Chapitre III

PRINCIPES DE MODÉLISATION : LA « PRAGMATIQUE SIMULÉE »

Dans le chapitre précédent, nous avons examiné les éléments essentiels connus à ce jour des mécanismes biologiques et sociaux du langage, et avons décrit les principales tentatives de modélisation existantes. La distance entre les réalités et leur simulation reste, nous l'avons vu, très importante ; or, une des possibilités de faire progresser l'intelligence artificielle et le traitement de la langue est précisément de faire diminuer cette distance. Autrement dit, le choix de modéliser un certain nombre de propriétés clés du langage, relativement négligées jusqu'ici, devrait permettre une meilleure appréhension du langage par les systèmes artificiels.

III.A Tentative de définition

Ce chapitre met en place les éléments d'une simulation à l'aide d'agents, visant à reproduire le caractère *situé* du langage, c'est-à-dire sont rapport avec la *situation* ou contexte d'énonciation, dans laquelle il faut inclure à la fois les agents linguistiques eux-mêmes, notamment leurs besoins ou valeurs, et le monde de référence. La communication langagière modélisée doit permettre aux agents une meilleure satisfaction des tâches liées à leur survie, et à ce titre le langage doit se rapporter à l'environnement, et donner aux agents une meilleure maîtrise de celui-ci.

III.A.1 Une nouvelle approche

Nous proposons d'appeler « pragmatique simulée » l'approche dont se réclame la simulation que nous allons décrire. Ce terme fait en effet référence à la pragmatique, étude de la communication en contexte (Moeschler et Reboul 1994), et à ce titre il occupe traditionnellement le niveau le plus élevé dans les niveaux de traitement de la langue : morphologie, syntaxe, sémantique, pragmatique. La « pragmatique simulée » réaliserait ainsi des programmes simulant le fonctionnement naturel de cet aspect, par comparaison avec les programmes manipulant des « connaissances pragmatiques », dans un cadre formel dépourvu en général d'une modélisation réaliste des agents linguistiques et d'un environnement.

Notons que la « pragmatique simulée » vise en réalité la modélisation à l'aide d'agents – donc la réduction naturaliste – de phénomènes aussi bien lexicaux, que syntaxiques ou sémantiques. Cependant, les propositions de modélisation qui se limitent

à ceux-ci, négligent précisément la *pertinence* de ces niveaux pour les agents linguistiques, qui est pour nous centrale, et qui est probablement antérieure, dans l'évolution, au lexique et à la syntaxe.

III.A.2 Mise en œuvre

La difficulté de décrire le plan de la simulation que nous préconisons vient du fait que ses nombreux éléments sont liés. En effet, simuler les échanges linguistiques demande un environnement relativement riche, des agents capables d'y évoluer, et un code de communication reliant des états internes des agents à des éléments de l'environnement.

De plus, il serait nuisible au concept même de l'expérience de donner explicitement aux agents les connaissances linguistiques. Il s'agit précisément d'un des points les plus épineux du TAL, contre lequel se sont élevées les modélisations par agents (cf. II.D.2), qui préconisent toutes une *évolution* du code, et un caractère *conventionnel* et *dynamique*. La « pragmatique simulée » défend naturellement l'idée d'une *émergence* des conventions communicatives, mais invoque les nombreux résultats de la biologie pour chercher un ancrage structurel plus fort, dans les agents, de certaines capacités.

Les différents éléments préconisés par la « pragmatique simulée » seront donc d'abord décrits dans une perspective d'ensemble, essentiellement fonctionnelle (III.B) ; la brièveté de la description permettra de ne pas perdre de vue cet ensemble assez complexe. La communication elle-même fera l'objet de propositions, en détaillant d'abord les « actes de langage » élémentaires prévus (III.B.3.i), puis le lien du langage avec l'environnement grâce à la référence (III.B.3.ii.a), et enfin un modèle d'émergence et de transmission inspiré par la créolisation (III.B.3.ii.b). Ensuite, les agents seront décrits plus en détail, à l'aide de considérations plus structurelles. Nous ferons des propositions en ce qui concerne leur microstructure (III.C.1), leurs perceptions et actions (III.C.2.i), leurs valeurs (III.C.2.ii) et discuterons la structure de contrôle (III.C.2.iii). Dans la mesure du possible toutes ces propositions tiendront compte des conclusions actuelles des sciences naturelles, en particulier la neuropsychologie, et de la linguistique, dans un souci de « réalisme ».

Il faut d'emblée prévenir le lecteur des vastes ambitions de la « pragmatique simulée », visible dans des propositions de simulation qui, si elles forment un tout unitaire et intégré, ne peuvent être ici amenées, chacune à part, jusqu'à un niveau de détail suffisant. Les dimensions de la modélisation que nous proposons dépassent donc les possibilités concrètes d'implémentation au niveau de cette thèse. Comme expliqué en conclusion du chapitre (III.D), nous nous concentrerons dans le reste de la thèse sur certains aspects particuliers de la « pragmatique simulée », et notamment sur l'étude fine d'un phénomène clé, la référence.

III.B Description d'ensemble de la simulation

III.B.1 Organisation fonctionnelle d'un agent

Il ressort du chapitre précédent que, pour réaliser un agent linguistique, il faut d'abord réaliser un agent, qui soit muni des trois capacités essentielles : perception, action, et valeurs (ou besoins, ou objectifs). L'utilisation du langage par un tel agent pourra s'inscrire dans la « pragmatique simulée », du moment où ce sera à propos d'entités *perçues*, que l'agent peut *manipuler* et qui acquièrent une importance par rapport à ses *valeurs*. Une représentation possible de l'agent et de ses « organes relationnels » est donnée sur la figure ci-dessous.

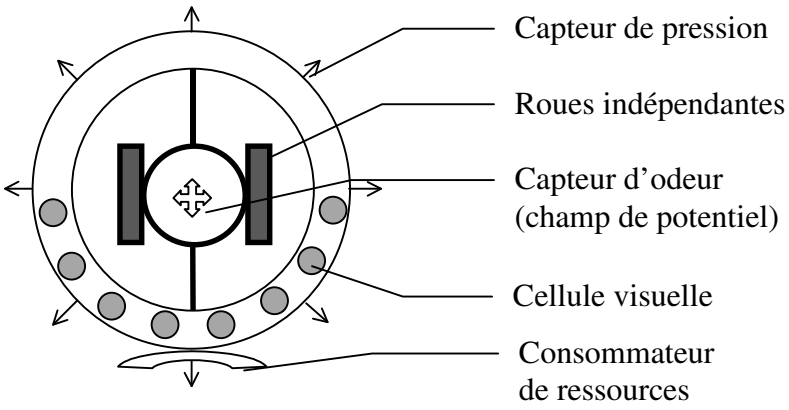

Figure III-1. Aspect d'un agent (vu d'en haut) ; capacités de perception et d'action.

