes the programming language and builds its operational semantics. Section 4 derives the observational semantics of programs from their operational semantics and contains the main results of the paper.

The proofs of the results are scattered over six appendices which introduce all the required material.

2. PRINCIPLES OF OBSERVATION

Programs in our scope are pure synchronization programs. A particular system comes from joining a program with an initial agent. An agent is said composite when it is programmed as the parallel composition of other agents, else it is atomic. M, the vocabulary of actions $\{\mu_1,\ldots,\mu_i,\ldots\}$, is the disjoint union of two subsets Δ and $\overline{\Delta}$, related to each other by a pair of bijections $^{-}$: $\lambda \in \Delta \mp \overline{\lambda} \in \overline{\Delta} \mp \overline{\overline{\lambda}} = \lambda \in \Delta$. Given an agent p composed of atomic agents q_i, $i=1,\ldots,n$, an operation of p is either an action μ of one of q_i's, which responds to some demand from the outer, or an interaction $(\mu,\overline{\mu})$ between a pair of agents (q_i,q_j). Actions of atomic agents are directed according to the communication capabilities $(\mu_1,\ldots,\mu_n)$ allowed by their programs. When using such a capability μ_i, an atomic agent disappears from the embedding agents to the benefit of a new agent, possibly atomic or composite. The presentation of capabilities $(\mu_1,\ldots,\mu_n)$ by one of the agents amounts for the other agents to a complementary action demand $(\overline{\mu}_1,\ldots,\overline{\mu}_n)$ issued from the global environment of the system. Bounded parallelism and bounded sequential non determinism are assumed.

Experimenting over a system amounts to involve it in an environment made of one or several agents, the observers, which it may consequently interact with. The system of observers obeys the same communication laws as the observed system does obey, but the behaviour of an observer is not constrained by a program. Observers submit to the observed system freely chosen action demands $(\mu_1,\ldots,\mu_n)$, possibly answered by the observed agents which display sufficient capabilities to perform one of the actions μ_i. Since an action demand $(\mu_1,\ldots,\mu_n)$ of an observer amounts, for the other observers, to the presentation of complementary capabilities $(\overline{\mu}_1,\ldots,\overline{\mu}_n)$ by the observed system, observers cannot avoid to interact by mutual answering. That property of observers is quite essential : under the assumption of a fair execution of the closed system made of both kinds of agents, infinite sequences of interactions between observers provide default information upon the observed system. We call an experience any fair processing of such a closed system : an atomic agent who infinitely often has the possibility to interact with other agents in the course of an experience will necessarily do so. We shall assume a constant number of observers and exclude the case of observers which could compete on non disjoint action demands. Let the observers date and record their individual acts, then the history of the system of observers may be gathered into a finite or infinite word over the alphabet of demands and responses, on condition that pairs of identically dated responses $\lambda,\overline{\lambda}$ are confused into simple elements $\tilde{\lambda}$ which represent interactions between observers. For a program p with sort $\Lambda \subseteq M$, let $Exp_\Lambda(p)$ be the set of the words which are constructed so. $Exp_\Lambda(p)$ will be called a language of experiments.

Let w in $Exp_\Lambda(p)$ - e.g. $w = (\alpha,\beta)(\gamma,\delta)\alpha(\alpha)(\overline{\gamma},\beta)\hat{\gamma}$ - ; let Fail(w) be the set union of unsuccessful demands which occur in w - e.g. (α) - intersected with ($\Lambda \cup \overline{\Lambda}$) ; let Div(w) be the subset of the action names which infinitely often occur in the demands of w intersected with ($\Lambda \cup \overline{\Lambda}$) ; let Resp(w) be the sequence of responses μ_i in Λ which occur in w, postfixed with a special symbol χ and excluding $\hat{\chi}$ symbols - e.g. $\alpha\chi$ -. For such a sequence ρ, let Act(ρ) - resp. Ult(ρ) - be the subset of the action names which occur - resp. occur infinitely often - in ρ. Using the notations $d = Div(w)$, $\delta = Fail(w)$ and $\rho = Resp(w)$, we define the application Ψ : $\Psi(w) = (d,\delta,\rho)$ then the following properties are verified by $\Psi(w)$ for any word of experiments w :

1. $d \subseteq \Lambda \cup \overline{\Lambda}$, $\delta \subseteq \Lambda \cup \overline{\Lambda}$, $\rho \in \Lambda^\omega \cup \Lambda^* \chi$
2. $Ult(\rho) \subseteq d$
3. $d \cap (\delta \cup \overline{\delta}) = \emptyset$

Define Obs(Λ) as the set of triples (d,δ,ρ) which satisfy conditions 1 to 3, then Obs(Λ) may be ordered by the relation $(d,\delta,\rho) \leq (d',\delta',\rho')$ if and only if ($\rho = \rho'$ and $\delta \subseteq \delta'$ and $d \subseteq d' \cup \delta'$) or ($\rho = \rho''\chi$ and $\rho'' < \rho'$ and $d \cup \delta = \emptyset$) where $<$ is the prefix order over words. We call a language of observations any downwards and non empty subset of Obs(Λ). A huge amount of combinatorial developments shows that the fonction Ψ is actually a bijection between languages of experiments and languages of observations. Let $Obs_\Lambda(p)$ denote $\Psi(Exp_\Lambda(p))$; for any program p with sort Λ, $Obs_\Lambda(p)$ will be considered from now on as the observational semantics associated with program p.

3. THE PROGRAMMING LANGUAGE AND ITS OPERATIONAL SEMANTICS

3.1. THE SYNTAX OF PROGRAMS

Given the set $M = \Delta \cup \overline{\Delta}$ of action names and a set X of variables, the syntactic categories of terms and programs in the language are the least families constructed from the following rules, where we let MS(t) and FV(t) respectively denote the minimal sort and set of free variables of term t (μ_i, x_i, t_i stand for an action name, a variable, a term) :

- NIL is a term, $MS(NIL)=\emptyset$, $FV(NIL)=\emptyset$
- x is a term , $MS(x)=\emptyset$, $FV(x)=\{x\}$
- $(\mu_1,\ldots,\mu_n)(t_1,\ldots,t_n)$ is a term, let t that term,
  $MS(t) = \{\mu_1,\ldots,\mu_n\} \cup (\bigcup_i MS(t_i))$, $FV(t) = \bigcup_i FV(t_i)$
- $Y(x_1 \leftarrow t_1,\ldots,x_n \leftarrow t_n)$ is a term if $\bigcup_i FV(t_i) \subseteq \{x_1,\ldots,x_n\}$,
  Let t that term then $MS(t) = \bigcup_i MS(t_i)$ and $FV(t) = \emptyset$
- $(t_1 | t_2)$ is a term if $FV(t_1) = FV(t_2) = \emptyset$, let t that term then

$MS(t) = MS(t_1) \cup MS(t_2)$ and $FV(t) = \emptyset$
- if $FV(t) = \emptyset$ then $t[/\mu_1 \ldots \mu_n]$ is a term, let t', $MS(t') = MS(t) \setminus (\{\mu_1 \ldots \mu_n\} \cup \{\overline{\mu}_1 \ldots \overline{\mu}_n\})$ and $FV(t') = \emptyset$
- if $FV(t) = \emptyset$ and $MS(t) \subseteq \Lambda$ then t is a program with $MS(t)$ as its minimal sort, and t_Λ is a program with minimal sort Λ.

Two kinds of construction operations are provided in the syntax :
- operations which define the *elementary* behaviours of programs (constant NIL, n-ary guarded selection $(\mu_1, \ldots, \mu_n)$, recursion (Y)) ;
- *flow-operations* which allow the composition of programs (parallel composition $|$, and restrictions $[/\mu_1 \ldots \mu_n]$.

3.2. HISTORIES

We call histories of a program the records of operation of the observed system in every possible experience upon a system initialized with that program. Given an history h_p of some program p of sort Λ, and an action name μ in $\Lambda \cup \overline{\Lambda}$, we say that μ is : *blocked* in h_p if some observed agent has remained endlessly inactive while displaying capability μ, *transient* in h_p if no observed agent shows capability μ beyond some step of the execution, *persistent* in h_p if neither blocked nor transient, *satiated* in h_p if corresponding responses μ have been issued infinitely often. From the assumption of fairness, we take it for granted that satiated labels are persistent and therefore not blocked. It follows that an history h_p of a program p with sort Λ may be represented by a triple (d,δ,ρ) : d, resp. δ, is the set of the action labels which are persistent, resp. transient, in h_p, so that $(\Lambda \cup \overline{\Lambda}) \setminus (d \cup \delta)$ contains exactly the labels of actions which are blocked in h_p. $H(\Lambda)$ will denote the set of such histories (d,δ,ρ) which verify conditions 1 and 2 of section 2 together with condition (3') : $d \cap \delta = \emptyset$. Define $H_\Lambda(p)$ as the set of histories of a program p with sort Λ ; from now on, $H_\Lambda(p)$ will be regarded as the operational semantics of p.

3.3. THE OPERATIONAL SEMANTICS

<u>definition 1</u>. For $W \subseteq M$ and $\mu \in W \cup \{\mathbb{1}\}$, we let $\xrightarrow{W,\mu}$ be the least binary relation over programs such that
- $(\mu_1, \ldots, \mu_n)(t_1, \ldots, t_n) \xrightarrow{(\mu_1, \ldots, \mu_n),\ \mu_i} t_i$, $1 \leq i \leq n$
- $Y(x_1 \leftarrow t_1, \ldots, x_n \leftarrow t_n) \xrightarrow{W,\mu} t$ iff

$W = \emptyset$, $\mu = \mathbb{1}$, $t = t_1$ and t_1 is the result of flow-operation, or

$t_1[Y(x_j \leftarrow t_j, \ldots, x_n \leftarrow t_n, x_1 \leftarrow t_1, \ldots)/x_j] \xrightarrow{W,\mu} t$
where the replacement of free variables applies for every j, $1 \leq j \leq n$.

<u>definition 2</u>. For elementary programs of sort Λ, $H_\Lambda(p)$ is the least set of histories determined by the following rules.

- if $p \xrightarrow{W,\mu} q$ for no W,μ and q, then $(\emptyset, \Lambda \cup \bar{\Lambda}, \chi) \in H_\Lambda(p)$
- if $p \xrightarrow{W,\mu} q$, then $(d,\delta,\mu\rho) \in H_\Lambda(p)$ for any (d,δ,ρ) in $H_\Lambda(q)$, and $(\emptyset, (\Lambda \cup \bar{\Lambda})\backslash W, \chi) \in H_\Lambda(p)$ if $\mu \neq 1\!\!1$
- if $p \xrightarrow{W_1,\mu_1} p_1 \xrightarrow{W_2,\mu_2} p_2 \ldots \xrightarrow{W_i,\mu_i} p_i \ldots$, $i \in \mathbb{N}$, then $(\lim_i(\bigcup_{j\geq i} W_j), \lim_i(\bigcap_{j\geq i}(\Lambda \cup \bar{\Lambda})\backslash W_j), \mu_1\mu_2\ldots\mu_i\ldots) \in H_\Lambda(p)$

<u>definition 3</u>. Given histories h_p and h_q in $H(\Lambda)$, h_p and h_q are compatible ($h_p \# h_q$) iff for any action μ blocked in h_p (resp. h_q), neither μ is satiated in h_q (resp. h_p) nor $\bar{\mu}$ is blocked or persistent in h_q (resp. h_p).
Clearly, given programs p and q, incompatible histories h_p and h_q cannot record the individual behaviors of p and q in a common experience upon their parallel compound $p|q$.

<u>definition 4</u>. Let f and g be two words of M^*, their parallel composition $f|g$ is inductively defined as follows, $\mu,\nu \in M$: either $f|g = \mu(f'|g)+\nu(f|g')$ with $f = \mu f'$, $g = \nu g'$ and $\mu \neq \bar{\nu}$ or $f|g = \mu(f'|g)+\nu(f|g')+(f'|g')$ with $f = \mu f'$, $g = \nu g'$ and $\mu = \bar{\nu}$.

<u>definition 5</u>. Let f and g be two words of M^∞, their parallel composition $f|g$ is the greatest subset of M^∞ such that $f|g = \sum_{n,m>0}(f\langle n\rangle|g\langle m\rangle)(f>n|g>m)$
where $f\langle n\rangle$ is the longest left factor of f of length less than or equal to n, and $f = (f\langle n\rangle)(f>n)$.

<u>definition 6</u>. Given compatible histories $h_p = (d_p,\delta_p,\rho_p)$ and $h_q = (d_q,\delta_q,\rho_q)$ in $H(\Lambda)$ $(h_p|h_q)$ is the set of histories (d,δ,ρ) which verify conditions i to iii :
i) $\delta = \delta_p \cap \delta_q$ ii) $d \cup \delta = (d_p \cup \delta_p) \cap (d_q \cup \delta_q)$
iii) $\rho\backslash\chi \in ((\rho_p\backslash\chi)|(\rho_q\backslash\chi))$

Let h_p and h_q record behaviours of p and q in a common experience on their compound $p|q$, then the set of blocked (resp. transient) actions in the history of $p|q$ is the union (intersection) of their respective sets of blocked (transient) actions.

<u>definition 7</u>. $H_\Lambda(p|q) = \bigcup\{(h_p|h_q) \mid h_p \in H_\Lambda(p), h_q \in H_\Lambda(q), h_p \# h_q\}$.

<u>definition 8</u>. Let the restriction $R \equiv /\mu_1\ldots\mu_n$ and let sets of labels $\Lambda,\Lambda',\Lambda''$ be such that $\Lambda' = \{\mu_1\ldots\mu_n\}$, $\Lambda'' = \Lambda \cup \Lambda' \cup \bar{\Lambda}'$, then
$H_\Lambda(q[R]) = ((H_{\Lambda''}(q)) \downarrow (\Lambda' \cup \bar{\Lambda}')) \uparrow ((\Lambda \cup \bar{\Lambda}) \cap (\Lambda' \cup \bar{\Lambda}'))$

where $(d,\delta,\rho) \uparrow \Omega = (d,\delta \cup \Omega,\rho)$ and $(d,\delta,\rho) \downarrow \Omega$ is equal to $(d\backslash\Omega,\delta\backslash\Omega,\rho)$ if $\rho \in (M\backslash\Omega)^{\omega} \cup (M\backslash\Omega)^{*}\chi$ or else is empty.

proposition 9. For any program p and for any triple (d,δ,ρ) in $H_{\Lambda}(p)$, $\overline{d} \subseteq d \cup \delta$. (In clear, the complement of a persistent label cannot be blocked).

4. THE OBSERVATIONAL SEMANTICS OF PROGRAMS

Let the order relation $\leqslant$ of section 2 be extended from $Obs(\Lambda)$ to $H(\Lambda)$; we state that for any possible history $h_p = (d_p,\delta_p,\rho_p)$ of a program p with sort Λ, the observations which are produced by experiences in which p behaves according to h_p are exactly the elements of the set $\{(d,\delta,\rho) \in Obs(\Lambda) \mid (d,\delta,\rho) \leqslant (d_p,\delta_p,\rho_p)\}$.

As an example, let Λ be $\{\alpha,\beta\}$ and h_p be $(\emptyset,\alpha,\beta\chi)$; let us consider some pair of complementary labels $\gamma,\overline{\gamma}$ which do not belong to Λ, then a possible experience upon p is described by the infinite word w equal to $(\beta)\beta(\alpha,\gamma)(\overline{\gamma})\tilde{\gamma}(\alpha,\gamma)(\overline{\gamma})\tilde{\gamma}$ ..., which produces the observation $\Psi(w)$ equal to $(\alpha,\emptyset,\beta\chi) < (\emptyset,\alpha,\beta\chi)$. At the opposite, let now h_p be $(\overline{\alpha}\beta\overline{\beta},\alpha,\beta^{\omega})$, then $(\overline{\alpha}\beta\overline{\beta},\alpha,\beta^{\omega})$ is not an observation of p : no word such as $(\alpha)(\overline{\alpha},\beta,\overline{\beta})\beta(\overline{\alpha},\beta,\overline{\beta})\beta$ ... describes a possible experience since one of the observers would be endlessly deprived of a possible interaction with the other observers.

At the present time, the observational meanings of programs are indirectly defined by the law $Obs_{\Lambda}(p) = \{o \in Obs(\Lambda) \mid (\exists h \in H_{\Lambda}(p))(o \leqslant h)\}$ that has just been assumed. The remaining of the paper intends to show that these meanings are in fact a pre-semantics and to derive a direct calculus of that semantics. Preliminary definitions are needed.

4.1. PARTIAL OBSERVATIONS AND RATIONAL LANGUAGES

definition 1. pre-$Obs(\Lambda)$, the set of pre-observations, is constituted by the triples (d,δ,ρ) whose elements verify :

1'. $d \subseteq \Lambda \cup \overline{\Lambda}$, $\delta \subseteq \Lambda \cup \overline{\Lambda}$, $\rho \in \Lambda^{\omega} \cup \Lambda^{*}\chi \cup \Lambda^{*}$.

2'. $Ult(\rho) \subseteq d$ if ρ is complete, i.e. $\rho \in \Lambda^{*}\chi \cup \Lambda^{\omega}$, or else $Act(\rho) \subseteq d$

3'. $d \cap \delta = \emptyset$

3". $Ult(\rho) \cap \overline{\delta} = \emptyset$ if ρ is complete, or else $Act(\rho) \cap \overline{\delta} = \emptyset$

<u>definition 2.</u> We call a language of pre-observations any non-empty subset of pre-Obs(Λ) downwards closed for the generalized order $\leqslant$ of section 2. We adopt the convention that a language of pre-observations (resp. observations) may be given the notation $\lfloor L \rfloor$ (resp. $\lfloor L \rfloor_o$) where L is any subset of the language which includes its maximal elements and $\lfloor\ \rfloor$ (resp. $\lfloor\ \rfloor_o$) is the downwards closure operation in pre-Obs(Λ) (resp. Obs(Λ)).

<u>definition 3.</u> For any pair of pre-observations $o_1 = (d_1, \delta_1, \rho_1)$ and $o_2 = (d_2, \delta_2, \rho_2)$, the association $o_1 . o_2$ of o_1 and o_2 is the pre-observation o given by :
$o = o_1$ if o is complete, i.e. $\rho_1 \in \Lambda^*\chi \cup \Lambda^\omega$, or else
$o = (d_2, \delta_2, \rho_1\rho_2)$ if o_1, o_2 are respectively incomplete and complete, or else
$o = (d_1 \cup d_2 \cup (\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1),\ \delta_1 \cap \delta_2,\ \rho_1\rho_2)$

<u>definition 4.</u> P-obs(Λ), the set of partial observations, is the monoid with carrier pre-Obs(Λ) $\cup$ $\{\mathbb{1}\}$, neutral element $\mathbb{1}$, concatenation . extending the association . in pre-Obs(Λ), and order $\leqslant$ defined as the minimal extension of the order $\leqslant$ on pre-Obs(Λ). By way of enlargement, we call a language of partial observations any non empty and downwards closed subset $\lfloor L \rfloor$ of P-obs(Λ) such that $\lfloor L \rfloor = \{\mathbb{1}\}$ or $\mathbb{1} \notin \lfloor L \rfloor$.

<u>definition 5.</u> I : $(\text{pre-Obs}(\Lambda))^\infty \to$ P-obs(Λ) is the function s.t.
- for finite words $o_1 o_2 \ldots o_k$, $I(o_1 o_2 \ldots o_k) = o_1 . o_2 . \ \ldots\ . o_k$ if $k \geqslant 1$ or else $I(\mathbb{1}) = \mathbb{1}$, the neutral element of P-obs(Λ),
- for infinite words $o_1 o_2 \ldots o_i \ldots$, let $o_i = (d_i, \delta_i, \rho_i)$, then $I(o_1 o_2 \ldots o_i \ldots) = I(o_1 o_2 \ldots o_k)$ if ρ_k is complete for some k, or else $I(o_1 o_2 \ldots o_i \ldots)$ is the triple (d, δ, ρ) defined by :

$$d = \operatorname{Lim}_j(\bigcup_{i \geqslant j} d_i) \cup \operatorname{Lim}_j(\bigcup_{i \geqslant j} \delta_i) \setminus \operatorname{Lim}_j(\bigcap_{i \geqslant j} \delta_i)$$

$$\delta = \operatorname{Lim}_j(\bigcap_{i \geqslant j} \delta_i)$$

$$\rho = \rho_1 \rho_2 \ldots \rho_i \ldots \chi$$

<u>proposition 6.</u> I is a monoid homomorphism.

<u>definition 7.</u> According to [Ei74], we note Rat(Z^∞) the least family of subsets of Z^∞ which contains the finite subsets of Z^* and is closed under concatenation, set union, star and ω-star operations. (The following characterization is proved in [Ei74]).

<u>proposition 8.</u> Let L be a language over Z, L is a rational set of Z^∞ iff there exist finitary rational sets of Z^*, say B, B_i, C_i, $1 \leqslant i \leqslant n$, such that $L = B + \sum_i B_i . C_i^{\ \omega}$.

definition 9. For $\mathcal{L} \subseteq \Lambda^\omega \cup \Lambda^*\chi \cup \Lambda^*$, we define $\hat{\mathcal{L}} = \mathcal{L} \cap (\Lambda^\omega \cup \Lambda^*\chi)$ and $\dot{\mathcal{L}} = \mathcal{L} \cap \Lambda^*$. A response-language $\mathcal{L}$ is bi-rational if both its complete and incomplete parts $\hat{\mathcal{L}}$ and $\dot{\mathcal{L}}$ are rational sets in $\mathrm{Rat}((\Lambda \cup \{\chi\})^\infty)$. $\mathcal{L}$ is rational if $\hat{\mathcal{L}}$ is rational and $\dot{\mathcal{L}}$ is empty.

Notational equivalence will be assumed in the sequel between $(d,\delta,\mathcal{L})$ and $\{(d,\delta,\rho) \mid \rho \in \mathcal{L}\}$ for any response-language $\mathcal{L}$.

definition 10. A language of partial observations (resp. observations) is rational if it can be expressed as $\lfloor L \rfloor$ (resp. $\lfloor L \rfloor_o$), $L = \sum_i (d_i,\delta_i,\mathcal{L}_i)$, $1 \leq i \leq n$, where the $\mathcal{L}_i$'s are non empty bi-rational (resp. rational) response-languages.

proposition 11. Let $X \in \mathrm{Rat}(\text{pre-Obs}(\Lambda)^\infty)$ such that $\mathbb{1} \notin X$ and $X \neq 0$, then $\lfloor I(X) \rfloor$ is a rational language of partial observations. Moreover, there exists an effective procedure which, given the rational expression of X, computes I(X) in the form $\sum_i ((d_i,\delta_i,\hat{\mathcal{L}}_i) + (d_i,\delta_i,\dot{\mathcal{L}}_i))$, where $\mathcal{L}_i$'s are bi-rational response-languages.

proposition 12. If $\lfloor L \rfloor$ is a rational language of partial observations, then $\lfloor L \rfloor \cap \mathrm{Obs}(\Lambda)$ is a rational language of observations, let $\lfloor \varphi(L) \rfloor_o$. Moreover, there exists an effective procedure which, given the expression of L as in proposition 11, computes $\varphi(L)$ in the form $\sum_i (d_i,\delta_i,\mathcal{L}_i)$ where $\mathcal{L}_i$'s are rational response-languages.

proposition 13. Let S be a system of linear equations over Z^∞, such as $X_i = A_{i1}X_1 + \ldots + A_{in}X_n + C_i$; $1 \leq i \leq n$; $A_{ij} \in \mathrm{Rat}(Z^+)$, $C_i \in \mathrm{Rat}(Z^\infty)$. Let Y(S) denote the greatest solution of S, then Y(S) is a vector of rational languages, and there exists an effective procedure for computing that extremal solution.

definition 14. Let L and L' be two languages on M^∞, their parallel composition $L|L'$ is the set $\Sigma((f|g)$, f in L and g in L').

proposition 15. If L and L' are rational, their parallel composition $L|L'$ is rational; moreover, there exists an effective procedure for computing the parallel composition of rational languages.

4.2. THE SEMANTIC LAWS

definition 16. For any elementary program p of sort Λ, we let the associated language of partial observations $\text{P-obs}_\Lambda(p)$ be equal to $\lfloor I(\mathcal{M}(p)) \rfloor$ where $\mathcal{M}$ is inductively defined as follows, using X_i's as variables ranging over subsets of $(\text{pre-Obs}(\Lambda))^\infty$:

$\mathcal{M}(\text{NIL}) = (\emptyset, \Lambda\cup\overline{\Lambda}, \chi)$

$\mathcal{M}(x_i) = (\emptyset, \Lambda\cup\overline{\Lambda}, \mathbf{1}).X_i$

$\mathcal{M}((\mu_1,\ldots,\mu_n)(t_1,\ldots,t_n)) = (\emptyset, (\Lambda\cup\overline{\Lambda})\setminus\{\mu_1,\ldots,\mu_n\}, \chi) +$

$\sum_{i=1}^{n} (\{\overline{\mu}_i\} \cup \{\mu_1,\ldots,\mu_n\}, (\Lambda\cup\overline{\Lambda})\setminus\{\overline{\mu}_i\}\setminus\{\mu_1,\ldots,\mu_n\}, \mu_i).\mathcal{N}(t_i)$

where either $\mathcal{N}(t_i)$ is taken as a constant given by $\text{Obs}_\Lambda(t_i) = \lfloor \mathcal{N}(t_i) \rfloor_o$ if t_i is the result of a flow-operation, or else $\mathcal{N}(t_i) \equiv \mathcal{M}(t_i)$.

$\mathcal{M}(Y(x_1 \leftarrow t_1,\ldots,x_n \leftarrow t_n)) = Y_1(X_1 = \mathcal{M}(t_1),\ldots,X_n = \mathcal{M}(t_n))$

where Y_1 denotes the first component of the greatest solution of the corresponding system of linear equations over $(\text{pre-Obs}(\Lambda))^\infty$.

<u>proposition 17.</u> Let p be an elementary program of sort Λ, then the following relations hold : $\text{Obs}_\Lambda(p) = \text{P-obs}_\Lambda(p) \cap \text{Obs}(\Lambda) = \lfloor \varphi(I(\mathcal{M}(p))) \rfloor_o$. Let $q_1 \ldots q_n$ be the outermost subprograms of p which are direct results of flow operations ; if $\text{Obs}_\Lambda(q_i)$ is rational for any i, then $\mathcal{M}(p)$, $\text{P-obs}_\Lambda(p)$ and $\text{Obs}_\Lambda(p)$ are rational, and there exists an effective procedure for computing $\varphi(I(\mathcal{M}(p)))$ in the rational form $\sum_j (d_j, \delta_j, \mathcal{L}_j)$, given the syntax of p and the rational expressions of the sets $\text{Obs}_\Lambda(q_i)$.

Next result shows a little more suprising property, since it states that the set of observations of a system of parallel processes can be synthesized from the sets of observations of the parallel components : fairness conditions can still be taken into full account despite the loss of information on operational properties which comes from considering observations instead of histories (for instance, given the history $(\alpha, \overline{\alpha}, \chi)$, none of the corresponding observations $(\alpha\overline{\alpha}, \emptyset, \chi)$ and $(\emptyset, \overline{\alpha}, \chi)$ tells us that the system can indefinitely escape action α without offering $\overline{\alpha}$).

<u>proposition 18.</u> Let observations $o_p = (d_p, \delta_p, \rho_p)$ and $o_q = (d_q, \delta_q, \rho_q)$ in $\text{Obs}(\Lambda)$; o_p and o_q are compatible $(o_p * o_q)$ iff the following property holds for any $\lambda \in \Lambda$: $(\{\lambda,\overline{\lambda}\} \subseteq \delta_p$ or $\{\lambda,\overline{\lambda}\} \subseteq \delta_q$ or $\lambda \in \delta_p \cap \delta_q$ or $\overline{\lambda} \in \delta_p \cap \delta_q$ or $\{\lambda,\overline{\lambda}\} \subsetneq d_p \cap d_q)$.

<u>definition 19.</u> Given rational response languages $\mathcal{L}$ and $\mathcal{L}'$, we let $\mathcal{L} \| \mathcal{L}'$ stand for $((\mathcal{L} \setminus \chi) | (\mathcal{L}' \setminus \chi))\chi$, where operations $|$ and $\setminus$ are respectively the parallel composition and right division in $\text{Rat}((\Lambda \cup \{\chi\})^\infty)$.

<u>proposition 20.</u> If $\text{Obs}_\Lambda(p)$ and $\text{Obs}_\Lambda(q)$ are rational languages of observations, let $\text{Obs}_\Lambda(p) = \lfloor L_p \rfloor_o$, $L_p = \sum_i (d'_i, \delta'_i, \mathcal{L}'_i)$ and $\text{Obs}_\Lambda(q) = \lfloor L_q \rfloor_o$, $L_q = \sum_j (d''_j, \delta''_j, \mathcal{L}''_j)$ where the $\mathcal{L}'_i$'s and $\mathcal{L}''_j$'s are rational, then $\text{Obs}_\Lambda(p|q)$ is the rational language of observations $\lfloor L_p \| L_q \rfloor_o$ defined by $L_p \| L_q =$

$\sum_{ij}((d'_i \cap d''_j) \cup (d'_i \cap \delta''_j) \cup (\delta'_i \cap d''_j),\ \delta'_i \cap \delta''_j,\ \mathcal{L}'_i \,||\, \mathcal{L}''_j)$, i and j such that $(d'_i, \delta'_i, \chi) * (d''_j, \delta''_j, \chi)$.

As a consequence, there exists an effective procedure for computing $\mathrm{Obs}_\Lambda(p|q)$ in the rational form $\lfloor \sum_k (d_k, \delta_k, \mathcal{L}_k) \rfloor_o$, given $\mathrm{Obs}_\Lambda(p)$ and $\mathrm{Obs}_\Lambda(q)$ in similar forms.

<u>proposition 21</u>. Let programs p and q such that $p \equiv q[R]$ where $R \equiv /\mu_1 \ldots \mu_n$. Let sets of labels Λ, Λ', Λ'' be such that $\Lambda' = \{\mu_1 \ldots \mu_n\}$ and $\Lambda'' = \Lambda \cup \Lambda' \cup \overline{\Lambda}'$. If $\mathrm{Obs}_{\Lambda''}(q)$ is a rational language of observations, put $\mathrm{Obs}_{\Lambda''}(q) = \lfloor L_q \rfloor_o$ and $L_q = \sum_i (d_i, \delta_i, \mathcal{L}_i)$ where the $\mathcal{L}_i$'s are rational, then $\mathrm{Obs}_\Lambda(p)$ is the rational language of observations $\lfloor (L_q \Downarrow (\Lambda' \cup \overline{\Lambda}')) \uparrow ((\Lambda \cup \overline{\Lambda}) \cap (\Lambda' \cup \overline{\Lambda}')) \rfloor_o$, where $\uparrow$ is the same as in section 3, and $(d_i, \delta_i, \mathcal{L}_i) \Downarrow \Omega$ equals $(d_i \setminus \Omega, \delta_i \setminus \Omega, \mathcal{L}_i \cap (M \setminus \Omega)^\infty \chi) + (\emptyset, \emptyset, (\mathrm{pref}(\mathcal{L}_i) \cap (M \setminus \Omega)^*)\chi)$ - $\mathrm{pref}(\mathcal{L}_i)$ is the set of the proper left factors of words φ in $\mathcal{L}_i$ - As a consequence, there exists an effective procedure for computing $\mathrm{Obs}_\Lambda(p)$ in the rational form $\lfloor \sum_k (d'_k, \delta'_k, \mathcal{L}'_k) \rfloor_o$, given $\mathrm{Obs}_{\Lambda''}(q)$ in similar form.

The induction on the structure of programs may now be used to prove the following facts.

<u>proposition 22</u>. For any program p of sort Λ, $\mathrm{Obs}_\Lambda(p)$ is rational, and there exists an effective procedure which computes Obs_Λ in rational form.

<u>proposition 23</u>. Let programs q and q' with respective minimal sorts MS(q) and MS(q'). If $\mathrm{Obs}_\Lambda(q) = \mathrm{Obs}_\Lambda(q')$ for $\Lambda = MS(q) \cup MS(q')$, then $\mathrm{Obs}_{\Lambda'}(p[q]) = \mathrm{Obs}_{\Lambda'}(p[q'])$ for any program context p[ ] and for any set Λ' s.t. $MS(p[q]) \cup MS(p[q']) \subseteq \Lambda'$.

For our simple language with bounded parallelism, we have precisely proved that languages of observations may be composed according to derived semantic laws, and that they moreover remain in the well known class of rational languages. The outcome is two-sided. First, we obtain an observational congruence of programs under the assumption of fairness : programs p and q are observationally congruent iff they are observationally equivalent, that is $\mathrm{Obs}_\Lambda(p) = \mathrm{Obs}_\Lambda(q)$ for some Λ including $MS(p) \cup MS(q)$. Second, due to the effectiveness of the semantic calculus, and since there exists a decision procedure for the equality of infinitary rational expressions, we can affirm the following

<u>theorem 24</u>. There exists a decision procedure for the observational congruence of programs.

This result motivates further work towards the axiomatization of the observational congruence of programs under the assumption of fairness, which task is perhaps unfeasible for more general programming languages without resorting to arithmetics or to ordinals.

5. SHORT EXAMPLES

Let $p \equiv Y(x \leftarrow (\alpha,\bar{\alpha})(x,x))$, $q \equiv Y(y \leftarrow (\alpha)(y))$, $r \equiv Y(z \leftarrow (\bar{\alpha})(z))$, then the following equalities hold for sort Λ equal to $\{\alpha,\bar{\alpha}\}$.

