MATHÉMATIQUES ET SCIENCES HUMAINES

G. FERRAND

Modèles d'apprentissage « stimulus-réponse » en termes d'automates

Mathématiques et sciences humaines, tome 84 (1983), p. 5-43

<http://www.numdam.org/item?id=MSH_1983__84__5_0>

Math. Sci. hum. (*21*^{e} **année, n°84,** *1983*, **p.5-43)**

MODELES D'APPRENTISSAGE "STIMULUS-REPONSE" EN TERMES D'AUTOMATES

G. FERRAND*

I. INTRODUCTION

On propose ici une représentation en termes d'automates de situations d'apprentissage, sous la forme d'un réseau d'automates probabilistes qu'on peut voir comme l'interconnexion d'un automate "sujet" et d'un automate "expérimentateur". C'est à John Derrick que l'auteur doit l'idée initiale d'une approche en termes d'automates des problèmes d'apprentissage (3).

Plus précisément, on commence par décrire une certaine formalisation de l'idée de réseau d'automates et d'évolution de ce réseau, qui a pour but de constituer des "modèles en termes d'automates". La notion d'automate dont il s'agit ici (1 , 10 , 12) est suffisamment large pour s'appliquer à des domaines variés. Mais l'idée d'interconnecter de tels automates pour constituer un réseau pose des problèmes de fondements logiques, tout particulièrement quant à la signification des probabilités conditionnelles associées à des automates qui figurent dans une "boucle".

En définissant les notions de multi-automate, et d'actualisation d'un multi-automate, nous avons étudié ces problèmes dans (4) et (5). Le contexte probabiliste, et la nécessité de distinguer, pour mieux les relier, aussi bien

---

* Département de Mathématiques et Informatique, Université d'Orléans.

mathématiquement qu'intuitivement, des notions qui risquent d'être confondues ou disjointes, imposent un formalisme assez complexe et des développements assez longs.

Mais ici, pour que les idées essentielles concernant des processus d'apprentissage ne soient pas masquées par des détails techniques fastidieux, nous nous sommes affranchi de certaines contraintes de ce formalisme. Certains développements formels sont seulement suggérés à travers des sous-entendus et des abus de langage.

En particulier, dans la Partie II, l'introduction de la notion de multi-automate a volontairement un style semi-naïf, semi-formel. Pour le détail des démonstrations on peut se reporter à [4] ou [5]

Dans la Partie III on "représente en terme d'automates" les "stimulus-sampling models" d'Estes et Suppes [13]. Comme Suppes dans [14] , on peut considérer ces modèles comme point de départ pour diverses variantes de "Stimulus-Response (= S-R) Models".

Le résultat est, en un certain sens, une réciproque au Théorème de Représentation de Suppes [14], lequel, dans le cadre de la controverse entre les courants behaviouriste et cognitiviste [2 , 6 , 8 , 9 , 15], ramène la notion d'automate "accepteur" à celle de S-R Modèle : ici, inversement, la notion de S-R Modèle est ramenée à un certain type de multi-automate, qui fait apparaître une division naturelle entre deux multi-automates, l'un "sujet" et l'autre "expérimentateur" (ou "élève" et "enseignant").

De plus on obtient (Partie IV) un résultat qui est le parallèle, en termes d'automates, du théorème de Suppes : on définit une classe de multi-automates "élèves" capables d'apprendre le comportement de n'importe quel automate accepteur.

L'interprétation en termes d'automates fournit un cadre dans lequel on peut représenter et discuter de façon naturelle et commode, d'une part les situations d'apprentissage, d'autre part les modèles déjà existants de telles situations. C'est par l'exemple des S-R Modèles que ce point est illustré ici :

Les premiers modèles semblaient a priori limités à des comportements du type fonction : stimulus $\mapsto$ réponse, et une des critiques qu'on leur opposait était de ne pas rendre compte de comportements plus élaborés. Le théorème de Suppes surmonte cette limitation en montrant que des modèles plus complexes, mais toujours S-R, peuvent s'appliquer au comportement du type automate "accepteur".

On verra (III.4.2 et IV.2) que la représentation de ces modèles en termes d'automates fait clairement apparaître la nature des modifications apportées : la structure générale du multi-automate, qui traduit la conception S-R, est maintenue ; et c'est la division du multi-automate en deux qui est modifiée, ce qui met en évidence une nouvelle conception du sujet.

Estes et Suppes (13) ont mis en évidence des chaînes de Markov dans certains "Stimulus-Sampling Models", correspondant à une idée de mémoire finie. On verra (III.4.1) que la représentation en termes d'automates permet d'éclaircir et de généraliser cette intervention de Chaînes de Markov.

C'est parce que le théorème de Suppes concerne les S-R Modèles que seuls ces modèles sont étudiés ici. Cependant l'emploi des automates comme outil mathématique n'est pas lié exclusivement aux S-R Modèles : il pourrait permettre aussi de reformuler d'autres conceptions et modèles de l'apprentissage et de l'activité psychologique (par exemple pour décrire des conceptions de Chomsky). On peut espérer que le cadre introduit ici conviendra pour formuler dans un même langage des modèles différents ou opposés, et pour approfondir et clarifier la discussion et la comparaison entre ces modèles.

L'auteur est reconnaissant à John Derrick d'avoir guidé ses premiers pas dans ce domaine.

II. MULTI-AUTOMATES

II.1. Automates

DEFINITION : Un automate (probabiliste et fini) M est défini par la donnée de trois ensembles (finis non vides) X, Y, Q appelés respectivement ensembles des inputs, outputs, états, et d'une probabilité de transition de Q x X à Q x Y, c'est-à-dire simplement d'une fonction p à valeurs réelles $\geqslant 0$ notées $p(q',y|q,x)$ définies pour tous q',y,q,x où $q'\varepsilon Q$, $y\varepsilon Y$, $q\varepsilon Q$, $x\varepsilon X$, et vérifiant la relation

$$\sum_{q'\varepsilon Q, y\varepsilon Y} p(q',y|q,x) = 1 \quad \text{pour tous q et x fixés.}$$

Formellement $M = (X,Y,Q,p)$. $p(q',y|q,x)$ est usuellement interprété comme une probabilité conditionnelle : sachant que l'automate est dans l'état q et qu'il reçoit l'input x, probabilité pour qu'il passe dans l'état q' en donnant l'output y. Mais l'usage que nous faisons de la notion d'automate conduit à donner un statut plus formel à cette interprétation:

Considérons un processus probabiliste qui associe à chaque entier naturel t un "input à l'étape t", noté I_t, un "output à l'étape t", noté O_t, un "état à l'étape t", noté E_t.

I_t doit être un élément aléatoire de X, c'est-à-dire qu'on peut parler, pour chaque $x (x\varepsilon X)$, de l'événement $\{I_t = x\}$. De même O_t et E_t sont des éléments aléatoires de Y et Q respectivement.

Ce processus doit aussi nous permettre de parler de la probabilité de chaque événement grâce à une mesure de probabilité P.

DEFINITION : Nous disons que ce processus est une actualisation de l'automate M s'il vérifie :

$$\begin{aligned} P(E_{t+1}=q',\ O_t = y | E_t = q,\ I_t = x,\ & E_{t-1} = q_{t-1},\dots,E_o = q_o, \\ & I_{t-1} = x_{t-1},\dots,I_o = x_o, \\ & O_{t-1} = y_{t-1},\dots,O_o = y_o) \end{aligned}$$

(si définie) = $p(q',y|q,x)$

(ceci quels que soient $t,q',y,q,x,q_{t-1},\dots,q_o,x_{t-1},\dots,x_o,y_{t-1},\dots,y_o$) ,

ce qui revient à dire que :

- d'une part

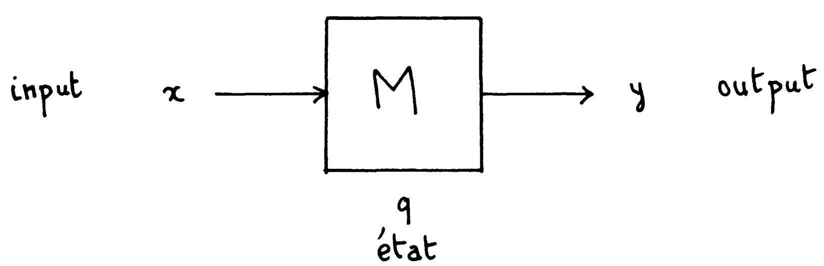
$P(E_{t+1} = q',\ O_t = y|E_t = q,\ I_t = x)$(si définie) = $p(q',y\ q,x)$

- d'autre part

le couple (E_t,I_t) étant donné, le couple $(E_{t+1},\ O_t)$ est indépendant de la suite formée par tous les $E_{t'}$, $I_{t'}$, $O_{t'}$ pour tous les $t'<t$. Ceci est une propriété d'<u>indépendance conditionnelle</u>.

Intuitivement, cela signifie que, si l'on connaît l'état q et l'input x à l'étape t, alors la probabilité pour que, à l'étape t l'output soit y et à l'étape t+1 l'état soit q', est $p(q',y|q,x)$, et que cette probabilité n'est pas modifiée par la connaissance d'information supplémentaire sur les états, inputs, outputs, aux étapes précédant l'étape t.

Ce processus formalise l'évolution au cours du temps d'une "machine"



qui, à chaque étape, se trouve dans un certain état, reçoit un certain input, donne un certain output et passe à l'état suivant (état à l'étape suivante). Dans son <u>état</u> est concentrée toute la mémoire qu'elle peut retenir de son histoire passée et qui peut influencer son histoire future (un ensemble d'états Q réduit à un seul élément formalise l'idée d'une absence de mémoire).

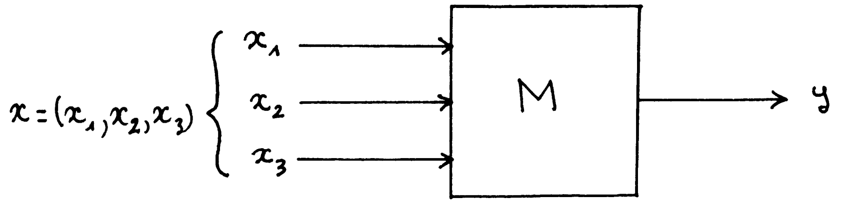
L'automate M formalise la machine, et nous formalisons son évolution par une actualisation de M.

Il est clair que les notions ici introduites généralisent la notion de chaîne de Markov. Plus précisément un automate généralise une matrice de transition M, une actualisation de l'automate généralise une chaîne de Markov associée à la matrice M.