III.B.1.i Perception

En ce qui concerne la perception, trois voies seraient à considérer dans un premier temps : (1) la vision, ou détection de caractéristiques R, V, B (rouge, vert, bleu) à certains endroits d'une rétine artificielle ; (2) l'odorat, ou la détection du gradient d'un certain potentiel ; (3) le canal de communication, ou la réception, pour simplifier, d'un nombre réduit de « symboles » (« tokens ») possible.

Ces trois modalités permettent des interactions suffisamment complexes, et si leur réalisation perfectionnée est complexe, leur simulation à un niveau élémentaire peut être simplifiée. Le rôle de « l'odorat » peut être considéré comme secondaire ; il peut permettre une localisation plus facile d'objets « odorants », et peut simuler une capacité de catégorisation « innée » (réflexes de suivi de certaines odeurs).

III.B.1.ii Valeurs

Les valeurs ou besoins sont simulées au niveau le plus simple par des réservoirs de ressources, dont la consommation est proportionnelle au temps et au niveau d'activité

de l'agent. Ce sont ces réservoirs qui conditionnent la survie de l'agent, et dont le remplissage sera son unique objectif, qui pourra se traduire par des buts intermédiaires.

III.B.1.iii Action

Trois classes d'actions doivent être simulées. Le déplacement (motricité) est bien sûr essentiel aux agents évoluant dans des environnements dimensionnels (un plan par exemple). La satisfaction des besoins nécessite un organe consommateur, qui assimile une certaine quantité d'une ressource présente ; cet organe aura une activité innée importante, mais doit être contrôlé à cause de l'existence d'objets nocifs. Enfin, à la fonction de communication correspond la capacité à émettre un nombre réduit de « tokens », à une certaine distance autour de soi.

Les éléments introduits jusqu'ici sont résumés dans la Table III-1.

Perceptions	*vision* (intensités R,V,B sur une « rétine »)
	odorat (gradient d'un potentiel)
	ouïe (détection de « tokens »)
Valeurs / Besoins	ressources – plusieurs réservoirs
Actions / Effecteurs	*moteurs* (déplacement plan)
	consommateur (assimilation d'une ressource présente)
	parole (émission de « tokens »)

Table III-1. Fonctions d'un agent (les mots en italiques sont des analogues plus ou moins lointains de ces fonctions chez l'homme).

III.B.2 L'environnement et les ressources

L'environnement est surtout conçu comme un réceptacle étendu, et accidenté, de ressources indispensables aux agents. On peut penser à un modèle de type labyrinthe (faibles distances mais nombreux obstacles) ou à un modèle plus « naturel » (distances plus importantes). À un niveau élémentaire, le type exact importe peu, mais si les agents possèdent des capacités de localisation et de navigation, il faut proposer un environnement discernable, i.e. dont chaque emplacement soit muni de propriétés caractéristiques.

Les objets que les agents rencontrent sont des fragments de ressources, ou d'autres substances. Ils peuvent être ainsi utiles à l'agent s'ils sont consommés (déclenchement de l'organe consommateur et remplissage des réservoirs), ou indifférents/inconsommables, ou bien nocifs. Une des principales fonctions du langage sera d'aider à effectuer ces différences.

III.B.3 La communication

III.B.3.i La fonction du langage : actes de langage élémentaires

Afin que le langage soit utilisé par une population en évolution, celui-ci doit avoir une utilité : en d'autres termes, les agents qui utilisent cette possibilité devront être favorisés par rapport aux autres. Il faut rappeler ici la vive discussion dans la communauté des chercheurs sur le critère exact de sélection pour la faculté de langage (communication, représentation, cohésion sociale, etc., cf. II.B.2.i).

D'un point de vue fonctionnel, le langage sera utilisé par les agents *à propos* des objets rencontrés dans l'environnement, et consistera en la production d'un ou plusieurs « tokens ». Nous pensons qu'il faut fixer de façon « innée » les différents actes de langage possibles, car les faire eux-mêmes évoluer semble très long. Nous avons établi une liste possible, en commençant par les plus simples :

- *commentaire* – en présence d'une ressource, un agent produit par association un signal diffusé autour de lui ; une version de cet acte consiste en « jeux de nommage » (cf. travaux de Steels, II.D.2.iv). Toutefois, ces jeux ne semblent pas avoir d'utilité directe pour la satisfaction des besoins de nos agents ; une version différente consiste à interpréter les signes émis comme *associés à la valeur* de la ressource rencontrée (« miam » vs. « pouah »). Cela permettrait alors une transmission d'expérience entre les agents, pour éviter la consommation des substances nocives et favoriser celle des ressources utiles.

  Ce système assez primaire peut être comparé à l'usage d'attributs accompagnés de déictiques ; l'étape suivante serait l'émergence de « noms » pour différencier les objets d'intérêt axiologique équivalent. C'est à ce stade qu'on retrouverait les jeux de nommage, pour aboutir à des assertions du type nom + attribut. Le nom permet de distinguer un des objets parmi ceux présents, mais on peut aussi imaginer une transmission d'information en l'absence de l'objet.

  Remarquons que cet acte ne nécessite pas une réponse linguistique, mais a seulement une conséquence sur l'état cognitif de l'agent qui le décode.

- *information* – l'agent indique la présence d'une ressource à un autre agent, plus ou moins lointain, à l'aide d'un attribut (ressource utile ou nocive) ou d'un nom. On peut voir cet acte comme un prolongement du commentaire, à la fois dans le temps (émission différée) et dans l'espace (émission à un endroit voisin) ; cet acte ne serait pas directement lié à la consommation de la ressource. Cet acte est le premier élément d'une coopération plus immédiate à l'aide du langage.

  Une version plus élaborée de l'acte devrait ajouter au nom ou attribut utilisé pour la désignation la possibilité d'indiquer la localisation de la ressource. Cela suppose des

capacités de représentation spatiale et de navigation, dont la reproduction n'est pas facile (mais cf. acte suivant).

- *demande_ressource* – cette fois-ci, un agent demanderait à un autre la localisation d'une ressource nécessaire dans l'immédiat ; cela suppose une réponse indiquant, même de façon sommaire, un emplacement. Il faut aussi que les agents mémorisent l'emplacement des ressources rencontrées par le passé. Cette tâche paraît difficile dans un environnement de type labyrinthe, muni d'un nombre faible d'indices. Toutefois, un apprentissage par association serait efficace dans un environnement plus « naturel », où les caractéristiques sont nombreuses, et certains emplacements faciles à désigner (p.ex. : « au sommet de cette colline »).

  Une version plus élaborée serait la demande d'aide d'un agent pour un autre ; il faudrait alors prévoir de telles tâches dans l'environnement, et la capacité des agents à les décrire.

III.B.3.ii L'évolution du langage

Nous venons de décrire en quelque sorte la « sémantique », très primitive, d'un certain nombre d'actes de langage ; de nombreux autres éléments restent à décrire, telle bien sûr la très problématique implémentation de ces actes, qu'on ne conçoit aisément que pour les versions les plus faciles. Cette implémentation devra s'intégrer à la structure d'agent autonome, que nous esquisserons au III.C.1 ci-dessous.