$H_\Lambda(p) = (\emptyset,\emptyset,(\alpha+\bar{\alpha})*\chi) + (\alpha\bar{\alpha},\emptyset,(\alpha+\bar{\alpha})^\omega)$
$H_\Lambda(q) = (\emptyset,\bar{\alpha},\alpha*\chi) + (\alpha,\bar{\alpha},\alpha^\omega)$
$H_\Lambda(r) = (\emptyset,\alpha,\bar{\alpha}*\chi) + (\bar{\alpha},\alpha,\bar{\alpha}^\omega)$
$H_\Lambda(q|r) = (\alpha\bar{\alpha},\emptyset,(\alpha^\omega \| \bar{\alpha}^\omega)) = (\alpha\bar{\alpha},\emptyset,(\alpha+\bar{\alpha})*\chi) + (\alpha\bar{\alpha},\emptyset,(\alpha+\bar{\alpha})^\omega)$
$Obs_\Lambda(p) = \lfloor(\emptyset,\emptyset,(\alpha+\bar{\alpha})*\chi) + (\alpha\bar{\alpha},\emptyset,(\alpha+\bar{\alpha})^\omega)\rfloor_o$
$Obs_\Lambda(q) = \lfloor(\emptyset,\bar{\alpha},\alpha*\chi) + (\alpha\bar{\alpha},\emptyset,\alpha^\omega)\rfloor_o$
$Obs_\Lambda(r) = \lfloor(\emptyset,\alpha,\bar{\alpha}*\chi) + (\alpha\bar{\alpha},\emptyset,\bar{\alpha}^\omega)\rfloor_o$
$Obs_\Lambda(q|r) = \lfloor(\alpha\bar{\alpha},\emptyset,(\alpha^\omega \| \bar{\alpha}^\omega)\rfloor_o = \lfloor(\alpha\bar{\alpha},\emptyset,(\alpha+\bar{\alpha})*\chi) + (\alpha\bar{\alpha},\emptyset,(\alpha+\bar{\alpha})^\omega\rfloor_o$
p and $(q|r)$ are therefore not equivalent.

Let now $p = Y(x \leftarrow (\alpha,\beta)(x,x)$, $q = Y(y \leftarrow (\alpha)(y))$, $r = Y(z \leftarrow (\beta)(z))$, with $\alpha \neq \bar{\beta}$. Taking $\Lambda = \{\alpha,\beta\}$, one gets $Obs_\Lambda(p) = \lfloor U \rfloor_o$ and $Obs_\Lambda(q|r) = \lfloor V \rfloor_o$ with U and V as follows :

$V = (\emptyset,\bar{\alpha}\ \bar{\beta},(\alpha+\beta)*\chi) + (\alpha\bar{\alpha}\beta\bar{\beta},\emptyset,(\alpha^+\beta^+ + \beta^+\alpha^+)^\omega) + (\beta\bar{\beta},\bar{\alpha},(\alpha+\beta)*\beta^\omega) + (\alpha\bar{\alpha},\bar{\beta},(\alpha+\beta)*\alpha^\omega)$

$U = (\emptyset,\bar{\alpha}\ \bar{\beta},(\alpha+\beta)*\chi) + (\alpha\bar{\alpha}\beta\bar{\beta},\emptyset,(\alpha+\beta)^\omega) + (\beta\bar{\beta},\bar{\alpha},(\alpha+\beta)*\beta^\omega) + (\alpha\bar{\alpha},\bar{\beta},(\alpha+\beta)*\alpha^\omega)$

As a consequence, $(\alpha\bar{\alpha}\beta\bar{\beta},\emptyset,\alpha^\omega) \in \lfloor U \rfloor_o \setminus \lfloor V \rfloor_o$, which shows that parallelism cannot be reduced to sequential non-determinism.

REFERENCES

[AuB82] Austry,D. and Boudol,G. Algèbre de processus et synchronisation. (private communication).

[CFM82] Castellani,I., Franceschi,P. and Montanari,U. Labelled event structures : a model for observable concurrency. IFIP TC-2 Working Conference, Garmisch-Partenkirchen, 1982.

[Da80] Darondeau,Ph. Processus non séquentiels et leurs observations en univers non centralisé. in LNCS 83, 1980.

[Da82] Darondeau, Ph. An enlarged definition and complete axiomatization of observational congruence of finite processes. in LNCS 137, 1982.

[Ei74] Eilenberg,S. Automata, Languages and Machines, Vol. 1. Academic Press ed.

[He80] Hennessy,M. and Milner,R. On observing non determinism and concurrency. in LNCS 85, 1980.

[He80] Hennessy,P. and Plotkin,G. A term model for CCS. in LNCS 88, 1980.

[Ho81] Hoare,C.A.R., Brookes,S.D., and Roscoe,A.D. A theory of communicating sequential processes. Technical Monograph PRG-16, Computing Laboratory, University of Oxford, 1981.

[KuR82] Kuiper,R. and de Roever,W.P. Fairness assumptions for CSP in a temporal logic framework. IFIP TC-2 Working Conference, Garmisch-Partenkirchen, 1982.

[Mi80] Milner,R. A calculus of communicating systems. LNCS 92, 1980.

[Mi82] Milner,R. Calculi for synchrony and asynchrony. CSR-104-82, Computer Science Department, Edinburgh, 1982.

[Pa80] Park, D. On the semantics of fair parallelism. in LNCS 86, 1980.

[P181] Plotkin,G. A structural approach to operational semantics. Daimi FN-19, Computer Science Department, Aarhus University, 1981.

[Wi80] Winskel,G. Events in computation. PhD Thesis, CST-10-80, Edinburgh, 1980.

APPENDIX 1 : MORE ON INFINITARY RATIONAL LANGUAGES.

The present appendix extends the classical resolution of systems of linear equations over rational languages to the case of infinitary rational languages, and gives a proof of the existence and computability of the least and greatest solutions of such extended systems (as stated by proposition 4.1.13 in the body of the paper). Are also provided additional definitions and propositions about infinitary rational languages, needed later on for proving the computability of the parallel composition operation defined in 4.1.14 . For the sake of clarity, the latter task will be carried out in a separate appendix.

First, we recall the basic definitions and properties of infinite words and infinitary (rational) languages.

Words

Let Z be an alphabet, each element of Z is called a letter. Z^* denotes the set of finite sequences of letters ; such a sequence is called a (finite) word, say f, and $|f|$ is the length of the word f.
Z^ω denotes the set of countably infinite sequences of letters ; elements of Z^ω are called (infinite) words also.
Z^∞ is the union $Z^* \cup Z^\omega$ of the sets of finite and infinite words over Z. That set may be equipped with a concatenation operation defined as follows :

- $\forall f \in Z^\omega, \forall g \in Z^\infty$: $fg = f$
- $\forall f \in Z^*, \forall g \in Z^\infty, \forall i \in \mathbb{N}$:

. $1 \leqslant i \leqslant |f| \Rightarrow fg(i) = f(i)$

. $i > |f| \Rightarrow fg(i) = g(i - |f|)$

(where f(i) is the $i^{\underline{th}}$ letter of f according to the functional definition of sequences).
Clearly, the concatenation operation is associative and admits $\mathbb{1}$ - the empty word - as a neutral element.

The following notations will be used in the sequel.
- for any f in Z^∞ and for any integer n, $f\langle n\rangle$ is the longest left factor ϕ of f with length $|\phi|$ less than or equal to n ;
- for any f in Z^∞ and for any integer n, $f > n$ is the unique word in Z^∞ such that $f = f\langle n\rangle\, f > n$.

Languages

A subset L of Z^∞ is called an (infinitary) language, and we put $L^{fin} = L \cap Z^*$ and $L^{inf} = L \cap Z^\omega$ for any $L \subseteq Z^\infty$. The concatenation operation is naturally extended from Z^∞ to $\mathcal{P}(Z^\infty)$, the set of subsets of Z^∞ , according to the definition :

$\forall A, B \subseteq Z^\infty$: $AB = \{fg \mid f \in A, g \in B\}$.

The following properties obviously hold :

$\forall A \subseteq Z^{\infty} : A.\emptyset = \emptyset.A = \emptyset$,
$\forall A \subseteq Z^{\omega}, \forall B \neq \emptyset : AB = A$.

Using the additive notation for the set union, we can define the star-operation over $\mathcal{P}(Z^{\infty})$ by
$\forall A \subseteq Z^{\infty} : A^{*} = \sum_{n\geq 0} A^{n}$,
and the omega-operation over $\mathcal{P}(Z^{\infty})$ by

$\forall A \subseteq Z^{\infty} : A = \{\mathbb{1}\} \Rightarrow A^{\omega} = \{\mathbb{1}\}$
$\forall A \subseteq Z^{\infty} : A \neq \{\mathbb{1}\} \Rightarrow A^{\omega} = (A \setminus \{\mathbb{1}\})^{\omega}$
$\forall A \subseteq Z^{\infty} : \mathbb{1} \notin A \Rightarrow A^{\omega} = \{f_0 \in Z^{\omega} \mid$
$\exists (f_i)_{i\geq 1}, f_i \in Z^{\omega}, \exists (a_i)_{i\geq 1}, a_i \in A :$
$(\forall i \geq 1)\ (a_i f_i = f_{i-1})\}$

Roughly speaking, if $A \neq \{\mathbb{1}\}$ then

$A^{\omega} = \{a_1 a_2 \ldots a_n \ldots \mid (\forall n)\ (a_n \in A \setminus \{\mathbb{1}\})\}$.
As a consequence $A = A^{\omega}$ if and only if
$A = \{\mathbb{1}\}$ or $A \subseteq Z^{\omega}$.

Rational languages

According to (Ei 74) , we define the set of the infinitary rational languages over Z as the least family $Rat(Z^{\infty})$ of subsets of Z^{∞} which contains the finite subsets of Z^{*} and is closed under concatenation, set union, star-operation and omega-operation (a re-statement of definition 4.1.7) .

In the above referenced volume, Eilenberg gives a rigorous proof of an important characterization of rational sets, originally due to Mac Naughton :

Proposition 1 Let L be a language over Z, L is a rational set of Z^{∞} if and only if there exist finitary rational languages of Z^{*}, say B, B_i, C_i $1 \leq i \leq n$ such that L is equal to $B + \sum_{i=1}^{n} B_i\ (C_i)^{\omega}$.
(a re-statement of proposition 4.1.8).

Clearly, for any rational set L of Z^{∞} , it is always possible to find finitary rational languages B, B_i, C_i such that $L^{fin} = B$ and $L^{inf} = \sum_{i=1}^{n} B_i\ (C_i)^{\omega}$.

The remaining of the appendix is dedicated to the study of systems of linear equations over rational sets of Z^{∞}. Consider the linear equation $X = AX + C$ where A and C are subsets of Z^{*} and respectively verify $\mathbb{1} \notin A$ and $C \neq \emptyset$. It is well known that the above equation has a unique solution in Z^{*}, namely the set $A^{*}C$, but it is also obvious that several solutions may exist in Z^{∞}. We shall examine below that set of possible solutions for generalized equations $X = AX + C$ where A and C are subsets of Z^{∞}. Several cases will be considered, according to the verification or non verification of properties $\mathbb{1} \in A$ and $C = \emptyset$. Next coming lemmas show that the solutions of a generalized equation $X = AX + C$ can always be deduced from the solutions of a simpler equation $X = A'X + C'$, $A' \subseteq Z^*$, whose coefficient A' and constant C' are rational if A and C are rational.

Lemma 2 Let (1) $X = AX + C$ be a linear equation over $\mathcal{P}(Z^\infty)$ and assume that C is a non empty set, then the set of solutions of (1) is equal to the set of solutions of the equation (1') $X = A^{fin} X + (A^{inf} + C)$.

Proof Since C is not empty, neither (1) nor (1') admits the empty set as a possible solution.

- Let L be a solution of (1), then the following equalities hold :

$$L = A\,L + C \quad (\neq \emptyset)$$
$$L = (A^{fin} + A^{inf})\,L + C$$
$$L = A^{fin}\,L + A^{inf}\,L + C$$
$$L = A^{fin}\,L + A^{inf} + C$$

- On the other hand, let L be a solution of (1'), then the following equalities hold :

$$L = A^{fin}\,L + A^{inf} + C \quad (\neq \emptyset)$$
$$A\,L + C = (A^{fin} + A^{inf})\,L + C$$
$$A\,L + C = A^{fin}\,L + A^{inf}\,L + C$$
$$A\,L + C = A^{fin}\,L + A^{inf} + C$$
$$L = A\,L + C \qquad \square$$

Lemma 3 Let (2) $X = AX$ be a linear equation over $\mathcal{P}(Z^\infty)$, then the set of the non-empty solutions of (2) is equal to the set of the non-empty solutions of (2') $X = A^{fin} X + A^{inf}$

Proof rewrite the last proof with the obvious replacements of C by $\emptyset$ and "solution" by "non-empty solution". $\square$

We can now focus on the solutions in Z^∞ of linear equations $X = AX + C$ which verify $A \subseteq Z^*$.

Lemma 4 Let $X = AX + C$ be a linear equation over $\mathcal{P}(Z^\infty)$, where A is a subset of Z^* and is free from the empty word, then there exists a least solution A^*C and a greatest solution $A^\omega + A^*C$ in Z^∞.

Proof The fact that A^*C and $A^\omega + A^*C$ are solutions of the equation is obvious. Now let L denote a particular solution of $X = AX + C$. An easy induction over the integers shows that A^nC is included in L for every n in $\mathbb{N}$, thus A^*C is included in L. In order to prove the inclusion $L \subseteq A^\omega + A^*C$, it suffices to show that any word f_0 of L^{inf} which does not belong to A^*C is an element of A^ω.

Consider such a word f_0 of L^{inf} :

- since $L = A\,L + C$ and $f_0 \notin A^*C$, there must exist two words a_1 in A and f_1 in L such that $f_0 = a_1\,f_1$;

- moreover, f_1 is in L^{inf} and does not belong to A^*C, thus the process may be iterated endlessly and, by the definition, f_0 $(=a_1a_2\ldots)$ is an element of A^ω □

Lemma 5 Let $X = AX + C$ be a linear equation over $\mathcal{P}(Z^\infty)$, where A is a subset of Z^* and contains $\mathbb{1}$, then there exist a least solution A^*C and a greatest solution Z^∞ in Z^∞

Proof it is obvious. □

To sum up the above lemmas 2 to 5, one may state the following

Proposition 6 Let $X = AX + C$ be a linear equation over $\mathcal{P}(Z^\infty)$. If A and C are rational sets of Z^∞, then the extremal solutions of $X = AX + C$ are rational languages. The least solution is always A^*C. The greatest solution is $A^\omega + A^*C$ if $\mathbb{1} \notin A$, or else is Z^∞.

Proof it is again obvious. □

The remaining of the appendix extends the above results from equations to systems of linear equations over $\mathcal{P}(Z^\infty)$, which systems always have least and greatest fixpoints. Only two special kinds of linear systems will be considered, namely the y-standard and Y-standard systems according to the

Definition 7 Let S denote the following system of linear equations over $\mathcal{P}(Z^\infty)$:

$$S \equiv \{X_i = (\sum_{j=1}^{n} A_{ij} X_j) + C_i \mid 1 \leqslant i \leqslant n\} \quad \text{, then}$$

- S is y-standard iff

$\forall i, \forall j : A_{ij} \in Rat(Z^*)$ and $\forall i : C_i \in (Rat(Z^\infty) \setminus \emptyset)$

- S is Y-standard iff

$\forall i, \forall j : A_{ij} \in Rat(Z^*)$ and $\forall i : C_i \in Rat(Z^\infty)$.

Let us consider first y-standard systems. Letting S as in definition 7, S may be rewritten in the matrix style into $X = AX + C$, where A is the n x n matrix with elements A_{ij}, and X, C are n-vectors. The least fixed point of S may still be written A^*C as it was the case for a simple equation, since it is possible to define the star-operation upon language - matrices . Let y(S) denote the least fixed point of S, it is obvious that the elements $y_i(S)$ of that vector are rational languages which can be effectively computed by stepwise resolution of the simple equations $X_i = A_iX + C_i$ in any order.

Now turning to Y-standard systems, our intend is to establish similar results for their greatest fixed points. Unhappily, some complications arise : although the matrix-style notation can still be used for a Y-standard system S, it is not possible to identify the greatest fixed point Y(S) of S with $A^\omega + A^*C$, since the ω-operator has no adequate extension on language matrices. (Notice that infinitary expressions built upon vectors and matrices would not obey the ordinary laws of infinitary expressions if A^ω was defined as $Y(X = AX)$!).

Lemma 8 Let S be the following Y-standard system made out of two linear equations over $\mathcal{P}(Z^\infty)$:

$$S \left\{ \begin{array}{ll} X = AX + BY + C & (1) \\ Y = DX + EY + F & (2) \end{array} \right.$$

Call (X_1, Y_1) the greatest solution obtained in the following way : solve (1) for X , then substitute the result in (2) and solve (2), last substitute the solution of (2) for Y in the solution of (1).
Similarly, let (X_2, Y_2) denote the greatest solution obtained by successively solving the equations in the converse order. If neither A nor E contains $\mathbb{1}$, then (X_1, Y_1) and (X_2, Y_2) are vectors of rational languages and are both equal to the greatest fixed point Y(S) of S.

Proof Simple calculi lead to the following statements :

$$\left\{ \begin{array}{l} X_1 = A^\omega + A^* (BY_1 + C) \\ Y_1 = (DA^*B + E)^\omega + (DA^*B + E)^* \, [D(A^\omega + A^*C) + F] \end{array} \right.$$

$$\left\{ \begin{array}{l} X_2 = (BE^*D + A)^\omega + (BE^*D + A)^* \, [B(E^\omega + E^*F) + C] \\ Y_2 = E^\omega + E^* (DX_2 + F) \end{array} \right.$$

By separation of the finitary and infinitary parts of these rational languages, one obtains :

$$\left\{ \begin{array}{l} X_1^{fin} = A^* (B(DA^*B + E)^* (DA^* C^{fin} + F^{fin}) + C^{fin}) \\ Y_1^{fin} = (DA^*B + E)^* (DA^* C^{fin} + F^{fin}) \\ X_2^{fin} = (BE^*D + A)^* (BE^*F^{fin} + C^{fin}) \\ Y_2^{fin} = E^*(D(BE^*D + A)^*(BE^*F^{fin} + C^{fin}) + F^{fin}) \end{array} \right.$$

$$\left\{ \begin{array}{l} X_1^{inf} = A^\omega + A^*(B\,[(DA^*B + E)^\omega + (DA^*B + E)^* (D(A^\omega + A^* C^{inf}) + F^{inf})] + C^{inf}) \\ Y_1^{inf} = (DA^*B + E)^\omega + (DA^* B + E)^* (D(A^\omega + A^* C^{inf}) + F^{inf}) \\ X_2^{inf} = (BE^*D + A)^\omega + (BE^*D + A)^* (B(E^\omega + E^* F^{inf}) + C^{inf}) \\ Y_2^{inf} = E^\omega + E^* (D\,[(BE^*D + A)^\omega + (BE^*D + A)^* (B(E^\omega + E^* F^{inf}) + C^{inf})] + F^{inf}) \end{array} \right.$$

The equalities $X_1^{fin} = X_2^{fin}$ and $Y_1^{fin} = Y_2^{fin}$ are easily verified, since (X_1^{fin}, Y_1^{fin}) and (X_2^{fin}, Y_2^{fin}) are both equal to the least fixed point $\mathbf{y}(S^{fin})$ of the following system S^{fin} of linear equations over $\mathcal{P}(Z^*)$:

$$S^{fin} \left\{ \begin{array}{l} X' = AX' + BY' + C^{fin} \\ Y' = DX' + EY' + F^{fin} . \end{array} \right.$$

In order to prove $(X_1, Y_1) = (X_2, Y_2)$, it is now enough to show that X_1^{inf} and X_2^{inf} are equal, since the equality $Y_1^{inf} = Y_2^{inf}$ then follows by the consideration of symmetry. To make the task easier, we rewrite the characteristic equations of X_1^{inf} and X_2^{inf} into

$$\begin{aligned} X_1^{inf} = {} & (A^*B(DA^*B + E)^* D + \mathbb{1})A^\omega + A^*B\,(DA^*B + E)^\omega \\ & + A^*(B(DA^*B + E)^* DA^* + \mathbb{1})\, C^{inf} \\ & + A^*B(DA^*B + E)^* F^{inf} \end{aligned}$$

$$\begin{aligned} X_2^{inf} = {} & (BE^*D + A)^\omega + (BE^*D + A)^* BE^\omega \\ & + (BE^*D + A)^* C^{inf} \\ & + (BE^*D + A)^* BE^* F^{inf} \quad . \end{aligned}$$

We leave to the reader to convince himself that the following equalities hold :

$$A^*(B(DA^* B + E)^* DA^* + \mathbb{1}) = (A^*BE^*D)^* A^* = (BE^*D + A)^*$$

$$(BE^*D + A)^* BE^* = A^*B(E^*DA^*B)^* E^* = A^*B\,(DA^*B + E)^*$$

Thus, to prove $X_1^{inf} = X_2^{inf}$, it is sufficient to establish the equality (1) :

$$(BE^*D + A)^* A^\omega + A^*B(DA^*B + E)^\omega = (BE^*D + A)^\omega + (BE^*D + A)^* BE^\omega.$$

Since $U^\omega = U^* U^\omega$ and $(U + V)^\omega = (U + V)^* (U^\omega + V^\omega + (U^+ V^+)^\omega)$ for any U and V in $Rat(Z^+)$, (1) is equivalent to each of the following equalities (2), (3) and (4) :

$$(2) \begin{cases} (BE^*D + A)^* A^\omega + A^*B(DA^*B + E)^* (DA^*B + E)^\omega \\ = (BE^*D + A)^\omega + (BE^*D + A)^* BE^\omega \end{cases}$$

$$(3) \begin{cases} (BE^*D + A)^* A^\omega + (BE^*D + A)^* BE^*(DA^*B + E)^\omega \\ = (BE^*D + A)^\omega + (BE^*D + A)^* BE^\omega \end{cases}$$

$$(4) \begin{cases} (BE^*D + A)^*(A^\omega + BE^*(DA^*B)^\omega + BE^*E^\omega + BE^*((DA^*B)^+E^+)^\omega) \\ = (BE^*D + A)^*(A^\omega + (BE^*D)^\omega + ((BE^*D)^+ A^+)^\omega + BE^\omega) \end{cases}$$

The truth of (4) follows from the truth of the equality

$$(5) \begin{cases} (BE^*D + A)^* (BE^*(DA^*B)^\omega + BE^*((DA^*B)^+ E^+)^\omega) \\ = (BE^*D + A)^* ((BE^*D)^\omega + ((BE^*D)^+A^+)^\omega) \end{cases}$$

The verification of (5) makes no difficulty and is left to the reader.

One has therefore proved that (X_1, Y_1) equals (X_2, Y_2). As a consequence, one may write down the equalities

(6) $X_1 = A^\omega + A^* (BY_1 + C)$ and

(7) $Y_1 = E^\omega + E^* (DX_1 + F)$.

There still remains to prove that (X_1, Y_1) is equal to the greatest fixed point $Y(S)$ of S, which fact we establish below by taking advantage of (6) and (7).

Put $Y(S) = (\overline{X}, \overline{Y})$. From (6) and (7), one draws :

(8) $(\overline{X} \setminus X_1) \subseteq A^*B\ (\overline{Y} \setminus Y_1)$

(9) $(\overline{Y} \setminus Y_1) \subseteq E^*D\ (\overline{X} \setminus X_1)$.

$((\overline{X} \setminus X_1), (\overline{Y} \setminus Y_1))$ is therefore a lower bound of the fixed point $Y(\{Z_1 = A^*BZ_2,\ Z_2 = E^*DZ_1\})$ which obviously amounts to the vector $((A^*B\ E^*D)^\omega, (E^*D\ A^*B)^\omega)$.

From (8) and (9), one can infer :

(10) $\overline{X} \subseteq X_1 + (A^*B\ E^*D)^\omega$

(11) $\overline{Y} \subseteq Y_1 + (E^*D\ A^*B)^\omega$.

Now, by the first series of statements at the beginning of the proof, one has the inclusions

$(BE^*D + A)^\omega \subseteq X_1$ and $(DA^*B + E)^\omega \subseteq Y_1\ (= Y_2)$.

Thus finally $\overline{X} \subseteq X_1$ and $\overline{Y} \subseteq Y_1$, which implies $(X_1, Y_1) = Y(S)$ since $(\overline{X}, \overline{Y})$ is the greatest fixed point of S. □

Lemma 9 All the properties which have been stated in the above lemma 8 remain valid when either A or E or both contain the empty word $\mathbb{1}$.

Proof Suppose first that both A and E contain $\mathbb{1}$, then $(X_1, Y_1) = (Z^\infty, Z^\infty) = (X_2, Y_2)$. Now consider the case where $\mathbb{1} \in (A + E) \setminus (A \cap E)$. For the reason of symmetry, one can freely assume $\mathbb{1} \in A$. The following equalities are then verified :

$(X_1, Y_1) = (Z^\infty, E^\omega + E^*(DZ^\infty + F)) = (X_2, Y_2)$. It is enough to notice that $E^\omega + E^*(DZ^\infty + F)$ is the upper bound of the set made out of the greatest solutions of $Y = EY + (DX + F)$ for X ranging over $\mathcal{P}(Z^\infty)$. □

We can now restate proposition 4.1.13 into the more precise

Proposition 10 Let S be the following Y-standard system of linear equations :

$$S \equiv \{X_i = (\sum_{j=1}^{n} A_{ij} X_j) + C_i \mid 1 \leqslant i \leqslant n\}.$$

The greatest fixed point Y(S) of S is a vector of rational languages and can be computed by stepwise resolution of the equations in any order. More precisely, given i in [1,n], Y(S) may be computed according to the following process :

- solve the $i^{\underline{th}}$ equation for X_i, giving

$$\tilde{X}_i = (A_{ii})^\omega + (A_{ii})^* ((\sum_{j \neq i} A_{ij} X_j) + C_i) + (A_{ii} \cap \{\mathbb{1}\})\ Z^\infty$$

- substitute $\tilde{X}_i$ for X_i in the remaining equations, giving a new Y-standard system S_i with (n-1) equations
- solve S_i for X_j's, $j \neq i$
- finally substitute the resulting values for X_j's in $\tilde{X}_i$.

<u>Proof</u> By induction on n. Proposition 6 and lemma 9 give the basis of the induction. Now consider $n > 2$ and suppose that the proposition is valid until n-1.

Take arbitrary different values k and l in $[1,n]$. Let expressions $\tilde{X}_k$, $\tilde{X}_l$ and Y-standard systems S_k, S_l be defined as in the proposition. According to the induction hypothesis, the greatest fixed points $Y(S_k)$ and $Y(S_l)$ can be computed by stepwise resolution of the corresponding systems S_k, S_l in any order of the equations. In particular, one is free to solve first the equations with left members X_l in S_k resp. X_k in S_l, giving solutions $\approx X_l$ resp. $\approx X_k$. By the way, the two-dimensional system $S' \equiv \{X_k = \sum_{j=1}^{n} A_{kj} X_j + C_k, X_l = \sum_{j=1}^{n} A_{lj} X_j + C_l\}$

has been solved twice, and the induction hypothesis tells us that the greatest fixed point $Y(S')$ has been reached each time. More precisely, let $Y(S') = (\hat{X}_k, \hat{X}_l)$, then the following equalities hold :

(1) $\hat{X}_k = \overset{\approx}{X}_k = \tilde{X}_k\ [\overset{\approx}{X}_k \,/\, X_k]$

(2) $\hat{X}_l = \overset{\approx}{X}_l = \tilde{X}_l\ [\overset{\approx}{X}_l \,/\, X_l]$

Now let S_{kl} and S_{lk} be the (n-2) dimensional systems got from S_k resp. S_l by substituting $\overset{\approx}{X}_l$ for X_l, resp. $\overset{\approx}{X}_k$ for X_k, and then dropping the equation with left member X_l resp. X_k. By the equalities (1) and (2), S_{kl} and S_{lk} are equivalent systems and, again by the induction hypothesis, their common greatest fixed point can be computed by stepwise resolution of the equations in any order. Denote that fixed point as $(\overline{X}_1, \ldots, \overline{X}_j, \ldots, \overline{X}_n)$ with $k \neq j \neq l$.

Define $\overline{X}_k = \overset{\approx}{X}_k\ \overrightarrow{[\overline{X}_j \,/\, X_j]}$ and $\overline{X}_l = \overset{\approx}{X}_l\ \overrightarrow{[\overline{X}_j \,/\, X_j]}$.

From the equalities (1) and (2), it is not difficult to see

$\overline{X}_k = \tilde{X}_k\ \overrightarrow{[\overline{X}_j \,/\, X_j]}$, $\overline{X}_l = \tilde{X}_l\ \overrightarrow{[\overline{X}_j \,/\, X_j]}$.

Hence, we have proved that the solution $(\overline{X}_1,\ldots,\overline{X}_n)$, which is obviously a n-vector of rational languages, does not depend on the order in which the equations are solved.

There remains to prove that $(\overline{X}_1, \ldots, \overline{X}_n)$ is the greatest fixed point $Y(S)$ of S.

Put $Y(S) = (\overline{\overline{X}}_1, \ldots, \overline{\overline{X}}_n)$.

Let us show that the following equation (3) is verified for every i in $[1,n]$:

(3) $\overline{\overline{X}}_i = A_{ii}^{\omega} + A_{ii}^{*}\,(\sum_{j \neq i} A_{ij}\,\overline{\overline{X}}_j + C_i) + (A_{ii} \cap \{1\})\,Z^{\infty}$

In order to establish the above fact, suppose for a moment that there exists some i in $[1,n]$ for which (3) does not hold.

Define $\check{X}_i = A_{ii}^{\omega} + A_{ii}^{*} \left(\sum_{j \neq i} A_{ij} \overline{\overline{X}}_j + C_i \right) + (A_{ii} \cap \{\mathbb{1}\})\, Z^{\infty}$, then certainly $\overline{\overline{X}}_i \subseteq \check{X}_i$, and define $\check{X}_j = \overline{\overline{X}}_j$ for $j \neq i$. The following inclusions are then obvious :

(4) $\check{X}_i \subseteq A_{ii} \check{X}_i + \sum_{j \neq i} A_{ij} \check{X}_j + C_i$

(5) $\check{X}_p \subseteq A_{pi} \check{X}_i + \sum_{j \neq i} A_{pj} \check{X}_j + C_p$ for $p \neq i$

The inclusions (4) and (5) make it clear that $(\check{X}_1, \ldots , \check{X}_n)$ is a pre-fixed-point of S. Since Y(S) is also the greatest pre-fixed-point of S, $\overline{\overline{X}}_i$ cannot be less than $\check{X}_i$, thus $\overline{\overline{X}}_i = \check{X}_i$ and (3) holds for every i in $[1,n]$.

In particular, (3) holds for $i = 1$, which shows that $(\overline{\overline{X}}_2, \ldots , \overline{\overline{X}}_n)$ is a solution of S_1 since $\tilde{X}_1$ equals $A_{11}^{\omega} + A_{11}^{*} \left(\left(\sum_{j \neq 1} A_{1j} X_j \right) + C_1 \right) + (A_{11} \cap \{\mathbb{1}\})\, Z^{\infty}$.

Now, by the induction hypothesis , $(\overline{X}_2 , \ldots , \overline{X}_n)$ is the greatest fixed point of S_1, thus $\overline{\overline{X}}_j \subseteq \overline{X}_j$ for $j > 1$, and it follows that $\overline{\overline{X}}_1$ is also included in $\overline{X}_1$ by the equality $\overline{X}_1 = \tilde{X}_1 \overrightarrow{(\overline{X}_j / X_j)}$.

Since $(\overline{\overline{X}}_1, \ldots , \overline{\overline{X}}_n)$ is the greatest fixed point of S, and since $(\overline{X}_1, \ldots , \overline{X}_n)$ is also a solution of S, one can finally conclude $(\overline{X}_1, \ldots, \overline{X}_n) = Y(S)$. $\square$

<u>Remarks</u>

1- The result stated in proposition 10 is a tight analogue of the well known result which applies to the finitary rational case.

2- Henceforth, given a linear system $X = AX + C$ written in matrix style notation, $Y\,(X = AX + C)$ and $\mathbf{y}\,(X = AX + C)$ will be used to denote its greatest resp. least fixed point.

3- It is easy to see that the i^{th} component of $Y(X = AX + C)$ may be written $[R^{\infty}(A)]_i + \sum_{j=1}^{n} [A^{*}]_{ij}\, C_j$, where $R^{\infty}(A)$ is a n-vector of languages in $\mathrm{Rat}(Z^{\omega})$ and does not depend on C.

APPENDIX 2 : MORE ON THE PARALLEL COMPOSITION OF LANGUAGES.

The goal of the appendix is to characterize the parallel composition of languages by a combined use of least and greatest fixed points of systems of linear equations.

For technical reasons, we shall in fact introduce two different kinds of operations over infinitary rational languages : a parallel composition and a quasi-parallel composition respectively denoted by $|$ and $\|$.

The finite alphabet Z used in the sequel is assumed to be the disjoint union of two subsets Z' and $\overline{Z}'$, related to each other by a pair of reciprocal bijections named overbar. We shall also assume of special symbol τ not in Z and an associated erasing morphism ψ from $(Z \cup \tau)^*$ to Z^* such that

$\forall z \in Z \quad \psi(z) = z$ and $\psi(\tau) = 1$

Definition 1 Let f and g be two words of Z^*, their parallel composition $f|g$ is the subset of Z^* given by

- if $g = 1$ then $f|g = \{f\}$
- if $f = 1$ then $f|g = \{g\}$
- otherwise, let $f = xf'$ and $g = yg'$ with $x,y \in Z$ and $f', g' \in Z^*$ then

$$f|g = \begin{cases} x\,(f'|g) + y(f|g') \text{ if } \overline{x} \neq y \\ x\,(f'|g) + \overline{x}(f|g') + (f'|g') \text{ if } \overline{x} = y \end{cases}$$

Definition 2 Let f and g be two words of Z^ω, their parallel compositon $f|g$ is the subset of Z^∞ defined by

$\forall h \in Z^\infty \quad h \in f|g \quad$ iff $\forall i \in \mathbb{N} \quad \exists f_i, g_i \in Z^+ \ \exists h_i \in Z^*$ s.t.