II.2. La définition d'un multi-automate

L'idée d'un multi-automate est celle d'un réseau construit en connectant entre eux certains automates. (Pour simplifier, on ne va considérer ici que des multi-automates que nous qualifions de fermés, car ils ne reçoivent pas d'information de l'extérieur).

Pour définir ces connexions sans rentrer dans les détails formels, nous allons encore représenter un automate par une "boite" comme précédemment d'où part une flèche suggérant la sortie de l'output y. Mais l'input x pourra être constitué de plusieurs composantes $x_1,..,x_n$ c'est-à-dire que x sera une suite finie $x = (x_1,..,x_n)$, l'ensemble des inputs étant un produit cartésien $X = X_1 x...x X_n$. Alors, au lieu de suggérer l'entrée de x par une flèche, c'est l'entrée de chaque composante x_i qui sera suggérée par une flêche entrant dans la boite.



(On admet même $n = 0$: alors aucune flèche n'entre. Formellement l'exigence X non-vide est respectée car X a pour unique élément la suite vide).

Un multi-automate est alors construit à partir d'un certain nombre d'automates ainsi représentés, en reliant et identifiant chaque flèche d'entrée de chaque boite à la flèche de sortie d'une et d'une seule boite. Plus précisément chaque automate sera repéré par un symbole d'automate inscrit dans la boite, et on appellera schéma l'indication des symboles d'automates et des flèches (voir comme exemple la figure 1) : la flèche de sortie d'une boite peut se ramifier, la fin de chaque ramification est une flèche d'entrée d'une boite.

Plus formellement un schéma est ici un couple (A,C) où A est l'ensemble des symboles d'automates, et C l'application qui à chaque symbole a ($a \in A$) fait correspondre l'ensemble C(a) des symboles auxquels est connecté a, c'est-à-dire desquels part une flèche qui aboutit au symbole a. Par exemple dans le schéma de la figure 1,

A = {COND, PRES, SAM, REP, OUT, RENF}

C(COND) = {SAM, RENF}, C(PRES) = {OUT, REP}, C(SAM) = {PRES},

C(REP) = {SAM, COND}, C(OUT) = {PRES, REP}, C(RENF) = {OUT, REP}.

On appellera système d'états l'indication, pour chacun de ces automates, de l'ensemble de ses états. Plus formellement un système d'états pour le schéma (A,C) est une famille $(Q_a)_{a \in A}$ où chaque Q_a est un ensemble fini non-vide.

On appellera système de valeurs l'indication, pour chacun de ces automates, de l'ensemble de ses outputs. Plus formellement c'est encore une famille de la forme $(Y_a)_{a \in A}$.

Pour chacun de ces automates l'input est formé d'un certain nombre de composantes qui chacune sont aussi un output d'un de ces automates. Le système de valeurs définit donc en même temps l'ensemble des inputs de chacun de ces automates. Plus formellement l'ensemble d'inputs associés à a est l'ensemble

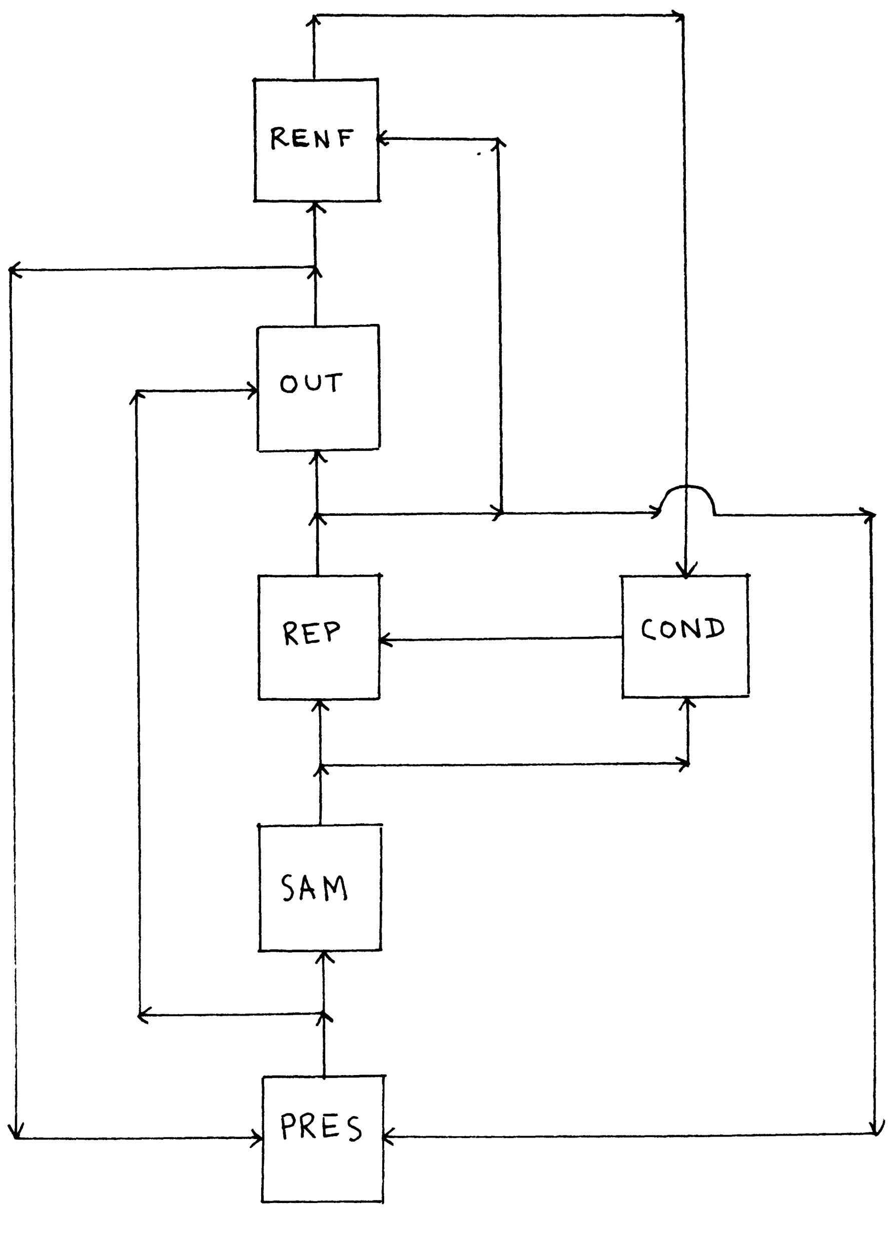


figure 1. Un exemple de schéma

produit $X_a = \prod_{s \in C(a)} Y_s$.

Chacun de ces automates est finalement défini en indiquant sa probabilité de transition. Plus formellement à chaque a($a \in A$) est associée une probabilité de transition p_a de $Q_a \times X_a$ à $Q_a \times Y_a$.

En résumé, un multi-automate $\mathcal{M}$ est défini par la donnée du schéma, des systèmes d'états et de valeurs et des probabilités de transition :

$\mathcal{M} = (A, C, (Q_a)_{a \in A}, (Y_a)_{a \in A}, (p_a)_{a \in A})$.

II.3. Actualisation d'un multi-automate bien bouclé

De même qu'à un automate on a associé un processus : l'actualisation de l'automate, on doit chercher à définir une notion d'actualisation pour un multi-automate. Cela conduit à des problèmes de fondements logiques qu'il n'est pas nécessaire de développer ici. On va seulement esquisser quelques résultats :

On définit de façon évidente la notion de boucle du schéma : c'est une suite de symboles d'automates dans laquelle chacun des symboles est relié au suivant par une flèche du schéma et dans laquelle les premiers et derniers symboles sont identiques. Par exemple, sur le schéma de la figure 1 la suite (RENF, COND, REP, OUT, RENF) est une boucle.

Ce sont essentiellement les boucles qui sont à l'origine des problèmes de fondements logiques. On peut résoudre ces problèmes pour certains multi-automates que nous avons appelés "bien bouclés". Les multi-automates qui vont intervenir ici seront des cas particuliers simples de multi-automates bien bouclés.

Dans ces cas particuliers, la simplification est dûe à certains automates que nous allons maintenant décrire :

DEFINITION : Un automate $M = (X,Y,Q,p)$ est déterministe si $p(q',y|q,x)$ ne prend que les valeurs 0 ou 1.

On définit alors deux fonctions $\delta : Q \times X \to Q$ et $\lambda : Q \times X \to Y$ par : $\delta(q,x) = q'$ et $\lambda(q,x) = y$ si $p(q',y|q,x) = 1$

Traditionnellement un tel automate déterministe est alors plutôt noté $M = (X,Y,Q,\delta,\lambda)$.

Intuitivement, sachant que l'automate est dans l'état q et qu'il reçoit l'input x, la fonction λ nous donne son output et la fonction δ son état suivant.

DEFINITION : Un automate déterministe $M = (X,Y,Q,\delta,\lambda)$ sera de plus un automate <u>retard</u> si $\lambda(q,x)$ ne dépend que de q, c'est-à-dire s'il existe une fonction $\lambda' : Q \to Y$ telle que $\lambda(q,x) = \lambda'(q)$ pour tous q et x ($q \in Q$, $x \in X$).

Intuitivement, sachant que l'automate est dans l'état q, la fonction λ' nous donne déjà son output ; la détermination de son input peut être retardée mais sera nécessaire pour que la fonction δ donne son état suivant.

DEFINITION : On appellera automate <u>délai</u> un automate déterministe $M = (X,Y,Q,\delta,\lambda)$ avec $X = Y = Q$, $\delta(q,x) = x$ et $\lambda(q,x) = q$. C'est donc un cas particulier d'automate retard.

Intuitivement, un automate délai garde en mémoire l'input qu'il reçoit, sous forme d'état suivant, pour le redonner à l'étape suivante sous forme d'output.

Les multi-automates qui vont intervenir ici auront la particularité suivante : dans chaque boucle figure (au moins) un automate (déterministe) retard. Nous appelons <u>repère</u> du multi-automate l'ensemble de tous ces automates (déterministes) retards. Par exemple sur le schéma de la figure 1 un repère peut être constitué par COND et PRES.

La notion d'actualisation qu'on peut alors définir pour de tels multi-automates est celle d'un processus qui associe à chaque entier t et à chaque symbole d'automate a($a \in A$) d'une part un <u>état</u> noté $E_{a,t}$ et d'autre part une <u>valeur</u> notée $V_{a,t}$.