Un point important lié à l'évolution du langage est celle de la population. Il serait bien sûr irréaliste de transmettre d'une génération à l'autre des conventions linguistiques. Simplement, il faudra remplacer régulièrement une partie de la population, afin d'introduire des agents vierges de toute connaissance linguistique. Dans un premier temps, il n'y aurait donc pas besoin de faire évoluer les structures de traitement linguistique, dans les agents. Si on se réfère au modèle d'évolution de Bickerton (cf.II.B.2.ii), il ne serait pas utile d'observer expérimentalement l'apparition d'une « mutation unique », et on peut tenter de définir manuellement une structure interne capable de traitement syntaxique, qui s'instancie par contact avec d'autres agents, comme dans la seconde expérience décrite ci-après.

Deux scénarios sont à étudier expérimentalement, correspondant aux deux phases d'émergence décrites notamment dans le modèle de Bickerton (II.B.2.ii)

III.B.3.ii.a Protolangage et référence

Les simulation de l'apparition ou du développement du protolangage consistent à mettre en évidence l 'établissement d'un lexique, dont les agents se serviront pour référer aux objets, au fur et à mesure que ce lexique devient plus discriminant. Ce lexique comprend des « adjectifs » qui qualifient les ressources en fonction des valeurs, les noms de différents traits et de différentes catégories de ressources, et des mots pour désigner la nature de l'acte de langage, si les confusions sont possibles.

Lorsque la tâche devient plus complexe, et le nombre d'objets environnants augmente considérablement, les agents ne possèdent plus de noms uniques pour les objets. Il faut alors étudier de manière plus fine le lien le code et les objets perçus, ce qui est un des aspects de la référence.

III.B.3.ii.b La créolisation

Un modèle de la créolisation mettra en évidence la régularisation du protolangage par un processus rapide, conditionné par l'apparition de nouvelles capacités structurelles dans les agents. Il faut à ce stade mettre en présence des agents « instructeurs », connaissant le lexique du protolangage, mais incapables d'organiser (ordonner) les mots dans les syntagmes (essentiellement les syntagmes nominaux servant à désigner les objets), et des agents « appreneurs », munis d'une structure interne qui tend naturellement à ordonner les syntagmes (le mécanisme exact est à déterminer, mais il serait fondé sur un mécanisme d'inhibition séquentielle entre mots, cf. III.C.1.iii et III.B.3.ii.b). Cette nouvelle génération établirait des conventions syntaxiques qui n'existaient pas auparavant ; les agents instructeurs peuvent être doués *par le programmeur* de conventions lexicales proche de celles du français ou de l'anglais.

Le modèle de la créolisation serait à utiliser pour transmettre aux agents un système de noms et d'articles proche du français. Les agents « instructeurs » pourront être programmés de façon plus élémentaire, pour produire simplement des noms en rapport avec des objets de l'environnement, dans le désordre. Les agents « appreneurs » seront munis d'une structure permettant l'usage de la référence en fonction de leurs « représentations » (nous y reviendrons longuement). Leur module linguistique sera capable de représenter des « règles grammaticales » (nous y reviendrons aussi) initialement non instanciées. Leur système représentationnel permettrait à ces agents de régulariser l'emploi des noms entendus depuis les instructeurs, et d'ajouter des articles pour signaler les différents cas référentiels.

III.C La structure des agents

Nous venons de présenter les principes de simulation dans leur ensemble, et nous avons vu les nombreux liens entre les différents niveaux. Nous allons maintenant apporter des précisions sur l'organisation des agents en vue de satisfaire les fonctions décrites.

III.C.1 Microstructure des agents

Une question qui doit être posée avant toute réalisation est celle de la structure interne des agents. En effet, l'implémentation des structures correspondant aux fonctions précédentes dépend beaucoup du paradigme choisi.

III.C.1.i Implémentation homogène ou hybride

Dans une perspective « animat », de simulation des propriétés du vivant, on peut soutenir que la microstructure des agents doit être, comme pour les humains, *homogène* : la même technique devrait permettre d'obtenir les différentes fonctions prévues, des aptitudes sensori-motrices au contrôle et à la communication. Une imitation plus poussée de la réalité devrait conduire à l'adoption d'une architecture connexionniste (discussion ci-dessous). Mais une implémentation symbolique n'est pas à exclure, moyennant une transformation des paramètres d'entrée/sortie (stimuli et actions) en couples attribut valeur. Toutefois, le risque est réel d'adopter, au niveau linguistique, une technique proche des grammaires formelles, dont on ignore la forme exacte dans le cerveau humain.

Une autre stratégie consiste à réaliser un modèle *hybride*, où les différentes fonctions soient desservies par des architectures différentes : par exemple connexionniste pour la perception, à base de règles pour le contrôle, et utilisant un modèle statistique pour la communication. Cette option nous paraît intéressante dans une perspective provisoire, lorsque le développement de la simulation en est à son début. Et, de fait, nos différentes réalisations, qui explorent différents aspects de la « pragmatique simulée », utilisent aussi des implémentations différentes.

III.C.1.ii Connexionnismes

III.C.1.ii.a Une variété de modèles

Nous avons déjà évoqué, aux chapitres précédents (notamment au I.C.2 et au II.C.3), l'existence de plusieurs approches à l'intérieur du courant connexionniste. On peut ainsi distinguer les modèles non-contraints, qui appliquent des algorithmes d'apprentissage classiques à des réseaux peu différenciés, que le Groupe PDP a utilisés pour des « explorations dans la microstructure de la cognition » (I.C.2.ii). Nous avons vu aussi le « connexionnisme contraint » défendu par Regier (II.C.3.iii).

Les neurobiologistes défendent quant à eux des modèles plus proches des réalités qu'ils observent, mais les capacités de traitement de ces modèles sont en général plus réduites que celles des premiers. Les modèles (par exemple ceux de la vision) considèrent souvent des chaînes de traitement dédiées, linéaires (monosynaptiques), le paramètre significatif étant *le taux individuel de décharge* (ce qui supposerait l'existence de « neurones grand-mère »). Au niveau sensoriel, on aurait une convergence de ces chaînes, alors qu'au niveau moteur, il y aurait divergence à partir des neurones de commande vers les effecteurs.

Edelman préfère attribuer, à travers la TSGN, une signification à l'activation *moyenne* de groupes localisés de neurones, mais ensuite leur organisation semble reprendre ce principe de ces chaînes linéaires, d'abord convergentes, puis divergentes, tout en leur ajoutant l'idée de réentrance, selon ces mêmes chaînes. Des études plus récentes (Philips et Singer 1997) étudient les connexions latérales (une variété de réentrance transversales permettant l'association entre stimuli, et la facilitation latérale

(amplification de la sensibilité). Toutefois, ces modèles font en général abstraction des propriétés temporelles fines des décharges neuronales ; nous allons voir maintenant, par souci d'exhaustivité, un modèle qui prend celles-ci en compte, puis sa comparaison avec les « chaînes dédiées ».