(i) $f = f_0 f_1 \dots f_n \dots , \ g = g_0 g_1 \dots g_n \dots$
(ii) $h = h_0 h_1 \dots h_n \dots$
(iii) $h_i \in f_i | g_i$

Definition 3 Let f be in Z^* and g be in Z^ω ,their parallel composition $f|g$ is the subset of Z^ω defined by

$$f|g = \sum_{n \geq 0} (f|g\,\langle n\rangle)\,.\,(g>n)$$

Definition 4 Let f and g be in Z^*, their quasi-parallel composition $f\|g$ is the subset of $(Z \cup \tau)^*$ given by

- if $g = 1$ then $f\|g = \{f\}$
- if $f = 1$ then $f\|g = \{g\}$
- otherwise let $f = xf'$, $g = yg'$ with $x,y \in Z$ and $f',g' \in Z^*$ then

$$f \Vert g = \begin{cases} x\,(f' \Vert g) + y\,(f \Vert g') & \text{if } \bar{x} \neq y \\ x\,(f' \Vert g) + \bar{x}\,(f \Vert g') + \tau(f' \Vert g') & \text{if } \bar{x} = y \end{cases}$$

<u>Definition 5</u> Let f and g be in Z^{ω}, their quasi-parallel composition $f \Vert g$ is the subset of $(Z \cup \tau)^{\omega}$ defined by

$$\forall h \in (Z \cup \tau)^{\omega} \quad h \in f \Vert g \text{ iff } \forall i \in \mathbb{N} \; \exists f_i, g_i \in Z^{+} \, \exists h_i \in (Z \cup \tau)^{+} \text{ s.t.}$$

$$\text{(i) } f = f_0 f_1 \ldots f_n \ldots, \; g = g_0 g_1 \ldots g_n \ldots$$

$$\text{(ii) } h = h_0 h_1 \ldots h_n \ldots$$

$$\text{(iii) } h_i \in f_i \Vert g_i$$

<u>Definition 6</u> Let f be in Z^{*} and g be in Z^{ω}, their quasi-parallel composition $f \Vert g$ is the subset of $(Z \cup \tau)^{\omega}$ defined by

$$f \Vert g = \sum_{n \geq 0} (f \Vert g\langle n \rangle)\,(g > n)$$

Following these definitions the parallel composition may be viewed as a fair shuffle with erasing rendez-vous whereas the quasi-parallel composition appears as a fair shuffle with flagged rendez-vous.

<u>Proposition 7</u> Let f and g be in Z^{∞} then $\psi(f \Vert g) = f | g$

<u>proof</u> obvious. □

<u>Lemma 8</u> Let f and g be in Z^{∞}, if h belongs to $f|g$ (resp. $f \Vert g$) then, for any integer p, $p \leq |h|$, there exists at least one pair of integers m and n such that $h\langle p \rangle$ belongs to $f\langle n \rangle | g\langle m \rangle$ (resp. $f\langle n \rangle \Vert g\langle m \rangle$).

<u>proof</u> obvious. □

<u>Proposition 9</u> Let f and g be in Z^{∞}, then the following identities hold :

$$f|g = \sum_{n,m>0} (f\langle n \rangle | g\langle m \rangle)(f > n | g > m)$$

$$f \Vert g = \sum_{n,m>0} (f\langle n \rangle \Vert g\langle m \rangle)(f > n \Vert g > m)$$

<u>proof</u> : This statement is a corollary of lemma 8 with the convention (see appendix 1) : if f is a finite word and n is an integer greater than $|f|$ then $f\langle n \rangle = f$ and $f > n = \mathbb{1}$. □

<u>Proposition 10</u> The parallel composition is an associative and commutative operation over Z^{∞}. The quasi-parallel composition is a commutative operation over Z^{∞}.

Definition 11 The parallel composition $L|L'$ of two languages, L and L', of Z^∞ is defined as $L|L' = \{f|g \mid f \in L, g \in L'\}$.

The quasi-parallel composition $L \| L'$ of two languages, L and L', of Z^∞ is defined as $L \| L' = \{f \| g \mid f \in L, g \in L'\}$.

Proposition 12 The parallel composition of languages is a commutative and associative operation. The quasi-parallel composition of languages is a commutative operation. Both composition operations distribute over the set union.

It should be clear from proposition 9 that our current definition of the parallel composition agrees with the earlier statements of section 3.3.

In order to proceed we need to introduce the notions of residuals and co-residuals.

Definition 13 Let L be a subset of Z^∞ and let f be a word of Z^*, the residual of L by f is the language defined by

$$L\backslash f = \{g \in Z^\infty | fg \in L\}$$

Definition 14 Given two languages L and L', let L' be the residual of L by some word f, then the co-residual of L' w.r.t. L is the language K defined by

$$K = \{g \in Z^* \mid L\backslash g = L'\}$$

Proposition 15 Let L be an infinitary rational language, then L admits a finite number of residuals.

proof : Any rational language L may be written as

$$L = B + \sum_{1 \leqslant i \leqslant n} B_i \, C_i^\omega$$

where B, B_i, C_i are finitary rational languages. Now for any finite word f, the following identity holds

$$L\backslash f = B\backslash f + \sum_{1 \leqslant i \leqslant n} (B_i C_i^*)\backslash f \, . \, C_i^\omega$$

and it is well-known that any finitary rational language has only a finite number of residuals (it is even a characteristic property). □

corollary 16 Let L be in Rat (Z^∞), let $\{L_1, \ldots, L_n\}$ be the set of non empty residuals of L and let $\{K_1, \ldots, K_n\}$ be the set of the respective co-residuals, then the following holds

$$L = \sum_{1 \leqslant i \leqslant n} K_i \, L_i \quad \text{(residual normal form)}$$

<u>Proposition 17</u> Let L and K be in Rat (Z^∞) such that there exist B, C, B', C' in $\mathrm{Rat}(Z^*)$ satisfying

(i) $L = BC^\omega$ and $K = B'C'^\omega$

(ii) C and C' do not contain the empty word

(iii) C and C' are given in residual normal form

$$C = \sum_{1\leqslant i\leqslant n} D_i\, C_i \quad \text{and } C' = \sum_{1\leqslant j\leqslant m} D'_j\, C'_j$$

then

$$BC^\omega \| B'C'^\omega = \sum_{i,j} (BC^* D_i \| B'C'^* D'_j).(C_i C^\omega \| C'_j C'^\omega)$$

$$C_i C^\omega \| C'_j C'^\omega = \sum_{k,l} (C_i C^+ D_k \| C'_j C'^+ D_l).(C_k C^\omega \| C'_l C'^\omega)$$

<u>proof</u> : obvious from proposition 9 and corollary 16. □

From now on, we shall make precise a systematic way to compute the (quasi) parallel composition of two rational languages L_1 and L_2. Owing to the distributivity of the (quasi) parallel composition over set union, it is sufficient to study the three following cases :

(i) L_1 and L_2 are both elements of $\mathrm{Rat}(Z^*)$

(ii) L_1 is an element of (Z^*) and L_2 is an element of $\mathrm{Rat}(Z^\infty)$

(iii) L_1 and L_2 are both in $\mathrm{Rat}(Z^\omega)$

In cases (i) and (ii), both the parallel composition and the quasi-parallel composition are computed as least fixed points of linear systems ; but only the quasi-parallel composition may be computed as a greatest fixed point in case (iii).

Let us go further in the presentation of the involved system of linear equations for cases (i) and (ii). Let L_1 be an element of $\mathrm{Rat}(Z^*)$ and L_2 be an element of $\mathrm{Rat}(Z^\infty)$, let n_1 (resp. n_2) be the number of non empty residuals of L_1 (resp. L_2), and let n be equal to $n_1.n_2$. A pair of corresponding applications - $1 : \{1,\dots,n\} \to \{1,\dots,n_1\}$ and $2 : \{1,\dots,n\} \to \{1,\dots,n_2\}$- are assumed to define a one-to-one correspondence - i <—> (1(i), 2(i)) - between {1,...,n} and the cartesian product $\{1,\dots,n_1\} \times \{1,\dots,n_2\}$.

Given the above elements, let us define three n x n matrices, A,B,C : for any integers i, j in {1,...,n},

$$A_{ij} = \begin{cases} \{\tau\} & \text{iff } \exists z \in Z \text{ s.t. } L_{1(j)} = L_{1(i)} \backslash z,\ L_{2(j)} = L_{2(i)} \backslash \overline{z} \\ \emptyset & \text{otherwise} \end{cases}$$

$$B_{ij} = \{z \in Z \mid L_{1(j)} = L_{1(i)} \backslash z \ ;\ L_{2(j)} = L_{2(i)}\}$$

$$C_{ij} = \{z \in Z \mid L_{1(j)} = L_{1(i)} \ ;\ L_{2(j)} = L_{2(i)} \backslash z\}$$

Let us last define the n vector D : for any integer i in $\{1,\ldots,n\}$,

$$D_i = \begin{cases} L_{1(i)} + L_{2(i)} & \text{if } \mathbb{1} \in L_{1(i)} \text{ and } \mathbb{1} \in L_{2(i)} \\ L_{2(i)} & \text{if } \mathbb{1} \in L_{1(i)} \text{ and } \mathbb{1} \notin L_{2(i)} \\ L_{1(i)} & \text{if } \mathbb{1} \notin L_{1(i)} \text{ and } \mathbb{1} \in L_{2(i)} \\ \emptyset & \text{if } \mathbb{1} \notin L_{1(i)} \text{ and } \mathbb{1} \notin L_{2(i)} \end{cases}$$

We shall name $S\boldsymbol{|}$ the following system of linear equations

$$(S\boldsymbol{|}) \quad X = [(B^* + C^*)\, A + B^+C + C^+B] \ \ X + (B^* + C^*)D$$

and $X\boldsymbol{|}$ the n-vector of languages :

$$\forall\, i \in \{1,\ldots,n\} \qquad X\boldsymbol{|}_i = L_{1(i)} \boldsymbol{|} L_{2(i)}$$

<u>lemma 18</u> For any integers i,j in $\{1,\ldots,n\}$ the following identities hold :

$$B^*_{ij} = \{f \in Z^* \mid L_{1(j)} = L_{1(i)} \backslash f,\ L_{2(j)} = L_{2(i)}\}$$

$$B^+_{ij} = \{f \in Z^+ \mid L_{1(j)} = L_{1(i)} \backslash f,\ L_{2(j)} = L_{2(i)}\}$$

$$C^*_{ij} = \{f \in Z^* \mid L_{1(j)} = L_{1(i)} \ ,\ L_{2(j)} = L_{2(i)} \backslash f\}$$

$$C^+_{ij} = \{f \in Z^+ \mid L_{1(j)} = L_{1(i)},\ L_{2(j)} = L_{2(i)} \backslash f\}$$

<u>proof</u> : By definition of the concatenation operation over language matrices and by construction of B and C. □

<u>Proposition 19</u> Let $y(S\boldsymbol{|})$ be the least fixed point of $(S\boldsymbol{|})$ then $X\boldsymbol{|}$ is equal to $y(S\boldsymbol{|})$.

<u>proof</u> : We show both inclusions $X\boldsymbol{|} \subseteq y(S\boldsymbol{|})$ and $y(S\boldsymbol{|}) \subseteq X\boldsymbol{|}$

(1) to prove $y(S\boldsymbol{|}) \subseteq X\boldsymbol{|}$, we check that $X\boldsymbol{|}$ is a fix point of $(S\boldsymbol{|})$. Or, in other words : $\forall\, i \in \{1,\ldots,n\}$

$$X\boldsymbol{|}_i = \sum_{1 \leqslant j \leqslant n} [(B^* + C^*)\, A + B^+C + C^+B]_{ij}\, X\boldsymbol{|}_j \ + \sum_{1 \leqslant j \leqslant n} (B^* + C^*)_{ij}\, D_j$$

and this equality clearly holds by lemma 18 and proposition 9.

(2) the converse inclusion $X| \subseteq y(S|)$ may be rephrased into $X| \subseteq ((B^* + C^*)A + B^+C + C^+B)^* (B^* + C^*)D$ or, in other words,
$\forall\, i \in \{1,\dots,n\}\quad X|_i \subseteq \sum_{1 \leqslant j \leqslant n} \{((B^* + C^*)A + B^+C + C^+B)^* (B^* + C^*)\}_{ij} D_j$

and this inclusion is proved by using lemma 18 and proposition 9. □

If $(S|)$ is the system

$(S|)\quad X = ((B^* + C^*)\psi A + B^+C + C^+B)X + (B^* + C^*)D$

and $X|$ the n-vector such that $X|_i$ is equal to $L_{1(i)}|L_{2(i)}$, then we get the

Proposition 20 Let $y(S|)$ be the least fixed point of $(S|)$ then $X|$ is equal to $y(S|)$.

proof analogous to the proof of proposition 19. □

Now turn to the case where both L_1 and L_2 are elements of $Rat(Z^\omega)$. It is sufficient to assume $L_1 = B_1C_1^\omega$ and $L_2 = B_2C_2^\omega$ with B_1,B_2,C_1,C_2 elements of $Rat(Z^+)$ (not containing the empty word).

We let n_1 (resp. n_2) be the number of non empty residuals of C_1 (resp. C_2), and n be equal to $n_1.n_2$; we assume corresponding applications, 1 and 2, defined as above from $\{1,\dots,n\}$ to $\{1,\dots,n_1\}$ respectively $\{1,\dots,n_2\}$. For any integer i in $\{1,\dots,n\}$, we let $K_{1(i)}$ (resp. $K_{2(i)}$) be the coresidual associated to $L_{1(i)}$ (resp. $L_{2(i)}$). Now define E, the n-vector such that, for any integer i in $\{1,\dots,n\}$, E_i is the quasi-parallel composition of $B_1C_1^*K_{1(i)}$ and $B_2C_2^*K_{2(i)}$. Last define F, the n x n matrix such that, for any integers i,j in $\{1,\dots,n\}$, F_{ij} is the quasi-parallel composition of $L_{1(i)}\ C_1^+K_{1(j)}$ and $L_{2(i)}\ C_2^+K_{2(j)}$. Clearly, $F_{ii}^*\,F_{ij}$ is included into F_{ij} for any i and j in $\{1,\dots,n\}$. We shall name **S** **the** following system of linear equations :

$$(S)\begin{cases} X = {}^tE\ X' \\ X' = F\ X' \end{cases}$$

(where X and Y are n-vector variables).

Proposition 21 Let $Y(S)$ be the greatest fixed point of (S), then $Y(S)$ is equal to the quasi-parallel composition of $B_1C_1^\omega$ and $B_2C_2^\omega$.

proof Firstly, it is clear that $Y(S)$ is equal to ${}^{t}EY(S')$, where (S') is the subsystem $X' = FX'$.
By proposition 17, a fixed-point of (S') is given by the n-vector $X'\|$ with elements, for any i in $\{1,\dots,n\}$:

$$X'\|_i = L_{1(i)}\, C_1^{\omega} \,\|\, L_{2(i)}\, C_2^{\omega}$$

Therefore the inclusions $X'\| \subseteq Y(S')$ holds.
To prove the converse inclusion, assume for a moment that there exists an integer i in $\{1,\dots,n\}$ and a word f such that f belongs to $Y(S')_i$ and does not belong to $X'\|_i$. Let us show that these assumptions lead to non sense.

As mentionned in proposition 10 of appendix 1, since F_{ii} does not contain the empty word (by definition of the quasi-parallel composition), the following equality must hold :

$$Y(S')_i = F_{ii}^{\omega} + F_{ii}^{*} \Big(\sum_{\substack{j=1 \\ j\neq i}}^{n} F_{ij}\, Y(S')_j \Big).$$

So, either f belongs to F_{ii}^{ω} or $f = f_1 f^1$ with f_1 in F_{ij} (for some j) and f^1 in $Y(S')_j$ and not in $X'\|_j$. By iterating this process one can construct a sequence $\{f_j\}(j>0)$ of words f_j in $(Z \cup \tau)^*$ such that

(i) $\forall_j \; f_j \in F_{i_{j-1}\, i_j} \quad (i_o = i)$

(ii) $f = f_1 \dots f_j \dots$

From (i), there must exist two sequences $\{g_j\}$ and $\{h_j\}$, $j > 0$, of words, $g_j \in Z^*$ and $h_j \in Z^*$, such that

(iii) $\forall_j \; f_j \in g_j \| h_j$

(iv) $g_j \in L_{1(i_{j-1})}\, C_1^{+}\, K_{1(j)}$ and $h_j \in L_{2(i_{j-1})}\, C_2^{+}\, K_{2(i_j)}$

from (iv) we draw

(v) $\forall_j \quad g_1 \dots g_j \in L_{1(i)}\, C_1^{+}\, K_{1(i_j)}$ and

$h_1 \dots h_j \in L_{2(i)}\, C_2^{+}\, K_{2(i_j)}$

Let g be the infinite word $g_1 \dots g_j \dots$ $(=\lim_{j \to +\infty} g_1 \dots g_j)$ and let h be the infinite word $h_1 \dots h_j \dots (=\lim_{j \to +\infty} h_1 \dots h_j)$.

From (iv) and (v), we know that g and h respectively belong to $L_{1(i)}\ C_1^{\omega}$ and to $L_{2(i)}\ C_2^{\omega}$; thus f belongs to $L_{1(i)}\ C_1^{\omega} \,\|\, L_{2(i)}\ C_2^{\omega}$. Since the last expression amounts to $X'\|_i$, the required contradiction has been reached. □

To compute the parallel composition of $B_1C_1^{\omega}$ and $B_2C_2^{\omega}$, it is not adequate to take the greatest fixed point of the system obtained from (S) when replacing the quasi-parallel composition operator with the parallel composition operator. But proposition 7 leads to the following

<u>Proposition 22</u> The parallel composition of $B_1C_1^{\omega}$ and $B_2C_2^{\omega}$ is the morphic image $\psi(Y(S))$ of the greatest fixed point Y(S) of (S).

To complete this appendix, the following points must be quoted.

1. Let $B_1C_1^{\omega} \| B_2C_2^{\omega}$ be equal to $\Sigma K_iL_i^{\omega}$ with K_i and L_i elements of Rat $((Z \cup \tau)^*)$; we define L'_i and L''_i such that $L_i = L'_i + L''_i$ with $L'_i \in$ Rat $((Z \cup \tau)^*)$ and $L''_i \in \mathrm{Rat}(Z^*)$; then $B_1C_1^{\omega} | B_2C_2^{\omega} = \psi(B_1C_1^{\omega} \| B_2C_2^{\omega})$

$$=\Sigma\psi(K_i)\,(\psi(L_i)^*\psi(L'_i)^{\omega}+(\psi(L'_i)^*\psi(L''_i))^{\omega})$$

2. All the constructions we have suggested for computing the (quasi) parallel composition of infinitary rational languages can be achieved in a purely syntactical way, given the rational expressions of the languages.

3. In Park's article (Pa80) , the fair shuffle of two infinite words is computed as a greatest fixed point. The same fact may be observed here as regards the quasi-parallel composition of languages, but not their parallel composition. The reason lays in the intrinsic discontinuity of the latter operation which may involve an infinite erasure of complementary letters inside the fair shuffle.

APPENDIX 3 : MORE ON HISTORIES.

The main purpose of this appendix is to show that the operational semantics of an elementary program, as defined in section 3, may be equivalently characterized as the morphic image, in a monoid of "partial histories", of the greatest fixed point of an associated system of linear equations over Z^{∞}(for Z a finite alphabet of "action records").This alternative characterization will be later used for proving the results of section 4. The proof of proposition 3.9 may also be found here.

Let us first define the monoid of partial histories and the needed morphism. Λ is assumed a fixed finite subset of M.

Definition 1 Let ρ in $\Lambda^{\omega} \cup \Lambda^{*}\chi \cup \Lambda^{*}$, ρ is a complete sequence (of responses) if it belongs to $\Lambda^{\omega} \cup \Lambda^{*}\chi$, else it is incomplete.

Definition 2 pre-H (Λ), the set of pre-histories, is the set of triples (d,δ,ρ) which verify the following conditions :

1'. $d \subseteq \Lambda \cup \overline{\Lambda}$, $\delta \subseteq \Lambda \cup \overline{\Lambda}$, $\rho \in \Lambda^{\omega} \cup \Lambda^{*}\chi \cup \Lambda^{*}$

2'.Ult $(\rho) \subseteq d$ if ρ is complete, or else Act$(\rho) \subseteq d$

3'. $d \cap \delta = \emptyset$.

Definition 3 pre-h (Λ), the set of action records, is the subset of pre-histories (d,δ,ρ) for which $\rho \in \Lambda \cup \{\chi\} \cup \{\mathbb{1}\}$.

Definition 4 P-h(Λ), the set of partial histories, is the monoid with carrier pre-H(Λ) $\cup$ $\{\mathbb{1}\}$, neutral element $\mathbb{1}$, and concatenation $\bullet$ as follows :

$(d,\delta,\rho) \bullet (d',\delta',\rho') = (d,\delta,\rho)$ if ρ is complete, or else $(d',\delta',\rho\rho')$ if ρ' is complete, or else $(d \cup d', \delta \cap \delta', \rho\rho')$.

Definition 5 K : $(\text{pre-H}(\Lambda))^{\infty} \to$ P-h (Λ) is the function s.t.

- for finite words $h_1 \dots h_k$, $K(h_1 \dots h_k) = h_1 \bullet \dots \bullet h_k$ if $k \geqslant 1$, or else $K(\mathbb{1}) = \mathbb{1}$, the neutral element of P-h(Λ)

- for infinite words $h_1\ h_2 \dots h_i \dots$, let $h_i = (d_i,\delta_i,\rho_i)$, then $K(h_1\ h_2 \dots h_i) =$ $K(h_1\ h_2 \dots h_k)$ if ρ_k is complete for some k, or else $K(h_1\ h_2 \dots h_i \dots)$ is the triple (d, δ, ρ) given by :

$$d = \lim_{i} \left(\bigcup_{j \geqslant i} d_j\right)$$

$$\delta = \lim_{i} \left(\bigcap_{j \geq i} \delta_j\right)$$

$$\rho = \rho_1 \, \rho_2 \ldots \rho_i \ldots \chi$$

Proposition 6 K is a surjective monoid homomorphism ; moreover, the restriction of K to $(\text{pre-h}(\Lambda))^{\infty}$ is also a surjective monoid homomorphism.

Proof We shall first verify that K is a total function, i.e. $K(h_1 \, h_2 \ldots h_i \ldots)$ is univocally determined and belongs to P-h (Λ). This fact is immediate for finite words $h_1 \ldots h_k$. Let $\mathscr{W}$ be the infinite word $h_1 \, h_2 \ldots h_i \ldots$, $h_i = (d_i, \delta_i, \rho_i)$ ε pre-H(Λ). Suppose that there exists k and l such that ρ_k and ρ_l are complete sequences, let $k < l$, then $h_1 \bullet h_2 \bullet \ldots \bullet h_k \bullet \ldots \bullet h_l = h_1 \bullet h_2 \bullet \ldots \bullet h_k \;\varepsilon$ P-h (Λ) by definition 4. Suppose now that ρ_i is incomplete for any i and put $\rho = \rho_1 \rho_2 \ldots \rho_i \ldots \chi$: ρ is a complete sequence. By the hypothesis $\mathrm{Act}(\rho_j) \subseteq d_j$, $\mathrm{Ult}(\rho) = \lim_i \left(\bigcup_{j \geq i} \mathrm{Act}(\rho_j)\right)$ is included in $\lim_i \left(\bigcup_{j \geq i} d_j\right)$; by the hypothesis $d_j \cap \delta_j = \emptyset$, $\lim_i \left(\bigcup_{j \geq i} d_j\right) \cap \lim_i \left(\bigcap_{j \geq i} \delta_j\right)$ is empty ; thus $K(\mathscr{W}) \;\varepsilon$ P-h(Λ).

Let us prove that K is a monoid homomorphism.
It is enough to show $K(uv) = K(u) \bullet K(v)$ for non empty words u and v in $(\text{pre-H}(\Lambda))^{\infty}$. The fact is immediate for u and v in $(\text{pre-H}(\Lambda))^{*}$. For u in $(\text{pre-H}(\Lambda))^{\omega}$, K(u) is a complete history in H(Λ), thus $K(u) \bullet K(v) = K(u)$, and the result follows by the identity $uv = u$ in $(\text{pre-H}(\Lambda))^{\infty}$. Now consider the remaining case $u = h_1 \, h_2 \ldots h_k$, $v = h_{k+1} \, h_{k+2} \ldots \;\varepsilon\; (\text{pre-H}(\Lambda))^{\omega}$.

- Suppose $(\exists\, n \leq k)\ (h_n \;\varepsilon\; H(\Lambda))$, then $K(uv) = K(h_1 \ldots h_n)$ is in H (Λ), thus $K(h_1 \ldots h_n) \bullet V = K(h_1 \ldots h_n)$ for any V in P - h(Λ). In particular, $K(u) = K(h_1 \ldots h_n) \bullet K(h_{n+1} \ldots h_k)$, whence $K(uv) = K(u) = K(u) \bullet K(v)$.

- Suppose $(\exists\, n > k)\ (h_n \;\varepsilon\; H(\Lambda))$, then $K(uv) = K(h_1 \ldots h_k \ldots h_n) = K(u) \bullet K(h_{k+1} \ldots h_n) = K(u) \bullet K(v)$.

- Suppose at last $(\nexists\, i)\ (h_i \;\varepsilon\; H(\Lambda))$. Let $K(u) = (d,\delta,\rho)$ and $K(v) = (d',\delta',\rho')$, then $K(uv) = (d',\delta',\rho\rho')$ may be verified from definition 5, whence $K(uv) = K(u) \bullet K(v)$ by definition 4.

In order to complete the proof, there remains to show that P-h(Λ) is the K-image of $(\text{pre-h}(\Lambda))^{\infty}$. Clearly, it is sufficient to verify that for any $h = (d,\delta,\rho)$ in pre-H(Λ), h equals $K(W_h)$ for some corresponding W_h in $(\text{pre-h}(\Lambda))^{\infty}$. Cases are as follows :
- if $|\rho| \leq 1$, then $K(h) = h \;\varepsilon$ pre-h(Λ).

- if $\rho = \mu_1 \dots \mu_n$, with $\mu_i \in \Lambda$ for any i, then (d,δ,μ_i) belongs to pre-h(Λ) for any i (since $\mu_i \in Act(\rho) \subseteq d$), and letting $h_i = (d,\delta,\mu_i)$, $K(h_1 \dots h_n) = h$
- if $\rho = \mu_1 \dots \mu_n \ \chi$, with $\mu_i \in \Lambda$ for any i, then $(\mu_i, \emptyset , \mu_i)$ belongs to pre-h(Λ) for any i, and letting $h_i = (\mu_i, \emptyset , \mu_i)$, one has $h = K(h_1 \dots h_n(d,\delta,\chi))$
- if ρ is the infinite sequence $\mu_1 \mu_2 \dots \mu_i \dots$, $\mu_i \in \Lambda$, then there exists some integer k such that $(\forall n \geqslant k)\ (\mu_n \in d)$ – since Ult $(\rho) \subseteq d$ –, and letting $h_i = (\mu_i , \emptyset , \mu_i)$ for $i < k$, $h_i = (d,\delta,\mu_i)$ for $i \geqslant k$, one has $h = K(h_1 h_2 \dots h_i \dots)$ □

We turn now to the solving of equational systems over sets of partial histories. Let X = A.X + C be a system of linear equations over $\mathscr{P}$(P-h(Λ)), where elements of the matrix A and vector C respectively range over $\mathscr{P}(K(\text{pre-h}(\Lambda)^+))$ and $\mathscr{P}$(P-h(Λ)). Since the above defined K is onto, one can always build an associated system of linear equations over $\mathscr{P}(\text{pre-h}(\Lambda)^\infty)$, let X' = A'X' + C', which verifies K(A'[i,j]) = A[i,j] and K(C'[i]) = C[i] for any values of the indexes. Since K is a monoid homomorphism, it is clear that K(Y(X' = A'X' + C')) does not depend on the particular choice of A' and C' in $K^{-1}(A)$ resp. $K^{-1}(C)$, and is moreover a solution of the original system (X = A.X+C). Henceforth, we let Y(X=A.X+C) denote that solution whenever the equations bear upon sets of partial histories. It is worth noting that Y(X=A.X+C) is <u>not</u> the greatest solution of X = A.X+C, at least for the inclusion of subsets. Consider for instance the elementary system of equations $S \equiv (X_1 = (\mu,\bar{\mu},\mu)\bullet X_1 + (\emptyset,\bar{\mu},\chi))$. One has $Y(S) = (\mu,\bar{\mu},\mu^\omega) + (\emptyset,\bar{\mu},\mu^*\chi)$, whereas another possible solution is $(\mu\bar{\mu},\mu,\mu^\omega) + (\mu,\bar{\mu},\mu^\omega) + (\emptyset,\bar{\mu},\mu^*\chi)$.

Equipped with the above tools, we shall supply programs with an alternative operational semantics $\mathscr{H}_\Lambda$ appealing to fixed points, before proving it equivalent to H_Λ . Definitions below make use of a new set $\mathscr{X}$ of meta-variables X_i in one-one correspondance with the syntactic variables $x_i \in X$.

<u>Definition 7</u> For t a term or a program of sort Λ which is not the result of a flow-operation, $\mathscr{H}_\Lambda(t)$ is inductively defined by the following set of rules, where X_i's range over $\mathscr{P}$(P-h(Λ)) :

$$\mathscr{H}_\Lambda(\text{NIL}) = (\emptyset, \Lambda\cup\bar{\Lambda},\chi)$$

$$\mathscr{H}_\Lambda(x_i) = (\emptyset, \Lambda\cup\bar{\Lambda}, \mathbb{1})\bullet X_i$$

$$\mathscr{H}_\Lambda((\mu_1, \dots , \mu_n)\ (t_1, \dots , t_n)) = (\emptyset,(\Lambda\cup\bar{\Lambda})\setminus\{\mu_1 \dots \mu_n\}, \chi) + \sum_{i=1}^{n} ((\mu_1 \dots \mu_n), (\Lambda\cup\bar{\Lambda})\setminus\{\mu_1 \dots \mu_n\} , \mu_i)\bullet \mathscr{H}_\Lambda(t_i)$$

$$\mathscr{H}_\Lambda(Y(x_1 \leftarrow t_1, \dots, x_n \leftarrow t_n)) = Y_1(X_1 = \mathscr{H}_\Lambda(t_1), \dots, X_n = \mathscr{H}_\Lambda(t_n))$$

N.B. $\mathcal{H}_\Lambda(Y(x\leftarrow x)) = Y_1(X = (\emptyset,\ \Lambda\cup\bar{\Lambda},\ \mathbb{1}).\ X)$

$= K(Y(X'=(\emptyset,\ \Lambda\cup\bar{\Lambda},\ \mathbb{1})X'))$

$= K((\emptyset,\ \Lambda\cup\bar{\Lambda},\ \mathbb{1})^{\omega})$

$= (\emptyset,\ \Lambda\cup\bar{\Lambda}, \chi)$

Definition 8 given programs p and q, $\mathcal{H}_\Lambda(p|q) = \bigcup\{(h_p|h_q)\mid h_p \in H(\Lambda)\cap\mathcal{H}_\Lambda(p),$ $h_q \in H(\Lambda)\cap\mathcal{H}_\Lambda(q),\ h_p \# h_q\}$.

Definition 9 Let the restriction $R \equiv /\mu_1 \dots \mu_n$ and let sets of labels $\Lambda,\Lambda',\Lambda''$, be such that $\Lambda' = \{\mu_1 \dots \mu_n\}$, $\Lambda'' = \Lambda\cup\Lambda'\cup\bar{\Lambda}'$, then $\mathcal{H}_\Lambda(q[R]) = ((H(\Lambda'')\cap\mathcal{H}_{\Lambda''}(q))\downarrow(\Lambda'\cup\bar{\Lambda}'))\uparrow((\Lambda\cup\bar{\Lambda})\cap(\Lambda'\cup\bar{\Lambda}'))$.

Proposition 10 For any program p of sort Λ, $\mathcal{H}_\Lambda(p)$ is a set of histories, i.e. $\mathcal{H}_\Lambda(p) \subseteq H(\Lambda)$

Proof By induction on the structure of programs, using the definition of Y fixed points and the fact that K(W) belongs to $H(\Lambda)$ for any word W in $(\text{pre-h}(\Lambda))^{\omega}$ □

Before proving the equivalence between H_Λ and $\mathcal{H}_\Lambda$, we shall first establish separate properties of these two independently defined semantic functions.

Proposition 11 For any program p of sort Λ, $H_\Lambda(p)$ is not empty.

Proof Using induction on the structure of programs, we show that for any program p, $H_\Lambda(p)$ contains at least one history (d,δ,ρ) which verifies $d\cup\delta = \Lambda\cup\bar{\Lambda}$.

Induction basis

Let p be an elementary program, with none of its proper subprograms defined by recursion or resulting from a flow-operation.

- If p has no (W,μ) rewriting, then $(\emptyset,\ \Lambda\cup\bar{\Lambda},\ \chi) \in H_\Lambda(p)$ by definition 3.2
- If $p \xrightarrow{W_1\ ,\ \mu_1} \dots \xrightarrow{W_n\ ,\ \mu_n} q$ where q has no (W,μ) rewriting, then $(\emptyset,\ \Lambda\cup\bar{\Lambda},\ \mu_1 \dots \mu_n\ \chi) \in H_\Lambda(p)$ by repeated application of definitions 3.2 and 4.
- If $p \xrightarrow{W_1\ ,\ \mu_1} \dots \xrightarrow{W_n\ ,\ \mu_n} \dots$ is an infinite rewriting sequence, then by the third rule in definition 3.2, $(d,\delta,\ \mu_1\dots\ \mu_n\dots) \in H_\Lambda(p)$ for $d = \lim_i(\bigcup_{j\geq i} W_j)$, $\delta = \lim_i(\bigcap_{j\geq i}(\Lambda\cup\bar{\Lambda})\setminus W_j)$, thus $d\cup\delta = \Lambda\cup\bar{\Lambda}$.

Induction step. Three cases have to be examined.