$E_{a,t}$ et $V_{a,t}$ sont des éléments aléatoires de Q_a et Y_a respectivement.
$E_{a,t}$ est l'état à l'étape t de l'automate symbolisé par a.
$V_{a,t}$ sert à la fois d'output et de composante d'input : $V_{a,t}$ est l'output à l'étape t de l'automate symbolisé par a, c'est donc en même temps, s'il y a une flèche de a vers b, c'est-à-dire si $a \epsilon C(b)$, une composante de l'input à l'étape t de l'automate symbolisé par b, input noté $I_{b,t}$

Formellement $I_{b,t} = (V_{a,t})_{a \epsilon C(b)}$.

Par exemple, selon le schéma de la figure 1, l'output à l'étape t de l'automate RENF est $V_{RENF,t}$, son input est composé de $V_{OUT,t}$ et $V_{REP,t}$, son état est $E_{RENF,t}$.

Ce processus, actualisation du multi-automate, présente les aspects suivants :

(i) Pour chaque symbole d'automate a fixé, si l'on ne retient du processus que les $I_{a,t}$, $V_{a,t}$, $E_{a,t}$ on obtient une actualisation de l'automate symbolisé par a. (Par rapport à la notion générale d'actualisation [5] il s'agit là d'une simplification dûe au caractère déterministe des automates du repère).

(ii) Ce processus vérifie une propriété d'indépendance conditionnelle qui généralise celle qu'on a déjà vue dans le définition de l'actualisation d'un automate en II.1. Mais il s'agit d'une généralisation "selon deux dimensions" : le temps, et les connexions (le schéma).

Pour ne pas rentrer dans les détails formels, nous allons seulement en donner une interprétation intuitive, en indiquant comment, à chaque étape, sont "déterminés" états et valeurs (des guillements à cause du caractère probabiliste).

Considérons une étape t fixée. $V_{a,t}$ et $E_{a,t+1}$ dépendent de $E_{a,t}$ et $I_{a,t}$. Mais $I_{a,t}$ est composé d'un certain nombre de $V_{b,t}(b \epsilon C(a))$ et chacun de ces $V_{b,t}$ à son tour dépend de $E_{b,t}$ et $I_{b,t}$ etc... d'où le risque de "cercle vicieux" en cas de boucle.

Mais si b fait partie du repère, alors l'automate b est retard donc $V_{b,t}$ ne dépend que de $E_{b,t}$. Par contre $E_{b,t+1}$ dépend de $E_{b,t}$ et de $I_{b,t}$.

Alors, étant donnés les états $E_{a,t}$ pour tous les $a(a\varepsilon A)$, les valeurs $V_{a,t}$ sont évaluées "de proche en proche en suivant les connexions", en commençant par les $V_{b,t}$ pour les b du repère.

Pour un a qui ne fait partie du repère, $E_{a,t+1}$ est évalué en même temps que $V_{a,t}$.

Mais pour un b du repère, $E_{b,t+1}$ est évalué à la fin quand tous les $V_{a,t}$, pour tous les $a(a\varepsilon A)$ sont déjà évalués.

Par exemple, d'après le schéma de la figure 1 en supposant encore que le repère est formé de COND et PRES, on peut se représenter ainsi le "déroulement" d'une étape :

Etant donnés

$E_{PRES,t}$; $E_{SAM,t}$; $E_{COND,t}$; $E_{REP,t}$; $E_{OUT,t}$; $E_{RENF,t}$;

$E_{PRES,t}$ permet d'évaluer $V_{PRES,t}$; $E_{COND,t}$ permet d'évaluer $V_{COND,t}$.

$E_{SAM,t}$ et $V_{PRES,t}$ permettent d'évaluer $E_{SAM,t+1}$ et $V_{SAM,t}$;

$E_{REP,t}$; $V_{SAM,t}$ et $V_{COND,t}$ permettent d'évaluer $E_{REP,t+1}$ et $V_{REP,t}$;

$E_{OUT,t}$; $V_{REP,t}$ et $V_{PRES,t}$ permettent d'évaluer $E_{OUT,t+1}$ et $V_{OUT,t}$;

$E_{RENF,t}$; $V_{OUT,t}$ et $V_{REP,t}$ permettent d'évaluer $E_{RENF,t+1}$ et $V_{RENF,t}$;

$E_{COND,t}$; $V_{RENF,t}$ et $V_{SAM,t}$ permettent d'évaluer $E_{COND,t+1}$;

$E_{PRES,t}$; $V_{OUT,t}$ et $V_{REP,t}$ permettent d'évaluer $E_{PRES,t+1}$.

Nous appelons état du multi-automate à l'étape t, noté E_t, la famille constituée par tous les $E_{a,t}$ pour $a\varepsilon A$. (C'est un élément aléatoire de $\prod_{a\varepsilon A} Q_a$).
Alors on montre que :

(i) Pour définir une actualisation du multi-automate, il suffit de se donner une probabilité initiale, concernant l'état initial E_0.

(ii) La suite des E_t (pour t = 0,1,2...) constitue une chaîne de Markov dont la matrice de transition se calcule à partir des probabilités de transition p_a des automates du multi-automate.

III. REPRESENTATION DES "STIMULUS-SAMPLING MODELS"

III.1. Multi-automate "Stimulus-Sampling"

On va maintenant définir un multi-automate (en fait, toute une classe de tels multi-automates) qui permet de "représenter en termes d'automates" des modèles d'apprentissage Stimulus-réponse.

Plus précisément, une actualisation de notre multi-automate va représenter un "Stimulus-Sampling Model" (en abrégé SSM) défini par Estes et Suppes dans [13] . Mais il sera clair qu'on pourrait appliquer la même méthode à diverses variantes.

Une expérience d'apprentissage (au sens de [13]) est une suite d'essais dans lesquels interviennent un sujet et un expérimentateur. Chaque essai est lui-même constitué de 6 composantes :

1. Le sujet est le siège d'un certain conditionnement (interne au sujet, non-observable par l'expérimentateur).

2. L'expérimentateur présente au sujet un ensemble de stimuli.

3. Le sujet en retient un échantillon ("sampling") (non-observable).

4. Le sujet donne une certaine réponse, qui dépend de cet échantillon et de son conditionnement.

5. L'expérimentateur donne une certaine issue ("outcome") (indication, récompense...).

6. Il en résulte un certain renforcement, interne au sujet, qui dépend aussi de sa réponse, et qui va modifier éventuellement son conditionnement.

Chaque renforcement est, ou bien le renforcement d'une réponse (et alors formellement il sera identifié à cette réponse), ou bien un "renforcement nul".

Le conditionnement est formalisé par une fonction de l'ensemble des stimuli dans l'ensemble des réponses.

Les axiomes d'Estes et Suppes précisent, en termes de probabilités, les relations entre ces diverses composantes.

On pourrait prendre le schéma de la figure 1 (déjà vu précédemment à titre d'exemple) comme schéma de notre multi-automate. Ce schéma exprime la conception de l'expérience d'apprentissage, chaque essai correspond à une étape de l'actualisation du multi-automate. On peut y suivre le déroulement d'un essai en le parcourant de bas en haut. Les boucles qui reviennent vers le bas correspondent au passage à l'essai suivant :

1. Le conditionnement est l'output de COND.
2. L'ensemble de stimuli présenté est l'output de PRES.
3. L'échantillon retenu est l'output de SAM.
4. La réponse donnée est l'output de REP.
5. L'issue ("outcome") est l'output de OUT.
6. Le renforcement est l'output de RENF.

Mais pour que cette représentation de SSM par l'actualisation soit une propriété bien définie et rigoureusement démontrée, il est nécessaire de préciser ces 6 automates. Il est alors plus commode de remplacer le schéma de la figure 1 par celui dela figure 2, de façon à suivre de plus près le détail de certaines des composantes d'un essai. (Ce changement de schéma illustre la modularité de la notion de multi-automate : sans rentrer dans des détails formels, on peut considérer que les automates sont des modules du multi-automate, qui peuvent à leur tour être décomposés en modules sous forme de multi-automates eux-mêmes).

III.1.1. Définitions et notations (inspirées par (13))

Soient S,R,U trois ensembles finis. On dira que

S est l'ensemble des stimuli

R est l'ensemble des réponses

U est l'ensemble des issues

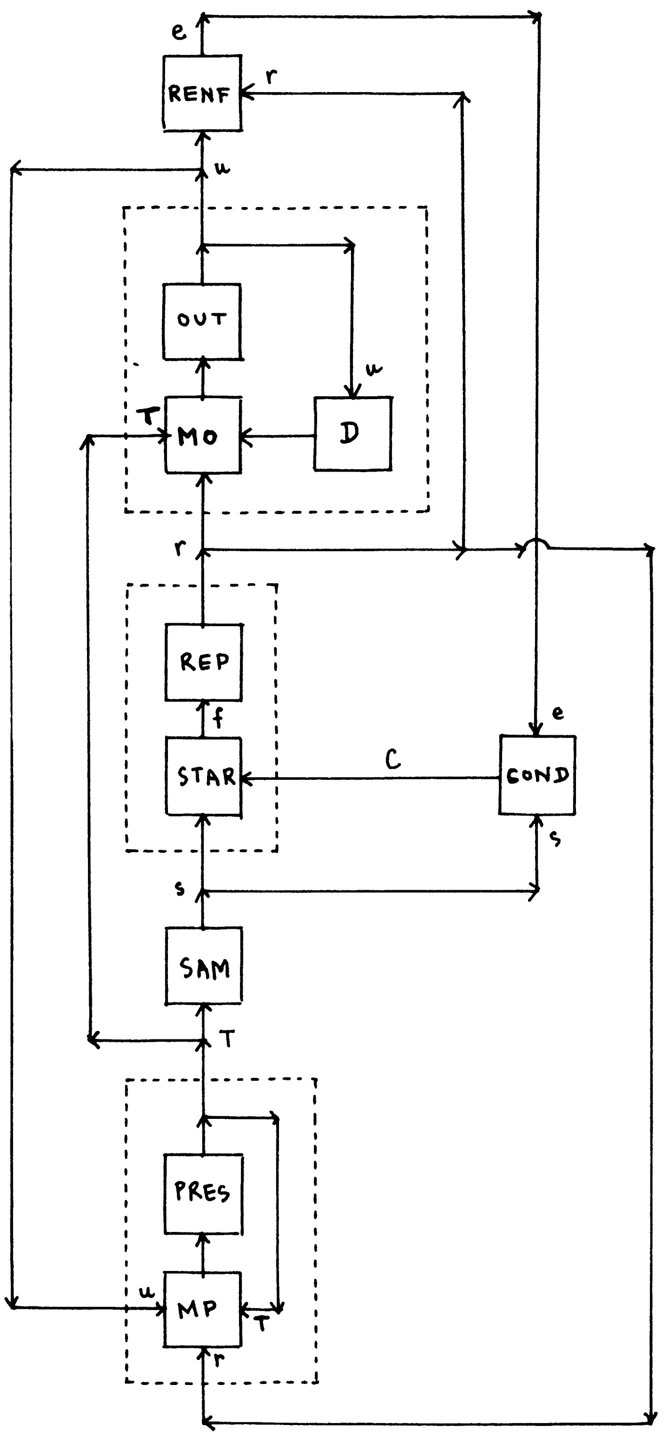


figure 2. Le schéma de $\mathcal{M}$

Soit $\nu \notin R$ et $E = R \cup \{\nu\}$. On dira que ν est le renforcement nul, et que E est l'ensemble des renforcements.