III.C.1.ii.b Les « chaînes synfire »

Moshe Abeles (1982, 1991) suppose que des *groupes de neurones* envoient des potentiels d'action de façon synchrone, ce qui déclenche un autre groupe, et ainsi de suite. Il appelle cela une « synfire chain », ou chaîne de déchargement synchrone. Chaque groupe décharge vers un nombre important de cellules, mais seul un nombre restreint cumulent suffisamment de potentiels d'action en entrée pour décharger à leur tour. Les neurones jouent donc le rôle de détecteurs de coïncidences, ce qui est plus conforme à ce qu'on connaît de leur morphologie. Ce modèle permet de rendre compte de propriétés d'associativité (des liens partiellement activés peuvent activer un ou plusieurs groupes « collatéraux »), qui ne sont pas sans évoquer la fonction de la réentrance latérale chez Edelman.

III.C.1.ii.c Comparaison

L'hypothèse des « lignes dédiées » s'inspire des nombreuses découvertes sur le système visuel, notamment celles de (Hubel et Wiesel 1979) ; il y a dans ces régions des neurones répondant (individuellement) à l'orientation, et aussi au mouvement d'atteinte d'un objet. Toutefois, aucun neurone « grand-mère » n'a été observé dans les régions associatives du cortex (contrairement à des neurones sensibles à la faim, dans l'hypothalamus latéral). De plus, si le système visuel périphérique semble bien organisé selon des « lignes dédiées », ce ne serait le cas ni pour l'odorat, ni pour l'ouïe. Une comparaison avec les chaînes « synfire » figure dans la table ci-dessous.

On peut supposer que certaines régions du cerveau utilisent un codage du premier type, alors que d'autres, peut-être plus centrales, utilisent un codage du second type. Il resterait alors, en neurosciences, à préciser comment se font le transcodage et la transition entre les deux codages, comment par exemple une « ligne » peut se développer en une chaîne « synfire ».

	Chaînes « synfire »	« Lignes dédiées »
Code	synchronisation d'un groupe	taux de décharge individuel
Champs récepteurs	brouillés	conservés ou réduits
Topographie	mélangée	préservée
Fonction d'un neurone	appartenance à plusieurs réseaux	appartenance à une seule ligne
Résistance	grande	faible
Parallélisme topologique	oui – intersection *simultanée* interdite	oui – *toute* intersection interdite

Table III-2. Comparaison de deux types de microstructure, d'après (Abeles 1982).

Ces considérations sont importantes pour le réalisme de notre simulation d'un agent : il n'est bien sûr pas question de prétendre à un quelconque degré d'exactitude dans l'imitation des structures, mais nous supposons que les chances de proposer une solution convenable et extensible sont plus grandes si l'on va *dans le sens de la réalité*, et pas à son encontre.

III.C.1.iii Proposition de mécanismes élémentaires

Nous proposons comme principe de modélisation l'utilisation de « briques » élémentaires inspirées des théories précédentes. Dans ces briques figurent des *unités*, qui sont à concevoir comme des groupes de neurones au sens d'Edelman ; leur implémentation informatique sera étudiée plus en détail au chapitre suivant, sur un exemple concret.

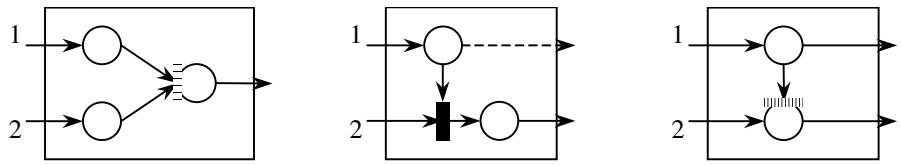

Figure III-2. Trois connexions élémentaires : association, séquence, facilitation.

Trois variétés de briques nous paraissent suffire, dans un premier temps, pour représenter une gamme étendue de connexions et de fonctionnalités (Figure III-2).

- *association* – il s'agit d'un détecteur de coïncidences : un signal est transmis si et seulement si les deux unités d'entrée sont actives au même moment. Cette brique est à la base de la catégorisation, telle qu'elle est proposée par Edelman (cf. II.A.2) : cela permet de construire, par combinaison, des détecteurs de traits complexes dans les signaux perceptifs, à conditions qu'ils soient (approximativement) synchrones.

- *séquence* – cette connexion permet de détecter la séquence d'activation « 1 puis 2 » ; en effet, l'activation de 1 permet le passage, pendant un certain laps de temps, de l'éventuelle activation 2. Ce type de connexion n'est pas présent chez Edelman. Il est en revanche inspiré de l'étude de (Dehaene, Changeux et Nadal 1987), qui propose un modèle capable d'apprendre une séquence de « notes » (analogie avec le chant des oiseaux).

- *facilitation* – cette connexion permet la facilitation contextuelle, c'est-à-dire l'activation de 1 augmente les chances d'activation de 2, ou autrement dit son taux d'amplification. Cela permet de rendre compte de certains « facteurs contextuels », comme l'illustrent Philips et Singer (1997). Sur ce même mode, on peut aussi concevoir une connexion inhibitrice.

Il est important de noter que le sens de ces connexions peut être inversé dans la zone responsable de l'action, aboutissant ainsi aux connexions *symétriques*. En effet, il s'agit d'une région de divergence des commandes, celles-ci correspondant à des séquences de gestes coordonnés.

L'activation de ces unités est numérique, tout comme les signaux transmis dans les connexions. Les unités peuvent amplifier ou non les signaux reçus, outre le traitement décrit par les trois connexions de base. Les connexions ont bien sûr des poids variables : un problème épineux est bien sûr leur évolution. Les algorithmes d'apprentissage classiques ne sont pas applicables, étant donnée la variété des connexions, et surtout l'impossibilité de fournir une « sortie désirable » du réseau. Il est plus facile d'imaginer, dans un premier temps, un mode de conception du système analogue aux réalisations d'Edelman et de son équipe, pour *Darwin III-IV* et le modèle *RCI* (II.C.2.i et II.C.2.ii) : tout est mis en place manuellement. La définition des briques permet d'envisager, à plus long terme, la définition d'un programme génétique qui fournirait le mode de connexion typique (« patron ») d'une région, programme évoluant comme dans les expériences neurogénétiques (II.C.1.ii).

Nos propositions ne remplacent bien sûr pas un définition précise à travers une implémentation concrète, puis le test de l'architecture neuronale réalisée. Toutefois, ce support microstructurel décrit ici indépendamment des fonctions associées n'est qu'un des aspects qui sont doivent être étudiés par la pragmatique simulée. Aspect déterminant, certes, dans la mesure où ils semble conditionner toute réalisation, nous l'étudions dans un modèle expérimental au chapitre suivant (IV.A) ; toutefois, pour progresser dans l'étude, certains comportement du réseau peuvent être simulés en attendant leur réalisation, comme nous le montrons avec l'étude de la syntaxe au IV.B, et *a fortiori* avec l'étude de la référence dans la deuxième partie.

III.C.2 Intégration des capacités de l'agent

Par rapport aux indications fonctionnelles données au début de ce chapitre (notamment au III.B.1) nous allons apporter ici quelques précisions sur la définition des différentes capacités des agents.