- p is an elementary program. The only rewriting sequences of p which have not yet been considered are of the form $p \xrightarrow{W1\ ,\ \mu 1} \xrightarrow{W2\ ,\ \mu 2} \dots \xrightarrow{Wn\ ,\ \mu n} q$ where either $\mu_i \neq \mathbb{1}$ for $i \leqslant n$ and q is a recursively defined subprogram of p, or $\mu_i \neq \mathbb{1}$ for $i < n$, $\mu_n = \mathbb{1}$, and q is the result of a flow-operation. In any case and by the induction hypothesis, there exists some history (d,δ,ρ) in $H_\Lambda(q)$ which verifies $d \cup \delta = \Lambda \cup \overline{\Lambda}$; but then $(d,\delta,\mu_1 \dots \mu_n\ \rho) \in H_\Lambda(p)$ by repeated application of definitions 3.2 and 4.

- $p = q\ (/\mu_1 \dots \mu_m)$: obvious from the induction hypothesis.

- $p = (q|r)$. By the induction hypothesis, there must exist (d,δ,ρ) in $H_\Lambda(q)$ resp. (d',δ',ρ') in $H_\Lambda(r)$ such that $d \cup \delta = \Lambda \cup \overline{\Lambda} = d' \cup \delta'$. Let $h_q = (d,\delta,\rho)$ and $h_r = (d',\delta',\rho')$, then h_q and h_r are obviously compatible, and it is easily shown that $H_\Lambda(p)$ contains at least one history (d'',δ'',ρ'') which verifies $d'' \cup \delta'' = (d \cup \delta) \cap (d' \cup \delta') = \Lambda \cup \overline{\Lambda}$ $\square$

Proposition 12 For any program p of sort Λ and for any history (d,δ,ρ) in $\mathscr{H}_\Lambda(p)$, $\overline{d} \subseteq d \cup \delta$

Proof By induction on the structure of programs.

Induction basis

p is an elementary program, with none of its proper subprograms defined by recursion or resulting from a flow-operation. Let us notice that any of the triples (d,δ,ρ) which explicity appear in definition 7 are members of pre-h(Λ) and verify either ($\rho = \chi$ and $d = \emptyset$) or $(d \cup \delta = \Lambda \cup \overline{\Lambda})$. Due to the definition of Y fixed points, it is then sufficient to show that $\overline{d} \subseteq d \cup \delta$ holds for any (d,δ,ρ) in K(W) where W is either

- an infinite word $h_1 h_2 \dots h_i \dots$, with each h_i equal to some (d_i,δ_i,ρ_i) verifying $d_i \cup \delta_i = \Lambda \cup \overline{\Lambda}$ and $\rho_i \in \Lambda \cup \{\mathbb{1}\}$, or
- a finite word $h_1 h_2 \dots h_n h$ with h_i' s as above, and h equal to some complete history (d',δ',ρ') s.t. $\overline{d'} \subseteq d' \cup \delta'$. In the first case, definition 5 shows $d \cup \delta = \Lambda \cup \overline{\Lambda}$, while definition 4 yields $d = d'$ and $\delta = \delta'$ in the second case.

Induction step Three separate cases have to be examined.
- p is an elementary program. Use proposition 10 together with the induction hypothesis and follow the same reasoning as above.
- $p = q\ (/\mu_1 \dots \mu_n)$. Obvious from the induction hypothesis.
- $p = (q|r)$. Let (d,δ,ρ) in $\mathscr{H}_\Lambda(p)$, then there exist compatible histories $h_q \in \mathscr{H}_\Lambda(q)$ and $h_r \in \mathscr{H}_\Lambda(r)$ such that $(d,\delta,\rho) \in h_q | h_r$. Put $h_q = (d_q,\delta_q,\rho_q)$, $h_r = (d_r,\delta_r,\rho_r)$. By

the induction hypothesis, $\overline{d}_q \subseteq d_q \cup \delta_q$ and $\overline{d}_r \subseteq d_r \cup \delta_r$. By definition 3.6, $\delta = \delta_q \cap \delta_r$ and $d \cup \delta = (d_q \cup \delta_q) \cap (d_r \cup \delta_r)$, which entails $d = (d_q \cap d_r) \cup (d_q \cap \delta_r) \cup (\delta_q \cap d_r)$ since $\mathscr{H}_\Lambda(p)$ is a set histories by proposition 10.

Let μ in $\overline{d}$, then $\mu \in \overline{d}_q \cup \overline{d}_r$. Take for instance $\mu \in \overline{d}_q$, then $\mu \in d_q \cup \delta_q$.

Suppose for a moment $\mu \notin d_r \cup \delta_r$, then $\overline{\mu} \in \delta_q$ by compatibility of h_q and h_r, whence $\overline{\mu} \notin d_q$ by definition of histories, a contradiction with $\mu \in \overline{d}_q$.

It follows that $\mu \in (d_q \cup \delta_q) \cap (d_r \cup \delta_r) = d \cup \delta$, q.e.d. $\square$

Notice that proposition 3.9 is now a direct consequence of the above proposition according to the following

Theorem 13 For any program p of sort Λ , $\mathscr{H}_\Lambda(p) = H_\Lambda(p)$.

Proof By induction on the structure of programs.

Induction basis

Let p = NIL, then p has no (W,μ) rewriting, thus $H_\Lambda(\mathrm{NIL}) = (\emptyset, \Lambda \cup \overline{\Lambda}, \chi) = \mathscr{H}_\Lambda(\mathrm{NIL})$ by definitions 3.2 and 7.

Let $p = Y(x_1 \leftarrow t_1, \dots, x_n \leftarrow t_n)$ where no subterm of the t_i's is a program, then the proof of $H_\Lambda(p) = \mathscr{H}_\Lambda(p)$ is a particular case of the general argument given in the induction step.

Induction step

For p resulting from a flow-operation, the induction is obvious, so that there only remains two nontrivial cases.

- $p = (\mu_1, \dots, \mu_n)(p_1, \dots, p_n)$ where the p_i's are programs.

By definition 3.1, the set of (W,μ) rewritings of p is

$$\{ p \xrightarrow{\{\mu_1 \dots \mu_n\},\ \mu_i} p_i \mid 1 \leqslant i \leqslant n\}.$$

By definition 3.2, $(\emptyset, (\Lambda \cup \overline{\Lambda}) \setminus \{\mu_1 \dots \mu_n\}, \chi) \in H_\Lambda(p)$. Aso by definition 3.2, $(d,\delta,\mu_i\rho)$ belongs to $H_\Lambda(p)$ for any history (d,δ,ρ) in $H_\Lambda(p_i)$. For any such history, one has $((\mu_1 \dots \mu_n), (\Lambda \cup \overline{\Lambda}) \setminus \{\mu_1 \dots \mu_n\}, \mu_i) \bullet (d,\delta,\rho) = (d,\delta,\mu_i\rho)$. Since $H_\Lambda(p_i) = \mathscr{H}_\Lambda(p_i)$ holds by the induction hypothesis, the inclusion $\mathscr{H}_\Lambda(p) \subseteq H_\Lambda(p)$ clearly follows from the two facts above.

Suppose for a moment that there exists some history h in $H_\Lambda(p) \setminus \mathscr{H}_\Lambda(p)$, and show that this assumption is nonsense. By the first part of the proof, one has

necessarily $(h \in H_\Lambda(p))$ provable from the third rule in definition 3.2 .
Thus $p_i \xrightarrow{W_1\ ,\ \nu_1} \xrightarrow{W_2\ ,\ \nu_2} \dots \xrightarrow{W_k\ ,\ \nu_k} \dots$ for some i such that
$h = (d,\delta,\mu_i\ \nu_1\ \nu_2 \dots \nu_k \dots)$ with d and δ respectively equal to $\lim_k\ (\bigcup_{j\geqslant k} W_j)$,
$\lim_k\ (\bigcap_{j\geqslant k} (\Lambda\cup\overline{\Lambda})\setminus W_j)$.

By the third rule in definition 3.2, $h_i = (d,\delta,\ \nu_1\ \nu_2 \dots \nu_k \dots)$ is an history in $H_\Lambda(p_i)$, and since $h_i \in H(\Lambda)$, one has $h = ((\mu_1 \dots \mu_n),\ (\Lambda\cup\overline{\Lambda}) \setminus \{\mu_1\dots\mu_n\}\ ,\ \mu_i)\bullet h_i$ by definition of the concatenation.
It follows by the induction hypothesis $H_\Lambda(p_i) = \mathscr{H}_\Lambda(p_i)$ that $h \in \mathscr{H}_\Lambda(p)$ since $h_i \in \mathscr{H}_\Lambda(p_i)$. Thus $H_\Lambda(p) = \mathscr{H}_\Lambda(p)$.

- $p = Y\ (x_1 \leftarrow t_1\ ,\ \dots\ ,\ x_m \leftarrow t_m)$.

Define Subst, the parallel substitution of free variables which makes each of x_j's replaced by the corresponding program $p_j = Y(x_j \leftarrow t_j\ ,\ \dots\ ,\ x_m \leftarrow t_m\ ,\ x_1 \leftarrow t_1\ ,\ \dots)$.
According to definitions 3.1, 3.2 , the following relations are valid for any j :
$H_\Lambda(p_j) = H_\Lambda(t_j(\mathrm{Subst}))$.
(Notice that for t_j the result of a flow-operation, $t_j \equiv t_j(\mathrm{Subst})$ and t_j has no (W,μ) rewriting, so that $p_j \xrightarrow{\emptyset\ ,\ \mathbb{1}} t_j$ is the unique rewriting sequence of p_j).
Now, for any possible subterm t of t_j :

- if t is a variable, let x_k , $H_\Lambda(x_k(\mathrm{Subst})) = H_\Lambda(p_k)$
  $= H_\Lambda(t_k(\mathrm{Subst}))$
- if t is a program, let q, $H_\Lambda(q(\mathrm{Subst})) = H_\Lambda(q)$
  $= \mathscr{H}_\Lambda(q)$ by the induction hypothesis
- in any other case, let $t = (\nu_1,\ \dots\ ,\nu_n)(t'_1,\ \dots\ ,\ t'_n)$, then $H_\Lambda(t(\mathrm{Subst})) =$
  $(\emptyset,\ (\Lambda\cup\overline{\Lambda}) \setminus \{\nu_1\dots\nu_n\},\ \chi)\ +$

$$\sum_{i=1}^{n} ((\nu_1\dots\nu_n),\ (\Lambda\cup\overline{\Lambda})\setminus\{\nu_1\dots\nu_n\},\ \nu_i)\bullet\ H_\Lambda(t'_i(\mathrm{Subst}))$$

by definitions 3.1, 3.2 and 4.

Denote $H_\Lambda(t_j(\mathrm{Subst}))$ as Z_j for $1 \leqslant j \leqslant m$, then $Z_j = (\emptyset,\ \Lambda\cup\overline{\Lambda}\ ,\ \mathbb{1})\bullet Z_j$ for any j, since $Z_j \subseteq H(\Lambda)$. It follows from all the above listed facts that the vector $Z = \langle Z_1,\ \dots,\ Z_m\rangle$ is a solution of the system of linear equations $\langle X\rangle = (A)\bullet\ \langle X\rangle + \langle C\rangle$ attached to the program p by definition 7, let S that system.
Moreover, a careful examination of definition 7 shows that Z is the <u>least</u> solution in $\mathscr{P}(H(\Lambda))^m$ of the modified system $S' \equiv \langle X\rangle = (A')\ \bullet\ \langle X\rangle + \langle C'\rangle$ which is obtained from S through the following transformations :

- substitute $\mathbb{1}$ for A (i,j) when equal to $(\emptyset, \Lambda\cup\bar{\Lambda}, \mathbb{1})$
- substitute $(\emptyset, \Lambda\cup\bar{\Lambda},\chi)$ for C(i) when x_i is engaged in a circular definition $(x_i = x_{i_0} \leftarrow x_{i_1}, \ldots, x_{i_k} \leftarrow x_{i_0})$, thus C(i) was empty in that case.

(Notice that the least solution of S' in $\mathscr{P}(H(\Lambda))^m$ generally differs from its least solution in $\mathscr{P}(P\text{-}h(\Lambda))^m$ when such a solution exists).

For any index i such that x_i is engaged in a circular definition, thus $Z_i = (\emptyset,\Lambda\cup\bar{\Lambda}, \chi)$ by construction of S', one finds out $Y_i(S) = K((\emptyset, \Lambda\cup\bar{\Lambda}, \mathbb{1})^\omega) = (\emptyset, \Lambda\cup\bar{\Lambda},\chi) = Z_i$.

Since $\mathbb{1}\bullet h = h = (\emptyset, \Lambda\cup\bar{\Lambda},\mathbb{1})\bullet h$ for any h in P-h(Λ), it clearly follows that Y(S) is a solution of S' in $\mathscr{P}(P\text{-}h(\Lambda))^m$. But then by proposition 10, $Y_j(S) = \mathscr{H}_\Lambda(p_j)$ is a set of histories for any j, whence Y(S) is also a solution of S' in $\mathscr{P}(H(\Lambda))^m$. Since Z is the least such solution, it follows that Z_j is included in $Y_j(S)$ for any j, and in particular $H_\Lambda(p) \subseteq \mathscr{H}_\Lambda(p)$ by taking j = 1.

Now show that the reverse inclusion also holds.

Put $\varepsilon = (\emptyset, \Lambda\cup\bar{\Lambda},\mathbb{1})$ and consider h in $\mathscr{H}_\Lambda(p)$, then one of the following situations must occur, by definitions 7 and 3.1 together with properties of Y-fixed points and morphism K.

▲ $h \in K(\varepsilon^*(\Delta_1, \nabla_1, \mu_1)\varepsilon^* \ldots (\Delta_n, \nabla_n, \mu_n)W)$ with

$\mu_i \in \Delta_i \cup \{\mathbb{1}\}$, $\nabla_i = (\Lambda\cup\bar{\Lambda})\setminus\Delta_i$ for any i,

$p = Y(x_1 \leftarrow t_1, \ldots, x_m \leftarrow t_m) \xrightarrow{\Delta_1,\mu_1} \ldots \xrightarrow{\Delta_n,\mu_n} q$,

and either 1, 2 or 3 as follows :

1) $\mu_i \neq \mathbb{1}$ for any i, $q = (\nu_1, \ldots, \nu_k)(t'_1, \ldots, t'_k)$, and $W = (\emptyset, (\Lambda\cup\bar{\Lambda})\setminus\{\nu_1 \ldots \nu_k\},\chi)$.

In this case, $q \xrightarrow{\{\nu_1 \ldots \nu_k\},\ \nu_1} q'$ for some q', and it follows by repeated application of the second rule in definition 3.2 that $h = (\emptyset, (\Lambda\cup\bar{\Lambda})\setminus\{\nu_1 \ldots \nu_k\}, \mu_1 \ldots \mu_n\chi)$ belongs to $H_\Lambda(p)$.

2) $\mu_i \neq \mathbb{1}$ for any i, none of the p_j's is a subprogram of q for $j \in [1,m]$, and $W \in K^{-1}(\mathscr{H}_\Lambda(q))$.

In this case, $K(W) \in \mathscr{H}_\Lambda(q) = H_\Lambda(q)$ by the induction hypothesis, since q must be a subprogram of p by the above assumption that none of p_j's is a subprogram of q. Put $K(W) = (d,\delta,\rho)$, then (d,δ,ρ) is an history, and $h = (d,\delta,\mu_1 \ldots \mu_n\rho) \in H_\Lambda(p)$ by repeated application of the second rule in definition 3.2

3) $\mu_i \neq \mathbb{1}$ for $i < n$, $\mu_n = \mathbb{1}$, q is the result of a flow-operation, and $W \in \mathbf{K}^{-1}(\mathscr{R}_\Lambda(q))$

Then apply the same reasoning as in the second case.

(Notice that all the above arguments apply as well to $h \in K(\mathcal{E}^* W)$, since $H_\Lambda(Y(y'_1 \leftarrow t'_1, \ldots, y'_k \leftarrow t'_k)) = H_\Lambda(t'_1[\text{Subst}'])$ for any recursively defined subprogram $Y(y'_1 \leftarrow t'_1, \ldots, y'_k \leftarrow t'_k)$ of p).

▲ $h \in K(\mathcal{E}^* (\Delta_1, \nabla_1, \mu_1) \mathcal{E}^* \ldots (\Delta_n, \nabla_n, \mu_n) \mathcal{E}^* \ldots)$, with $\mu_i \in \Delta_i$, $\nabla_i = (\Lambda \cup \overline{\Lambda}) \setminus \Delta_i$ for any i, thus $\mu_i \neq \mathbb{1}$, $p = Y(x_1 \leftarrow t_1, \ldots, x_m \leftarrow t_m) \xrightarrow{\Delta_1, \mu_1} \xrightarrow{\Delta_2, \mu_2} \ldots \xrightarrow{\Delta_n, \mu_n} \ldots$ and conditions considered so far are verified for no W.

Then h belongs to $H_\Lambda(p)$ by the third rule in definition 3.2

▲ $h \in K(\mathcal{E}^* (\Delta_1, \nabla_1, \mu_1) \mathcal{E}^* \ldots (\Delta_n, \nabla_n, \mu_n) \mathcal{E}^\omega)$, with $\mu_i \in \Delta_i, \nabla_i = (\Lambda \cup \overline{\Lambda}) \setminus \Delta_i$ for any i,

$p = Y(x_1 \leftarrow t_1, \ldots, x_m \leftarrow t_m) \xrightarrow{\Delta_1, \mu_1} \ldots \xrightarrow{\Delta_n, \mu_n} x_j[\text{Subst}]$ and x_j is engaged in a circular definition

$x_j = x_{j0} \leftarrow x_{j1}, \ldots, x_{jk} \leftarrow x_{j0}$.

In this case, $x_j[\text{Subst}]$ has no (W, μ) rewriting, thus $(\emptyset, \Lambda \cup \overline{\Lambda}, \chi) \in H_\Lambda(q)$, and $h = (\emptyset, \Lambda \cup \overline{\Lambda}, \mu_1 \ldots \mu_n \chi)$ belongs to $H_\Lambda(p)$ by repeated application of definition 3.2 □

In order to complete our review of the properties of the semantic function H_Λ, we shall finally mention the following

<u>Proposition 14</u> For any program p of sort Λ, the set $\{\rho \in \Lambda^\omega \cup \Lambda^* \chi \mid (\exists d)(\exists \delta)((d, \delta, \rho\chi) \in H_\Lambda(p))\}$ is a prefix-closed language.

<u>scheme of proof</u>: from definitions 3.3.2, 3.3.7, 3.3.8 and by the now usual induction on the structure of programs, noting that the parallel compound $\mathscr{L}' \mid \mathscr{L}''$ of prefix closed languages $\mathscr{L}'$ and $\mathscr{L}''$ in $\mathscr{P}(\Lambda^\infty)$ is also a prefix-closed language. □

APPENDIX 4 : MORE ON OBSERVATIONS.

This fourth appendix contains the full proofs of the properties of partial observations which have been stated in subsection 4.1, and establishes also some complementary facts to be used in later proofs. Λ denotes a fixed finite subset of M throughout the division. We begin with two lemmas which regard the association operation $\bullet$ in pre-Obs (Λ), the set of pre-observations.

Lemma 1 The association of pre-observations, as stated in 4.1.3, is a totally defined operation.

Proof The single non trivial case to be examined is the association of pre-observations O_1 and O_2 both incomplete, let $O_1 = (d_1,\delta_1,\rho_1)$ and $O_2 = (d_2,\delta_2,\rho_2)$. Define $\rho = \rho_1\ \rho_2$, $\delta = \delta_1 \cap \delta_2$, and $d = d_1 \cup d_2 \cup (\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1)$, and check (d,δ,ρ) for the conditions expressed in def. 4.1.1. Conditions 1' and 2' are satisfied, since $Act(\rho) = Act\ (\rho_1) \cup Act(\rho_2) \subseteq d_1 \cup d_2 \subseteq d$. Condition 3' is a direct consequence of properties $d_1 \cap \delta_1 = \emptyset$, $d_2 \cap \delta_2 = \emptyset$. Satisfaction of condition 3", i.e. $Act(\rho) \cap \overline{\delta} = \emptyset$, is guaranteed by properties $Act(\rho_1) \cap \overline{\delta}_1 = \emptyset$ and $Act(\rho_2) \cap \overline{\delta}_2 = \emptyset$.

Lemma 2 The association of pre-observations is associative.

Proof $(O_1 \bullet O_2) \bullet O_3 = O_1 \bullet (O_2 \bullet O_3)$ is shown below by case analysis.

- if O_1 is a complete pre-observation, then definition 4.1.3 brings $(O_1 \bullet O_2) \bullet O_3 = O_1 \bullet O_3 = O_1 = O_1 \bullet (O_2 \bullet O_3)$
- if O_1 resp. O_2 are incomplete resp. complete pre-observations, then $O_1 \bullet O_2$ is complete, thus $(O_1 \bullet O_2) \bullet O_3 = O_1 \bullet O_2 = O_1 \bullet (O_2 \bullet O_3)$
- if O_1, O_2, O_3 are respectively incomplete, incomplete and complete, then the following are verified, letting $O_i = (d_i,\delta_i,\rho_i)$: $(O_1 \bullet O_2) \bullet O_3 = (\,-,\ -,\rho_1\rho_2) \bullet O_3 = (d_3,\delta_3,\rho_1\rho_2\rho_3)$, $O_1 \bullet (O_2 \bullet O_3) = O_1 \bullet (d_3,\delta_3,\rho_2\rho_3) = (d_3,\delta_3,\rho_1\rho_2\rho_3)$.
- finally consider the case where pre-observations $O_i = (d_i,\delta_i,\rho_i)$ are incomplete for every $i = 1,2,3$.

$(O_1 \bullet O_2) \bullet O_3 = (d,\delta_1 \cap \delta_2 \cap \delta_3,\ \rho_1\rho_2\rho_3)$, letting

$$d = d_1 \cup d_2 \cup (\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1) \cup d_3 \cup ((\delta_1 \cap \delta_2) \setminus \delta_3) \cup (\delta_3 \setminus (\delta_1 \cap \delta_2))$$

$$= d_1 \cup d_2 \cup d_3 \cup ((\delta_1 \cup \delta_2 \cup \delta_3) \setminus (\delta_1 \cap \delta_2)) \cup ((\delta_1 \cap \delta_2) \setminus \delta_3)$$
$$= d_1 \cup d_2 \cup d_3 \cup ((\delta_1 \cup \delta_2 \cup \delta_3) \setminus (\delta_1 \cap \delta_2 \cap \delta_3)).$$

$O_1 \bullet (O_2 \bullet O_3) = (d',\delta_1 \cap \delta_2 \cap \delta_3,\ \rho_1\rho_2\rho_3)$, letting

$$d' = d_1 \cup d_2 \cup d_3 \cup (\delta_2 \setminus \delta_3) \cup (\delta_3 \setminus \delta_2) \cup (\delta_1 \setminus (\delta_2 \cap \delta_3)) \cup ((\delta_2 \cap \delta_3) \setminus \delta_1)$$
$$= d_1 \cup d_2 \cup d_3 \cup ((\delta_2 \cup \delta_3 \cup \delta_1) \setminus (\delta_2 \cap \delta_3)) \cup ((\delta_2 \cap \delta_3) \setminus \delta_1)$$
$$= d_1 \cup d_2 \cup d_3 \cup ((\delta_1 \cup \delta_2 \cup \delta_3) \setminus (\delta_1 \cap \delta_2 \cap \delta_3)). \quad \square$$

The two lemmas above immediately widen into the following proposition, which we state without proof :

Proposition 3 The concatenation of partial observations, as determined in 4.1.4, is a totally defined and associative operation.

We now turn to the solving of systems of linear equations over sets of partial observations. Our first task is to prove that I : (pre-Obs $(\Lambda))^{\infty} \to$ P-obs (Λ) is actually a monoid homomorphism as it has been announced in proposition 4.1.6 . We shall in fact strengthen that earlier proposition by stating the following definition 4 and proposition 5 :

Definition 4 pre-obs(Λ), the set of interaction records, is the subset of pre-observations (d,δ,ρ) for which $\rho \in \Lambda \cup \{\chi\} \cup \{\mathbb{1}\}$.

Proposition 5 I is a surjective monoid homomorphism ; moreover, the restriction of I to (pre-obs$(\Lambda))^{\infty}$ is also a surjective monoid homomorphism.

Proof We shall first verify that I is a total function, i.e., that I $(O_1O_2\ldots O_i\ldots)$ is univoquely determined and belongs to P-obs(Λ). The proof follows the same lines as in proposition 6 of appendix 3 apart from the case where $\mathcal{W} = O_1O_2\ldots O_i\ldots$ is an infinite sequence of incomplete pre-observations O_i, let $O_i = (d_i,\delta_i,\rho_i)$. In that case, let (d,δ,ρ) be the triple associated with $\mathcal{W}$ through the definition 4.1.5, that is $d = \lim\limits_{j} \ (\bigcup\limits_{i\geq j} d_i) \cup (\lim\limits_{j} \ (\bigcup\limits_{i\geq j} \delta_i) \setminus \lim\limits_{j} \ (\bigcap\limits_{i\geq j} \delta_i))$, $\delta = \lim\limits_{j} \ (\bigcap\limits_{i\geq j} \delta_i)$, and $\rho = \rho_1\rho_2\ldots\rho_i\ldots\chi$.
By the hypothesis $d_i \cap \delta_i = \emptyset$, $\lim\limits_{j} \ (\bigcup\limits_{i\geq j} d_i) \cap \lim\limits_{j} \ (\bigcap\limits_{i\geq j} \delta_i)$ is empty, and thus $d \cap \delta = \emptyset$.
Noticing that ρ is a complete sequence, there remains to show that conditions 2' and 3" of definition 4.1.1 are also verified under the assumption $\mathrm{Ult}(\rho) \neq \emptyset$. Let $\mu \in \mathrm{Ult}(\rho)$, then $(\forall j)\ (\exists i \geq j)\ (\mu \in \mathrm{Act}(\rho_i))$, and since the O_i's are incomplete, $(\forall j)\ (\exists i \geq j)$ $(\mu \in d_i$ and $\mu \notin \overline{\delta_i})$ is implied by the former property. It comes $\mu \in d$ and $\mu \notin \overline{\delta}$, whence one can conclude $(d,\delta,\rho) \in$ P-obs(Λ).

The proof that I is a monoid homomorphism follows exactly the same lines as in proposition 6 of appendix 3.

There remains to show that P-obs(Λ) is the I-image of (pre-obs$(\Lambda))^{\infty}$. Clearly, it is sufficient to verify that for any $O = (d,\delta,\rho)$ in pre-Obs(Λ), O equals $I(W_O)$ for some corresponding W_O in (pre-obs$(\Lambda))^{\infty}$. Cases are as follows :

- if $|\rho| \leq 1$, then $I(O) = O \in$ pre-obs(Λ).
- if $\rho = \mu_1\ldots\mu_n$, with $\mu_i \in \Lambda$ for any i, then (d,δ,μ_i) belongs to pre-obs(Λ) for any i, since $\mu_i \in \mathrm{Act}(\rho) \subseteq d$; letting $O_i = (d,\delta,\mu_i)$, $I(O_1\ldots O_n)$ equals O by defns 4.1.3, 4.1.5.
- if $\rho = \mu_1\ldots\mu_n\chi$, with $\mu_i \in \Lambda$ for any i, then $(\mu_i, \emptyset, \mu_i)$ belongs to pre-obs(Λ) for any i ; letting $O_i = (\mu_i, \emptyset, \mu_i)$, O is equal to $I(O_1\ldots O_n(d,\delta,\chi))$.

- if ρ is the infinite sequence $\mu_1\mu_2 \ldots \mu_i \ldots$, $\mu_i \in \Lambda$, then there exists some integer k such that $(\forall n \geq k)$ $(\mu_n \in d \setminus \overline{\delta})$, since $Ult(\rho)$ is included in $d \setminus \overline{\delta}$; letting $0_i = (\mu_i, \emptyset, \mu_i)$ for $i < k$, and $0_i = (d, \delta, \mu_i)$ for $i \geq k$, 0 is equal to $I(0_1 0_2 \ldots 0_i \ldots)$ □

Armed with the above property of I, we are able to solve equations over sets of partial observations along the way that has been already followed in the appendix 3.

Let $X = A \bullet X + C$ be a system of linear equations over sets of partial observations, where elements of the matrix A and vector C respectively range over $\mathcal{P}(I(\text{pre-obs}(\Lambda)^+))$ and $\mathcal{P}(\text{P-obs}(\Lambda))$. Since function I is onto, one can always build an associated system of linear equations over $\mathcal{P}(\text{pre-obs}(\Lambda)^\infty)$, let $X' = A'X' + C'$, which verifies $I(A'(i,j)) = A(i,j)$ and $I(C'(i)) = C(i)$ for any values of the indexes. Since I is a monoid homomorphism, $I(Y(X' = A'X'+C'))$ does not depend on the particular choice of A' and C' in $I^{-1}(A)$ resp. $I^{-1}(C)$, and is moreover a solution of the original system $X = A \bullet X + C$. Henceforth, we let $Y(X = A \bullet X + C)$ denote that solution whenever the equations bear upon sets of partial observations. As it was already the case with partial histories, $Y(X = A \bullet X + C)$ is <u>not</u> the greatest solution of $X = A \bullet X + C$ in $\mathcal{P}(\text{P-obs}(\Lambda))$. (Consider for instance the elementary equation $x = (\emptyset\ ,\ \Lambda \cup \overline{\Lambda},\ \mathbb{1}) \bullet x$)

Notice that elements $C'(i)$ of the vector C' are left unexplored through the calculation of $Y(X' = A'X'+C')$ by the resolution process given in appendix 1. As a consequence, we can still denote by $Y(X' = A'X'+C)$ the result obtained from the resolution process when blindly mistaking the elements $C(i)$ of vector C for subsets of $\text{pre-obs}(\Lambda)^\infty$ - instead of subsets of $\text{P-obs}(\Lambda)$. (This slight abuse of notation has already been admitted de facto in the last lines of definition 4.2.6). It is clear from the properties of I that the following equalities now hold :

$$I(Y(X' = A'X'+C')) = I(Y(X' = A'X'+C)) = Y\ (X = A \bullet X + C).$$

Our next series of propositions exhibits important properties of the concatenation operation $\bullet$ with respect to the ordering $\leq$ on pre-observations.

<u>Proposition 6</u> The association of pre-observations is monotonously increasing in each of its arguments.

<u>Proof</u>

<u>i) $0_1 \leq 0'_1 \Rightarrow 0_1 \bullet 0_2 \leq 0'_1 \bullet 0_2$</u>

We prove this fact by case analysis, letting $0_1 = (d_1, \delta_1, \rho_1)$, $0_2 = (d_2, \delta_2, \rho_2)$ and $0'_1 = (d'_1, \delta'_1, \rho'_1)$.

<u>case 1</u> $0'_1$ is a complete pre-observation.

Then 0_1 must be complete according to the definition of $\leq$, and therefore $0_1 \bullet 0_2 = 0_1 \leq 0'_1 = 0'_1 \bullet 0_2$

<u>case 2</u> O_1 and O'_1 are respectively complete and incomplete.

Then $\rho_1 = \rho''_1\,\chi$ with $\rho''_1 \prec \rho'_1$, and $O_1 = (\emptyset, \emptyset, \rho_1) = O_1 \bullet O_2$. Since O'_1 is incomplete, $O'_1 \bullet O_2$ has form $(- , - , \rho'_1\,\rho_2)$; $\rho''_1 \prec \rho'_1$ entails $\rho''_1 \prec \rho'_1\rho_2$, thus $(\emptyset, \emptyset, \rho_1) \leqslant O'_1 \bullet O_2$, that is still $O_1 \bullet O_2 \leqslant O'_1 \bullet O_2$.

<u>case 3</u> O_1 and O'_1 are incomplete, O_2 is complete.

$O_1 \leqslant O'_1$ then implies $\rho_1 = \rho'_1$, whence $O_1 \bullet O_2 = (d_2, \delta_2, \rho_1\rho_2) = (d_2, \delta_2, \rho'_1\rho_2) = O'_1 \bullet O_2$.

<u>case 4</u> O_1, O'_1 and O_2 are incomplete pre-observations. From $O_1 \leqslant O'_1$, one draws $\rho_1 = \rho'_1$, $\delta_1 \subseteq \delta'_1$, and $d_1 \subseteq d'_1 \cup \delta'_1$. $O_1 \bullet O_2$ and $O'_1 \bullet O_2$ are respectively given by :

$O_1 \bullet O_2 = (d_1 \cup d_2 \cup (\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1),\ \delta_1 \cap \delta_2,\ \rho_1\ \rho_2)$,

$O'_1 \bullet O_2 = (d'_1 \cup d_2 \cup (\delta'_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta'_1),\ \delta'_1 \cap \delta_2, \rho'_1\ \ \rho_2)$.

Relations $\rho_1\rho_2 = \rho'_1\rho_2$ and $\delta_1 \cap \delta_2 \subseteq \delta'_1 \cap \delta_2$ are obviously satisfied. There remains to prove the inclusion $d_1 \cup d_2 \cup (\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1) \subseteq d'_1 \cup d_2 \cup (\delta'_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta'_1) \cup (\delta'_1 \cap \delta_2)$, that is still $d_1 \cup d_2 \cup (\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1) \subseteq d'_1 \cup d_2 \cup \delta'_1 \cup \delta_2$.

In fact, $d_1 \cup d_2 \cup (\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1) \in (d'_1 \cup \delta'_1) \cup d_2 \cup (\delta'_1 \setminus \delta_2) \cup \delta_2 = d'_1 \cup \delta'_1 \cup d_2 \cup \delta_2$.

<u>ii) $O_2 \leqslant O'_2 \Rightarrow O_1 \bullet O_2 \leqslant O_1 \bullet O'_2$</u>

We proceed with case analysis, letting $O_1 = (d_1, \delta_1, \rho_1)$, $O_2 = (d_2, \delta_2, \rho_2)$ and $O'_2 = (d'_2, \delta'_2, \rho_2)$.