Soit $F = R^S$ l'ensemble de toutes les fonctions $C : S \to R$, appelées fonctions de conditionnement.

On appellera essai un sext-uple (C, T, s, r, u, e) où : $C \varepsilon F$, $s \subset T \subset S$, $r \varepsilon R$, $u \varepsilon U$, $e \varepsilon E$, et on dira que, pour cet essai,

C est le conditionnement

T est l'ensemble (de stimuli) présenté (au sujet)

s est l'échantillon (retenu par le sujet)

r est la réponse

u est l'issue

e est le renforcement.

Soit $\ell : \mathcal{P}(S) \times E \times F \to F$ la fonction définie par :

(i) $\ell(s, \nu, C) = C$

(ii) pour $r \varepsilon R$, $\ell(s,r,C)$ est la fonction : $S \to R$ qui, sur $S-s$ coïncide avec C, et sur s a pour restriction la fonction constante r.

ℓ formalise la façon dont un renforcement e ($e = \nu$ ou $e = r \varepsilon R$) modifie le conditionnement C en fonction de l'échantillon $s \subset S$ retenu par le sujet : le renforcement nul ν ne change rien ; le renforcement r "conditionne à r" chaque stimulus élément de s, et ne change rien en ce qui concerne les autres stimuli, qui sont éléments de $S-s$.

On appellera expérience une suite $x = (x_t)_{t \varepsilon \mathbb{N}}$ où chaque $x_t = (C_t, T_t, s_t, r_t, u_t, e_t)$ est un essai et telle que :

$C_{t+1} = \ell(s_t, e_t, C_t)$ pour tout $n \varepsilon \mathbb{N}$.

Enfin, pour $s \subset S$ et $C \varepsilon F$ soit

$$s * C : R \to \mathcal{P}(S)$$
$$r \mapsto s \cap C^{-1}(r)$$

$(S * C)(r)$ est l'ensemble des stimuli éléments de s qui sont "conditionnés à r" par C. Remarquons, pour la suite, que $s = \bigcup_r (S * C)(r)$.

III.1.2. Schéma

Le schéma de notre multi-automate $\mathcal{M}$ sera donc le schéma de la figure 2. Ce schéma traduit le déroulement d'un essai de la façon suivante (on y a ajouté les notations C,T,s,r,u,e ... pour mieux le suivre) :

L'expérimentateur présente, au sujet, $T = V_{PRES,t}$ (output de PRES à l'étape t), cela en fonction de la "mémoire" qu'il a gardée des $V_{PRES,t'}$; $V_{REP,t'}$; $V_{OUT,t'}$ des essais précédents $(t' < t)$. Cette mémoire est assurée par MP.

Le sujet retient de T l'échantillon s. Cela est assuré par SAM.

Sa réponse $r = V_{REP,t}$ ne dépend de $s = V_{SAM,t}$ et du conditionnement $C = V_{COND,t}$ que par l'intermédiaire de la fonction $f = s * C$, donnée par STAR.

L'expérimentateur donne au sujet l'issue $u = V_{OUT,t}$, cela en fonction de la "mémoire" qu'il a gardée des $V_{PRES,t'}$ et $V_{REP,t'}$ pour $t' \leq t$, et des $V_{OUT,t"}$ pour $t" < t$. Cette mémoire est assurée par MO, par l'intermédiaire de D en ce qui concerne $V_{OUT,t"}$: D représente un automate "délai" qui garde en mémoire son input pour le redonner comme output à l'étape suivante.

Chez le sujet se produit le renforcement $e = V_{RENF,t}$ qui dépend de $u = V_{OUT,t}$ et $r = V_{REP,t}$.

Enfin le conditionnement pour l'essai suivant $C' = V_{COND,t+1}$ dépend de $e = V_{RENF,t}$ et $s = V_{SAM,t}$ mais aussi de $C = V_{COND,t}$, d'où le rôle des états (mémoire) associés à COND.

III.1.3. Valeurs et états

Le système de valeurs de notre multi-automate $\mathcal{M}$ est défini par les égalités suivantes, où l'on utilise les notations introduites précédemment en II.2 :

$Y_{RENF} = E$, $Y_{OUT} = Y_D = U$, $Y_{REP} = R$, $Y_{SAM} = Y_{PRES} = \mathcal{P}(S)$, $Y_{COND} = F$,
$Y_{STAR} = \mathcal{P}(S)^R$ (ensemble de toutes les fonctions : $R \to \mathcal{P}(S)$) ,
Y_{MO} et Y_{MP} sont chacun des ensembles finis (non-vides) fixés, mais quelconques.

De même, le système d'états du multi-automate est défini par :

$Q_D = U$, $Q_{COND} = F$,
$Q_{MO} = Y_{MO}$, $Q_{MP} = Y_{MP}$,
Q_{RENF}, Q_{OUT}, Q_{REP}, Q_{STAR}, Q_{SAM}, Q_{PRES} n'ont chacun qu'un seul élément, qu'il ne sera pas utile de mentionner (absence de mémoire).

III.1.4. Probabilités de transition

Plutôt que de définir formellement chacune des probabilités de transition p_{RES}, p_{SAM}, ... etc ... il sera plus clair de procéder de la façon suivante, justifiée par les résultats généraux, esquissés précédemment en II.3 :

D'une part le multi-automate $\mathcal{M}$ sera bien bouclé, avec pour repère {D, COND, MP}.

D'autre part, les valeurs $V_{COND,t}$; $V_{PRES,t}$; ... etc... et les états $E_{COND,t}$ ... vérifient les relations suivantes dans toute actualisation de $\mathcal{M}$ (où P désigne la mesure de probabilité associée).

$P(V_{RENF,t} = e | V_{REP,t} = r, V_{OUT,t} = u) = p_{RENF}(e|r,u)$ où
p_{RENF} est une certaine probabilité de transition de R x U à E, qui fait partie de la donnée de $\mathcal{M}$.

$P(V_{OUT,t} = u | V_{MO,t} = y_{MO}) = p_{OUT}(u|y_{MO})$ où
p_{OUT} est une certaine probabilité de transition de Y_{MO} à U...

$P(V_{PRES,t} = T | V_{MP,t} = y_{MP}) = p_{PRES}(T|y_{MP})$ où
p_{PRES} est une certaine probabilité de transition de Y_{MP} à $\mathcal{P}(S)$...

$$P(V_{REP,t} = r \mid V_{STAR,t} = f) = \frac{\mathrm{Card}\ (f(r))}{\mathrm{Card}\ (\bigcup_r f(r))} \quad \text{si} \ \bigcup_r f(r) \neq \emptyset$$

$$= p_{REP}(r) \ \text{sinon} \qquad \text{où}$$

p_{REP} est une certaine probabilité sur R... (Card désigne le nombre d'éléments, voir la remarque sur l'opération $*$ à la fin de III.1.1).

$P(V_{SAM,t} = s \mid V_{PRES,t} = T) = p_{SAM}(s \mid T)$ où

p_{SAM} est une certaine probabilité de transition de $\mathcal{P}(S)$ à $\mathcal{P}(S)$ qui vérifie :

$p_{SAM}(s \mid T) = 0$ si $s \not\subset T$

$p_{SAM}(s \mid T) = p_{SAM}(s' \mid T)$ si $s \cup s' \subset T$ et $\mathrm{Card}(s) = \mathrm{Card}(s')$.

Les autres probabilités qu'il reste à indiquer sont "déterministes", c'est-à-dire qu'elles correspondent à des automates composants déterministes. Il est plus clair de les présenter sous la forme des égalités suivantes :

$V_{STAR,t} = V_{SAM,t} * V_{COND,t}$ (le rôle de l'automate symbolisé par STAR est simplement de calculer l'opération $*$).

$\begin{cases} V_{D,t} = E_{D,t} \\ E_{D,t+1} = V_{OUT,t} \end{cases}$ (l'automate symbolisé par D est un automate retard "délai")

$\begin{cases} V_{COND,t} = E_{COND,t} \\ E_{COND,t+1} = \ell(V_{SAM,t}, V_{RENF,t}, E_{COND,t}) \end{cases}$ où ℓ est la fonction de $\mathcal{P}(S) \times E \times F$ dans F définie en III.1.1 (l'automate symbolisé par COND est aussi retard.)

$\begin{cases} V_{MP,t} = E_{MP,t} \\ E_{MP,t+1} = \psi(V_{PRES,t}, V_{REP,t}, V_{OUT,t}, E_{MP,t}) \end{cases}$ où ψ est une certaine fonction de $\mathcal{P}(S) \times R \times U \times Q_{MP}$ dans $Q_{MP} = Y_{MP}$ (l'automate symbolisé par MP est aussi retard).

$V_{MO,t} = E_{MO,t+1} = \phi(V_{PRES,t}, V_{REP,t}, V_{D,t}, E_{MO,t})$ où ϕ est une certaine fonction de $\mathcal{P}(S) \times R \times U \times Q_{MO}$ dans $Y_{MO} = Q_{MO}$.

En récapitulant :

Le multi-automate $\mathcal{M}$ est entièrement déterminé par la donnée :

du schéma de la figure 2, de $S, R, U, \nu, Y_{MO}, Y_{MP}$ et de :

p_{RENF}, p_{OUT}, p_{PRES}, p_{REP}, p_{SAM}, ϕ et ψ.

III.2. Actualisation de ce multi-automate

Pour définir une actualisation de $\mathcal{M}$ il suffit de donner une probabilité initiale (voir II.3). Plus précisément on obtient le résultat suivant :

THEOREME ET DEFINITION : Soient $q_D \in Q_D$, $q_{MO} \in Q_{MO}$, $q_{MP} \in Q_{MP}$ fixés, et considérons une actualisation de $\mathcal{M}$ pour laquelle

$P(E_{D,o} = q_D,\ E_{MO,o} = q_{MO},\ E_{MP,o} = q_{MP}) = 1.$

Alors, si de cette actualisation on ne retient que les six éléments aléatoires

$V_{COND,t}$; $V_{PRES,t}$; $V_{SAM,t}$;

$V_{REP,t}$; $V_{OUT,t}$; $V_{RENF,t}$

on obtient un SSM au sens de [13], c'est-à-dire que les axiomes des SSM d'Estes et Suppes sont satisfaits, en convenant que, à l'essai t,

$V_{COND,t}$ est le conditionnement

$V_{PRES,t}$ est l'ensemble (de stimuli) présenté (au sujet)

$V_{SAM,t}$ est l'échantillon (retenu par le sujet)

$V_{REP,t}$ est la réponse

$V_{OUT,t}$ est l'issue

$V_{RENF,t}$ est le renforcement

On dira alors que cette actualisation représente ce SSM.