III.C.2.i Perception vs. motricité

Pour la perception, le principe de catégorisation par détection de coïncidence de traits fournit une bonne base de départ. La finesse du système de catégorisation peut être graduellement augmentée, mais à son début ce système sera fortement lié aux valeurs : les percepts seront catégorisés en fonction de leur pertinence par rapport aux ressources internes. Le caractère convergent des connexions, des perceptions vers les groupes chargés de reconnaître leurs coprésence, est visible sur la Figure III-3.

On peut imaginer un système analogue de catégorisation pour les signaux internes, tels les niveaux des ressources, mais le traitement « inné » des besoins sera évoqué ci-dessous. On a représenté donc aussi la prise en compte des valeurs comme des associations (on parle d'ailleurs volontiers des « aires associatives » du cortex).

En ce qui concerne le système moteur, nous dirons que les connexions constituent un système de « décatégorisation », dans le sens où elles vont du plus synthétique (unités de commande) vers le plus explicite (organes effecteurs). L'apprentissage des connexions reposerait en grande partie sur la proprioception, que nous n'avons toutefois pas représentée sur notre schéma (Figure III-3). Nous en donnons toutefois une analyse lors de notre implémentation de l'agent, et montrons comment elle peut être remplacée (cf. IV.A.3.iii.b et IV.A.4.i).

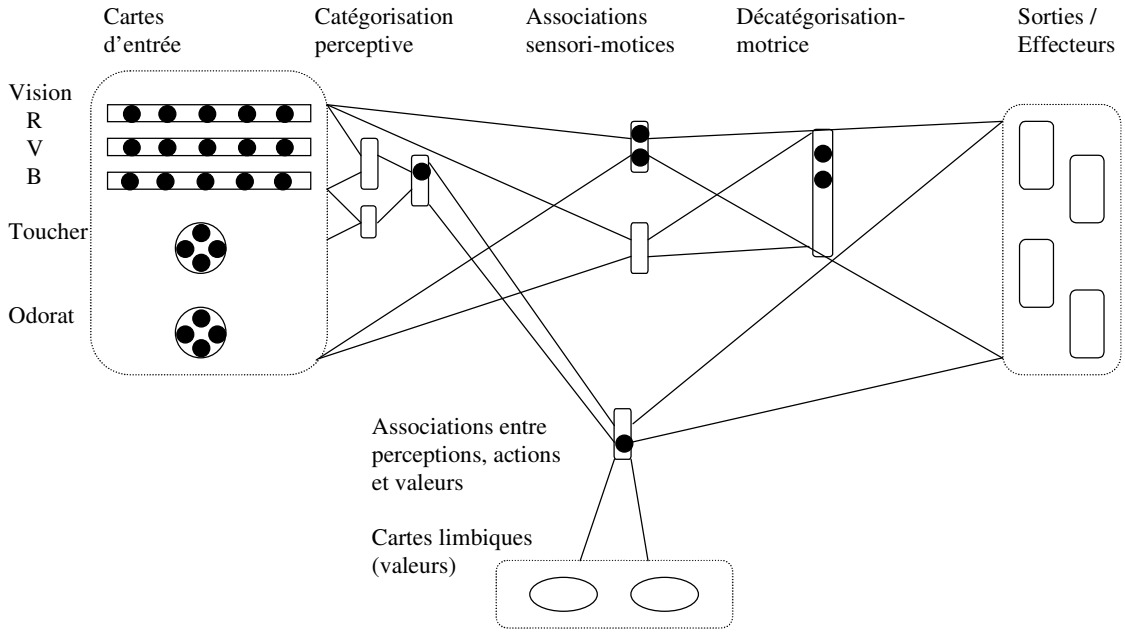

Figure III-3. Intégration perceptions-actions-valeurs grâce à la microstructure.

III.C.2.ii Valeurs ou besoins

La modélisation des besoins internes est difficile parce que leur fonctionnement neuronal est encore assez problématique ; on connaît certes des trajets nerveux *intéroceptifs*, et certains organes impliqués dans la régulation interne, mais ce qui pose problème est leur lien avec la *motivation* de l'individu (ici de l'agent) et ses comportements. En d'autres termes, on ignore comment sont déclenchés les comportements de satisfaction de besoins (p.ex. la recherche de nourriture).

Pour simplifier, la question peut se ramener à l'alternative : les zones cérébrales intéroceptives (projections dans l'hypothalamus et l'amygdale) sont elles directement l'origine de signaux nerveux répercutés au niveau moteur, ou bien activent-elles des liens sensori-moteurs spécifiques, responsables du comportement ?

La deuxième solution semble plus proche de la réalité. On a tendance à concevoir la motivation (animale) comme un réflexe complexe, déclenché à la fois par des stimuli internes et externes (la présence de l'élément satisfaisant le besoin). De façon générale, on sait que l'existence d'un besoin interne accroît le niveau général d'activité par la libération de certains neurotransmetteurs.

Pour notre part, nous avons représenté sur la Figure III-3 des connexions sensori-motrices dépendant de la valeur, pour suggérer que celle-ci déclenche un certain comportement (p.ex. la recherche d'une ressource). Toutefois, c'est le premier terme de l'alternative qui sera exploité dans notre première expérimentation.

III.C.2.iii Structure de contrôle et communication

L'intégration des capacités étudiées est peut-être un des points les plus problématiques, ce qui est déjà visible sur le schéma. En effet, si l'on conçoit assez bien une architecture de boucles sensori-motrices modulées par la valeur, il semble assez difficile de leur ajouter un contrôle plus planifié.

Nous proposons à ce stade deux directions. La première consiste bien sûr à développer le modèle dans le sens proposé par Edelman, vers une possible « conscience primaire » (cf. II.A.2.iii). Une difficulté est que ce développement n'a pas une fonction évidente pour nos agents, donc devra être introduit explicitement dans leur structure (pas d'émergence). Son lien avec la communication n'est pas immédiat, et, plus gênant, son lien avec l'action est peu abordé dans les travaux d'Edelman.

La seconde direction serait l'exploitation des travaux existants sur la planification des actions d'un agent. Ce sont moins les nombreux travaux symboliques qui nous intéressent, mais certains modèles de type conexionniste, par exemple celui, assez simple, développé par Dehaene et Changeux à propos d'une tâche simple de prévision connue en neuropsychologie (Dehaene et Changeux 1991). Un modèle plus réaliste, mais plus éloigné de l'implémentation, est le modèle de la planification de Tim Shallice (1988:419 sq.).

III.D Perspective

Ces nombreux points à approfondir sont à l'heure actuelle bien en avance sur les travaux observés dans la communauté de la modélisation par agents des échanges langagiers. Il s'agit en effet d'un programme de recherche, ou d'une démarche – celle de la « pragmatique simulée » – qui suppose de nombreux travaux de détails, puisque sur chacun des points précédents les données accumulées sont nombreuses, ainsi que les tentatives d'explication. Nous avons voulus esquisser ici le *tableau d'ensemble* qui regroupe tous ces éléments, et qu'on ne doit pas perdre de vue dans les travaux de détail.