<u>case 1</u> O_1 is a complete pre-observation.
Then simply $O_1 \bullet O_2 = O_1 = O_1 \bullet O'_2$

<u>case 2</u> O_1 and O_2 are respectively incomplete and complete, and ρ_2 ,ρ'_2 are identical. From the definition of $\leqslant$, one has $\delta_2 \subseteq \delta'_2$ and $d_2 \subseteq d'_2 \cup \delta'_2$, and since O'_2 is complete by the hypothesis $\rho_2 = \rho'_2$, it comes $O_1 \bullet O_2 = (d_2, \delta_2, \rho_1\rho_2) \leqslant (d'_2, \delta'_2, \rho_1\rho_2) = O_1 \bullet O'_2$.

<u>case 3</u> O_1 and O_2 are respectively incomplete and complete, but ρ'_2 differs from ρ_2. Then $\rho_2 = \rho''_2\,\chi$ with $\rho''_2 \prec \rho'_2$ and $d_2 \cup \delta_2 = \emptyset$, thus $O_1 \bullet O_2 = (\emptyset, \emptyset, \rho_1\rho_2)$. Since O_1 is incomplete, $O_1 \bullet O'_2$ has form $(-, -, \rho_1\ \rho'_2)$, and $O_1 \bullet O_2 \leqslant O_1 \bullet O'_2$ follows by $\rho_1\ \rho''_2 \prec \rho_1\ \rho'_2$.

<u>case 4</u> O_1, O'_1 and O_2 are incomplete pre-observations. Similar to case 4 in the first part. □

Proposition 7 For any pair $\lfloor L_1 \rfloor$, $\lfloor L_2 \rfloor$ of languages of pre-observations, $\lfloor L_1 \bullet L_2 \rfloor = \lfloor L_1 \rfloor \bullet \lfloor L_2 \rfloor$.

Proof The inclusion $\lfloor L_1 \rfloor \bullet \lfloor L_2 \rfloor \subseteq \lfloor L_1 \bullet L_2 \rfloor$ is verified by straightforward application of the above proposition 6. We shall establish the reverse inclusion by showing that for any $O_1 \in L_1$, $O_2 \in L_2$ and $O \in$ pre-Obs(Λ) such that $O \preccurlyeq O_1 \bullet O_2$, there exist corresponding pre-observations O'_1 and O'_2 which verify $O'_1 \preccurlyeq O_1$, $O'_2 \preccurlyeq O_2$ and $O = O'_1 \bullet O'_2$. Put $O_1 = (d_1, \delta_1, \rho_1)$, $O_2 = (d_2, \delta_2, \rho_2)$. As usual, the reasoning is by case analysis.

i) O_1 is a complete pre-observation

Then $O \preccurlyeq O_1 \bullet O_2 = O_1$ shows that O is also complete, thus $O = O'_1 \bullet O'_2$ letting $O'_1 = O$ and $O'_2 = O_2$

ii) O_1 and O_2 are respectively incomplete and complete

In this case, $O_1 \bullet O_2$ is equal to $(d_2, \delta_2, \rho_1\rho_2)$.
Possible subcases are listed below.

case 1 $\rho = \rho_1\rho_2$, $\delta \subseteq \delta_2$, $d \subseteq d_2 \cup \delta_2$

Define $O'_2 = (d, \delta, \rho_2)$, then O'_2 is a pre-observation, since $(d, \delta, \rho) \in$ pre-Obs(Λ) and $\mathrm{Ult}(\rho) = \mathrm{Ult}(\rho_2)$ together imply $\mathrm{Ult}(\rho) \subseteq d$ as required. $O'_2 \preccurlyeq O_2$ is obviously verified. Now, letting $O'_1 = O_1$, the following equalities hold by the hypothesis ii) :
$O = (d, \delta, \rho_1\rho_2) = (d_1, \delta_1, \rho_1) \bullet (d, \delta, \rho_2) = O'_1 \bullet O'_2$

case 2 $d \cup \delta = \emptyset$, $\rho = \rho' \chi$, $\rho' \prec \rho_1$

Put $O'_1 = (\emptyset, \emptyset, \rho'\chi)$, then $O'_1 \preccurlyeq O_1$, and letting $O'_2 = O_2$, $O = O'_1 = O'_1 \bullet O'_2$ since O'_1 is complete.

case 3 $d \cup \delta = \emptyset$, $\rho = \rho_1 \rho' \chi$, $\rho' \prec \rho_2$

One can freely assume that ρ' is a finite sequence, since $|\rho'| = \omega$ would imply $\rho = \rho_1\rho_2$ as in the above case 1. Thus $(\emptyset, \emptyset, \rho'\chi)$ is a pre-observation, let it O'_2, and verifies $O'_2 \preccurlyeq O_2$. Since ρ_1 is incomplete by the hypothesis ii) , one has the equalities $O = (\emptyset, \emptyset, \rho_1\rho'\chi) = O_1 \bullet (\emptyset, \emptyset, \rho'\chi) = O_1 \bullet O'_2$

iii) O_1 and O_2 are incomplete pre-observations

In this case, $O_1 \bullet O_2$ is equal to $(d_3, \delta_3, \rho_1\rho_2)$, letting $d_3 = d_1 \cup d_2 \cup (\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1)$ and $\delta_3 = \delta_1 \cap \delta_2$.

Possible subcases are as follows.

case 1 $\rho = \rho_1\rho_2$, $\delta \subseteq \delta_3$, $d \subseteq d_3 \cup \delta_3$

Put $d'_1 = d \cap (d_1 \cup \delta_1)$, $\delta'_1 = \delta \cap \delta_1$, $O'_1 = (d'_1, \delta'_1, \rho_1)$, and $d'_2 = d \cap (d_2 \cup \delta_2)$, $\delta'_2 = \delta \cap \delta_2$, $O'_2 = (d'_2, \delta'_2, \rho_2)$.

$Act(\rho_1) \subseteq d_1$ and $Act(\rho_1\rho_2) \subseteq d$ are both verified since 0_1 and 0 are incomplete pre-observations, and thus $Act(\rho_1) \subseteq d \cap d_1 \subseteq d'_1$ by the obvious inclusions $Act(\rho_1) \subseteq Act(\rho_1) \cap Act(\rho_1)$, $Act(\rho_1) \subseteq Act(\rho_1\rho_2)$. $Act(\rho_1) \cap \overline{\delta}_1 = \emptyset$ also holds from hypothesis iii, whence it comes that $Act(\rho_1) \cap \overline{\delta'}_1 = \emptyset$. Since $d \cap \delta = \emptyset$ implies $d'_1 \cap \delta'_1 = \emptyset$, the above facts show that $0'_1$ is a pre-observation, which moreover satisfies $0'_1 \lessdot 0_1$ as required. Similar facts lead to similar conclusions for $0'_2$. Now, $0'_1 \bullet 0'_2$ is equal to $(d', \delta \cap \delta_1 \cap \delta_2, \rho_1\, \rho_2)$, letting $d' = (d \cap (d_1 \cup \delta_1)) \cup (d \cap (d_2 \cup \delta_2)) \cup ((\delta \cap \delta_1) \setminus (\delta \cap \delta_2)) \cup ((\delta \cap \delta_2) \setminus (\delta \cap \delta_1)) = (d \cap (d_1 \cup d_2 \cup \delta_1 \cup \delta_2)) \cup (\delta \cap ((\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1)))$.

$d \subseteq d_3 \cup \delta_3 = d_1 \cup d_2 \cup \delta_1 \cup \delta_2 \Rightarrow d = d \cap (d_1 \cup d_2 \cup \delta_1 \cup \delta_2)$;
$\delta \subseteq \delta_3 = \delta_1 \cap \delta_2 \Rightarrow \delta \cap ((\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1)) = \emptyset$;
$\delta \subseteq \delta_3 = \delta_1 \cap \delta_2 \Rightarrow \delta \cap \delta_1 \cap \delta_2 = \emptyset$;
thus $d' = d$ and $0'_1 \bullet 0'_2 = 0$.

<u>case 2</u> $d \cup \delta = \emptyset$, $\rho = \rho'\, \chi$, $\rho' \prec \rho_1$
Identical with case 2 in part ii)

<u>case 3</u> $d \cup \delta = \emptyset$, $\rho = \rho_1\, \rho'\, \chi$, $\rho' \prec \rho_2$
Then ρ' is necessarily finite since 0_2 is incomplete, and the remainder is as in case 3 of ii) $\square$

The remaining of the division pays attention to languages of partial observations, resp. observations, as regards the issues of rationality and computability.

We turn first to consider languages of partial observations.

<u>Definition 8</u> Let L be an expression which denotes a set of partial observations. L is in <u>rational normal form</u> if and only if it appears as a finite non-empty sum $\sum_i (d_i, \delta_i, \mathcal{L}_i)$ where the $\mathcal{L}_i$'s are non empty bi-rational languages.

<u>Lemma 9</u> There exist effective procedures which, given L' and L'' in rational normal form, compute each of the sets $I(L' + L'')$, $I(L'L'')$, $I(L'^{+})$ and $I(L'^{\omega})$ in rational normal form.

<u>proof</u> The above fact is immediate for the union of languages, so that we examine now the three remaining constructs, letting $L' = \sum_{i=1}^{n} (d'_i, \delta'_i, \mathcal{L}'_i)$ and $L'' = \sum_{j=1}^{m} (d''_j, \delta''_j, \mathcal{L}''_j)$.

<u>$L = L'\, L''$.</u>

$I(L) = (\sum_{i=1}^{n} (d'_i, \delta'_i, \mathcal{L}'_i)) \bullet (\sum_{j=1}^{m} (d''_j, \delta''_j, \mathcal{L}''_j)) = \hat{L} + \dot{L}$, letting

$$\hat{L} = \left(\sum_i (d'_i, \delta'_i, \hat{\mathcal{L}}'_i) \right) + \left(\sum_{i,j} (d''_j, \delta''_j, \dot{\mathcal{L}}'_i \hat{\mathcal{L}}''_j) \right),$$

$$\dot{L} = \sum_{i,j} ((d'_i \cup d''_j \cup \delta'_i \setminus \delta''_j \cup \delta''_j \setminus \delta'_i \ , \ \delta'_i \cap \delta''_j \ , \dot{\mathcal{L}}'_i \dot{\mathcal{L}}''_j)$$

Now, response languages $\hat{\mathcal{L}}'_i$, $\dot{\mathcal{L}}'_i \hat{\mathcal{L}}''_j$ and $\dot{\mathcal{L}}'_i \dot{\mathcal{L}}''_j$ are bi-rational, since restricted to their complete part for the first two and to the incomplete part for the last.

$L = L'^+$

$I(L) = I\left(\left(\sum_{i=1}^{n} (d'_i, \delta'_i, \mathcal{L}'_i)\right)^+\right) = \hat{L} + \dot{L}$, letting

$\hat{L} = \sum_i (d'_i, \delta'_i, (\sum_j \dot{\mathcal{L}}'_j)^* \hat{\mathcal{L}}'_i)$ and

$$\dot{L} = \sum_{1 \leqslant k \leqslant n} \ \sum_{1 \leqslant i_1 < i_2 \ldots < i_k \leqslant n} (((\bigcup_\ell d'_{i_\ell}) \cup (\bigcup_\ell \delta'_{i_\ell}) \setminus (\bigcap_\ell \delta'_{i_\ell})), \bigcap_\ell \delta'_{i_\ell} \ , \ Z^{\emptyset}_{\{i_1 \ldots i_k\}})$$

where $Z^{\emptyset}_{\{i_1 \ldots i_k\}}$ is the least solution of the following system of linear equations over $\mathscr{P}(\Lambda^\infty)$:

$$\left[\begin{array}{l}
Z^{\emptyset}_{\{i_1 \ldots i_k\}} = \sum_{m \leqslant k} \dot{\mathcal{L}}'_{i_m} \ Z^{\{i_m\}}_{\{i_1 \ldots i_k\} \setminus \{i_m\}} \\
Z^{\{j_1 \ldots j_\ell\}}_{\{i_1 \ldots i_k\}} = \sum_{p \leqslant \ell} \dot{\mathcal{L}}'_{j_p} \ Z^{\{j_1 \ldots j_\ell\}}_{\{i_1 \ldots i_k\}} + \sum_{q \leqslant k} \dot{\mathcal{L}}'_{i_q} \ Z^{\{j_1 \ldots j_\ell\} \cup \{i_q\}}_{\{i_1 \ldots i_k\} \setminus \{i_q\}} \\
Z^{\{j_1 \ldots j_\ell\}}_{\emptyset} = \mathbb{1} + \sum_{p \leqslant \ell} \dot{\mathcal{L}}'_{j_p} \ Z^{\{j_1 \ldots j_\ell\}}_{\emptyset}
\end{array}\right.$$

Since the least solutions of systems of linear equations are computable, so is $Z^{\emptyset}_{\{i_1 \ldots i_k\}}$, and since the $\dot{\mathcal{L}}'_i$ are rational, the same property holds for $Z^{\emptyset}_{\{i_1 \ldots i_k\}}$ as it has been proved in appendix 1. Clearly, either $\hat{L}$ or $\dot{L}$ differs from the empty set by the assumption $(\forall i)\ (\hat{\mathcal{L}}'_i \neq \emptyset$ or $\dot{\mathcal{L}}'_i \neq \emptyset)$. Now, response languages $(\sum_j \dot{\mathcal{L}}'_j)^* \hat{\mathcal{L}}'_i$ and $Z^{\emptyset}_{\{i_1 \ldots i_k\}}$ are birational, for they are respectively restricted to their complete and incomplete part.

$L = L'^\omega$

$I(L) = I\left(\left(\sum_{i=1}^{n} (d'_i, \delta'_i, \mathcal{L}'_i)\right)^{\omega}\right) = \hat{L}_1 + \hat{L}_2$, letting

$\hat{L}_1 = \sum_i (d'_i, \delta'_i, (\sum_{j=1}^{n} \dot{\mathcal{L}}'_j)^* \hat{\mathcal{L}}'_i)$, and

$$\hat{L}_2 = \sum_{1\leqslant k\leqslant n} \ \sum_{1\leqslant i_1<i_2\ldots<i_k\leqslant i_n} (((\bigcup_\ell d'_{i_\ell}) \cup (\bigcup_\ell \delta'_{i_\ell}) \setminus (\bigcap_\ell \delta'_{i_\ell})), \bigcap_\ell \delta'_{i_\ell}, X_{\{i_1\ldots i_k\}})$$

where $X_{\{i_1\ldots i_k\}}$ is defined as

$(\sum_{j=1}^{n} \dot{\mathcal{L}}'_j)^* \ (Z^{\emptyset}_{\{i_1\ldots i_k\}})^{\omega}$ if $\mathbb{1} \notin Z^{\emptyset}_{\{i_1\ldots i_k\}}$, or else

$(\sum_{j=1}^{n} \dot{\mathcal{L}}'_j)^* \ \chi$ if $Z^{\emptyset}_{\{i_1\ldots i_k\}} = \{\mathbb{1}\}$, or else

$(\sum_{j=1}^{n} \dot{\mathcal{L}}'_j)^* \ (\chi + (Z^{\emptyset}_{\{i_1\ldots i_k\}})^{\omega})$.

Clearly, either $\hat{L}_1$ or $\hat{L}_2$ differs from the empty set since for each i, either $\hat{\mathcal{L}}'_i$ or $Z^{\emptyset}_{\{i\}}$ differs from $\emptyset$ by the assumption $\hat{\mathcal{L}}'_i \neq \emptyset$ or $\dot{\mathcal{L}}'_i \neq \emptyset$.
Now, response languages $(\sum_j \dot{\mathcal{L}}'_j)^* \hat{\mathcal{L}}'_i$ and $X_{\{i_1\ldots i_k\}}$ are birational, since both restricted to their complete parts which are denoted by rational expressions. □

<u>Lemma 10</u> The following identities hold for any sets X and Y in $\mathscr{P}((\text{pre-obs}(\Lambda))^{\infty} \setminus \{\mathbb{1}\})$:

$I(X+Y) = I(I(X)+I(Y))$ (1)
$I(XY) = I(I(X)I(Y))$ (2)
$I(X^+) = I((I(X))^+)$ (3)
$I(X^\omega) = I((I(X))^\omega)$ (4)

<u>Proof</u>

$I(X+Y) = I(X) + I(Y) = I(I(X)) + I(I(Y)) = I(I(X) + I(Y))$.

$I(XY) = I(X) \bullet I(Y) = I(I(X)I(Y))$ by prop. 5 and defn. 4.1.5.

$I(X^+) = I(\sum_{n>0} X^n) = \sum_{n>0} (I_1(X) \bullet \ldots \bullet I_n(X))$ - by prop. 5 -

$= \sum_{n>0} I((I(X))^n)$ - by defn. 4.1.5. -

$= I(\sum_{n>0} (I(X))^n) = I((I(X))^+)$.

In order to prove the fourth equality, we have to show $I(x_1x_2\ldots x_i\ldots) = I(I(x_1)I(x_2)\ \ldots\ I(x_i)\ldots)$ for any infinite sequence $(x_i)_{i\in\omega}$ of words $x_i \in X$.

Put $x_i = O_{i1} \ldots O_{in_i}$, where O_{ij} ε pre-obs (Λ) for each j. First suppose that some interaction record O_{ij} is complete, then letting $x'_i = O_{i1} \ldots O_{ij}$, one obtains :

$$I(x_1x_2 \ldots x_i \ldots) = I(x_1x_2 \ldots x'_i) = I(x_1x_2 \ldots x_i)$$

$$I(I(x_1)I(x_2) \ldots I(x_i) \ldots) = I(I(x_1)I(x_2) \ldots I(x'_i) \ldots)$$

$$= I(I(x_1)I(x_2) \ldots I(x'_i)) = I(I(x_1)I(x_2) \ldots I(x_i))$$

by definition 4.1.5 together with definition 4.1.3.

But then $I(x_1 \ldots x_i) = I(x_1) \cdot I(x_2) \cdot \ldots \cdot I(x_i)$ by prop. 5, and $I(I(x_1) \ldots I(x_i)) = I(x_1) \cdot I(x_2) \cdot \ldots \cdot I(x_i)$ by the definition of I. Now turn to the case where every interaction records O_{ij} are incomplete. Define $(O'_k)_{k \in \omega}$ by $O_{ij} = O'_{n_1 + \ldots + n_{i-1} + j}$, and put $O_{ij} = (d_{ij}, \delta_{ij}, \rho_{ij})$, $O'_k = (d'_k, \delta'_k, \rho'_k)$. Applying definition 4.1.5, one obtains

$\underline{I\ (x_1 \ldots x_i \ldots)}$ = (d', δ', ρ') with

$$d' = \lim_{\ell} \left(\bigcup_{k \geqslant \ell} (d'_k \cup \delta'_k)\right) \setminus \lim_{\ell} \left(\bigcap_{k \geqslant \ell} \delta'_k\right) \qquad \delta' = \lim_{\ell} \left(\bigcap_{k \geqslant \ell} \delta'_k\right)$$

$$\rho' = \rho'_1 \, \rho'_2 \cdots \rho'_k \cdots \chi,$$

$\underline{I(x_i)}$ = (d_i, δ_i, ρ_i) with

$$d_i = \bigcup_{j \leqslant n_i} (d_{ij} \cup \delta_{ij}) \setminus \bigcap_{j \leqslant n_i} \delta_{ij} \qquad \delta_i = \bigcap_{j \leqslant n_i} \delta_{ij}$$

$$\rho_i = \rho_{i1} \cdots \rho_{in_i} ,$$

$\underline{I(I(x_1) \ldots I(x_i) \ldots)}$ = (d, δ, ρ) with

$$d = \lim_{m} \left(\bigcup_{i \geqslant m} (d_i \cup \delta_i)\right) \setminus \lim_{m} \left(\bigcap_{i \geqslant m} \delta_i\right) \qquad \delta = \lim_{m} \left(\bigcap_{i \geqslant m} \delta_i\right)$$

$$\rho = \rho_1 \rho_2 \cdots \rho_i \cdots \chi.$$

The equalities $\rho = \rho'$ and $\delta = \delta'$ are obvious, so that we are left with proving $d = d'$:

$$d = \lim_{m} \left(\bigcup_{i \geqslant m} \left(\bigcup_{j \leqslant n_i} (d_{ij} \cup \delta_{ij})\right)\right) \setminus \delta$$

$$= \lim_{\ell} \left(\bigcup_{k \geqslant \ell} (d'_k \cup \delta'_k)\right) \setminus \delta$$

$$= d' \quad (\text{since } \delta = \delta') \qquad \square$$

By the induction on the structure of rational expressions, the proof of proposition 4.1.11 is quite immediate in view of the above lemmas 9 and 10. Another outcome of lemma 9 is stated in the following

Proposition 11 Let $\mathcal{A}'_n$ denote the set of n x m matrices A' whose elements $A'(i,j)$ range over finite sets of finite non empty words over pre-obs(Λ), and let $\mathcal{C}_n$ denote the set of n-ranked vectors C whose elements C(i) either appear in rational normal form or denote the empty set. Let $\mathcal{S}_n$ be the set of equational systems $X' = A'X'+C'$ s.t. $A' \in \mathcal{A}'_n$, $C \in \mathcal{C}_n$, and $(\forall i \leqslant n)\ (\sum_j A'(i,j) + C(i) \neq \emptyset)$. There exists an effective procedure which, given S in $\mathcal{S}_n$, computes I(Y(S)) in rational normal form.

proof In view of the resolution technique that has been propounded in the appendix 1, $Y(X' = A'X'+C)$ can be computed in the form $L + \sum_{i=1}^{n} L_i\ C(i)$, where L and L_i's depend upon A' only, and

- L is a purely infinitary rational language in Rat(pre-obs$(\Lambda)^{\omega}$),
- L_i's are finitary rational languages in Rat(pre-obs$(\Lambda)^{*}$)

(thus the same holds for $E_i \overset{def}{=} L_i \setminus \{1\}$).

By proposition 4.1.11, I(L) and $I(E_i)$ - if not empty- can be computed in rational normal form.

By lemmas 9 and 10, $I(E_i\ C(i))$ -if not empty- can also be computed in rational normal form, since it amounts to $I(I(E_i)\ C(i))$ by the obvious identity $C(i) = I(C(i))$.

In order to complete the proof, it suffices to remark that Y(S) differs from $\emptyset$ by the assumption $(\forall i \leqslant n)\ (\sum_j A'(i,j) + C(i) \neq \emptyset)$ $\Box$

We finally turn to consider languages of observations and their relation to rational languages of partial observations.

Definition 12 Given the set of partial observations $L = \sum_i (d_i,\delta_i,\mathcal{L}_i)$ in rational normal form, we define $\theta(L) = \sum_i (d_i,\delta_i,\hat{\mathcal{L}}_i) + (\emptyset, \emptyset, \text{pref}\ (\sum_i \mathcal{L}_i)\chi)$, where pref(Z) is the set made out of the finite left factors of the words in $(Z\setminus\chi)$.

Definition 13 Given the set of partial observations $L = \sum_i (d_i,\delta_i,\mathcal{L}_i)$ in rational normal form, we define $\pi(L) = \sum_i \pi(d_i,\delta_i,\mathcal{L}_i)$, where we let $\pi(d,\delta,\mathcal{L})$ be equal to the sum $\Sigma(d',\delta',\mathcal{L})\ |\ d'$ and δ' such that : $(\delta\setminus\bar{d}) \subseteq \delta' \subseteq \delta$ and $d' = (d \cup \delta)\setminus(\delta' \cup \bar{\delta}')$.

Lemma 14 If the set of partial observations $L = \sum_i (d_i, \delta_i, \mathcal{L}_i)$ appears in rational normal form, then $\pi(\theta(L))$ is a non empty set of complete observations ; moreover, there exists an effective procedure which, given L, computes $\pi(\theta(L))$ in the rational normal form of sets of observations.

Proof The only non trivial part is to show that for any complete pre-observation (d,δ,ρ), (d',δ',ρ) is an observation in $Obs(\Lambda)$ if d' and δ' verify the conditions $(\delta \setminus \overline{d}) \subseteq \delta' \subseteq \delta$ and $d' = (d \cup \delta) \setminus (\delta' \cup \overline{\delta}')$.

Since $d' \cap (\delta' \cup \overline{\delta}')$ is obviously empty, we are left with checking $Ult(\rho) \subseteq d'$:

$Ult(\rho) \subseteq d$, $d \cap \delta = \emptyset$, $\delta' \subseteq \delta \Rightarrow Ult(\rho) \cap \delta' = \emptyset$;
$Ult(\rho) \cap \overline{\delta} = \emptyset$, $\delta' \subseteq \delta \Rightarrow Ult(\rho) \cap \overline{\delta}' = \emptyset$;
thus $Ult(\rho) \subseteq d \setminus (\delta' \cup \overline{\delta}') \subseteq d'$. $\square$

We are now able to make proposition 4.1.12 a little more precise through the following definition of φ :

Definition 15 For L a set of partial observations in rational normal form, we let $\varphi(L)$ be equal to $\pi(\theta(L))$.

Proof of proposition 4.1.12 We have only to establish the identity $\lfloor L \rfloor \cap Obs(\Lambda) = \lfloor \varphi(L) \rfloor$ for L in rational normal form, since the remaining of the proof follows by direct application of lemma 14.

Put $L = \sum_i (d_i, \delta_i, \mathcal{L}_i)$, where the $\mathcal{L}_i$'s are non-empty bi-rational response languages.

Let $\hat{L}$ denote the set of the complete pre-observations O' which belong to $\lfloor L \rfloor$. The inclusion $\theta(L) \subseteq \hat{L}$ is obvious. On the other side, let us prove that $\theta(L)$ contains all maximal complete pre-observations in $\hat{L}$.

Consider (d,δ,ρ) in $\hat{L}$. There must exist (d',δ',ρ') in L such that $(d,\delta,\rho) \preccurlyeq (d',\delta',\rho')$.

- if ρ' is complete, then (d',δ',ρ') belongs to $(d_i, \delta_i, \hat{\mathcal{L}}_i)$ for some i, whence $(d,\delta,\rho) \preccurlyeq (d',\delta',\rho') \in \theta(L)$
- if ρ' is incomplete, then necessarily $\rho = \rho'' \chi$, $\rho'' \prec \rho'$ and $d \cup \delta = \emptyset$ by the definition of the ordering $\preccurlyeq$, and since ρ' is in $\dot{\mathcal{L}}_i$ for some i one obtains $(d,\delta,\rho) \preccurlyeq (\emptyset , \emptyset , \rho'' \chi) \in \theta(L)$.

Since any particular observation is also a complete pre-observation, it follows from the above facts that $\lfloor L \rfloor \cap Obs(\Lambda)$ is equal to $\lfloor \theta(L) \rfloor \cap Obs(\Lambda)$.

Note that $\lfloor \theta(L) \rfloor$ is a set of complete pre-observations : no incomplete pre-observation can be less than a complete pre-observation. Also note that given $(d'',\delta'',\rho'') \in \lfloor \theta(L) \rfloor$, one can always find (d,δ,ρ) in $\theta(L)$ such that $\rho'' = \rho$ and $(d'',\delta'',\rho'') \preccurlyeq (d,\delta,\rho)$ - this fact is a consequence of the inclusion $(\emptyset, \emptyset, pref(\bigcup_i \mathcal{L}_i)\ \chi) \subseteq \theta(L)$-.

Put $\hat{\mathbb{L}} = \lfloor \theta(L) \rfloor \cap \mathrm{Obs}(\Lambda)$. The inclusion $\pi(\theta(L)) \subseteq \hat{\mathbb{L}}$ is easily verified :

- let (d',δ',ρ) in $\pi(\theta(L))$, then there exists (d,δ,ρ) in $\theta(L)$ such that $(\delta \setminus \bar{d}) \subseteq \delta' \subseteq \delta$ and $d' = (d \cup \delta)\setminus(\delta' \cup \bar{\delta}')$; $\delta' \subseteq \delta$ and $d' \subseteq d \cup \delta$ imply $(d',\delta',\rho) \preccurlyeq (d,\delta,\rho)$, whence we can conclude $(d',\delta',\rho) \in \hat{\mathbb{L}}$ by lemma 14.

In order to prove that $\hat{\mathbb{L}}$ equals $\lfloor \pi(\theta(L)) \rfloor_\circ$, it is sufficient to show that for any (d'',δ'',ρ) in $\hat{\mathbb{L}}$, there exists a corresponding observation (d',δ',ρ) in $\pi(\theta(L))$ which verifies $(d'',\delta'',\rho) \preccurlyeq (d',\delta',\rho)$ - remark that $\hat{\mathbb{L}} = \lfloor \hat{\mathbb{L}} \rfloor_\circ$ - .
Consider (d'',δ'',ρ) in $\hat{\mathbb{L}}$.
Since (d'',δ'',ρ) belongs to $\lfloor \theta(L) \rfloor$, there must exist some pre-observation (d,δ,ρ) in $\theta(L)$ for which $(d'',\delta'',\rho) \preccurlyeq (d,\delta,\rho)$, that is still $\delta'' \subseteq \delta$ and $d'' \subseteq d \cup \delta$. We will prove that there exists (d',δ',ρ) in $\pi(d,\delta,\rho)$ which verifies $(d'',\delta'',\rho) \preccurlyeq (d',\delta',\rho)$.

Let $\delta' = (\delta \setminus \bar{d}) \cup \delta''$ and $d' = (d \cup \delta) \setminus (\delta' \cup \bar{\delta}')$, then (d',δ',ρ) belongs to $\pi(d,\delta,\rho)$ since δ'' is included in δ - a consequence of $(d'',\delta'',\rho) \preccurlyeq (d,\delta,\rho)$ - .
The inclusion $\delta'' \subseteq \delta'$ is obvious, so that there remains to prove $d'' \subseteq d' \cup \delta'$.
$(d'',\delta'',\rho) \preccurlyeq (d,\delta,\rho) \Rightarrow d'' \subseteq d \cup \delta$,

$(d'',\delta'',\rho) \in \mathrm{Obs}(\Lambda) \Rightarrow d'' \cap (\delta'' \cup \bar{\delta}'') = \emptyset$,

thus $d'' \subseteq (d \cup \delta) \setminus (\delta'' \cup \bar{\delta}'')$.

$d' \cup \delta' = (d \cup \delta) \setminus (\delta' \cup \bar{\delta}') \cup (\delta \setminus \bar{d}) \cup \delta''$,
$\delta' \cup \bar{\delta}' = (\delta \setminus \bar{d}) \cup (\bar{\delta} \setminus d) \cup (\delta'' \cup \bar{\delta}'')$, thus :
$d' \cup \delta' = (d \cup \delta) \setminus ((\delta \setminus \bar{d}) \cup (\bar{\delta} \setminus d) \cup (\delta'' \cup \bar{\delta}'')) \cup (\delta \setminus \bar{d}) \cup \delta''$
$= (d \cup \delta) \setminus ((\bar{\delta} \setminus d) \cup \delta'' \cup \bar{\delta}'') \cup (\delta \setminus \bar{d}) \cup \delta''$
$= d \setminus (\delta'' \cup \bar{\delta}'') \cup \delta \setminus ((\bar{\delta} \setminus d) \cup \delta'' \cup \bar{\delta}'') \cup (\delta \setminus \bar{d}) \cup \delta''$
$= d \setminus (\delta'' \cup \bar{\delta}'') \cup \delta \setminus (\bar{\delta} \cup \delta'' \cup \bar{\delta}'') \cup (\delta \setminus \bar{d}) \cup \delta''$ - since $d \cap \delta = \emptyset$ -
$= d \setminus (\delta'' \cup \bar{\delta}'') \cup \delta \setminus (\delta'' \cup \bar{\delta}'') \cup (\delta \setminus \bar{d}) \cup \delta''$ - since $\delta \cap \bar{\delta} \subseteq \delta \setminus \bar{d}$ -
$= (d \cup \delta) \setminus (\delta'' \cup \bar{\delta}'') \cup (\delta \setminus \bar{d}) \cup \delta''$.

It follows that the inclusion $d'' \subseteq d' \cup \delta'$ is verified. One can therefore conclude that $\hat{\mathbb{L}}$ equals $\lfloor \pi(\theta(L)) \rfloor_\circ$, whence finally
$\lfloor \pi(\theta(L)) \rfloor_\circ = \lfloor \theta(L) \rfloor \cap \mathrm{Obs}(\Lambda) = \lfloor L \rfloor \cap \mathrm{Obs}(\Lambda)$. $\square$

<u>Proposition 16</u> For any sets of partial observations L, L' and L'' in rational normal form, the equality $\mathrm{Obs}(\Lambda) \cap (\lfloor L'' \rfloor \bullet \lfloor L \rfloor) = \mathrm{Obs}(\Lambda) \cap (\lfloor L'' \rfloor \bullet \lfloor L' \rfloor)$ holds if $L' \subseteq \mathrm{Obs}(\Lambda)$ and $\mathrm{Obs}(\Lambda) \cap \lfloor L \rfloor = \lfloor L' \rfloor_\circ$.

<u>Proof</u> $\mathrm{Obs}(\Lambda) \cap (\lfloor L'' \rfloor \bullet \lfloor L \rfloor) = \mathrm{Obs}(\Lambda) \cap \lfloor L'' \bullet L \rfloor$ - by prop. 7 -
$= \{ \omega \in \mathrm{Obs}(\Lambda) \mid (\exists O'' \in L'')(\exists O \in L)\ (\omega \preccurlyeq O'' \bullet O) \}$.
Let $\hat{L}''$ and $\dot{L}''$ respectively denote the complete and incomplete parts of L'', then the above set evaluates to the union
$\{ \omega \in \mathrm{Obs}(\Lambda) \mid (\exists O'' \in \hat{L}'')\ (\omega \preccurlyeq O'') \} \cup$
$\{ \omega \in \mathrm{Obs}(\Lambda) \mid (\exists O'' \in \dot{L}'')\ (\exists O \in L)\ (\omega \preccurlyeq O'' \bullet O) \}$.
In order to establish $\mathrm{Obs}(\Lambda) \cap (\lfloor L'' \rfloor \bullet \lfloor L \rfloor) = \mathrm{Obs}(\Lambda) \cap (\lfloor L'' \rfloor \bullet \lfloor L' \rfloor)$,

it suffices to prove that the following equality holds for any O" in $\dot{L}''$:

$$\{ \omega \in Obs(\Lambda) \mid (\exists O \in L) (\omega \leqslant O''\bullet O)\} = \{ \omega \in Obs(\Lambda) \mid (\exists O' \in L') (\omega \leqslant O''\bullet O')\}.$$

From the definitions of the ordering $\leqslant$ and concatenation $\bullet$, it may be seen that the above property does not depend at all upon O", so that $O'' = (\emptyset, \emptyset, \mathbb{1})$ may be freely assumed. Hence, the above equality is equivalent to the statement

$$\{ \omega \in Obs(\Lambda) \mid (\exists O \in L) (\omega \leqslant O)\} = \{ \omega \in Obs(\Lambda) \mid (\exists O' \in L') (\omega \leqslant O')\} ,$$

that is still $Obs(\Lambda) \cap \lfloor L \rfloor = \lfloor L' \rfloor_{\circ}$. $\square$

APPENDIX 5 = HISTORIES AND OBSERVATIONS.