La démonstration rigoureuse nécessite que de part et d'autre (actualisation et SSM) il s'agisse d'objets mathématiques rigoureusement définis, et donc en particulier nécessite une définition complète et précise des SSM. Cela conduit à de longs développements formels (voir [4]).

On va plutôt donner ici un aperçu des SSM à travers un commentaire de ce théorème.

Les axiomes des SSM sont de trois sortes.

Les axiomes d'une première sorte spécifient de quelle façon certaines variables dépendent directement d'autres variables :

Un axiome impose que, à chaque essai, si s,e,C sont respectivement l'échantillon, le renforcement, le conditionnement, alors le conditionnement à l'essai suivant soit ℓ(s,e,C). Dans notre actualisation de $\mathcal{M}$ cet axiome est vérifié grâce au choix de l'automate COND.

Un axiome impose que, à chaque essai, l'échantillon s retenu par le sujet, soit une partie de l'ensemble T présenté par l'expérimentateur. Cet axiome est vérifié grâce au choix de l'automate SAM.

Sont aussi vérifiés grâce au choix de SAM un axiome qui exprime une homogénéité par rapport au temps (la probabilité de s ne dépend pas du numéro de l'essai), ainsi qu'un axiome qui exprime que cette probabilité ne dépend de cet échantillon s que par l'intermédiaire du nombre d'éléments de s.

Un axiome impose que, si l'échantillon s est non-vide, alors la probabilité pour que la réponse soit r est égale à la proportion des éléments de s qui sont "conditionnés à r". Cette proportion elle-même ne dépend de s et du conditionnement C que par l'intermédiaire de la fonction f = S $*$ C. Cet axiome est vérifié grâce au choix des automates STAR et REP.

Les axiomes d'une deuxième sorte spécifient l'indépendance de certaines variables par rapport à d'autres variables.

Ainsi à chaque essai le renforcement e ne dépend que de la réponse r et de l'issue u, par exemple, r et u étant donnés, e ne dépend pas de l'échantillon s ni d'aucune des variables (réponse, issue, etc...) aux essais précédents.

De façon générale, ces axiomes expriment des propriétés d'indépendance conditionnelle. Ils sont ici vérifiés grâce au choix du schéma (en vertu des propriétés générales d'indépendance conditionnelle d'une actualisation d'un multi-automate évoquées en II.3).

Enfin une troisième sorte d'axiomes formalise la façon dont l'expérimentateur contrôle l'expérience à partir des seules variables observables, qui sont l'ensemble présenté T, la réponse r, et l'issue u. Ces axiomes font intervenir une notion qu'Estes et Suppes appellent partition de l'expérimentateur et dont la définition générale est assez compliquée.

Par exemple un axiome vise à exprimer que, puisque les probabilités des issues sont du ressort de l'expérimentateur, elle ne peuvent dépendre, à chaque essai, que des observables précédents, à savoir les ensembles présentés, réponses, issues aux essais précédents. C'est d'un mécanisme de mémoire que vise à rendre compte la notion de partition de l'expérimentateur. Ici, dans l'actualisation de $\mathcal{M}$ ce mécanisme de mémoire est obtenu grâce aux états des automates MO, MP,D.
Et ces axiomes sont vérifiés grâce aux choix des automates MO,MP,D.

Cette représentation (du SSM par l'actualisation de $\mathcal{M}$) revient à rendre compte du comportement des variables du SSM (conditionnement, etc...) en faisant intervenir des variables supplémentaires, notamment les états $E_{COND,t}$, etc... Ceci permet en particulier d'expliciter un mécanisme de mémoire...

Une telle représentation ne se situe pas au niveau de la concrétisation physique des mécanismes psychologiques étudiés. D'autre part la distinction entre états et valeurs (ou encore entre d'un côté état d'un automate et de l'autre input et output) ne correspond pas à la distinction entre variables observables et non-observables en psychologie. Il s'agit ici d'une description mathématique (donc abstraite et formelle) de relations entre des variables, qui n'implique pas de prise de position sur la réalité des phénomènes psychologiques non-observables.

De façon générale la "représentation en termes d'automates" est un outil mathématique, neutre vis-à-vis des controverses entre psychologues, mais pouvant éventuellement contribuer à mieux les comprendre.

Notre multi-automate $\mathcal{M}$ dépend de S,R,U... p_{RENF}, p_{OUT},... ϕ et ψ (voir la fin de III.1). On a donc en fait défini toute une classe de multi-automates qui permettent de représenter, par des actualisations, des SSM. On dira que ce sont des <u>multi-automates "Stimulus-Sampling"</u>.

III.3. <u>Problème de la représentation dans le cas général</u>

Le théorème précédent appelle une réciproque : tout SSM peut-il être ainsi représenté ?

S'il ne s'agit que des <u>exemples</u> "concrets" de SSM, par exemple ceux donnés dans [13], la réponse est oui. Mais s'il s'agit de la <u>définition générale</u> et formelle des SSM donnée dans [13], la réponse est non, la classe des multi-automates $\mathcal{M}$ définis précédemment n'est pas assez large. On obtient cependant un résultat général, grâce à un élargissement de cette classe de multi-automates, qui permet de "représenter en termes d'automates", en un certain sens, tous les SSM ; cela nécessite une analyse détaillée et une discussion de la notion de SSM (voir [4]). De ce résultat général on ne va retenir ici que deux aspects :

Tout d'abord, par la notion de SSM, Estes et Suppes ont cherché à exprimer un certain nombre de principes de base relevant de la doctrine behaviouriste "Stimulus-Réponse". Mais, pour définir une notion très générale, ils ont dû formuler un système d'axiomes qui admet des modèles "limites". Par exemple, intuitivement parlant, aucune borne n'est fixée a priori pour la capacité de mémoire de l'expérimentateur. Non seulement il y a des modèles pour lesquels la capacité de mémoire est finie et arbitrairement grande (toujours intuitivement parlant) mais il y a aussi des modèles pour lesquels la capacité de mémoire est infinie. Pour représenter ces modèles en termes d'automates on doit alors généraliser la notion d'automate (fini) en admettant par exemple pour certains automates $M = (X,Y,Q,p)$ un ensemble d'états Q infini dénombrable. Avec cette notion plus générale d'automate on obtient une notion plus générale de multi-automate et une classe de multi-automates "Stimulus-Sampling" plus large que celle définie précédemment.

Mais par ailleurs il semble que certaines contraintes imposées aux SSM ne soient dûes qu'aux difficultés que soulève le langage des SSM pour exprimer des propriétés plus générales, et que certaines intentions sous-jacentes aux axiomes soient plus facilement exprimées dans le langage des automates. Par exemple on pourrait accorder au sujet une certaine mémoire concernant les réponses et les renforcements : il suffirait pour cela de prendre pour Q_{REP} et Q_{RENF} des ensembles non-nécessairement réduits à un unique élément (voir III.1.3) et des probabilités de transition appropriées tenant compte de ces nouveaux états. On obtiendrait alors des modèles qui, de part le schéma du multi-automate respecteraient toujours un certain nombre de principes de base "Stimulus-Sampling", mais qui sous certains aspects seraient plus généraux que les SSM.

De façon générale, il semble que l'intérêt principal d'une "représentation en termes d'automates" des SSM réside dans la possibilité d'explicitation et de discussion qu'elle offre grâce au langage des automates. C'est ce qu'illustre la suite de ce travail.

La démarche qui consiste à représenter les SSM en termes d'automates est une démarche réciproque de celle du "Théorème de Représentation de Suppes"(14) : Certaines objections opposées aux modèles behaviouristes (8 , 9) étant formulées en termes d'automates, ce théorème visait à montrer que la "théorie Stimulus-Réponse" est aussi puissante que la "théorie des automates" (en ce qui concerne ces questions de psychologie de l'apprentissage). En fait ce théorème ramène tout automate déterministe du type "accepteur" (l'output identifié à l'état) à un S-R modèle qui l'approche "asymptotiquement" (l'aspect "Sampling" n'intervient pas ici, c'est pourquoi on parle de S-R modèle). On va revenir sur ce théorème de Suppes dans le IV qui suit.

III.4. Conséquences de cette représentation

III.4.1. Propriété de Markov

Dans [13] Estes et Suppes montrent que pour une très large classe de SSM, une suite d'éléments aléatoires bien choisis constitue une chaîne de Markov finie. Il s'agit des SSM ayant "une règle de l'expérimentateur de caractère fini".

La représentation des SSM en termes d'automates montre l'origine de cette chaîne de Markov : c'est la chaîne formée par la suite de tous les états du multi-automate (voir la fin de II.3). Si le SSM a "une règle de l'expérimentateur de caractère fini", le multi-automate qui permet de le représenter est tel que cette chaîne de Markov peut se définir dans le langage des SSM. Cela est dû à une propriété que vérifient alors les automates M^O et M^P : intuitivement parlant, ce sont des automates pour lesquels existe un entier k tel que l'état à l'étape t est défini par la liste des inputs reçus aux étapes t-1, t-2,...,t-k ; ce qui permet d'exprimer cet état à l'aide de ces inputs et finalement d'exprimer l'état du multi-automate à l'aide des seules variables du langage des SSM.

Mais les automates finis ne vérifient pas tous cette propriété. Intuitivement parlant les automates qui gardent en mémoire, sous forme d'état, une liste de k inputs reçus précédemment ne sont pas les seuls automates pour lesquels on peut parler de capacité de mémoire finie.

En prenant pour MO et MP des automates finis qui ne vérifient pas cette propriété, on a donc la possibilité d'avoir un SSM, encore représenté par l'actualisation d'un multi-automate comme en III.2, auquel est donc encore associée une chaîne de Markov finie (qui s'exprime en termes d'automates) mais qui ne soit pas un SSM ayant "une règle de l'expérimentateur de caractère finie". (Pour plus de détails voir [4]).

En résumé, l'idée d'une capacité de mémoire de l'expérimentateur finie se traduit par l'intervention d'automates finis, qui sont à l'origine d'une chaîne de Markov finie ; et la "règle de l'expérimentateur de caractère finie" n'en est qu'un cas particulier, dans lequel le langage des SSM est suffisant pour définir cette chaîne de Markov.