Dans la suite de cette thèse, nous allons explorer ce paradigme de la « pragmatique simulée » à travers un modèle d'agent adaptatif mettant en œuvre une architecture conforme aux esquisses ci-dessous, puis un modèle d'émergence de la syntaxe qui utilise cette fois-ci une grammaire formelle ayant une pertinence linguistque reconnue.

Puis, dans la deuxième partie de la thèse, nous aborderons un problème central, et original dans le cadre de la modélisation multi-agents du langage : la référence, qui est l'étude fine de la désignation (ici par les agents) d'entités de l'environnement à l'aide d'expressions du code de communication. Afin de franchir des étapes plus avancées sur la voie de la « pragmatique simulée », nous proposons un modèle d'usage de la référence par un agent, puis un modèle représentationnel, implémenté dans un système de résolution que nous étudions et évaluons dans le détail.

Chapitre IV

DEUX RÉALISATIONS : ÉTUDES D'UN AGENT ET D'UNE SOCIÉTÉ

Nous allons maintenant décrire deux études que nous avons réalisées dans la direction de la « pragmatique simulée » décrite au chapitre précédent. Il ne s'agit pas encore de réalisations intégrant toutes les propriétés touchant au langage, mais de deux expériences partielles, s'intéressant chacune à un aspect différent. La première étudie un modèle simple d'agent isolé fondé sur le « darwinisme neuronal »[1]. La seconde étudie une société d'agents qui établissent des conventions syntaxiques au travers de dialogues[2]. Ces réalisations nous permettent d'évaluer un certain nombre d'idées et de techniques, et montrent la faisabilité du modèle plus riche et plus intégré présenté au Chapitre III, et dont nous étudierons l'aspect référentiel au Chapitre IV.

IV.A Un agent adaptatif simple inspiré par le « darwinisme neuronal »

Nous allons rappeler brièvement les principes que nous cherchons à étudier ici, puis nous décrirons l'architecture conçue, avant de rendre compte des résultats obtenus, et discuter l'« intentionnalité élémentaire » observée. Des enseignements retenus pour les futures modélisations seront donnés à la fin.

IV.A.1 Principes edelmaniens retenus

Nous avons déjà donné un aperçu des idées de Gerald Edelman (cf. **II.A.2**), dont la théorie du « darwinisme neuronal » part de quelques principes simples et parvient à rendre compte de phénomènes sensori-moteurs complexes, et même, selon son auteur, de l'émergence du langage et de la conscience. Rappelons donc que pour la TSGN (Théorie de Sélection des Groupes de Neurones), les couches perceptives du cerveau, et le cortex

[1] Une version préliminaire de ce travail a été publiée en français (Popescu-Belis 1996), et résumée en anglais (Popescu-Belis 1997).

[2] Nous avons bénéficié, pour la programmation de l'expérience, de l'aide de Christophe Allexandre, stagiaire au LIMSI, qui a implémenté nos idées sur l'architecture de la simulation et l'utilisation de TAG, et a proposé à son tour des algorithmes concrets. C'est pourquoi nous avons co-signé la publication de ce travail (Allexandre et Popescu-Belis 1998), que nous-même avons réalisée postérieurement.

en général, se comportent vis-à-vis des signaux fournis par les organes récepteurs comme des systèmes sélectionnistes. Les *groupes de neurones* en constituent les unités de sélection, puisqu'ils sont en pour reconnaître une certaine forme du signal. Grâce à la sélectivité des groupes et aux connexions qu'ils établissent, un ensemble de groupes appelé *carte* peut *cartographier* une couche d'entrée (sensorielle) ou de sortie (motrice).

Bien que la théorie ait pour but une explication globale du cerveau, les modèles informatiques proposés par Edelman et ses collaborateurs (cf. II.C.2) ont des ambitions plus limitées, plus réalisables. Rappelons que la série des *Darwin I–IV* illustre la catégorisation perceptive à travers deux voies sensorielles, ainsi qu'un type de lien sensori-moteur. L'idée fondamentale est de transformer le patron d'activation d'un répertoire d'entrée, à un instant donné, en des patrons de plus en plus abstraits, dans des cartes de niveau plus élevés – puis de relier entre elles deux telles cartes, de façon réentrante. La configuration résultante est ensuite utilisée comme une description abstraite de l'entrée, et peut déclencher une séquence motrice.

IV.A.2 Propositions d'amélioration des modèles

Les modèles informatiques précédents apportent une justification pratique au « darwinisme neuronal ». Néanmoins, ces systèmes nous paraissent peu susceptibles d'être étendus, à cause de leur complexité et de leur étroite intégration (par exemple *Darwin III* possède environ 50000 « unités » regroupées en 43 répertoires). Nous essayons ici de tirer un meilleur parti de certains concepts de base, afin de nous donner les moyens de satisfaire nos ambitions.

IV.A.2.i Modélisation des groupes de neurones à l'aide d'*acteurs*

En premier lieu, nous proposons de remplacer les « unités intégratives » (assez proches des neurones formels) utilisées dans l'équipe d'Edelman par des *acteurs*. Ces acteurs sont de petits modules informatiques capables d'être *déclenchés*, c'est-à-dire de traiter les *messages structurés* qu'ils ont reçus et d'en envoyer d'autres à leurs *accointances* ou acteurs connus[3]. Ces messages remplacent donc les signaux numériques échangés, et les accointances jouent le rôle de connexions.

Cette technique nous paraît plus conforme à l'idée de *groupe neuronal*, qui est une unité capable de traitements plus évolués que ceux d'un seul neurone, comme par exemple la reconnaissance d'une configuration complexe (dans une carte de niveau inférieur), et l'émission retardée ou non d'un signal lui aussi plus élaboré, en rapport avec la configuration reconnue. Biologiquement, un tel groupe se compose de nombreux neurones, capables d'un traitement complexe. Les communications complexes peuvent avoir lieu grâce à des faisceaux de plusieurs axones, ou à des séquences de potentiels sur

[3] Le terme d'acteur a déjà été employé en ce sens en informatique, ayant conduit à la notion de « programmation par acteurs » (Briot 1989, Ferber 1995 :153-55). Nous retiendrons en particulier le concept de message asynchrone, mais pas la possibilité de créer dynamiquement de nouveaux acteurs.

un même axone. L'existence et le rôle possible de ces groupes est analysée aussi dans (Burnod 1988, Edelman et Mountcastle 1978)[4].

L'épigenèse des groupes neuronaux est modélisée remarquablement dans (Reeke, Finkel, Sporns et Edelman 1990:616-26), et ceux-ci ont un rôle important dans une première version de la TSGN (Edelman 1981). Néanmoins, ils semblent moins utilisés en tant que structures spécifiques dans *Darwin II* ou *III*, qui leur préfèrent des « unités intégratives » plus proche en vérité des neurones individuels.