Some ingredients are still missing for the full proof of the propositions which connect the operational and observational semantics of our simple programming language. We shall now complete our technical preparation for the final proofs by writing down some lemmas which investigate the relationship between pre-histories and pre-observations.

Lemma 1 Let $h = (d,\delta,\rho)$ in pre-$H(\Lambda)$, then h has a greatest lower bound O in pre-$Obs(\Lambda)$, given by : $O = (d \cup (\delta \cap \overline{Act(\rho)}), \delta \setminus \overline{Act(\rho)}, \rho)$ if ρ is incomplete, or $O = (d \cup (\delta \cap \overline{Ult(\rho)}), \delta \setminus \overline{Ult(\rho)}, \rho)$ if ρ is complete.

Proof immediate from definitions 4.1.1 and App. 3.2 □

Definition 2 Given O and h in pre-$Obs(\Lambda)$, resp. pre-$H(\Lambda)$, $O \sqsubset h$ holds iff O is the greatest lower bound of h in pre-$Obs(\Lambda)$.

Definition 3 pre$\sim H(\Lambda)$ is the subset of pre-$H(\Lambda)$ made out of the pre-histories (d,δ,ρ) which verify $d \cup \delta = (\Lambda \cup \overline{\Lambda})$.

Lemma 4 Let h_i in pre$\sim H(\Lambda)$, $i \in \omega$, then the following are verified :

$K(h_1h_2 \ldots h_n) \in \text{pre}\sim H(\Lambda)$,

$K(h_1h_2 \ldots h_i \ldots) \in \text{pre}\sim H(\Lambda)$.

Proof From definition 5 and proposition 6 of appendix 3, the first property can be established by just considering $n = 2$. Put $h_1 = (d_1,\delta_1,\rho_1)$ and $h_2 = (d_2,\delta_2,\rho_2)$.

- if ρ_1 is complete, then $K(h_1h_2) = h_1 \bullet h_2 = h_1 \in \text{pre}\sim H(\Lambda)$;
- if ρ_1 and ρ_2 are respectively incomplete and complete, then $K(h_1h_2) = h_1 \bullet h_2 = (d_2,\delta_2,\rho_1\rho_2) \in \text{pre}\sim H(\Lambda)$;
- if ρ_1 and ρ_2 are both incomplete, then $K(h_1h_2)$ is equal to $(d,\delta,\rho_1\rho_2)$, letting $d = d_1 \cup d_2$ and $\delta = \delta_1 \cap \delta_2$, hence $d \cup \delta = d_1 \cup d_2 \cup (\delta_1 \cap \delta_2) = (\Lambda \cup \overline{\Lambda}) \setminus \delta_1 \cup (\Lambda \cup \overline{\Lambda}) \setminus \delta_2 \cup (\delta_1 \cap \delta_2)$
$= \Lambda \cup \overline{\Lambda}$.

Let us consider now the infinitary case. If some element of the sequence $(h_i)_{i \in \omega}$ is a complete history, then according to the definition of K, $K(h_1h_2 \ldots h_i \ldots)$ belongs to pre$\sim H(\Lambda)$ by the first part of the proof. In the converse case, put $h_i = (d_i,\delta_i,\rho_i)$ for every i then $K(h_1h_2 \ldots h_i \ldots)$ is equal to $(d,\delta,\rho_1\rho_2 \ldots \rho_i \ldots \chi)$, letting

$$d = \lim_i \left(\bigcup_{j \geq i} d_j\right) \text{ and } \delta = \lim_i \left(\bigcap_{j \geq i} \delta_j\right) .$$

Now, $d = \lim_i \left(\bigcup_{j \geq i} ((\Lambda \cup \overline{\Lambda}) \setminus \delta_j)\right) = (\Lambda \cup \overline{\Lambda}) \setminus \lim_i \left(\bigcap_{j \geq i} \delta_j\right)$,

so that $d \cup \delta$ amounts to $\Lambda \cup \overline{\Lambda}$. □

<u>Proposition 5</u> For $i = 1\ldots n$, let h_i in pre$\sim$H(Λ) and O_i in pre-Obs(Λ) such that $O_i -\!\sqsubset h_i$, then $O_1 \bullet O_2 \bullet \ldots \bullet O_n -\!\sqsubset h_1 \odot h_2 \odot \ldots \odot h_n$

where $\bullet$ and $\odot$ denote the concatenation operations in pre-Obs(Λ) resp. pre-H(Λ).

<u>Proof</u> First notice that $h_1 \odot h_2 \odot \ldots \odot h_n$ belongs to pre$\sim$H(Λ) by the above lemma 4, since equal to $K(h_1 h_2 \ldots h_n)$. By the associativity of $\bullet$ and $\odot$, it is enough to consider $n = 2$.

Put $O_1 = (d_1, \delta_1, \rho_1)$, $h_1 = (d'_1, \delta'_1, \rho_1)$,

and $O_2 = (d_2, \delta_2, \rho_2)$, $h_2 = (d'_2, \delta'_2, \rho_2)$.

- if ρ_1 is complete, then $O_1 \bullet O_2 = O_1 -\!\sqsubset h_1 = h_1 \odot h_2$.

- if ρ_1 and ρ_2 are respectively incomplete and complete, then $O_1 \bullet O_2 = (d_2, \delta_2, \rho_1\rho_2)$, $h_1 \odot h_2 = (d'_2, \delta'_2, \rho_1\rho_2)$, and $\mathrm{Ult}(\rho_1\rho_2) = \mathrm{Ult}(\rho_2)$; hence, $O_1 \bullet O_2 -\!\sqsubset h_1 \odot h_2$ derives directly from the hypothesis $O_2 -\!\sqsubset h_2$.

- there remains to analyse the case where both ρ_1 and ρ_2 are incomplete sequences. According to definitions 4.1.3 and App.3.4, $O_1 \bullet O_2$ and $h_1 \odot h_2$ are then respectively given by $O_1 \bullet O_2 = (d, \delta, \rho_1\ \rho_2)$ and $h_1 \odot h_2 = (d', \delta', \rho_1\rho_2)$, letting :

$$d = d_1 \cup d_2 \cup \delta_1 \setminus \delta_2 \cup \delta_2 \setminus \delta_1 \quad , \qquad \delta = \delta_1 \cap \delta_2$$

$$d' = d'_1 \cup d'_2 \quad , \qquad \delta' = \delta'_1 \cap \delta'_2 \ .$$

$$\delta_1 \cap \delta_2 = (\delta'_1 \setminus \overline{\mathrm{Act}(\rho_1)}) \cap (\delta'_2 \setminus \overline{\mathrm{Act}(\rho_2)})$$

$$= (\delta'_1 \cap \delta'_2) \setminus (\overline{\mathrm{Act}(\rho_1)} \cup \overline{\mathrm{Act}(\rho_2)}) = (\delta'_1 \cap \delta'_2) \setminus \overline{\mathrm{Act}(\rho_1\rho_2)}.$$

$$d_1 \cup d_2 \cup (\delta_1 \setminus \delta_2) \cup (\delta_2 \setminus \delta_1) =$$

$$d'_1 \cup (\delta'_1 \cap \overline{\mathrm{Act}(\rho_1)}) \cup d'_2 \cup (\delta'_2 \cap \overline{\mathrm{Act}(\rho_2)}) \cup (\delta'_1 \setminus \overline{\mathrm{Act}(\rho_1)}) \setminus \delta_2 \cup (\delta'_2 \setminus \overline{\mathrm{Act}(\rho_2)}) \setminus \delta_1$$

$$= d'_1 \cup (\delta'_1 \cap \overline{\mathrm{Act}(\rho_1)}) \cup d'_2 \cup (\delta'_2 \cap \overline{\mathrm{Act}(\rho_2)}) \cup \delta'_1 \setminus \delta_2 \cup \delta'_2 \setminus \delta_1 =$$

$$d'_1 \cup (\delta'_1 \cap \overline{\mathrm{Act}(\rho_1)}) \cup d'_2 \cup (\delta'_2 \cap \overline{\mathrm{Act}(\rho_2)}) \cup (\delta'_1 \setminus (\delta_2 \setminus \overline{\mathrm{Act}(\rho_1)}) \cup (\delta'_2 \setminus (\delta_1 \setminus \overline{\mathrm{Act}(\rho_2)})$$

$$= d'_1 \cup (\delta'_1 \cap \overline{\mathrm{Act}(\rho_1)}) \setminus d'_2 \cup d'_2 \cup (\delta'_2 \cap \overline{\mathrm{Act}(\rho_2)}) \setminus d'_1 \cup$$

$$\delta'_1 \setminus (\delta'_2 \setminus \overline{\mathrm{Act}(\rho_1\rho_2)}) \cup \delta'_2 \setminus (\delta'_1 \setminus \overline{\mathrm{Act}(\rho_1\rho_2)})$$

$$= d'_1 \cup (\delta'_1 \cap \delta'_2 \cap \overline{\mathrm{Act}(\rho_1)}) \cup d'_2 \cup (\delta'_1 \cap \delta'_2 \cap \overline{\mathrm{Act}(\rho_2)}) \cup$$

$$\delta'_1 \setminus (\delta'_2 \setminus \overline{\mathrm{Act}(\rho_1\rho_2)}) \cup \delta'_2 \setminus (\delta'_1 \setminus \overline{\mathrm{Act}(\rho_1\rho_2)})$$

$$= d'_1 \cup d'_2 \cup (\delta'_1 \cap \delta'_2 \cap \overline{\mathrm{Act}(\rho_1\rho_2)}) \cup \delta'_1 \setminus (\delta'_2 \setminus \overline{\mathrm{Act}(\rho_1\rho_2)}) \cup \delta'_2 \setminus (\delta'_1 \setminus \overline{\mathrm{Act}(\rho_1\rho_2)})$$

Let μ in $\delta'_1 \setminus (\delta'_2 \setminus \overline{\mathrm{Act}(\rho_1\rho_2)})$, then

either $\mu \in (\Lambda \cup \overline{\Lambda}) \setminus \delta'_2 = d'_2$

or $\mu \in \delta'_1 \cap \delta'_2 \cap \overline{\mathrm{Act}(\rho_1\rho_2)}$.

According to that remark, the above expression may be simplified into
$d'_1 \cup d'_2 \cup (\delta'_1 \cap \delta'_2) \cap \overline{Act(\rho_1\rho_2)}$.

From $\delta = \delta' \setminus \overline{Act(\rho_1\rho_2)}$ and $d = d' \cup (\delta' \cap \overline{Act(\rho_1\rho_2)})$,

one finally draws $O_1 \bullet O_2 \sqsubset h_1 \odot h_2$ by lemma 1. $\square$

<u>Proposition 6</u> For $i \in \omega$, let h_i in pre-H(Λ) and O_i in pre-Obs(Λ) such that $O_i \sqsubset h_i$, then :
$I\ (O_1O_2 \ldots O_i \ldots) \sqsubset K\ (h_1h_2 \ldots h_i \ldots)$.

<u>Proof</u> Put $O_i = (d_i, \delta_i, \rho_i)$ and $h_i = (d'_i, \delta'_i, \rho_i)$ for every i. If some element of $(\rho_i)_{i \in \omega}$ is a complete sequence, let ρ_n, then the result comes by straightforward application of prop. 5, since $I\ (O_1O_2 \ldots O_i \ldots)$ and $K(h_1h_2 \ldots h_i \ldots)$ are respectively equal to $O_1 \bullet O_2 \bullet \ldots \bullet O_n$ and $h_1 \odot h_2 \odot \ldots \odot h_n$.

In the converse case, $I(O_1O_2 \ldots O_i \ldots)$ and $K(h_1h_2 \ldots h_i \ldots)$ respectively amount to (d,δ,ρ) and (d',δ',ρ), letting :

$$\rho = \rho_1\rho_2 \ldots \rho_i \ldots \chi,$$

$$\delta = \lim_j \left(\bigcap_{i \geq j} \delta_i\right) \quad , \delta' = \lim_j \left(\bigcap_{i \geq j} \delta'_i\right) ,$$

$$d = \lim_j \left(\bigcup_{i \geq j} d_i\right) \cup \lim_j \left(\bigcup_{i \geq j} \delta_i\right) \setminus \lim_j \left(\bigcap_{i \geq j} \delta_i\right) ,$$

$$d' = \lim_j \left(\bigcup_{i \geq j} d'_i\right).$$

By the hypothesis $(\forall i)\ (O_i \sqsubset h_i)$, one obtains the equalities :

$$\delta = \lim_j \left(\bigcap_{i \geq j} (\delta'_i \setminus \overline{Act(\rho_i)})\right) = \lim_j \left(\bigcap_{i \geq j} \delta'_i\right) \setminus \lim_j \left(\bigcup_{i \geq j} \overline{Act(\rho_i)}\right)$$

$$= \delta' \setminus \overline{Ult(\rho)} ,$$

$$d = \lim_j \left(\bigcup_{i \geq j} (d'_i \cup (\delta'_i \cap \overline{Act(\rho_i)}))\right) \cup \lim_j \left(\bigcup_{i \geq j} (\delta'_i \setminus \overline{Act(\rho_i)})\right) \setminus \delta$$

$$= \lim_j \left(\bigcup_{i \geq j} d'_i\right) \cup \lim_j \left(\bigcup_{i \geq j} (\delta'_i \cap \overline{Act(\rho_i)})\right) \setminus \delta \cup \lim_j \left(\bigcup_{i \geq j} (\delta'_i \setminus \overline{Act(\rho_i)})\right) \setminus \delta$$

– from $d \cap \delta = \emptyset$ –

$$= \lim_j \left(\bigcup_{i \geq j} d'_i\right) \cup \lim_j \left(\bigcup_{i \geq j} \delta'_i\right) \setminus \delta \,.$$

$$= \lim_j \left(\bigcup_{i \geq j} d'_i\right) \cup \left(\lim_j \left(\bigcup_{i \geq j} \delta'_i\right) \setminus \lim_j \left(\bigcup_{i \geq j} d'_i\right)\right) \setminus \delta$$

$$= \lim_j \left(\bigcup_{i \geq j} d'_i\right) \cup \left(\lim_j \left(\bigcap_{i \geq j} \delta'_i\right)\right) \setminus \delta$$

– from $d'_i \cup \delta'_i = \Lambda \cup \overline{\Lambda}$ and $d'_i \cap \delta'_i = \emptyset$ –

$$= d' \cup \delta' \setminus (\delta' \setminus \overline{Ult(\rho)}) = d' \cup (\delta' \cap \overline{Ult(\rho)}).$$

Thus, $(d,\delta,\rho) \sqsubset (d',\delta',\rho)$. $\square$

Proposition 7 Let D be a set of incomplete pre-observations in pre-Obs(Λ), and let D' be a set of incomplete pre-histories in pre-H(Λ), related to each other by the property

$$D = \{O \in \text{pre-Obs}(\Lambda) \mid (\exists h \in D')\ (O \mathrel{-\!\!\sqsubset} h)\}.$$

Let $C \subseteq \text{Obs}(\Lambda)$ and $C' \subseteq H(\Lambda)$ such that

$$\lfloor C \rfloor_{\circ} = \{O \in \text{Obs}(\Lambda) \mid (\exists h \in C')\ (O \preccurlyeq h)\}\text{ , then}$$

$$\lfloor D{\bullet}C \rfloor_{\circ} = \{O \in \text{Obs}(\Lambda) \mid (\exists h \in D' \odot C')\ (O \preccurlyeq h)\}$$

Proof Immediate from the fact that for any ρ in Λ^* ,

$$(\exists\, d)\ (\exists\, \delta)\ ((d,\delta,\rho) \in D) \Leftrightarrow (\exists\, d')\ (\exists\, \delta')\ ((d',\delta',\rho) \in D'). \qquad \square$$

APPENDIX 6 : FINAL PROOFS.

We give here the full proofs of the results which have been stated in the second part of section 4, concerning the issue of the observational semantics.

Proof of proposition 4.2.17

We appeal to the induction on the structure of programs.

Induction basis : p is an elementary program, with none of its proper subprograms defined by recursion or resulting from a flow-operation.

Define end-h(Λ) as the set of the triples of the form $(\emptyset,\delta,\chi)$,$\delta \subseteq \Lambda \cup \bar{\Lambda}$, and remark that any such triple belongs to the intersection $Obs(\Lambda) \cap H(\Lambda)$.

Next define pre$\sim$h(Λ) as the set of the triples (d,δ,ρ) in pre$\sim$H(Λ) which verify $\rho \in \Lambda \cup \bar{\Lambda} \cup \{\mathbb{1}\}$, and remark that for any such triple,

$(d,\delta,\rho) \sqsubset (d,\delta,\rho)$ if $\rho = \mathbb{1}$, or else
$(d \cup \{\bar{\mu}\}\ ,\delta\setminus\{\bar{\mu}\}\ ,\rho) \sqsubset (d,\delta,\rho)$ if $\rho = \mu \neq \mathbb{1}$.

Accounting for the obvious similarity between the equations which define $\mathcal{M}(p)$ and $\mathcal{H}_\Lambda(p)$ - defns 4.2.16 and App. 3.7 -, and according to the facts which have been established in the appendix 1, we can make the following claim.

- There exist rational languages,
let $L_\circ \subseteq ((\text{pre}\sim h(\Lambda))^* \text{ end-h}(\Lambda))$, $L_\omega \subseteq (\text{pre}\sim h(\Lambda))^\omega$
and $L'_\circ \subseteq (\text{pre-obs}(\Lambda))^+$,$L'_\omega \subseteq (\text{pre-obs}(\Lambda))^\omega$
such that relations i to iv are satisfied :

i) $\mathcal{H}_\Lambda(p) = K(L_\circ) + K(L_\omega)$

ii) $I(\mathcal{M}(p)) = I(L'_\circ) + I(L'_\omega)$

iii) $I(L'_\omega) = \{O \in \text{pre-Obs}(\Lambda) \mid (\exists h \in K(L_\omega))\ (O \sqsubset h)\}$

- by prop. 6 of app.5 and the second remark above-

iv) $\lfloor I(L'_\circ)\rfloor_\circ = \{O \in Obs(\Lambda) \mid (\exists h \in K(L_\circ))\ (O \leqslant h)\}$

- by prop. 5 and 7 of app. 5 and the two remarks above-

(notice that the use of notation $\lfloor I(L'_\circ)\rfloor_\circ$ implicitly means that $I(L'_\circ)$ is included in $Obs(\Lambda)$)

Now, $\lfloor I(L'_\circ)\rfloor_\circ = \lfloor I(L'_\circ)\rfloor \cap Obs(\Lambda) = \lfloor \varphi(I(L'_\circ))\rfloor_\circ$

$= \{O \in Obs(\Lambda) \mid (\exists h \in K(L_\circ))(O \leqslant h)\}$

- by prop. 4.1.11 and 4.1.12, and by iv -

and $\lfloor \varphi(I(L'_\omega))\rfloor_\circ = \lfloor I(L'_\omega)\rfloor \cap Obs(\Lambda)$

$= \{O \in Obs(\Lambda) \mid (\exists h \in K(L_\omega))\ (O \leqslant h)\}$

- by defn. 2 of app.5, and by iii -

thus $\lfloor \varphi(I(\mathcal{M}(p)))\rfloor_\circ = \{O \in Obs(\Lambda) \mid (\exists h \in \mathcal{H}_\Lambda(p))(O \leqslant h)\}$

$= \mathrm{Obs}_\Lambda\ (p)$

- by the definition of Obs_Λ and by App. 3.13 -

The remaining of the proof is a straightforward application of propositions 4.1.13, 4.1.11, 4.1.12 (in the order).

Induction step p is an elementary program, and $p_1 \dots p_m$ are the outermost proper subprograms of p such that for any i, p_i is defined by recursion Y or results from a flow-operation.

We shall freely assume that $p_1 \dots p_\ell$ are recursively defined programs whereas $p_{\ell+1} \dots p_m$ are results of flow-operations. The recursion hypothesis may therefore be expressed as $(\forall i \leqslant \ell)$: $\lfloor \varphi(I(\mathscr{M}(p_i)))\rfloor_\circ = \mathrm{Obs}_\Lambda(p_i) = \{O \in \mathrm{Obs}(\Lambda) \mid (\exists h \in H_\Lambda(p_i))(O \leqslant h)\}$

For the same reasons as in the first part of the proof, we can make the following affirmation.

- There exist indexes $I \subseteq (1,\ell)$, $J \subseteq (\ell+1,m)$ and there exist rational languages, let

$L_\circ \in ((\mathrm{pre}\sim h(\Lambda))^* \ \mathrm{end}\text{-}h(\Lambda))$, $L'_\circ \subseteq (\mathrm{pre\text{-}obs}(\Lambda))^+$

$L_\omega \subseteq (\mathrm{pre}\sim h(\Lambda))^\omega$, $L'_\omega \subseteq (\mathrm{pre\text{-}obs}(\Lambda))^\omega$

$L_i \subseteq (\mathrm{pre}\sim h(\Lambda))^+$, $L'_i \subseteq (\mathrm{pre\text{-}obs}(\Lambda))^+$ $(i \in I \cup J)$

such that the following relations are all satisfied :

i) $\mathscr{H}_\Lambda(p) = K(L_\circ) + K(L_\omega) + \sum_{i \in I \cup J} K(L_i) \odot \mathscr{H}_\Lambda(p_i)$

ii) $I(\mathscr{M}(p)) = I(L'_\circ) + I(L'_\omega) + \sum_{i \in I} I(L'_i) \bullet I(\mathscr{M}(p_i)) + \sum_{j \in J} I(L'_j) \bullet \mathscr{N}(p_j)$

iii) $I(L'_\omega) = \{O \in \mathrm{pre\text{-}Obs}(\Lambda) \mid (\exists h \in K(L_\omega))\ (O \sqsubset h)\}$

iv) $\lfloor I(L'_\circ)\rfloor_\circ = \{O \in \mathrm{Obs}(\Lambda) \mid (\exists h \in K(L_\circ))\ (O \leqslant h)\}$

v) $I(L'_i) = \{O \in \mathrm{pre\text{-}Obs}(\Lambda) \mid (\exists h \in K(L_i))\ (O \sqsubset h)\}$

for every i in $I \cup J$

- thus $I(L'_i)$ is a set of incomplete pre-observations, since $K(L_i)$ is a set of incomplete pre-histories-

The equalities $\lfloor \varphi(I(L'_\circ))\rfloor_\circ = \{O \in \mathrm{Obs}(\Lambda) \mid (\exists h \in K(L_\circ))\ (O \leqslant h)\}$ and $\lfloor \varphi(I(L'_\omega))\rfloor_\circ = \{O \in \mathrm{Obs}(\Lambda) \mid (\exists h \in K(L_\omega))\ (O \leqslant h)\}$ are established the same way as in the first part of the proof. Now consider j in J, then $I(L'_j) \bullet \mathscr{N}(p_j)$ is a set of observations (since $\mathscr{N}(p_j) \subseteq \mathrm{Obs}(\Lambda)$). One obtains :

$\lfloor \varphi(I(L'_j) \bullet \mathscr{N}(p_j))\rfloor_\circ = \lfloor I(L'_j) \bullet \mathscr{N}(p_j)\rfloor \cap \mathrm{Obs}(\Lambda)$ - by 4.1.12 -

$= \lfloor I(L'_j) \bullet \mathscr{N}(p_j)\rfloor_\circ$ - from $I(L'_j) \bullet \mathscr{N}(p_j) \subseteq \mathrm{Obs}(\Lambda)$

$= \{O \in \mathrm{Obs}(\Lambda) \mid (\exists h \in K(L_j) \odot H_\Lambda(p_j))\ (O \leqslant h)\}$

– from condition v, definition of $\mathcal{N}$, and prop. 7 of app. 5 –

$= \{O \in Obs(\Lambda) \mid (\exists h \in K(L_j)\ \Theta \mathcal{H}_\Lambda(p_j))\ (O \leqslant h)\}$ – by app. 3.13 –

Last consider i in I. Using relation ii and by the induction on the structure of programs, it appears that $I(\mathcal{M}(p_i))$ is a set of complete pre-observations. Using relation v and by the definition of $\dashv\!\!\!\sqsubset$ (app.5.2), it also appears that $I(L'_i)$ is a set of incomplete pre-observations. One can verify from the definition of φ (app.4.15) that the equality $\lfloor \varphi(O \bullet O') \rfloor_\circ = \lfloor O \bullet \varphi(O') \rfloor_\circ$ for O and O' respectively being incomplete and complete pre-observations. Hence, $\lfloor \varphi(I(L'_i) \bullet I(\mathcal{M}(p_i))) \rfloor_\circ =$ $\lfloor I(L'_i) \bullet \varphi(I(\mathcal{M}(p_i))) \rfloor_\circ = \{O \in Obs(\Lambda) \mid (\exists h \in K(L_i)\ \Theta\ H_\Lambda(p_i))\ (O \leqslant h)\}$

– from condition 5, induction hypothesis, and by app. 5.7 –

$= \{O \in Obs(\Lambda) \mid (\exists h \in K(L_i)\ \Theta \mathcal{H}_\Lambda(p_i))\ (O \leqslant h)\}$ – by app. 3.13 –

Gathering the above results, one obtains :

$\lfloor \varphi(I(\mathcal{M}(p))) \rfloor_\circ = \{O \in Obs(\Lambda) \mid (\exists h \in \mathcal{H}_\Lambda(p))\ (O \leqslant h)\}$

$= \{O \in Obs(\Lambda) \mid (\exists h \in H_\Lambda(p))\ (O \leqslant h)\}$ – by app. 3.13 –

$= Obs_\Lambda(p)$ – by the definition of Obs_Λ –

The remaining of the proof is a straightforward application of propositions 11 of app. 4 and 4.1.12 (in the order) □

The proof of the crucial proposition 4.2.20 appeals to some rather complex lemmas which we introduce below as propositions 1 to 5. The main auxiliary result is stated in the

Proposition 1 Let (d,δ,ρ) be a maximal element of $Obs_\Lambda(p|q)$, then there exist observations $(d''_p,\delta''_p,\rho_p)$ and $(d''_q,\delta''_q,\rho_q)$ in $Obs_\Lambda(p)$ resp. $Obs_\Lambda(q)$ such that the following conditions are verified :

i) $\delta = \delta''_p \cap \delta''_q$

ii) $d = (d''_p \cap d''_q) \cup (d''_p \cap \delta''_q) \cup (\delta''_p \cap d''_q)$

iii) $\rho \setminus \chi \ \ \varepsilon \ \ (\rho_p \setminus \chi) \mid (\rho_q \setminus \chi)$

iv) $(d''_p,\delta''_p,\rho_p) * (d''_q,\delta''_q,\rho_q)$

Proof

• From the definition of Obs_Λ and by 3.3.7, there exist histories h_p and h_q in $H_\Lambda(p)$ resp. $H_\Lambda(q)$, let $h_p = (d_p,\delta_p,\rho_p)$ and $h_q = (d_q,\delta_q,\rho_q)$, which are compatible ($h_p \# h_q$) and satisfy the relations :

– $\rho \setminus \chi \ \ \varepsilon \ (\rho_p \setminus \chi) \mid (\rho_q \setminus \chi)$

– $\delta \subseteq \delta' \overset{\text{def}}{=} \delta_p \cap \delta_q$

- $d \setminus \delta' \subseteq d' \overset{\text{def}}{=} (d_p \cap d_q) \cup (d_p \cap \delta_q) \cup (\delta_p \cap d_q)$.

Let $O'_p = (d'_p, \delta'_p, \rho_p) \sqsubset h_p$ and $O'_q = (d'_q, \delta'_q, \rho_q) \sqsubset h_q$. Define $\delta'' = \delta'_p \cap \delta'_q$ and $d'' = (d'_p \cap d'_q) \cup (d'_p \cap \delta'_q) \cup (\delta'_p \cap d'_q)$.
We shall show that (d'', δ'', ρ) is a (complete) pre-observation.

According to the lemma 1 of app.5, the (complete) pre-observations O'_p and O'_q are given by :

$$\delta'_p = \delta_p \setminus \overline{\mathrm{Ult}(\rho_p)} \text{ , } d'_p = d_p \cup (\delta_p \setminus \delta'_p)$$

$$\delta'_q = \delta_q \setminus \overline{\mathrm{Ult}(\rho_q)} \text{ , } d'_q = d_q \cup (\delta_q \setminus \delta'_q) \text{ .}$$

The inclusions $\overline{\mathrm{Ult}(\rho_p)} \subseteq d_p \cup \delta_p$ and $\overline{\mathrm{Ult}(\rho_q)} \subseteq d_q \cup \delta_q$ hold by proposition 3.9, thus $\overline{\mathrm{Ult}(\rho_p)} \subseteq d'_p$ and $\overline{\mathrm{Ult}(\rho_q)} \subseteq d'_q$. Also notice that the compatibility of histories h_p and h_q validates the following implications, which will be used later on :

- $(\mu \notin d'_p \cup \delta'_p) \supset (\overline{\mu} \in \delta_q \wedge \mu \notin \mathrm{Ult}(\rho_q)) \supset (\overline{\mu} \in \delta'_q)$

- $(\mu \notin d'_q \cup \delta'_q) \supset (\overline{\mu} \in \delta_p \wedge \mu \notin \mathrm{Ult}(\rho_p)) \supset (\overline{\mu} \in \delta'_p)$

for any μ in $(\Lambda \cup \overline{\Lambda})$.

Now come back to our task. Since d'' and δ'' are clearly disjoint, we are left with proving $\mathrm{Ult}(\rho) \subseteq d''$ and $\overline{\mathrm{Ult}(\rho)} \cap \delta'' = \emptyset$.
- let μ in $\mathrm{Ult}(\rho)$, then either $\mu \in \mathrm{Ult}(\rho_p)$ or $\mu \in \mathrm{Ult}(\rho_q)$. Assume $\mu \in \mathrm{Ult}(\rho_p)$, then $\mu \in d_p \subseteq d'_p$ and by the compatibility of h_p and h_q $\mu \in (d_q \cup \delta_q) = (d'_q \cup \delta'_q)$, thus $\mu \in d''$. By the consideration of symmetry, it follows that the condition $\mathrm{Ult}(\rho) \subseteq d''$ is verified.
- let μ in $\overline{\mathrm{Ult}(\rho)}$, then either $\mu \in \overline{\mathrm{Ult}(\rho_p)} \subseteq d'_p$ or $\mu \in \overline{\mathrm{Ult}(\rho_q)} \subseteq d'_q$, whence $\mu \notin \delta''$ since $d'_p \cap \delta'_p = \emptyset = d'_q \cap \delta'_q$. The condition $\overline{\mathrm{Ult}(\rho)} \cap \delta'' = \emptyset$ is therefore verified.

• We shall now show that (d,δ,ρ) is one of the maximal lower bounds of (d'',δ'',ρ) in $\mathrm{Obs}(\Lambda)$. From the hypothesis, (d,δ,ρ) is a maximal element of $\mathrm{Obs}_\Lambda(p|q)$; by the construction of (d',δ',ρ), that history belongs to $(h_p|h_q)$; suppose that we can prove $(d,\delta,\rho) \lessdot (d'',\delta'',\rho) \lessdot (d',\delta',\rho)$, then the above fact is clearly established.
Relation $(d'',\delta'',\rho) \lessdot (d',\delta',\rho)$ holds from the obvious inclusion $\delta'' \subseteq \delta'$ and from the chain of equalities
$d'' \cup \delta'' = (d'_p \cup \delta'_p) \cap (d'_q \cup \delta'_q) = (d_p \cup \delta_p) \cap (d_q \cup \delta_q) = d' \cup \delta'$.

Suppose for a moment that $(d,\delta,\rho) \leqslant (d'',\delta'',\rho)$ does not hold, and show that this supposition is nonsense.

By the construction of d' and δ', one has $(d \cup \delta) \subseteq (d' \cup \delta') = (d'' \cup \delta'')$. It follows from the assumption $(d,\delta,\rho) \nleqslant (d'',\delta'',\rho)$ that there exists some label μ in δ which does not belong to δ''. Hence μ belongs also to $\delta' \setminus \delta''$ by the inclusion $\delta \subseteq \delta'$.

The equalities $\delta' \setminus \delta'' = (\delta_p \cap \delta_q) \setminus (\delta'_p \cap \delta'_q)$,
$\delta'_p = \delta_p \setminus \overline{\mathrm{Ult}(\rho_p)}$ and $\delta'_q = \delta_q \setminus \overline{\mathrm{Ult}(\rho_q)}$
clearly show that μ belongs to $\overline{\mathrm{Ult}(\rho_p)} \cup \overline{\mathrm{Ult}(\rho_q)}$.

Besides, $\mu \in \delta$ implies $\mu \notin \overline{\mathrm{Ult}(\rho)}$ - since $(d,\delta,\rho) \in \mathrm{Obs}(\Lambda)$ - By the definition of $(\rho_p \setminus \chi) | (\rho_q \setminus \chi)$, one of the following cases must occur :

- $\mu \in \overline{\mathrm{Ult}(\rho_p)}$ and $\mu \in \mathrm{Ult}(\rho_q)$, or
- $\mu \in \overline{\mathrm{Ult}(\rho_q)}$ and $\mu \in \mathrm{Ult}(\rho_p)$.

Consider the latter case for instance, then $\mu \in \mathrm{Ult}(\rho_p) \supset \mu \in d_p \supset \mu \notin \delta_p \supset \mu \notin \delta'$, a contradiction with $\mu \in \delta' \setminus \delta''$.
Hence $(d,\delta,\rho) \leqslant (d'',\delta'',\rho)$.