III.4.2. Sujet et expérimentateur

Ce qui est modélisé par un SSM, puis par l'actualisation d'un multi-automate, c'est l'expérience d'apprentissage. Mais sur le schéma de la figure 2 on peut clairement distinguer ce qui correspond au sujet et ce qui correspond à l'expérimentateur. Pour simplifier on va plutôt considérer le schéma de la figure 1 qui, comme on l'a présenté en III.1, peut être vu comme une version simplifiée du schéma de la figure 2. Sur la figure 3 nous avons représenté le même schéma que celui de la figure 1, mais en mettant en évidence ce qu'on pourrait appeler un "multi-automate sujet", avec SAM, REP, COND, RENF, et un "multi-automate expérimentateur" avec PRES, OUT. (Il s'agirait de multi-automates "non fermés", plus généraux que ceux que nous avons introduits en II).

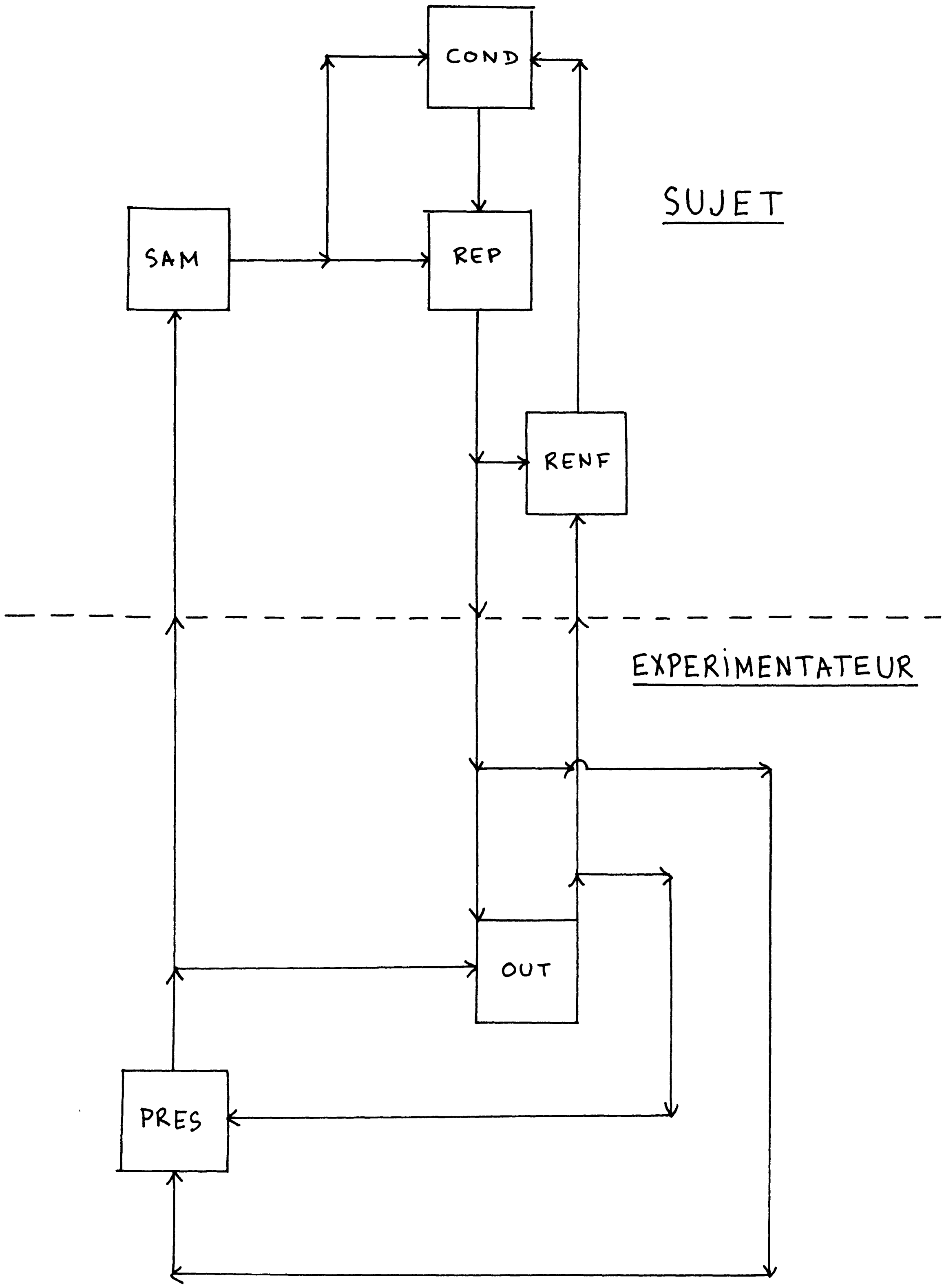


figure 3. Même schéma que celui de la figure 1

Sur cette figure 3, on visualise ainsi une certaine conception "Stimulus-Sampling"

- du sujet
- de l'expérimentateur (plus exactement de leur comportement)
- du système sujet-expérimentateur.

Cette conception, nous allons la comparer avec celle qui résulte d'un autre découpage du schéma, dans le IV qui suit.

IV. AUTOMATE OBTENU PAR CONDITIONNEMENT D'UN AUTOMATE

IV.1. Conditionnement, fonction, automate "accepteur"

De la même façon que Suppes pour son théorème de représentation (14), on va maintenant laisser de côté l'aspect "Sampling" et ne plus parler que de Modèles Stimulus-Réponse (= S.R.).

En termes de SSM, cela revient à ne plus considérer que le cas particulier où l'ensemble de Stimuli T présenté au sujet est toujours une partie de S contenant exactement un élément, donc de la forme $T = \{\sigma\}$ où $\sigma \varepsilon S$, et où l'échantillon s retenu par le sujet est toujours égal à cet ensemble présenté : $s = \{\sigma\}$.

En termes d'automates, et pour simplifier l'écriture, cela revient à remplacer le schéma de la figure 3, dans lequel on peut maintenant se passer de SAM (automate qui ne ferait que transmettre son input comme output) par le schéma de la figure 4. On peut aussi considérer maintenant que les outputs de l'automate PRES sont des éléments de S, c'est-à-dire poser $Y_{PRES} = S$.

Alors on peut aussi considérer que la définition de l'automate REP (en III.1.4) se simplifie au point de rendre cet automate déterministe : dans une actualisation où $V_{COND,t} = E_{COND,t} = C$ (où C est une certaine fonction de conditionnement : $S \rightarrow R$) et $V_{PRES,t} = s$ (où $s \varepsilon S$), on a

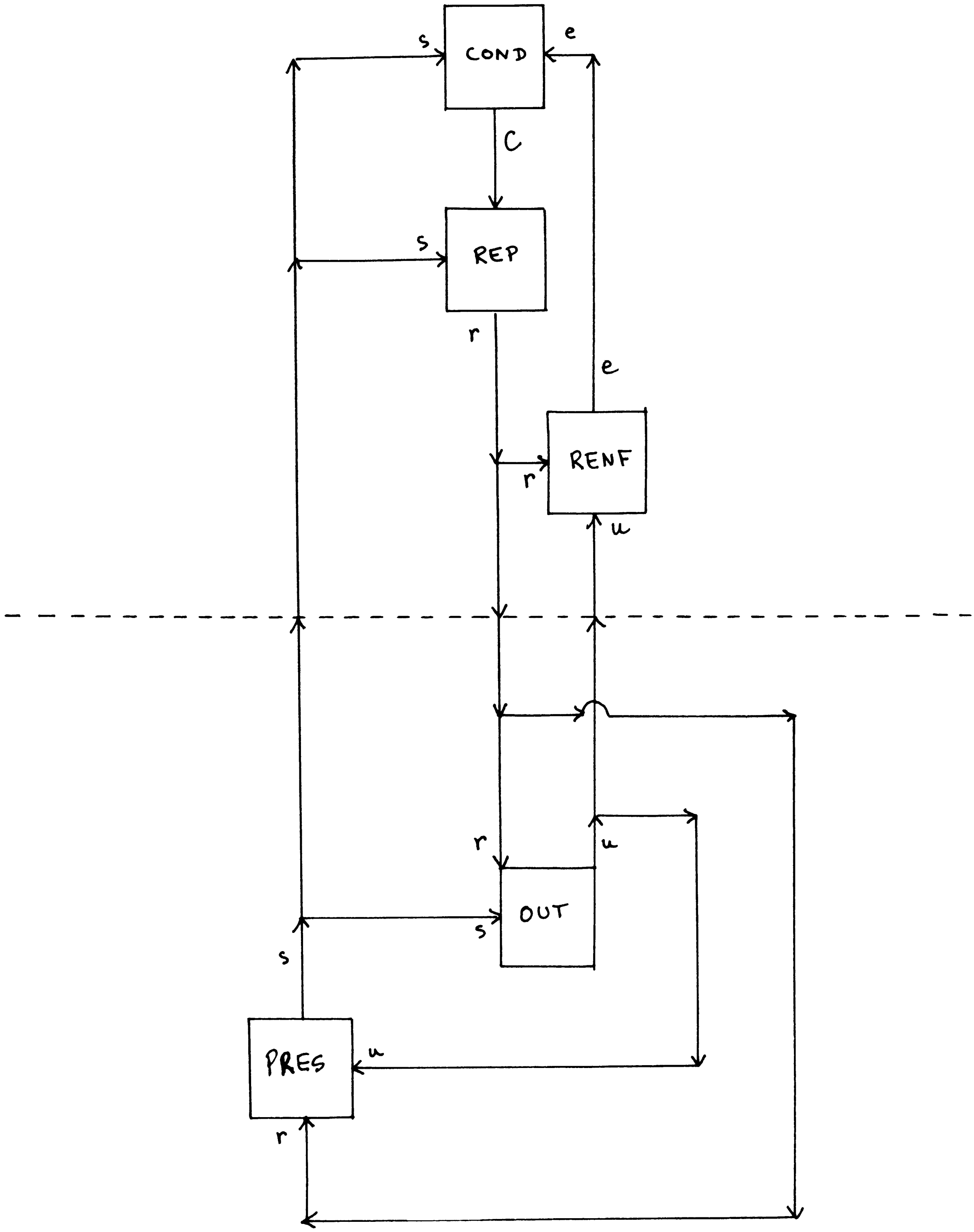


figure 4. Analogue à la figure 3, mais sans "sampling".

$V_{REP,t} = C(s)$.