IV.A.2.ii Simplification des entrées/sorties

Il peut paraître curieux de prétendre qu'une simplification de la capacité à traiter des stimuli visuels (réel ou simulé) soit une « amélioration ». En réalité, il s'agit plutôt de signaler que cette capacité n'est probablement pas essentielle pour faire progresser les modèles vers des fonctions plus élaborées. Le système visuel, pour être relativement bien connu chez le singe, n'en est pas moins d'une grande complexité, à la mesure de l'espace tridimensionnel qu'il appréhende. En implémenter un modèle réaliste signifierait que l'on maîtrise le domaine de la *reconnaissance des formes*, ce qui montre que la tâche est démesurée, lorsque l'on connaît les efforts déployés par les informaticiens dans ce domaine.

Nous pensons qu'une modélisation fondée sur le « darwinisme neuronal » de certaines fonctions élaborées du cerveau est possible sans reproduire en détail le système visuel, en utilisant des *entrées perceptives simplifiées*, puisqu'aucun des principes de la théorie (sélection, catégorisation, réentrance) ne présuppose une forme déterminée pour les entrées.

IV.A.2.iii Représentation explicite des valeurs

Enfin, dans l'ouvrage d'Edelman (1989), les valeurs (ou besoins) du système jouent un rôle décisif, dans la catégorisation, la mémoire et l'action. Or, dans *Darwin III*, les valeurs sont codées implicitement, par un réflexe : le bras repousse une classe de formes définies par avance (Reeke, Finkel, Sporns et Edelman 1990:692).

Les études neurobiologiques du fonctionnement des besoins ne sont pas encore suffisamment avancées pour permettre de trancher entre une représentation des besoins par des unités ou groupes neuronaux *générateurs de signaux* vers d'autres régions sensori-motrices, et une représentation par groupes *modulateurs de connexions sensori-motrices*. Nous voulons explorer la première possibilité, et *représenter explicitement des valeurs* ou besoins à l'aide de groupes neuronaux spécifiques, et d'étudier à un niveau élémentaire leur rôle dans le *déclenchement* des actions, compte tenu de l'état perceptif courant.

Les trois propositions formulées nous éloignent quelque peu du terrain neurobiologique, mais elles fournissent, comme nous allons le montrer, des avantages

[4] Il s'agit de deux études sur les *colonnes corticales*, que nous avons déjà commentées au II.A.2.i.a. Nos acteurs sont capables de micro-traitements spécifiques, à l'image de ces colonnes.

considérables sur le plan informatique, essentiel à toute modélisation. La présente étude vise à préciser ce qui doit être conservé de la réalité à modéliser (qui est ici une partie du système nerveux), et ce qui peut et doit être adapté.

IV.A.2.iv Comparaison avec d'autres modélisations

Nous avons déjà passé longuement en revue différentes réalisations dans le domaine des agents isolés ou communicants (II.C et II.D). Rappelons ici les aptitudes comportementales perfectionnées des robots de Luc Steels (II.D.2.iv), capables d'évoluer parmi des obstacles dans un monde plan réel, et d'engager des dialogues. Le système de contrôle de ces robots établit simplement des liaisons fonctionnelles entre paramètres d'entrée et de sortie, les faisant varier selon un critère de survie ; le mécanisme est donc relativement élémentaire.

Les agents ou robots évoqués, pour frappants qu'ils soient, semblent peu extensibles et peu flexibles, à cause justement de leur simplicité structurale. La pertinence cognitive de leur structure paraît très réduite, celle-ci étant trop simple pour modéliser n'importe quel phénomène cognitif. Au contraire, les modèles d'Edelman s'inscrivent dans une théorie globale du système nerveux, et fournissent des principes sur lesquels construire des modèles plus cognitifs.

Par rapport au connexionnisme classique, les acteurs que nous proposons permettent des traitements plus structurés, et échangent des messages plus élaborés ; la réalisation d'un tel système est donc plus aisée. Quant aux différences de principe entre le connexionnisme et la TSGN, elles sont exposées dans (Reeke, Sporns et Edelman 1989) : les systèmes sélectionnistes utilisant la TSGN ne sont pas « instructivistes », i.e. ne présupposent pas une forme de résultats déterminée qu'il s'agit d'atteindre après apprentissage, contrairement aux réseaux connexionnistes. Notre agent n'est pas non plus guidé par des configurations prédéfinies « correctes » de ses répertoires d'acteurs, mais par les niveaux de ses ressources, qu'il s'agit de maintenir élevés.

IV.A.3 Architecture de l'agent

IV.A.3.i Vue d'ensemble

L'agent proposé évolue dans un environnement simulé, avec lequel il interagit à travers des organes simulés. On simule aussi des besoins, satisfaits par des ressources extérieures. L'agent consiste en une structure de contrôle et des organes relationnels ; il est baptisé Ève (« être vivant élémentaire »).

Une vue d'ensemble de la simulation réalisée (Figure IV-1) montre que la structure de contrôle est l'élément le plus complexe de cette simulation. Cette structure regroupe des *acteurs* (en grisé sur la figure), que l'on peut voir aussi comme des micromodules chargés d'opérations relativement simples, liées à une fonction particulière. Ces *acteurs* sont groupés en répertoires selon leurs fonctions. On distingue trois grandes familles : la perception (système visuel), l'intéroception (besoins N,B) et la

motricité. À chacun de ces aspects est associé un « organe », qui est relié à un répertoire d'acteurs.

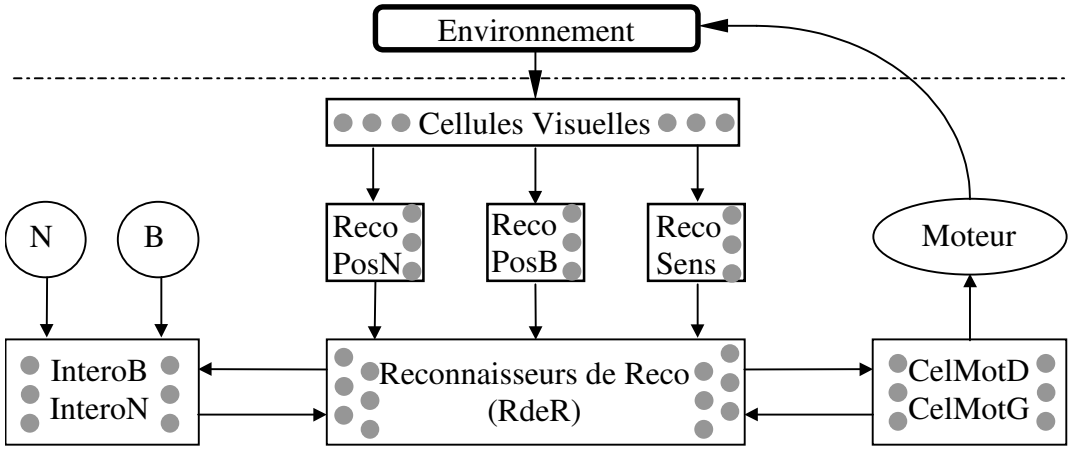

Figure IV-1. Structure de contrôle de l'agent (et lien avec l'environnement).

L'action d'un acteur consiste à faire varier son état d'activité courant en fonction des messages reçus, et à envoyer certains messages à certains acteurs connus, en fonction de son état. Cet état possède en général une valeur binaire (actif / inactif) ; l'existence de plusieurs acteurs dans les répertoires *intéroceptif* et *moteur* permet de représenter l'intensité du stimulus reçu ou émis.