The above facts show that (d,δ,ρ) is one of the maximal lower bounds of (d'',δ'',ρ) in $\mathrm{Obs}(\Lambda)$, which property can be more precisely expressed by the statements
$\delta'' \setminus \overline{d}'' \subseteq \delta \subseteq \delta''$ and $d = (d'' \cup \delta'') \setminus (\delta \cup \overline{\delta})$
- cf prop. 4.1.12 together with defns. 12, 13, 15 of app. 4 -

- Put $\nabla = \delta'' \setminus \delta$ and define the following sets of labels :
- $d''_p = (d'_p \setminus \overline{\delta}'_p) \cup \{\mu, \overline{\mu} \mid \mu \in \overline{d}'_p \cap \nabla \text{ or } \mu \in \overline{d}'_p \cap (\delta'_p \setminus \delta'_q)\}$
- $\delta''_p = (\delta'_p \setminus \overline{d}'_p) \cup \{\mu \mid \mu \in \overline{d}'_p \cap (\delta'_p \cap \delta'_q) \setminus \nabla\}$
- $d''_q = (d'_q \setminus \overline{\delta}'_q) \cup \{\mu, \overline{\mu} \mid \mu \in \overline{d}'_q \cap \nabla \text{ or } \mu \in \overline{d}'_q \cap (\delta'_q \setminus \delta'_p)\}$
- $\delta''_q = (\delta'_q \setminus \overline{d}'_q) \cup \{\mu \mid \mu \in \overline{d}'_q \cap (\delta'_p \cap \delta'_q) \setminus \nabla\}$.

Finally let $O_p = (d''_p, \delta''_p, \rho_p)$ and $O_q = (d''_q, \delta''_q, \rho_q)$. Our next task is to prove that O_p and O_q are observations and moreover verify $O_p \leqslant h_p$ and $O_q \leqslant h_q$. By the consideration of symmetry, it suffices to establish these properties for O_p.
The emptiness of the intersections $d''_p \cap \delta''_p$ and $d''_p \cap \overline{\delta}''_p$ can be established by a careful analysis, using $d'_p \cap \delta'_p = \emptyset$ as an auxiliary property. Now consider μ in $\mathrm{Ult}(\rho_p)$, then :

$\mu \in d'_p$ since O'_p is a pre-observation, and

$\bar{\mu} \in d'_p$ since $\overline{\mathrm{Ult}(\rho_p)} \subseteq d'_p$ has been established in the first step of the proof,

thus $\{\mu,\bar{\mu}\} \subseteq d'_p \setminus \bar{\delta}'_p$ follows by $d'_p \cap \delta'_p = \emptyset$,

that is still $\{\mu,\bar{\mu}\} \subseteq d''_p$.

The above facts show that O_p is an observation, i.e. $O_p \in \mathrm{Obs}(\Lambda)$.

Let us prove $O_p \leqslant h_p$. The inclusions $\delta''_p \subseteq \delta'_p \subseteq \delta_p$ are obvious, so that there remains to check $d''_p \cup \delta''_p \subseteq d_p \cup \delta_p$. We shall prove $d''_p \cup \delta''_p = \delta'_p \cup d'_p \setminus (\bar{\delta}'' \setminus \bar{\nabla})$, which entails

$d''_p \cup \delta''_p \subseteq d'_p \cup \delta'_p = d_p \cup \delta_p$.

$$d''_p \cup \delta''_p = (d'_p \setminus \bar{\delta}'_p) \cup d'_p \cap \bar{\nabla} \cup d'_p \cap (\bar{\delta}'_p \setminus \bar{\delta}'_q)$$
$$\cup\, \bar{d}'_p \cap \nabla \cup \bar{d}'_p \cap (\delta'_p \setminus \delta'_q) \cup \bar{d}'_p \cap (\delta'_p \cap \delta'_q) \setminus \nabla$$
$$\cup\, \delta'_p \setminus \bar{d}'_p .$$

$$(d'_p \setminus \bar{\delta}'_p) \cup d'_p \cap \bar{\nabla} \cup d'_p \cap (\bar{\delta}'_p \setminus \bar{\delta}'_q) =$$
$$d'_p \setminus (\bar{\delta}'_p \cap \bar{\delta}'_q) \cup d'_p \cap \bar{\nabla} =$$
$$d'_p \setminus \bar{\delta}'' \cup d'_p \cap \bar{\nabla} =$$
$$d'_p \setminus (\bar{\delta}'' \setminus \bar{\nabla}) \quad \text{- from } \nabla \subseteq \delta'' \text{ -}$$

$$\bar{d}'_p \cap \nabla \cup \bar{d}'_p \cap (\delta'_p \setminus \delta'_q) \cup \bar{d}'_p \cap (\delta'_p \cap \delta'_q) \setminus \nabla =$$
$$\bar{d}'_p \cap \nabla \cup \bar{d}'_p \cap (\delta'_p \setminus \delta'_q) \cup \bar{d}'_p \cap (\delta'_p \cap \delta'_q) =$$
$$\bar{d}'_p \cap (\delta'_p \setminus \delta'_q) \cup \bar{d}'_p \cap (\delta'_p \cap \delta'_q) \quad \text{- from } \nabla \subseteq \delta'' \text{ -}$$
$$= \bar{d}'_p \cap \delta'_p .$$

Thus, $d''_p \cup \delta''_p = d'_p \setminus (\bar{\delta}'' \setminus \bar{\nabla}) \cup \bar{d}'_p \cap \delta'_p \cup \delta'_p \setminus \bar{d}'_p$

$= \delta'_p \cup d'_p \setminus (\bar{\delta}'' \setminus \bar{\nabla})$.

• Before checking O_p, O_q for conditions i to iv of the proposition, let us state some additional facts about ∇. From $\nabla = \delta'' \setminus \delta$ and $\delta'' \setminus \bar{d}'' \subseteq \delta \subseteq \delta''$, one draws $\nabla \subseteq \delta'' \cap \bar{d}''$, which validates the following assertions :

- $(\forall \mu \in \nabla)$ $(\mu \in \delta'_p$ and $\bar{\mu} \in (d'_p \cup \delta'_p))$
- $(\forall \mu \in \nabla)$ $(\mu \in \delta'_q$ and $\bar{\mu} \in (d'_q \cup \delta'_q))$

- $(\forall \mu \in \nabla)\ (\overline{\mu} \notin \delta'_p \cap \delta'_q)$ - by $d'' \cap \delta'' = \emptyset$ - .

In particular, $\nabla \cap \overline{\delta}'' = \emptyset = \nabla \cap \overline{\nabla}$, and

$d \cup \delta = d'' \setminus (\overline{\delta}'' \setminus \overline{\nabla}) \cup \delta''$

as proved by the chain of equalities

$d \cup \delta = (d'' \cup \delta'') \setminus (\delta \cup \overline{\delta}) \cup \delta'' \setminus \nabla$

$= d'' \setminus (\delta \cup \overline{\delta}) \cup \delta'' \setminus (\delta \cup \overline{\delta}) \cup \delta'' \setminus \nabla$

$= d'' \setminus (\delta'' \setminus \nabla \cup \overline{\delta}'' \setminus \overline{\nabla}) \cup \delta'' \setminus (\delta'' \setminus \nabla \cup \overline{\delta}'' \setminus \overline{\nabla}) \cup \delta'' \setminus \nabla$

$= d'' \setminus (\overline{\delta}'' \setminus \overline{\nabla}) \cup \nabla \setminus (\overline{\delta}'' \setminus \overline{\nabla}) \cup \delta'' \setminus \nabla$ - by $d'' \cap \delta'' = \emptyset$, $\nabla \subseteq \delta''$ -

$= d'' \setminus (\overline{\delta}'' \setminus \overline{\nabla}) \cup \nabla \cup \delta'' \setminus \nabla$ - by $\nabla \cap \overline{\delta}'' = \emptyset$ -

$= d'' \setminus (\overline{\delta}'' \setminus \overline{\nabla}) \cup \delta''$ - by $\nabla \subseteq \delta''$ -

- Verification of condition i of the proposition.

$\delta''_p \cap \delta''_q$ is given by the expression

$((\delta'_p \setminus \overline{d}'_p) \cup (\overline{d}'_p \cap (\delta'' \setminus \nabla))) \cap ((\delta'_q \setminus \overline{d}'_q) \cup (\overline{d}'_q \cap (\delta'' \setminus \nabla)))$.

We shall establish the converse inclusions

$\delta''_p \cap \delta''_q \subseteq \delta'' \setminus \nabla$, $\delta'' \setminus \nabla \subseteq \delta''_p \cap \delta''_q$,

whence $\delta''_p \cap \delta''_q = \delta$ - by $\delta = \delta'' \setminus \nabla$ -

- let $\mu \in \delta''_p \cap \delta''_q$, and suppose $\mu \notin \delta'' \setminus \nabla$.
The above expression of $\delta''_p \cap \delta''_q$ shows that μ must belong to $(\delta'_p \setminus \overline{d}'_p) \cap (\delta'_q \setminus \overline{d}'_q)$, and since $\delta'_p \cap \delta'_q = \delta''$, that property may be expressed equivalently by the assertion

$\mu \in (\delta'_p \setminus \overline{d}'_p) \cap (\delta'_q \setminus \overline{d}'_q) \cap \nabla$ - by $\mu \notin \delta'' \setminus \nabla$ -

$\mu \in \nabla$ now implies $\overline{\mu} \in (d'_p \cup \delta'_p) \cap (d'_q \cup \delta'_q)$,

which amounts to $\overline{\mu} \in \delta'_p \cap \delta'_q$ by the above assertion.
Hence $\mu \in \nabla \cap \overline{\delta}''$, which is contradicted by the fact $\nabla \cap \overline{\delta}'' = \emptyset$ (established in the previous step of the proof).

- let $\mu \in \delta'' \setminus \nabla$, hence $\mu \in \delta'_p \cap \delta'_q$.
If $\mu \notin \overline{d}'_p$, then $\mu \in \delta'_p \setminus \overline{d}'_p \subseteq \delta''_p$;
if $\mu \in \overline{d}'_p$, then $\mu \in \overline{d}'_p \cap (\delta'' \setminus \nabla) \subseteq \delta''_p$;
therefore $\mu \in \delta''_p$.
$\mu \in \delta''_q$ may be shown in a similar way,
hence $\mu \in \delta''_p \cap \delta''_q$.

- Verification of condition ii of the proposition.

One has to prove $(d''_p \cap d''_q) \cup (d''_p \cap \delta''_q) \cup (\delta''_p \cap d''_q) = d$.

The left member amounts to $((d''_p \cup \delta''_p) \cap (d''_q \cup \delta''_q)) \setminus (\delta''_p \cap \delta''_q)$.

The right member is equal to $(d \cup \delta) \setminus \delta$.

From condition i, it is enough to establish the relation $(d''_p \cup \delta''_p) \cap (d''_q \cup \delta''_q) = d \cup \delta$, which is equivalent to $(d''_p \cup \delta''_p) \cap (d''_q \cup \delta''_q) = \delta'' \cup d'' \setminus (\overline{\delta''} \setminus \overline{\nabla})$ – by the fourth step of the proof –

Now, the following equalities have been established in the third step of the proof :

$d''_p \cup \delta''_p = \delta'_p \cup d'_p \setminus (\overline{\delta''} \setminus \overline{\nabla})$,

$d''_q \cup \delta''_q = \delta'_q \cup d'_q \setminus (\overline{\delta''} \setminus \overline{\nabla})$.

Thus $(d''_p \cup \delta''_p) \cap (d''_q \cup \delta''_q) =$

$(\delta'_p \cap \delta'_q) \cup ((d'_p \cap d'_q) \cup (d'_p \cap \delta'_q) \cup (\delta'_p \cap d'_q)) \setminus (\overline{\delta''} \setminus \overline{\nabla}) =$

$\delta'' \cup d'' \setminus (\overline{\delta''} \setminus \overline{\nabla})$

- Verification of condition iii : immediate.
- Verification of condition iv of the proposition.

We shall first consider pairs of labels $(\mu, \overline{\mu})$ in $\Lambda \cup \overline{\Lambda}$ such that $\{\mu, \overline{\mu}\} \not\subseteq (d_p \cup \delta_p) \cap (d_q \cup \delta_q)$. We can freely assume $\mu \notin (d_p \cup \delta_p)$, since the other cases may be derived from that case through the consideration of symmetries. The compatibility of h_p and h_q then implies $\overline{\mu} \in \delta'_q$ (refer to step one).

By proposition 3.3.9, $\mu \notin d_p \cup \delta_p \Rightarrow \overline{\mu} \notin d_p$.

$\mu \notin d_p \Rightarrow \mu \notin \mathrm{Ult}(\rho_p) \Rightarrow \overline{\mu} \notin \delta_p \setminus \delta'_p$

(since $\delta'_p = \delta_p \setminus \overline{\mathrm{Ult}(\rho_p)}$).

On account of the equality $d'_p = d_p \cup (\delta_p \setminus \delta'_p)$, it follows that $\overline{\mu}$ does not belong to d'_p.

To sum up : $\mu \notin (d'_p \cup \delta'_p)$, $\overline{\mu} \in \delta'_q$, and $\overline{\mu} \notin d'_p$

(recalling that $d_p \cup \delta_p$ equals $d'_p \cup \delta'_p$) .

We proceed with separation of cases $\overline{\mu} \in \delta'_p$, $\overline{\mu} \notin \delta'_p$.

- $\underline{\overline{\mu} \in \delta'_p}$, that is still $\overline{\mu} \in \delta'_p \cap \delta'_q = \delta''$.

$\overline{\mu} \in \delta'_p \wedge \mu \notin (d'_p \cup \delta'_p) \Rightarrow \overline{\mu} \in \delta'_p \setminus \overline{d'}_p \Rightarrow \overline{\mu} \in \delta''_p$.

 - if $\mu \notin d'_q$, then $\overline{\mu} \in \delta'_q \setminus \overline{d'}_q \Rightarrow \overline{\mu} \in \delta''_q$,
 and thus $\overline{\mu} \in \delta''_p \cap \delta''_q$.

 - if $\mu \in d'_q$, then

 $\mu \notin (d'_p \cup \delta'_p) \Rightarrow \mu \notin d''$ (by the definition of d'', step 1)
 $\Rightarrow \overline{\mu} \notin \delta'' \cap \overline{d}''$
 $\Rightarrow \overline{\mu} \notin \nabla$ (by the inclusion $\nabla \subseteq \delta'' \cap \overline{d}''$, step 4)

$\Rightarrow \bar{\mu} \in \bar{d}'_q \cap (\delta'_p \cap \delta'_q) \setminus \nabla$

$\Rightarrow \bar{\mu} \in \delta''_q$ (by the definition of δ''_q, step 3),

and thus $\bar{\mu} \in \delta''_p \cap \delta''_q$.

- $\underline{\bar{\mu} \notin \delta'_p}$, that is still $\bar{\mu} \notin (d'_p \cup \delta'_p)$.

The compatibility of h_p and h_q then implies

$\mu \in \delta'_q$ (refer to step one) ; therefore,

$\{\mu,\bar{\mu}\} \subseteq \delta'_q$ and $\{\mu,\bar{\mu}\} \cap d'_q = \emptyset$,
whence $\{\mu,\bar{\mu}\} \subseteq \delta'_q \setminus \bar{d}'_q \subseteq \delta''_q$.

We consider now pairs of labels $(\mu,\bar{\mu})$ in $(\Lambda \cup \bar{\Lambda})$ such that $\{\mu,\bar{\mu}\} \subseteq (d_p \cup \delta_p) \cap (d_q \cup \delta_q)$, or equivalently, $\{\mu,\bar{\mu}\} \subseteq (d'_p \cup \delta'_p) \cap (d'_q \cup \delta'_q)$.

If $\{\mu,\bar{\mu}\} \subseteq \delta'_p$, then $\{\mu,\bar{\mu}\} \cap d'_p = \emptyset \Rightarrow \{\mu,\bar{\mu}\} \subseteq \delta'_p \setminus \bar{d}'_p \subseteq \delta''_p$.

If $\{\mu,\bar{\mu}\} \subseteq \delta'_q$, then $\{\mu,\bar{\mu}\} \cap d'_q = \emptyset \Rightarrow \{\mu,\bar{\mu}\} \subseteq \delta'_q \setminus \bar{d}'_q \subseteq \delta''_q$.

If $\{\mu,\bar{\mu}\} \subseteq d'_p \cap d'_q$, then $\{\mu,\bar{\mu}\} \cap \delta'_p = \emptyset = \delta'_q \cap \{\mu,\bar{\mu}\}$,

hence $\{\mu,\bar{\mu}\} \subseteq (d'_p \setminus \bar{\delta}'_p) \cap (d'_q \setminus \bar{\delta}'_q) \subseteq d''_p \cap d''_q$.

By the consideration of symmetries, the remains only three cases to examine

- $\underline{\mu \in d'_p \cap d'_q}$, $\underline{\bar{\mu} \in \delta'_p \cap \delta'_q}$.

Obviously, $\bar{\mu} \in \bar{d}'_p$ and $\bar{\mu} \in \bar{d}'_q$.

If $\bar{\mu} \notin \nabla$, then $\bar{\mu} \in \delta''_p \cap \delta''_q$ by the definitions of δ''_p, δ''_q.

If $\bar{\mu} \in \nabla$, then $\bar{\mu} \in \bar{d}'_p \cap \nabla$ and $\bar{\mu} \in \bar{d}'_q \cap \nabla$ imply

$\{\mu,\bar{\mu}\} \subseteq d''_p \cap d''_q$ by the definitions of d''_p ; d''_q.

- $\underline{\mu \in d'_p \cap \delta'_q}$, $\underline{\bar{\mu} \in \delta'_p \cap d'_q}$.

$\bar{\mu} \in d'_q \Rightarrow \bar{\mu} \notin \delta'_q \Rightarrow \bar{\mu} \in \bar{d}'_p \cap (\delta'_p \setminus \delta'_q) \Rightarrow \{\mu,\bar{\mu}\} \subseteq d''_p$;

$\mu \in d'_p \Rightarrow \mu \notin \delta'_p \Rightarrow \mu \in \bar{d}'_q \cap (\delta'_q \setminus \delta'_p) \Rightarrow \{\mu,\bar{\mu}\} \subseteq d''_q$;

thus $\{\mu,\bar{\mu}\} \subseteq d''_p \cap d''_q$.

- $\underline{\mu \in d'_p \cap d'_q}$, $\underline{\bar{\mu} \in d'_p \cap \delta'_q}$.

$\{\mu,\bar{\mu}\} \subseteq d'_p \Rightarrow \{\mu,\bar{\mu}\} \cap \delta'_p = \emptyset \Rightarrow \{\mu,\bar{\mu}\} \subseteq d'_p \setminus \bar{\delta}'_p \subseteq d''_p$;

$\bar{\mu} \in d'_p \Rightarrow \bar{\mu} \notin \delta'_p \Rightarrow \bar{\mu} \in \bar{d}'_q \cap (\delta'_q \setminus \delta'_p) \Rightarrow \{\mu,\bar{\mu}\} \subseteq d''_q$;

thus $\{\mu,\bar{\mu}\} \subseteq d''_p \cap d''_q$.

• Every conditions have been checked successfully, hence the proposition is now established. $\square$

<u>Definition 2</u> Given compatible observations $O_p \stackrel{def}{=} (d_p,\delta_p,\rho_p)$ and $O_q \stackrel{def}{=} (d_q,\delta_q,\rho_q)$, their parallel compound $O_p|O_q$ is the set of the observations (d,δ,ρ) which verify :

i) $\delta = \delta_p \cap \delta_q$

ii) $d = (d_p \cap d_q) \cup (d_p \cap \delta_q) \cup (\delta_p \cap d_q)$

iii) $\rho \setminus \chi \in (\rho_p \setminus \chi) | (\rho_q \setminus \chi)$.

<u>Proposition 3</u> Given programs p and q, let O_p in $Obs_\Lambda(p)$ and O_q in $Obs_\Lambda(q)$ such that $O_p * O_q$, then there exist corresponding histories h_p in $H_\Lambda(p)$ and h_q in $H_\Lambda(q)$ which are compatible ($h_p \# h_q$) and verify $O_p \leqslant h_p$, $O_q \leqslant h_q$ and $(\forall O \in (O_p|O_q))$ $(\exists h \in (h_p|h_q))$ $(O \leqslant h)$.

<u>Proof</u> Put $O_p = (d_p,\delta_p,\rho_p)$ and $O_q = (d_q,\delta_q,\rho_q)$. By the compatibility of O_p and O_q , the following chains of implications are valid for any label μ in $(\Lambda \cup \bar\Lambda)$:

$\mu \notin d_p \cup \delta_p \supset (\{\mu,\bar\mu\} \subseteq \delta_q$ or $\bar\mu \in \delta_p \cap \delta_q) \supset (\bar\mu \in \delta_q$ and $\mu \notin Ult(\rho_q))$

$\mu \notin d_q \cup \delta_q \supset (\{\mu,\bar\mu\} \subseteq \delta_p$ or $\bar\mu \in \delta_p \cap \delta_q) \supset (\bar\mu \in \delta_p$ and $\mu \notin Ult(\rho_p))$

- for $(d,\delta,\rho) \in Obs(\Lambda) \Rightarrow Ult(\rho) \subseteq d$ and $d \cap \bar\delta = \emptyset$ - .

By the definition of Obs_Λ , there exist h_p in $H_\Lambda(p)$ and h_q in $H_\Lambda(q)$ such that $O_p \leqslant h_p$ and $O_q \leqslant h_q$ - since O_p and O_q respectively belong to $Obs_\Lambda(p)$, $Obs_\Lambda(q)$ - . By proposition 14 of the third appendix, and from the defining properties of the ordering $\leqslant$, we are free to assume that h_p and h_q have respective forms $h_p = (d'_p, \delta'_p,\rho_p)$ and $h_q = (d'_q,\delta'_q,\rho_q)$.

The following conditions are then verified :

$\delta_p \subseteq \delta'_p$, $d_p \subseteq d'_p \cup \delta'_p$ - from $O_p \leqslant h_p$ -

$\delta_q \subseteq \delta'_q$, $d_q \subseteq d'_q \cup \delta'_q$ - from $O_q \leqslant h_q$ - .

The compatibility of h_p and h_q ($h_p \# h_q$) arises from the following implications, which are valid for any label $\mu \in (\Lambda \cup \bar\Lambda)$:

$\mu \notin (d'_p \cup \delta'_p) \supset \mu \notin (d_p \cup \delta_p) \supset (\bar\mu \in \delta_q \subseteq \delta'_q$ and $\mu \notin Ult(\rho_q))$

$\mu \notin (d'_q \cup \delta'_q) \supset \mu \notin (d_q \cup \delta_q) \supset (\bar\mu \in \delta_p \subseteq \delta'_p$ and $\mu \notin Ult(\rho_p))$.

In order to prove $(\forall O \in (O_p|O_q))$ $(\exists h \in (h_p|h_q))$ $(O \leqslant h)$,

it is enough to establish the inclusions :

. $(\delta_p \cap \delta_q) \subseteq (\delta'_p \cap \delta'_q)$ - immediate -

. $(d_p \cap d_q) \cup (d_p \cap \delta_q) \cup (\delta_p \cap d_q) \subseteq (d'_p \cup \delta'_p) \cap (d'_q \cup \delta'_q)$.

Now, the latter inclusion follows from

$$(d_p \cup \delta_p) \cap (d_q \cup \delta_q) \subseteq (d'_p \cup \delta'_p) \cap (d'_q \cup \delta'_q) \; . \qquad \Box$$

<u>Proposition 4</u> Let O_1, O_2, $O'_1 \in \mathrm{Obs}(\Lambda)$.

If $O_2 * O_1$ and $O_1 \preccurlyeq O'_1$, then $O'_1 * O_2$.

<u>Proof</u> Put $O_1 = (d_1,\delta_1,\rho_1)$, $O_2 = (d_2,\delta_2,\rho_2)$, $O'_1 = (d'_1,\delta'_1,\rho'_1)$. By the compatibility of O_1 and O_2, the following condition is verified for any label $\mu \in (\Lambda \cup \bar{\Lambda})$:

$\{\mu,\bar{\mu}\} \subseteq \delta_1$ or $\{\mu,\bar{\mu}\} \subseteq \delta_2$ or $\mu \in \delta_1 \cap \delta_2$ or $\bar{\mu} \in \delta_1 \cap \delta_2$ or $\{\mu,\bar{\mu}\} \subseteq d_1 \cap d_2$.

Suppose that the last alternative $\{\mu,\bar{\mu}\} \subseteq d_1 \cap d_2$ is verified for some particular label μ :

$O_1 \preccurlyeq O'_1 \Rightarrow d_1 \subseteq d'_1 \cup \delta'_1$

$O'_1 \in \mathrm{Obs}(\Lambda) \Rightarrow d'_1 \cap \bar{\delta}'_1 = \emptyset$

thus $(\{\mu,\bar{\mu}\} \subseteq d'_1$ or $\{\mu,\bar{\mu}\} \subseteq \bar{\delta}'_1)$.

Using the inclusion $\delta_1 \subseteq \delta'_1$ (another consequence of $O_1 \preccurlyeq O'_1$), it can therefore be verified that the following condition holds for any label μ :

$\{\mu,\bar{\mu}\} \subseteq \delta'_1$ or $\{\mu,\bar{\mu}\} \subseteq \delta_2$ or $\mu \in \delta'_1 \cap \delta_2$ or $\bar{\mu} \in \delta'_1 \cap \delta_2$ or $\{\mu,\bar{\mu}\} \subseteq d'_1 \cap d_2$.

Hence $O'_1 * O_2$. $\Box$

<u>Proposition 5</u> The parallel composition of compatible observations, as given in definition 2, is a monotonously increasing operation as more precisely stated below.

Let O_1, O_2 and O'_1, O'_2 in $\mathrm{Obs}(\Lambda)$ s.t. $O_1 * O_2$, $O_1 \preccurlyeq O'_1$, $O_2 \preccurlyeq O'_2$:

for any $O \in (O_1|O_2)$, there exists $O' \in (O'_1|O'_2)$ such that $O \preccurlyeq O'$.

<u>Proof</u> For obvious reasons, it is enough to show that the parallel composition of observations is monotonously increasing in the first argument.

Put $O_1 = (d_1,\delta_1,\rho_1)$, $O_2 = (d_2,\delta_2,\rho_2)$, $O'_1 = (d'_1,\delta'_1,\rho'_1)$. We proceed with case analysis.

<u>case 1</u> $\rho_1 = \rho'_1$.

$O_1 \preccurlyeq O'_1 \Rightarrow \delta_1 \subseteq \delta'_1 \Rightarrow \delta_1 \cap \delta_2 \subseteq \delta'_1 \cap \delta_2$.

$O_1 \preccurlyeq O'_1 \Rightarrow (d_1 \cup \delta_1) \subseteq (d'_1 \cup \delta'_1)$, whence

$(d_1 \cap d_2) \cup (d_1 \cap \delta_2) \cup (\delta_1 \cap d_2) \subseteq (d'_1 \cup \delta'_1) \cap (d_2 \cup \delta_2)$

The above facts may be considered as a sufficient proof for case 1.

<u>case 2</u> $\rho_1 \neq \rho'_1$.

$O_1 \preccurlyeq O'_1 \Rightarrow (d_1 \cup \delta_1) = \emptyset$ and $(\rho_1 \setminus \chi) \prec (\rho'_1 \setminus \chi)$.

$((O_1 * O_2)$ and $(d_1 \cup \delta_1) = \emptyset) \Rightarrow \delta_2 = (\Lambda \cup \overline{\Lambda})$.

Thus $d_2 = \emptyset$ and ρ_2 must be a finite sequence.

Let $O \in (O_1 | O_2)$: since $(d_1 \cup \delta_1) = \emptyset$, O can be written $(\emptyset, \emptyset, \rho\ \chi)$ for some word ρ in $((\rho_1 \setminus \chi) | (\rho_2 \setminus \chi))$. Since $(\rho_2 \setminus \chi)$ is finite and $(\rho_1 \setminus \chi) \prec (\rho'_1 \setminus \chi)$,there certainly exists ρ' in $(\rho'_1 \setminus \chi) | (\rho_2 \setminus \chi)$ such that $\rho \prec \rho'$ and thus $\rho \prec \rho'\ \chi$.

Define $O' = (d'_1 \cap d_2 \cup d'_1 \cap \delta_2 \cup \delta'_1 \cap d_2,\ \delta'_1 \cap \delta_2,\ \rho'\ \chi)$, then $O' \in (O'_1 | O'_2)$ - by construction -, and $O \preccurlyeq O'$. □

We are now ready to give the

<u>Proof of proposition 4.2.20</u>

By propositions 1 and 3, $Obs_\Lambda(p|q)$ is equal to $\lfloor \check{L}_p \| \check{L}_q \rfloor_\circ$ letting $\check{L}_p = \lfloor L_p \rfloor_\circ$ and $\check{L}_q = \lfloor L_q \rfloor_\circ$.

By propositions 4 and 5, the following assertion is valid :

$(\forall O \in \check{L}_p \| \check{L}_q)\ (\exists O' \in L_p \| L_q)\ (O \preccurlyeq O')$.

Thus $Obs_\Lambda(p|q) = \lfloor L_p \| L_q \rfloor_\circ$.

The remaining of the proof is a straightforward application of the results shown in the appendix 2 : the parallel composition of rational response languages is an effective operator, and this property naturally extends to the parallel composition of languages of observations. □

<u>Proof of proposition 4.2.21</u>

Let notations as follows : $E = (\Lambda' \cup \overline{\Lambda}')$, $F = \Lambda \setminus E$, $G = (\Lambda \cup \overline{\Lambda}) \cap E$, and $X = L_q \Downarrow E$, $Y = X \uparrow G$, $Z = (H_{\Lambda''}(q)) \downarrow E$, $W = Z \uparrow G$.

- We start with proving $X \subseteq Obs(F)$ and $Y \subseteq Obs(\Lambda)$.

Let $O \in X$, then one of the following cases must occur :

- $O = (\emptyset, \emptyset, \rho\chi)$ and there exists $(d', \delta', \rho') \in L_q$ such that $\rho \in (pref\ (\rho') \cap (M \setminus E)^*)$.

$L_q \subseteq Obs(\Lambda'') \Rightarrow \rho' \in (\Lambda'')^\infty \chi$, thus $\rho \in (\Lambda'' \setminus E)^*$.

Since $\Lambda'' \setminus E = F$, it follows that $(\emptyset, \emptyset, \rho\chi) \in Obs(F)$.

- $O = (d \setminus E, \delta \setminus E, \rho)$ with $(d, \delta, \rho) \in L_q$ and $\rho \in (M \setminus E)^\infty \chi$.

From $L_q \subseteq Obs(\Lambda'')$, one draws :

$(d \cap (\delta \cup \overline{\delta})) = \emptyset \Rightarrow (d \setminus E \cap (\delta \setminus E \cup \overline{\delta} \setminus \overline{E})) = \emptyset$,

$d \cup \delta \subseteq \Lambda'' \cup \overline{\Lambda}'' \Rightarrow (d \setminus E \cup \delta \setminus E) \subseteq (\Lambda'' \cup \overline{\Lambda}'') \setminus E = (F \cup \overline{F})$,

$\rho \in F^\infty \chi$ and $Ult(\rho) \subseteq d \Rightarrow Ult(\rho) \subseteq d \cap F \subseteq d \setminus E$.

Hence $O \in Obs(F)$.

Let $O \in Y$, then O may be written $(d, \delta \cup G, \rho)$ with $(d,\delta,\rho) \in X \subseteq Obs(F)$.
$O \in Obs(\Lambda)$ immediately follows from $\delta \cup G \subseteq (F \cup \bar{F}) \cup G = (\Lambda \cup \bar{\Lambda}) \setminus E \cup (\Lambda \cup \bar{\Lambda}) \cap E = \Lambda \cup \bar{\Lambda}$.

• We now prove $Y \subseteq Obs_\Lambda(p)$.

Let $O \in Y$, then one of the following cases i or ii must occur.
i) $O = (\emptyset, G, \rho\chi)$ and there exists $(d',\delta',\rho') \in L_q$ such that $\rho \in pref(\rho') \cap F^*$.
$(d',\delta',\rho') \in L_q \Rightarrow (\exists (d'',\delta'',\rho'') \in H_{\Lambda''}(q))((d',\delta',\rho') \leqslant (d'',\delta'',\rho''))$,
hence $\rho \in (pre(\rho'') \cap F^*)$ by the definition of the ordering.
By proposition 13 of the third appendix,
$((d'',\delta'',\rho'') \in H_{\Lambda''}(q)$ and $\rho \prec \rho'') \Rightarrow$

$(\exists d''',\delta''')((d''',\delta''',\rho\chi) \in H_{\Lambda''}(q))$.

$((d''',\delta''',\rho\chi) \in H_{\Lambda''}(q)$ and $\rho \in F^*) \Rightarrow$

$(d'''\setminus E, \delta'''\setminus E, \rho\chi) \in Z \Rightarrow$

$(d'''\setminus E, \delta'''\setminus E \cup G, \rho\chi) \in W$.
Now $W = H_\Lambda(p)$ and $(\emptyset, G, \rho\chi) \in Y \subseteq Obs(\Lambda)$, whence $(\emptyset, G, \rho\chi) \in Obs_\Lambda(p)$ on the account of the obvious relation $(\emptyset, G, \rho\chi) \leqslant (d'''\setminus E, \delta'''\setminus E \cup G, \rho\chi)$.

ii) $O = (d'\setminus E, \delta'\setminus E \cup G, \rho')$ with $(d',\delta',\rho') \in L_q$.
$(d',\delta',\rho') \in L_q \Rightarrow (\exists (d'',\delta'',\rho'') \in H_{\Lambda''}(q))((d',\delta',\rho') \leqslant (d'',\delta'',\rho''))$.
If $\rho' = \rho''$, then (d',δ',ρ') may also be written $(\emptyset, \emptyset, \rho\chi)$, and the remaining of the proof is as in the above case i. Now assume $\rho' \neq \rho''$. $O \in Y \Rightarrow \rho'' \in F^\infty \chi \subseteq (M\setminus E)^\infty \chi$, thus $(d''\setminus E, \delta''\setminus E, \rho'') \in Z$ and $(d''\setminus E, \delta''\setminus E \cup G, \rho'') \in W$.