Supposons, intuitivement parlant, que le résultat de l'expérience d'apprentissage soit le maintien du sujet dans l'état de conditionnement représenté par C, c'est-à-dire que $E_{COND,t} = C$. Alors au Stimulus $s = V_{PRES,t}$, le sujet réagit par la réponse $r = V_{REP,t} = C(s)$.

On peut considérer que le comportement du sujet, appris par conditionnement, est représenté par la fonction C. On va maintenant envisager un comportement plus "élaboré".

DEFINITION : Soit $M = (X,Y,Q,\delta,\lambda)$ un automate déterministe tel que $Y = Q$ et $\lambda = \delta$. Alors M est simplement défini par la donnée de X,Q et δ et on dira que M est un automate "accepteur".

Intuitivement parlant M sert à reconnaître ("accepter") les mots d'un certain langage formel de la façon suivante (1 , 7) : soient $q_o \varepsilon Q$ et $Q_o \subset Q$ fixés ; le langage accepté par (M, q_o, Q_o) est l'ensemble de tous les mots $x_o x_1 \ldots x_n$ sur l'alphabet X qui sont tels que, quand l'automate M part de l'état initial q_o et qu'on lui applique successivement les inputs $x_o, x_1, \ldots, x_n$, son état final (qui est aussi son dernier output) est élément de Q_o.

L'automate accepte ainsi un mot en le "lisant" séquentiellement.

Les langages ainsi acceptés sont les langages réguliers. Pour ces automates particuliers la distinction entre output et état est inutile (en fait, en théorie des langages on ne considère pas du tout d'output, mais ici l'output aura une signification).

Revenons maintenant à la situation d'apprentissage par conditionnement évoquée plus haut. A priori un comportement du type fonction : $s = \text{stimulus} \mapsto \text{réponse} = r$ ne semble pas pouvoir rendre compte de la reconnaissance d'un langage. Ce comportement, sans mécanisme de mémoire, est trop rudimentaire.

Par exemple, savoir reconnaître les mots d'un langage sur X ce n'est pas simplement savoir calculer une certaine fonction définie sur X. En un certain sens, c'est savoir calculer une certaine fonction définie sur l'ensemble de tous les mots, mais c'est un ensemble infini et on ne peut pas prendre les mots comme stimuli.

Dans le cas d'un langage régulier, le mécanisme de mémoire, représenté par un automate accepteur, a un caractère "observable" au sens où l'output produit par l'automate est aussi son état interne. C'est ce qui permet à Suppes (14) de "représenter" un automate accepteur en termes de SSM (par le comportement appris dans une expérience d'apprentissage). Ce résultat lui sert aussi d'argument dans la controverse entre courants "behaviouriste" et "cognitiviste" à propos d'apprentissage de langages (2 , 6 , 8 , 9 , 14 , 15).

La situation se présente alors de la façon suivante :

Le comportement qui doit être appris n'est plus représenté par une <u>fonction</u> : $S \to R$, mais par un <u>automate accepteur</u> M défini par S, R et $\delta : R \times S \to R$, c'est-à-dire que $M = (X,Y,Q,\delta,\lambda) = (S,R,R,\delta,\delta)$.

L'idée de départ est de se ramener à la situation précédente où le comportement appris est bien encore du type

fonction : stimulus $\mapsto$ réponse

mais où la fonction est maintenant $\delta : R \times S \to R$.

C'est-à-dire qu'on prend comme ensemble de stimuli $R \times S$. Mais les éléments de S doivent garder leur interprétation comme "stimuli pour le sujet" alors que les éléments de $R \times S$ sont plutôt des "stimuli pour l'expérience dans son ensemble".

De plus, pour que ce soit le comportement du sujet qui soit représenté par l'automate M, le "stimulus pour le sujet" étant l'input de M, la réponse du sujet étant l'output de M, c'est-à-dire aussi son état suivant, il faut que :

si au stimulus s le sujet réagit par la réponse r, alors cette réponse r doit être $r = \delta(r_1,s)$ où r_1 est l'état de M, c'est-à-dire l'output précédent de M, c'est-à-dire que r_1 doit être la réponse précédente.

D'où l'idée d'imposer à la situation d'apprentissage que le stimulus présenté (au sens de l'expérience dans son ensemble) ait la forme (r_1,s) où r_1 est la réponse précédente.

En termes d'automates, la situation se représente donc de la façon suivante :

Puisqu'on se ramène à la situation précédente on doitavoir un schéma qui se ramène à un cas particulier du schéma de la figure 4.

Mais puisque le nouvel ensemble de stimuli est R x S alors que l'ensemble de réponses reste R, on doit prendre

$Y_{PRES} = R \times S$ et $Y_{COND} = R^{(R \times S)}$

D'autre part, puisque la composante r_1 du nouveau stimulus $(r_1,s) = V_{PRES,t}$ doit être aussi $r_1 = V_{REP,t-1}$, il est plus simple de remplacer PRES par deux autres symboles, STI et DR :

$Y_{STI} = S$, l'automate associé à STI donne pour output le "stimulus pour le sujet" $s = V_{STI,t}$,

$Y_{DR} = R$, l'automate associé à DR est un automate "délai" qui garde en mémoire son input pour le redonner comme output à l'étape suivante $(r_1 = V_{REP,t-1} = V_{DR,t})$.

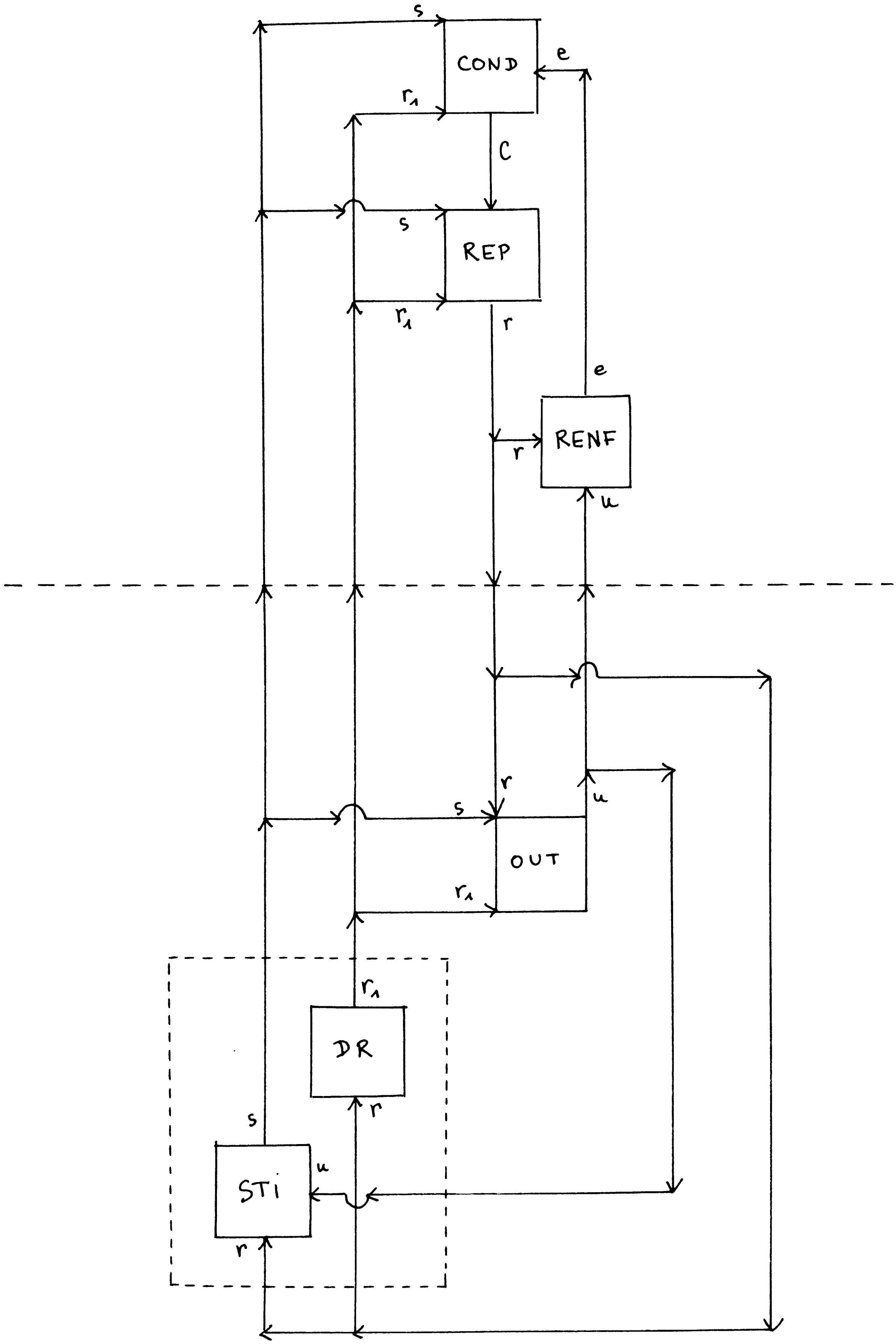


figure 5.

Finalement, on obtient le schéma de la figure 5. Sa structure est celle du schéma de la figure 4, la nouveauté étant le remplacement de PRES par STI et DR, et, pour les valeurs des variables, de s par le couple (r_1, s).

Mais ce schéma (figure 5) est un peu compliqué, et on peut le remplacer par le schéma de la figure 6, légèrement plus simple, sur lequel il n'y a plus de connexion allant de DR vers OUT : en effet, pour représenter la même situation, il suffit de modifier l'automate associé à OUT de façon à ce que, grâce à la connexion qui le relie à REP, il garde en mémoire l'output de l'automate REP, tâche qu'assurait l'automate "délai" DR.

IV.2. Nouveau "multi-automate sujet"

Sur la figure 6 on a reproduit, pour ce schéma, la ligne de démarcation qui fait apparaître un "multi-automate sujet" et un "multi-automate expérimentateur" telle qu'elle figurait déjà sur la figure 4, et même déjà sur la figure 3 (du III.4.2).

Mais pour le sujet dont on parle maintenant et dont le comportement appris doit être représenté par l'automate M, le stimulus est $s = V_{STI,t}$, il réagit en donnant la réponse $r = V_{REP,t}$, et il reçoit aussi, venant de l'expérimentateur, l'issue $u = V_{OUT,t}$.

C'est donc un nouveau découpage du même schéma qui doit faire apparaître un multi-automate, représentant ce sujet. Ce nouveau découpage est indiqué sur la figure 7. Dans le nouveau "multi-automate sujet" figure la boucle passant par DR qui, dans l'ancien découpage, faisait partie du multi-automate expérimentateur.