From $(d',\delta',\rho') \leqslant (d'',\delta'',\rho'')$, one draws :
$\delta' \subseteq \delta'' \Rightarrow \delta'\setminus E \cup G \subseteq \delta''\setminus E \cup G$,
$d' \subseteq d'' \cup \delta'' \Rightarrow (d'\setminus E) \subseteq (d''\setminus E) \cup (\delta''\setminus E \cup G)$.
Hence $O \leqslant (d''\setminus E, \delta''\setminus E \cup G, \rho'') \in W$.
By the definition of Obs_Λ, $O \in Obs_\Lambda(p)$ follows from $W = H_\Lambda(p)$ and $O \in Y \subseteq Obs(\Lambda)$.

• In order to complete the proof, there remains to show that every maximal elements of $Obs_\Lambda(p)$ also belong to Y.

Let O be a maximal element in the set $Obs_\Lambda(p)$.
By the definition of Obs_Λ, there exists h in $H_\Lambda(p)$ such that $O \leqslant h$. Let h denote one of the maximal histories which verify those assertions.
Since W equals $H_\Lambda(p)$, there exists some maximal element h' in Z, let $h' = (d',\delta',\rho')$, such that $h = (d',\delta' \cup G,\rho')$. By the definition of Z, there also exists h'' in $H_{\Lambda''}(q)$, let $h'' = (d'',\delta'',\rho'')$ such that $(d',\delta',\rho') = (d''\setminus E, \delta''\setminus E,\rho'')$ and thus $\rho'' = \rho' \in F^\infty \chi$.

For any observation (d,δ,ρ) in Obs(F) such that $(d,\delta,\rho) \preccurlyeq (d',\delta',\rho')$, $(d, \delta \cup G,\rho)$ belongs to Obs(Λ) and verifies $(d,\delta \cup G,\rho) \preccurlyeq (d',\delta' \cup G,\rho')$. Since O is maximal in $Obs_{\Lambda}(p)$ and since h belongs to $H_{\Lambda}(p)$, it is clear that O can be written $(d,\delta \cup G,\rho)$ with $(d,\delta,\rho) \in Obs(F)$ and $(d,\delta,\rho) \preccurlyeq (d',\delta',\rho')$. By the above remark, it follows that (d,δ,ρ) is one of the maximal lower bounds of h' in the set Obs(F). Now, $(d',\delta',\rho') \preccurlyeq (d'',\delta'',\rho'')$ holds for obvious reasons, whence also : $(d,\delta,\rho) \preccurlyeq (d'',\delta'',\rho'') \in H_{\Lambda''}(q)$.

We proceed with separate analysis of the two possible cases in which that last relation can hold.

i) $(d \cup \delta) = \emptyset$, $\rho \in F^{*}\chi$, $\rho\setminus\chi \prec \rho'' \in F^{\infty}\chi$.

$(d'',\delta'',\rho'') \in H_{\Lambda''}(q) \Rightarrow (\emptyset, \emptyset,\rho) \in Obs_{\Lambda''}(q)$, since $(\emptyset, \emptyset,\rho) \in Obs(\Lambda'')$. From $Obs_{\Lambda''}(q) = \lfloor L_q \rfloor_{\circ}$, one draws $(\exists (d''',\delta''',\rho''') \in L_q)\ ((\emptyset, \emptyset,\rho) \preccurlyeq (d''',\delta''',\rho'''))$. $\rho \in F^{*}\chi$, $F = (\Lambda \setminus E)$ and $\rho\setminus\chi \prec \rho'''$ now imply that $(\emptyset, \emptyset,\rho)$ belongs to $L_q \Downarrow E$ by the definition of $\Downarrow$, and one therefore obtains $O = (\emptyset , G,\rho) \in (L_q \Downarrow E) \uparrow G = Y$.

ii) $(d \cup \delta) \neq \emptyset$, $d \subseteq d'' \cup \delta''$, $\delta \subseteq \delta''$, and $\rho = \rho'' \in F^{\infty}\chi$.

Since $h' = (d''\setminus E, \delta''\setminus E, \rho'')$ and $F = \Lambda\setminus E$, any lower bound of h" in Obs(F) is also a lower bound of h'. Now, (d,δ,ρ) is one of the maximal lower bounds of h' in the set Obs(F). From $(d,\delta,\rho) \preccurlyeq h''$, it therefore follows that (d,δ,ρ) is one of the maximal lower bounds of (d'',δ'',ρ'') in the set Obs(F).

Suppose that (d'',δ'',ρ'') is not maximal in the set $H_{\Lambda''}(q)$. In that case, let $(d''',\delta''', \rho''')$ be one of the maximal upper bounds of h" in $H_{\Lambda''}(q)$, then $\rho''' = \rho''$ since $d'' \cup \delta'' \neq \emptyset$, and $h' = (d''\setminus E, \delta''\setminus E, \rho'') \preccurlyeq (d'''\setminus E, \delta'''\setminus E, \rho'') \in Z$. But h' is a maximal element of Z, and we therefore obtain $h' = (d'''\setminus E, \delta'''\setminus E, \rho'')$. As a consequence, we can freely assume that h" is a maximal element of $H_{\Lambda''}(q)$ (for h" can be freely replaced by (d''',δ''', ρ'')) . Now recall that (d,δ,ρ) is one of the maximal lower bounds of h" in the set Obs(F). Since $\rho = \rho''$ and $(d \cup \delta) \neq \emptyset$, the above statement implies that (d,δ,ρ) equals $(d'''\setminus E, \delta'''\setminus E, \rho)$ for (d''', δ''', ρ) some of the maximal lower bounds of h" in Obs(Λ'') (which is clear from the fact that $(d'''\setminus E, \delta'''\setminus E, \rho)$ belongs to the set Obs(F) for any such triple (d''',δ''',ρ)). Since $h'' \in H_{\Lambda''}(q)$, (d''', δ''', ρ) belongs to $Obs_{\Lambda''}(q)$ which is equal to $\lfloor L_q \rfloor_{\circ}$. As a consequence, there exists (D,Δ,R) in L_q such that $(d''',\delta''',\rho) \preccurlyeq (D,\Delta,R)$, whence $R = \rho$ (for $(d''' \cup \delta''')$ differs from $\emptyset$).

$(d,\delta,\rho) = (d'''\setminus E, \delta'''\setminus E,\rho)$ and $(d''', \delta''', \rho) \preccurlyeq (D,\Delta,\rho)$ clearly imply $(d,\delta,\rho) \preccurlyeq (D\setminus E, \Delta\setminus E,\rho) \in X$, and thus also $(d, \delta \cup G,\rho) \preccurlyeq (D\setminus E,\Delta\setminus E \cup G,\rho) \in Y$.

Now, Y is included in $Obs_{\Lambda}(p)$, and $(d,\delta \cup G,\rho)$ equals O which is a maximal element of $Obs_{\Lambda}(p)$.

Hence $O = (D\setminus E, \Delta\setminus E \cup G,\rho) \in Y$. □

By the induction on the structure of programs, the proof of proposition 4.2.22 is quite obvious from propositions 4.2.17, 4.2.20 and 4.2.21. Nevertheless, those propositions have not been completely proved, since it may be observed that we have omitted to verify the non-emptiness of $Obs_\Lambda(C_i)$ for each of the corresponding constructs C_i. Proposition 11 of the third appendix allows to remedy this insufficiency, since it shows that for any program p of sort Λ, $H_\Lambda(p)$ contains at least one element h which is of course an upper bound of $(\emptyset, \emptyset, \chi) \varepsilon\ Obs(\Lambda)$.

Our next series of propositions prepares the way for demonstrating proposition 4.2.23 .

<u>Proposition 6</u> Let p be an elementary program of sort Λ. Let q be a proper subprogram of p. Assume that q is not a proper subprogram of the other subprograms of p which result from flow-operations. If $Obs_\Lambda(q) = Obs_\Lambda(q')$, then $Obs_\Lambda(p) = Obs_\Lambda(p[\![q' \setminus q]\!])$.
– where $[\![q' \setminus q]\!]$ means the syntactic substitution of q' for q –

<u>Proof</u> By proposition 4.2.17 together with definition 4.2.16, $Obs_\Lambda(p)$ depends upon q exactly in the way that $\lfloor \varphi(I\ (Z\ \mathcal{N}(q)))\rfloor_\circ$ depends upon q for $Z \in Rat(\text{pre-obs}(\Lambda)^+)$.

Since I is a monoid homomorphism from $(\text{pre-Obs}(\Lambda)^\infty)$ onto $\text{P-obs}(\Lambda)$, $I(Z\mathcal{N}(q))$ equals $I(Z)\bullet I(\mathcal{N}(q))$ for any such Z. By proposition 4.1.12, $\lfloor \varphi(I(Z)\bullet I(\mathcal{N}(q)))\rfloor_\circ$ is equal to $\lfloor (I(Z)\bullet I(\mathcal{N}(q)))\rfloor \cap Obs(\Lambda)$, and therefore to $Obs(\Lambda) \cap (\lfloor I(Z)\rfloor \bullet \lfloor I(\mathcal{N}(q))\rfloor)$ – by prop. 7 of app. 4 –. Let $L_q = \varphi(I(\mathcal{N}(q)))$. According to definition 4.2.16, L_q equals $\mathcal{N}(q)$ if q is the result of a flow-operation. Appealing to proposition 4.1.12 in the converse case, we can conclude from the definition of $\mathcal{N}$ that $\lfloor L_q \rfloor_\circ$ equals $\lfloor I(\mathcal{N}(q))\rfloor \cap Obs(\Lambda)$ in both situations. It follows by proposition 16 of the fourth appendix that one has $Obs(\Lambda) \cap (\lfloor I(Z)\rfloor \bullet \lfloor I(\mathcal{N}(q))\rfloor) = Obs(\Lambda) \cap (\lfloor I(Z)\rfloor \bullet \lfloor L_q \rfloor)$.
Now, $\lfloor L_q \rfloor$ equals $\lfloor L'_q \rfloor$ for any set of observations L'_q which verifies $\lfloor L'_q \rfloor_\circ = \lfloor \varphi(I(\mathcal{N}(q)))\rfloor_\circ$, or yet equivalently $\lfloor L'_q \rfloor_\circ = Obs_\Lambda(q)$ (from the definition of $\mathcal{N}$ and by proposition 4.2.17).
It is now clear from the above facts that $Obs_\Lambda(q) = Obs_\Lambda(q')$ implies $\lfloor \varphi(I(Z\mathcal{N}(q)))\rfloor_\circ = \lfloor \varphi(I(Z\ \mathcal{N}(q')))\rfloor_\circ$, hence the proposition has been proved. □

<u>Proposition 7</u> Let Λ, Λ' be finite subsets of M s.t. $\Lambda \subseteq \Lambda'$, and let p denote a program s.t. $MS(p) \subseteq \Lambda$. Then $H_{\Lambda'}(p) = (H_\Lambda(p))\uparrow \nabla$ with $\nabla = (\Lambda' \cup \overline{\Lambda'}) \setminus (\Lambda \cup \overline{\Lambda})$.

<u>Proof</u> We shall establish the above fact by induction on the structure of programs, using the alternative characterization of H_Λ given in the third appendix (definitions 7 to 9 and theorem 13). The induction step is by case analysis. The induction basis is a particular instance of the first case in the induction step which we detail now.

<u>case 1</u> p is an elementary program.
Let $\{p_1 \ldots p_n\}$ denote the set made out of the outermost proper subprograms of p which are defined by recursion Y or by flow-operations. By the induction hypothesis,

$H_{\Lambda'}(p_i) = (H_\Lambda(p_i))\uparrow\nabla$ for any p_i in that set (whose possible emptiness gives a basis to the induction). Let us naturally extend function $\uparrow\nabla$ from $H(\Lambda)$ to pre-$H(\Lambda)$ and then accordingly to $(\text{pre-}H(\Lambda))^\infty$.
(i.e. $(h_1 h_2 \ldots h_i \ldots)\uparrow\nabla = ((h_1\uparrow\nabla)(h_2\uparrow\nabla)\ldots(h_i\uparrow\nabla)\ldots)$). Looking at definition 7 of app. 3, it then appears that there exists some language $\mathcal{L}$ in $\mathcal{P}(\text{pre-}H(\Lambda)^\infty)$ which verifies $H_\Lambda(p) = K(\mathcal{L})$ and $H_{\Lambda'}(p) = K(\mathcal{L}\uparrow\nabla)$.
$H_{\Lambda'}(p) = (H_\Lambda(p))\uparrow\nabla$ follows directly by the definition of the homomorphism K (defn. 5 in app. 3).

<u>case 2</u> $p \equiv q\ [/\mu_1\ldots\mu_n]$.

Put the following notations : $R = \{\mu_1\ldots\mu_n\}\cup\{\bar{\mu}_1\ldots\bar{\mu}_n\}$, $S = (\Lambda\cup\bar{\Lambda})\cap R$, $T = \Lambda\cup R$ and $S' = (\Lambda'\cup\bar{\Lambda}')\cap R$, $T' = \Lambda'\cup R$. According to definition 9 in app.3 , $H_\Lambda(p)$ and $H_{\Lambda'}(p)$ are respectively equal to $(H_T(q)\downarrow R)\uparrow S$ and $(H_{T'}(q)\downarrow R)\uparrow S'$.
By the induction hypothesis, $H_{T'}(q) = H_T(q)\uparrow((T'\cup\bar{T}')\setminus(T\cup\bar{T}))$. Thus $H_{\Lambda'}(p) = ((H_T(q)\uparrow\nabla')\downarrow R)\uparrow S'$, letting $\nabla' = (T'\cup\bar{T}')\setminus(T\cup\bar{T})$. Now, $\nabla'\cap R = \emptyset$ implies $(H_T(q)\uparrow\nabla')\downarrow R = (H_T(q)\downarrow R)\uparrow\nabla'$, hence $H_{\Lambda'}(p) = (H_T(q)\downarrow R)\uparrow(\nabla'\cup S')$.
Looking at the definitions, one obtains

$\nabla'\cup S' = ((\Lambda'\cup\bar{\Lambda}')\setminus(\Lambda\cup\bar{\Lambda}))\setminus R\cup((\Lambda'\cup\bar{\Lambda}')\cap R)$
$= (\Lambda'\cup\bar{\Lambda}')\setminus(\Lambda\cup\bar{\Lambda})\cup((\Lambda'\cup\bar{\Lambda}')\cap R)$
$= (\Lambda'\cup\bar{\Lambda}')\setminus(\Lambda\cup\bar{\Lambda})\cup((\Lambda\cup\bar{\Lambda})\cap R)$ - from $\Lambda\subseteq\Lambda'$
$= \nabla\cup S$.

$H_{\Lambda'}(p)$ is therefore equal to $((H_T(q)\downarrow R)\uparrow S)\uparrow\nabla$, that is still to $H_\Lambda(p)\uparrow\nabla$.

<u>case 3</u> $p \equiv (q|r)$

The step of induction is immediate from definitions 3.3.3 and 3.3.6 . □

<u>Proposition 8</u> Let Λ, Λ' be finite subsets of M s.t. $\Lambda\subseteq\Lambda'$, and let p denote a program s.t. $\Lambda\subseteq MS(p)$.
Then $Obs_\Lambda(p) = \lfloor\sum_i(d_i,\delta_i,\mathcal{L}_i)\rfloor_\circ$ <u>if and only if</u>
$Obs_{\Lambda'}(p) = \lfloor\sum_i(d_i,\delta_i\cup\nabla,\mathcal{L}_i)\rfloor_\circ$ with ∇ equal to $(\Lambda'\cup\bar{\Lambda}')\setminus(\Lambda\cup\bar{\Lambda})$.

<u>Proof</u> a straightforward corollary of proposition 7. □

By the induction on the structure of program-contexts, the proof of proposition 4.2.23 follows directly from the propositions 8 and 6 above together with propositions 4.2.20 and 4.2.21 .

Define $p\sim q$ iff $Obs_\Lambda(p) = Obs_\Lambda(q)$ for $\Lambda = MS(p)\cup MS(q)$. By proposition 8, equivalent definitions of $\sim$ may be : $p\sim q$ iff $Obs_\Lambda(p) = Obs_\Lambda(q)$ for some Λ s.t. $MS(p)\cup MS(q)\subseteq\Lambda$, $p\sim q$ iff $Obs_\Lambda(p) = Obs_\Lambda(q)$ for every Λ s.t. $MS(p)\cup MS(q)\subseteq\Lambda$.
These alternative characterizations show that $\sim$ is in fact an equivalence relation over the global set PROG of programs of every possible sorts. Hence $\sim$ is also a congruence over PROG (evidence of this fact is given by prop. 4.2.23).

We are left with proving our final theorem 4.2.24, which amounts to state the following remarks.

- Given programs p and q, the least upper bound of their respective minimal sorts, let $\Lambda = MS(p) \cup MS(q)$, is effectively computable.
- The semantic function Obs_Λ is effectively computable, and $Obs_\Lambda(p)$, $Obs_\Lambda(q)$ can be obtained in the rational form $\lfloor E \rfloor_\circ$, $E \equiv \sum_{i \varepsilon I} (d_i, \delta_i, \mathcal{L}_i)$, with $(d_i, \delta_i) \neq (d_j, \delta_j)$ for $i \neq j$.
- Given E as above, another rational expression $\tilde{E}$, such that $\lfloor E \rfloor_\circ = \lfloor \tilde{E} \rfloor_\circ$ and $(\forall O_1, O_2 \ \varepsilon \ \tilde{E}) \ (O_1 \leqslant O_2 \Rightarrow O_1 = O_2)$, can be computed in the form $\tilde{E} \equiv \sum_{i \varepsilon I} (d_i, \delta_i, \tilde{\mathcal{L}}_i)$

through the following process :

1) if $d_i = \delta_i = \emptyset$ for some i, then replace $\mathcal{L}_i$ by $\mathcal{L}_i \setminus ((plf(plf(\mathcal{L}_i)) \cup (\bigcup_{j \neq i} lf \ (\mathcal{L}_j)))\chi)$

(where plf(L), resp. lf(L), denotes the set of the proper left factors, resp. left factors, of words ϕ in L).

2) while there exist different indexes i and j in I such that $d_i \subseteq d_j \cup \delta_j$, $\delta_i \subseteq \delta_j$ and $\mathcal{L}_i \cap \mathcal{L}_j \neq \emptyset$, replace $\mathcal{L}_i$ by $\mathcal{L}_i \setminus \mathcal{L}_j$.

- Given $\tilde{E}$ and $\tilde{E}'$ obtained from the above process, the decision of $\lfloor \tilde{E} \rfloor_\circ = \lfloor \tilde{E}' \rfloor_\circ$ amounts to the decision of $\tilde{E} = \tilde{E}'$.

- There exists a decision procedure for the equality of infinitary rational languages over $(\Lambda \cup \{\chi\})$. $\square$.

APPENDIX 7 : FINAL COMMENTS.

We give here some justifications for the directions that have been taken in the study, and we point out at the same time the inheritance and the departure from other studies.

* First of all, we have to explain why the rational subset of pure CCS has been considered instead of full CCS. A convincing argument needs a preliminary recall of some important results about unbounded nondeterminacy.

In(Ch), Chandra has proved that almost every set in Σ_1^1 coincides with the family of inputs which may cause to diverge a corresponding program run on a register machine with random assignment. For an equivalent class of programs with random assignments and <u>or</u> and <u>while</u> statements, Park has shown in (Pa) that the set of inputs which ensure the convergence of a particular program is in general the least fixed point of a <u>weakly</u> continuous function. The latter work also studies the language obtained when taking away the random assignment and providing the <u>par</u> construct in exchange. A direct relational semantics has been built for that language, based upon the fair merge of infinite sequences. The fair merge appears as an adequate combination of least and greatest fixed points of weakly continuous functions (we shall come back over that fact later on).

A converse way to design an algebraic semantics for a language with parallel constructs is to give a set of axioms for transforming programs into equivalent programs of a sequential non deterministic sublanguage (Br). The transformational approach may in particular be applied to the fair parallel construct, provided that the random assignment is present in the sequential sublanguage.

For sequential programs with random assignment, Apt and Plotkin have shown in(ApP) that complete proof systems can be obtained as soon as the induction over the ordinals is allowed in the proofs (a counterpart of the weak continuity of the semantic functions). In the particular case of sequential programs with the repetitive non deterministic construct <u>do</u> $B_1 \rightarrow C_1 \square .. \square B_n \rightarrow C_n$ <u>od</u> , the assumption of fairness may be applied to the alternatives of each of the <u>do</u> statements exactly as it is commonly applied to finite families of concurrent agents. In a way which is very akin to the approach of transformational semantics, Apt and Olderog have shown how that fairness property can be enforced by a transformation which adds to the programs some kind of internal schedulers, with auxiliary variables submitted to random assignments (ApO). More importantly, the authors demonstrate that it is then possible to derive from the proof system which applies to the target programs another proof system which applies to the source programs in complete adequation with the assumption of fairness. Of course, the induction over the ordinals is explicitly called for in the resulting proof system. A similar situation would arise if the method was adapted to parallel programs, then using the random assignment of auxiliary variables as suggested by Plotkin for a different purpose in (Pl). Notice that strong connections exist with the result established by Queille and Sifakis in the framework of the temporal reasoning about transition systems (QuS). However, no complete proof system is expected there : each transition system defines a particular interpretation of the modal operators, and the authors are interested more in the truth of properties w.r.t. particular models of their logics than in the validity of such properties. That fact may explain why the ordinals are not called for.

Now, our general intention was to show that it is possible to obtain a complete proof system for the fair equivalence of programs of a non trivial subset of CCS, without appealing to the induction over the ordinals. Although a proof system for the equivalence of programs strongly departs as regards its objective from a proof system for properties of the input-output relations computed by programs, we felt that both categories of proof systems are not so distant as regards their complexity. Hence, we considered all the above results as a clear indication that our objective could not be achieved without placing sufficient restrictions on CCS to disallow the simulation of register machines or Turing machines. Clearly, neither of these simulations can be performed in our chosen subset of CCS, since we have left no means to program unbounded strings of agents each of which communicates with its left and right neighbours. However, less drastic restrictions could be imposed with an identical effect, and we shall give further justifications in the sequel.

Another important departure from CCS is the replacement of the + operator by n-ary guarding operators. This second kind of restriction aims at a clear distinction between explicitly programmed alternatives and implicit scheduling alternatives. Confusions which might arise between the two types of alternatives - e.g. in $(\alpha : p) + ((\beta : q) \mid (\gamma : r))$ - had to be avoided since our assumption of strong fairness applies to the scheduling alternatives only.

* Once the above decisions were taken, a strong hope in the reachability of our objective has been raised by the discovery that the fair parallel composition of infinitary rational languages is a rational and computable operation. Infinitary languages are in fact a well suited domain for the semantics of parallel programs (Fa) , and the existence of a decision procedure for the equality of infinitary rational languages indicates clearly that corresponding proof systems may be found. Notice that the algorithm which has been suggested for computing the parallel composition of languages may be encoded into the inference rules of a formal system for rational expressions, since the associated automata are not called for. There lays the main difference with the tools developped independently by de Simone for a more general kind of fair composition (Si) . A closer likeness may be seen with Park's characterization of the fair merge, in what our algorithm combines too the use of least and greatest fixed points.

It has been pointed out in (Je) that the closure of the class of finitary rational languages under the shuffle-closure operator of (Sh) is a proper subclass of context-sensitive languages. Now, the case is still worse when considering infinitary languages instead of finitary languages and the fair shuffle instead of the ordinary shuffle : there remains very little hope to find a complete proof system for the equivalence of programs with unbounded parallelism, as soon as their semantics are defined in terms of infinitary languages. This remark explains why, to the difference of CCS, we have defined the parallel composition on programs and not on terms.

* At this step in the development of our study, the critical point was to define a fair operational semantics of the programming language. Clearly, in the case of open systems of purely communicating agents, the prescription of finite delay of actions cannot be used as a basis for a fair operational semantics. Consider for instance the agent (α) (NIL) : that agent may either get satisfaction of its communication demand α or remain endlessly inactive, depending on the behaviour of the external environment, or observer. The situation is left unchanged when one considers the open system ((α) (NIL) | (β) (NIL)) made out of two agents : it may well be the case that only the second agent gets satisfaction of its communication demand β, depending on the observer. It therefore appears that the difficulty cannot be turned by the introduction of a finite delay operator as used in (Mi) for another class of behaviours, nor by the two level mechanism of (CS)whose rules express no more than the finite delay of actions.

Now, the inability to define an operational semantics for open systems is not redhibitory : the "a-priori" semantics of an open system may be indirectly defined on the basis of the "a-posteriori" semantics of a particular family of closed systems, in which it appears as a subsystem (BrW). That method has been used by Hennessy and de Nicola in a very elegant and constructive way to build a fully observational model of CCS, drawn from an associated complete proof system with heavy induction (HeN). Thus, it may suffice to define an operational semantics for closed systems of purely communicating agents. Even in that simplified case, the techniques of (Mi, CS) are not adequate, since they would leave some closed systems without any fair computation ! Nor does work the assignment to agents of random credits of action, suggested in(Pl) for independent processes. We feel that the provision of additional waiting queues is in fact necessary, whence a lot of complications arise. Nevertheless, the most serious problem with the above approach is probably to derive a fair semantics of open systems from an a-posteriori semantics of closed systems, since the technique used in (HeN) entails the assumption of ω-continuity.

To sum up, nothing can be gained from considering closed systems instead of open systems, and the prescription of finite delay of actions is inadequate for open systems. Our definition of histories and their laws of synthesis is a possible way out. The technical trick is the enrichment of the natural monoid of action sequences. The major justification for our precise definitions lays in the fact that both the operational and the observational semantics can be decided.
* At the time these lines are written, we cannot claim that our objective has been completely achieved : the proof system is still missing, although a significant advance has been made in that direction as regards the equality of infinitary rational expressions [DK].

Another issue which deserves further attention is the expression and verification of those properties of observable behaviours which cannot be reduced to congruence formulas. Since languages of observations capture all the relevant details, a decision procedure can certainly be found for all the properties which may be specified in the setting of rational expressions, but more attractive specification techniques may be prefered.

Our last remark concerns possible extensions of the present work to non-rational subsets of CCS. An interesting subset may be obtained by considering only those programs p which verify the following property :

- there exist a finite integer n, a n-ary program context $\mathscr{C}(x_1, \dots, x_n)$ and a family of programs $p_1, \dots, p_n$, such that any of the derivatives of p is equivalent to $\mathscr{C}((p_1)^{m_1}, \dots, (p_n)^{m_n})$ for some $m_i \geqslant 0$, letting $p_i^{m_i}$ stand for the parallel composition of m_i instances of p_i. That class of programs lays in between full CCS and its rational subset, and allows to simulate for instance unbounded semaphores but not Turing machines. A corresponding syntactic restriction on CCS is to limit flow-operations, except for the parallel composition, to be applied on programs but not on general terms. We guess that a fair operational semantics can be defined for the considered class of programs by a straightforward extension of the above presented principles. The point is that each of the finite families of agents $p_i^{m_i}$, which fill in the holes of the context $\mathscr{C}(x_1, \dots, x_n)$ to give the program $\mathscr{C}((p_1)^{m_1}, \dots, (p_n)^{m_n})$, can in fact be considered as an elementary agent as regards the realization of fairness : it does not matter which particular member of the family $p_i^{m_i}$ is concerned in an elementary action of $p_i^{m_i}$!

References of the appendix

(ApO) Apt K. & Olderog E. *Proof rules dealing with fairness.* Bericht 8104, Inst. Inf. Prakt. Math., University of Kiel (1981)

(ApP) Apt K. & Plotkin G. *Countable non determinism and random assignment,* TR 82-7, L.I.T.P. , University of Paris 7(1982)

(Br) Broy M. *Transformational Semantics for concurrent Programs,* IPL Vol 11 n° 2 (1980)

(BrW) Broy M. & Wirsing M. *On the Algebraic Specification of Finitary Infinite Communicating Sequential Processes,* IFIP-TC2 Work. Conf. , Garmisch-Partenkirchen, North-Holland Publishing Company (1982)

(Ch) Chandra A. *Computable Non deterministic Functions,* Proc. 19th Symposium on Found. of Comp.Sc., IEEE (1978)

(CS) Costa G. & Stirling S. *A fair Calculus of Communicating Systems,* Proc. FCT 83, LNCS 158 (1983)

(DK) Darondeau P. & Kott L. *A formal Proof System for Infinitary Rational Expressions,* TR 199, IRISA, University of Rennes (1983)

(Fa) Fauconnier H. *Application des Langages Infinitaires , à l'Etude de C.S.P.* (thesis), TR 82-26, L.I.T.P., University of Paris 7 (1982)

(HeN) Hennessy M. & de Nicola R. *Testing Equivalences for Processes,* CSR Proc. ICALP83, LNCS 154 (1983)

(Je) Jedrzejowicz J. *On the enlargement of the class of Regular languages by the Shuffle closure* ,IPL. Vol. 16 n° 2 (1983)

(Mi) Milner R. *A finite delay operator in synchronous CCS,* CSR 116-82, Edinburgh University (1982)

(Pa) Park D. *On the Semantics of Fair Parallelism,* in Abstract Software Specifications, LNCS Vol. 86 (1980)

(Pl) Plotkin G. *A powerdomain for countable nondeterminism,* LNCS Vol. 140 (1982)

(QuS) Queille J. & Sifakis J. *Fairness and related properties in transition systems, a time logic to deal with fairness,* TR.292, University of Grenoble (1982)

(Sh) Shaw A. *Software descriptions with flow-expressions,* IEEE Trans. on Soft.Eng. SE-4 (3) (1978)

(Si) de Simone R. *Langages infinitaires et produit de mixage (à paraître dans TCS).*

Liste des Publications Internes IRISA

PI 185 **Dialogue et représentation des informations dans un système de messagerie intelligent**
Philippe Besnard, René Quiniou, Patrice Quinton, Patrick Saint-Dizier, Jacques Siroux, Laurent Trilling , 45 pages ; Janvier 1983

PI 186 **Analyse classificatoire d'une correspondance multiple ; typologie et régression**
I.C. Lerman , 54 pages ; Janvier 1983

PI 187 **Estimation de mouvement dans une sequence d'images de télévision en vue d'un codage avec compensation de mouvement**
Claude Labit , 132 pages ; Janvier 1983

PI 188 **Conception et réalisation d'un logiciel de saisie et restitution de cartes élémentaires**
Eric Secher , 45 pages ; Janvier 1983

PI 189 **Etude comparative d'algorithmes pour l'amélioration de dessins au trait sur surfaces point par point**
M.A. ROY , 96 pages ; Janvier 1983

PI 190 **Généralisation de l'analyse des correspondances à la comparaison de tableaux de fréquence**
Brigitte Escofier , 35 pages ; Mars 1983

PI 191 **Association entre variables qualitatives ordinales «nettes» ou «floues»**
Israel-Cesar Lerman , 42 pages ; Mars 1983

PI 192 **Un processeur intégré pour la reconnaissance de la parole**
Patrice Frison , 80 pages ; Mars 1983

PI 193 **The Systematic Design of Systolic Arrays**
Patrice Quinton , 39 pages ; Avril 1983

PI 194 **Régime stationnaire pour une file M/H/1 avec impatience**
Raymond Marie et Jean Pellaumail , 8 pages ; Mars 1983

PI 195 **SIGNAL : un langage pour le traitement du signal**
Paul Le Guernic, Albert Benveniste, Thierry Gautier , 49 pages ; Mars 1983

PI 196 **Algorithmes systoliques : de la théorie à la pratique**
Françoise André, Patrice Frison, Patrice Quinton , 19 pages ; Mars 1983

PI 197 **HAVANE : un système de mise en relation automatique de petites annonces**
Patrick Bosc, Michèle Courant, Sophie Robin, Laurent Trilling , 79 pages ; Mai 1983

PI 198 **Une procédure de décision en logique non-monotone**
Philippe Besnard , 59 pages ; Mai 1983

PI 199 **A formal proof system for infinitary rational expressions**
Philippe Darondeau, Laurent Kott , 28 pages ; Mai 1983

PI 200 **Etude générale d'un réseau constitué de deux stations hyperexponentielles**
Jean-Yves Le Boudec , 12 pages ; Mai 1983

PI 201 **Langage de Dyck et groupe symétrique**
Yves Cochet , 13 pages ; Juin 1983

PI 202 **On the observational semantics of pair parallelism**
Philippe Darondeau, Laurent Kott , 14 pages ; Juin 1983

PI 203 **Les langages fonctionnels : caractéristiques, utilisation et mise en œuvre**
Daniel Le Métayer , 162 pages ; Juin 1983

PI 204 **On exact and approximate itterative methods for general queueing networks**
Raymond A. Marie, William J. Stewart , 27 pages ; Juin 1983

PI 205 **On quantifier hierarchy and its paraphrase in a semantic representation of natural language sentences**
Patrick Saint-Dizier , 17 pages ; Juillet 1983

PI 206 **Trois articles sur le traitement adaptatif du signal pour l'encyclopédie Pergamon sur l'automatique**
Albert Benveniste , 64 pages ; Septembre 1983

PI 207 **Sur l'existence et l'unicité du réseau homogène à un réseau ferme de files d'attente à lois générales**
Raymond Marie, Gérardo Rubino , 44 pages ; Septembre 1983

PI208 Problèmes d'implémentation du langage Prolog en vue de la réalisation d'une machine Prolog
Yves Bekkers, Bernard Canet, Olivier Ridoux, Lucien Ungaro
Octobre 1983, 63 pages

PI 209 La technique du suivi de contour en synthèse d'images et ses applications
Gérard HEGRON - Octobre 1983

PI 210 A new characterization of infinitary rational languages
Philippe DARONDEAU, Laurent KOTT- Octobre 1983 - 9 pages.

PI 211 On the observational semantics of fair parallelism
Philippe DARONDEAU, Laurent KOTT - Octobre 1983, 80 pages.