On a donc un nouveau multi-automate sujet, plus complexe que l'ancien, grâce au phénomène de mémoire que représente cette boucle.

La représentation en termes d'automates permet donc de visualiser sur

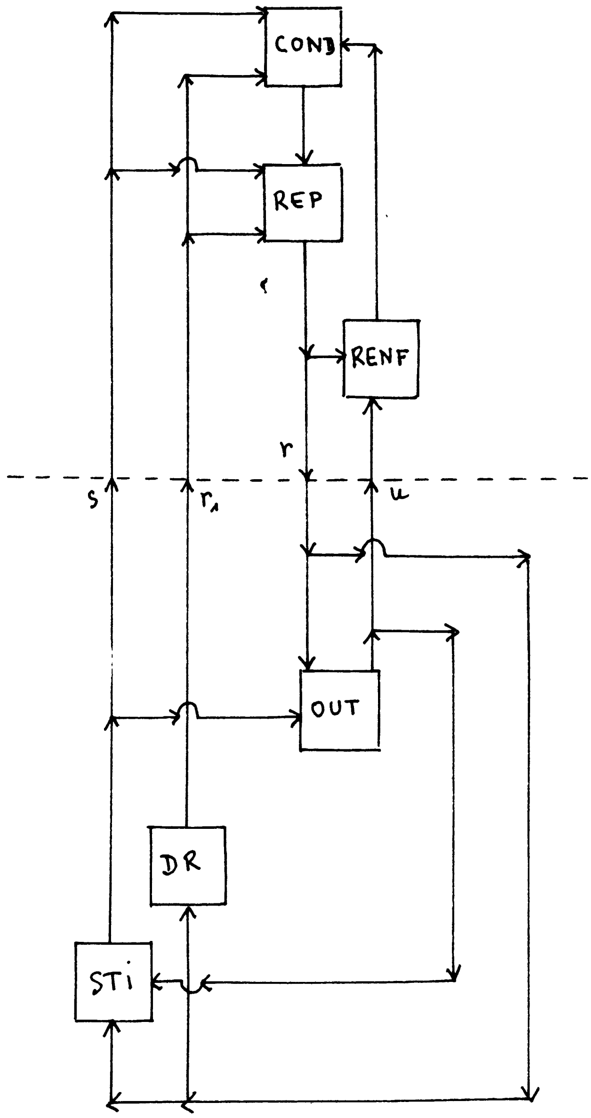


figure 6. légère simplification de la figure 5

NOUVEAU

SUJET

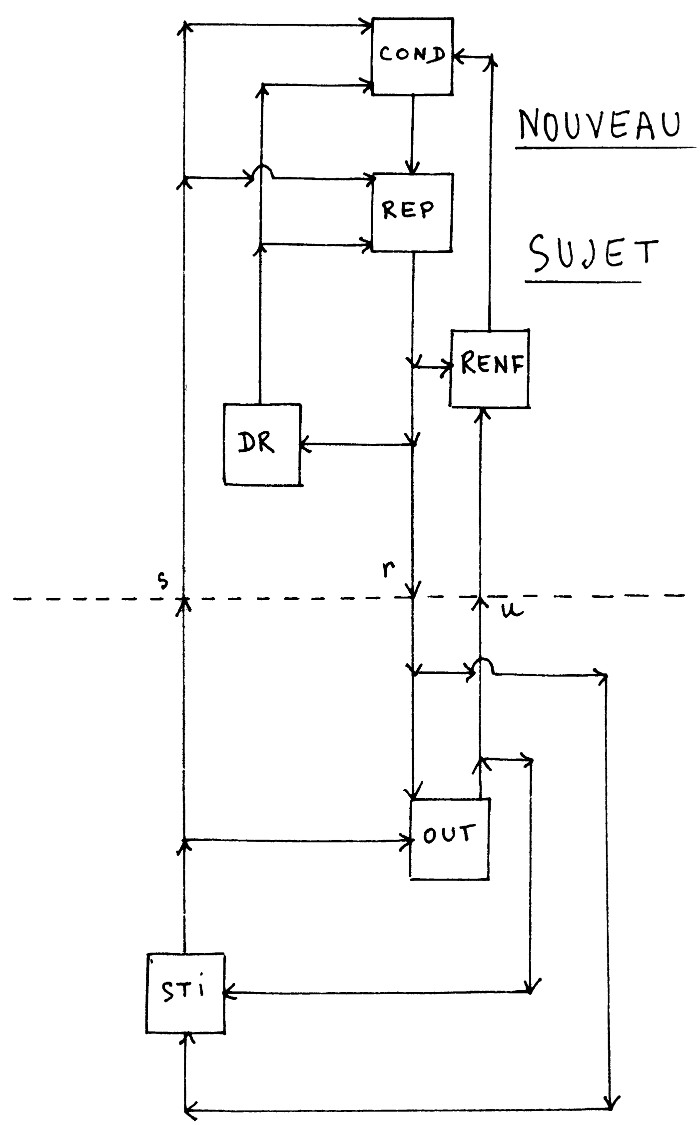


figure 7. Même schéma que la figure 6 mais nouveau découpage

un schéma l'idée du théorème de Suppes, en dissipant la contradiction apparente entre les deux idées suivantes :

1/ Par principe une situation d'apprentissage de ce genre est limitée à l'apprentissage d'un comportement du type "fonction", rudimentaire.

2/ Dans un cas particulier d'une telle situation, on obtient un comportement du type "automate", plus élaboré.

C'est qu'il faut distinguer entre deux points de vue :

D'un point de vue global, il s'agit toujours des mêmes principes S-R d'apprentissage. L'expérience d'apprentissage n'est qu'un cas particulier bien choisi d'expérience représentée de façon générale, en termes d'automates, par un certain type de multi-automates.

Du point de vue du sujet, il y a changement de conception. Le nouveau sujet, représenté par découpage du multi-automate, n'est pas le cas particulier correspondant de l'ancien sujet.

IV.3. Le théorème de représentation de Suppes représenté en termes d'automates

L'exposé qui précède (dans IV.1 et IV.2) n'a qu'un caractère spéculatif dans la mesure où il décrit une démarche conceptuelle possible mais ne donne pas de résultat précis rigoureusement démontré.

Mais on peut donner un exemple, avec démonstration, de réalisation de cette démarche, sous forme d'une version, en termes d'automates, du théorème de Suppes :

On a (dans le III précédent) représenté les SSM en termes d'automates, alors que le théorème de Suppes représente certains automates en termes de SSM.

Si maintenant le SSM que l'on représente en termes d'automates est précisément le SSM du théorème de Suppes, on obtient une version "tout automates" de ce théorème, où la notion d'automate intervient à deux niveaux :

- au niveau de l'expérience d'apprentissage, représentée par l'actualisation d'un multi-automate,
- au niveau du comportement appris, qui est celui d'un certain automate.

Et alors le résultat peut être formulé et démontré de façon autonome, sans référence aux SSM, et donc indépendamment du théorème de Suppes, quant à la logique, mais non quant à l'inspiration.

Ce qu'on obtient ainsi de nouveau c'est la mise en évidence du "multi-automate sujet" qui, globalement, peut être vu comme un automate qui, par conditionnement, apprend le comportement d'un certain autre automate (sous le contrôle d'un automate expérimentateur).

En fait, dans cette version en termes d'automates comme dans le théorème de Suppes, il ne s'agit pas d'une situation d'apprentissage très élaborée. Elle a surtout valeur d'exemple, pour illustrer une possibilité sans trop de complications.

L'outil mathématique essentiel est la chaîne de Markov associée à l'actualisation d'un multi-automate (voir la fin du II). C'est cette chaîne qui, par ses états récurrents, permet de formaliser l'idée d'un "comportement appris" au cours de l'expérience d'apprentissage. (Pour les détails, voir [4]).

Le résultat précis s'exprime plus facilement en termes d'automates qu'en termes de SSM (le langage des SSM reflète la tendance behaviouriste à s'intéresser surtout à l'expérience d'apprentissage dans son ensemble).

Ce résultat concerne des automates fortement connexes, et des automates initialisés connexes (voir [4]).

Intuitivement parlant, pour S et R fixés, on obtient une classe Λ d'automates telle que, pour tout automate accepteur M de la forme $M = (S,R,R,\delta,\delta)$ fortement connexe, tout automate de la classe Λ peut apprendre à se comporter comme M.

On obtient aussi une classe Λ_o telle que, pour tout $r_o \varepsilon R$ et tout automate accepteur M de la forme précédente mais seulement supposé connexe à partir de r_o, tout automate de la classe Λ_o peut apprendre à être ramené à l'état initial r_o par application d'un stimulus spécial s_o, pour ensuite se comporter comme M.

BIBLIOGRAPHIE

[1] ARBIB M.A., Theories of abstract automata, Englewood Cliffs, N.J., Prentice Hall, 1969.

[2] ARBIB M.A., "Memory limitations of stimulus-response models", SUPPES P., "Stimulus-response theory of automata : a reply to Arbib", Psychological Review, 76 (1969), 507-514.

[3] DERRICK J., Réflexions préalables à une étude de la représentation des situations d'apprentissage par les automates, Orleans et Leeds, IREM, 1975.

[4] FERRAND G., Modèles en termes d'automates : questions de fondements logiques, et application à des processus d'apprentissage "Stimulus-Réponse", Orléans, Thèse de 3ème cycle, 1982, Université d'Orléans.

[5] FERRAND G., "On logical foundations of probabilistic automata networks : syntax and semantic", à paraître.

[6] KIERAS D.E., "Finite automata and S-R Models", Journal of Math. psychology, 13 (1976), 127-147.

[7] NELSON R.J., Introduction to automata, New York, J. Wiley, 1968.

[8] NELSON R.J., "Behaviorism is false", The journal of philosophy, LXVI n° 14 (1969), 417-452.

[9] NELSON R.J., "Behaviorism, finite automata and stimulus-réponse theory", Theory and decision, 6 (1975), 249-267.

[10] PAZ A., Introduction to probabilistic automata, New York, Academic Press, 1971.

[11] ROUANET H., Les modèles stochastiques d'apprentissage, Paris, Gauthier-Villars, 1967.

[12] STARKE P.H., Abstract automata, Amsterdam, North-Holland, 1972.

[13] ESTES W.K., SUPPES P., Foundations of statistical learning theory II. The Stimulus-Sampling Model, Technical report n° 26, Stanford, Institute for mathematical studies in the social sciences, 1959.

[14] SUPPES P., "Stimulus response theory of finite automata", Journal of math. psychology, 6(1969), 327-355.

[15] SUPPES P., "From behaviorism to neobehaviorism", theory and decision, 6(1975), 269-285.