Pour le moment, les acteurs sont déclenchés séquentiellement, par famille, et les messages transitent par une boîte aux lettres centralisée, suivant le modèle décrit par (Ferber 1995:223-35). L'unité temporelle de base est le *cycle d'exécution*, au cours duquel chaque acteur exécute une fois ces opérations : lecture des messages reçus, traitement, envoi d'autres messages.

L'agent évolue dans des environnements unidimensionnels : un segment, une droite infinie ou un cercle (Figure IV-2). Un environnement contient deux ressources N et B (nourriture/boisson) satisfaisant les besoins de l'agent, situées en des points différents ; leur séparation empêche l'existence d'une zone de satisfaction permanente et impose donc le mouvement. Ève possède deux réservoirs N et B, qui se vident linéairement au cours du temps ; seul un passage sur la zone respective (N ou B) permet de remplir le réservoir, instantanément et en totalité.

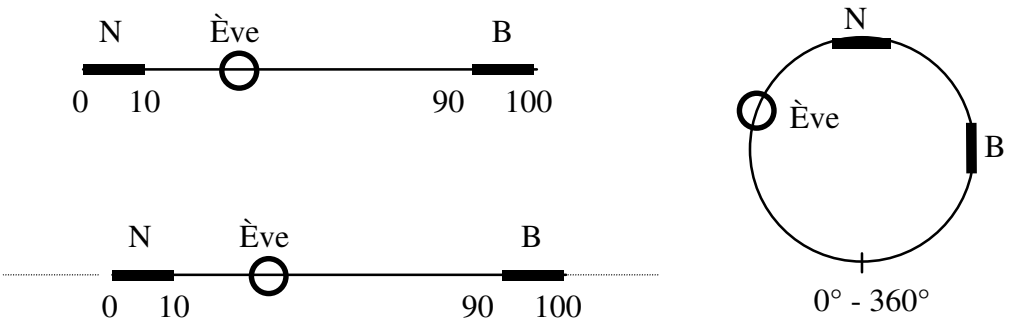

Figure IV-2. Les différents environnements.

IV.A.3.ii Perception visuelle : catégorisation

Cette fonction s'inspire des modèles *Darwin I* et *II*. Il s'agit de l'aspect le plus figé de la structure de contrôle, car nous avons réalisé directement un « répertoire secondaire », sans le faire émerger (justification au **II.A.2.ii**). C'est ce qui est fait aussi dans *Darwin II* (Reeke et Edelman 1987), après que *Darwin I* (Edelman 1981) eut montré le fonctionnement du mécanisme de sélection. Comme nous l'avons dit, l'émergence de ce répertoire n'est pas un problème central dans cette expérience.

Les cellules visuelles constituent à la fois l'organe visuel et la couche d'entrée (comme pour la rétine biologique), et forment une représentation topographique linéaire de l'environnement autour de la position d'Ève. Le champ total, couvert par environ 50 cellules, est suffisamment large pour que sur le segment, N et B soient en permanence visibles, et puissent activer les cellules visuelles ou acteurs correspondants. Plusieurs familles d'acteurs reçoivent des messages de ces acteurs d'entrée, et en fonction de l'envoyeur et de la nature du message, ces acteurs du second ordre sont capables de « catégoriser » la configuration d'entrée (cf. Figure IV-1). Pour l'instant cela se limite à l'émission des messages :

- 'N est à droite', 'N est à gauche' – par les acteurs *RecoPosN* aux acteurs *RdeR* ;
- 'B est à droite', 'B est à gauche' – par les acteurs *RecoPosB* aux acteurs *RdeR* ;
- 'je vais à droite', 'je vais à gauche' – par les acteurs *RecoSens* aux acteurs *RdeR*.

La couche ou le répertoire d'acteurs *RdeR* opère une étape de plus dans la catégorisation – *RdeR* signifie « reconnaisseurs de reconnaisseurs » (Edelman 1981, Edelman et Mountcastle 1978). Ce répertoire contient des acteurs réceptifs à toutes les combinaisons des messages précédents, ce qui permet de catégoriser toutes les configurations spatiales accessibles à la couche d'entrée ; à ce stade il y a 16 acteurs *RdeR*[5]. La fréquence de ces configurations n'est pas prise en compte, dans un premier temps ; une idée plus réaliste consisterait à ne créer d'acteurs *RdeR* que pour les configurations/combinaisons suffisamment fréquentes, et faire évoluer ce répertoire.

[5] On peut voir ces acteurs comme une matérialisation de connexions réentrantes qui s'établissent entre *RecoPosN* et *RecoSens*, respectivement *RecoPosB* et *RecoSens*.

IV.A.3.iii Les protocoles réentrants

IV.A.3.iii.a Principe

L'adaptation de l'agent, ou son apprentissage, résulte de l'établissement de liens entre acteurs. Les protocoles de communication correspondants doivent permettre à des acteurs actifs en même temps de faire connaissance, l'un enregistrant l'autre parmi ses accointances. L'apprentissage a lieu en permanence : il n'y pas de phase d'acquisition de connaissances, puis de phase d'application. Il n'y a pas non plus de superviseur : le seul critère est que les réservoirs N et B de l'agent soient maintenus à un niveau acceptable, à savoir supérieur à une limite donnée (non nulle, car un réservoir nul signifie la disparition de l'agent).

L'enregistrement d'un acteur dans une liste d'accointances n'est pas définitif, mais sujet à un lent oubli. Un « degré d'accointance » est donc géré pour chaque couple d'acteurs. L'enregistrement répété d'un acteur parmi les accointances d'un autre renforce fortement sa présence, et son score augmente. Un équilibre doit donc être trouvé entre l'oubli et le rappel. Ici, l'*oubli* est une fonction linéaire du temps, alors que chaque *réactivation* multiplie par un coefficient fixé la force de l'accointance, qui doit rester bornée[6].

Les acteurs *RdeR* constituent l'articulation centrale du contrôle, comme on le voit sur la Figure IV-1. Leur communication avec les acteurs *Intero* d'une part et *CelMot* de l'autre se déroule selon deux protocoles semblables, mais non identiques, puisque le premier doit être conditionné par les valeurs (*InteroB*, *InteroN*), mais pas le second. Ces protocoles matérialisent notre interprétation de la notion de réentrance (Edelman 1989), utilisée ici entre des répertoires non homologues. Lorsqu'un acteur A_1 contacte un acteur A_2 sous certaines conditions, un lien se forme et A_2 pourra plus tard contacter A_1 sous certaines autres conditions.

IV.A.3.iii.b Les communications entre RdeR et Intero

Ce protocole peut se décrire ainsi (cf. Figure IV-3) : un acteur *RdeR* actif, i.e. ayant décelé la configuration spatiale qui lui correspond, signale son activité à tous les acteurs *Intero* ; chacun mémorise l'envoyeur un court laps de temps dans un tampon (*ActeursConnusPotentiels*). Lorsque un acteur *Intero* actif vient de se désactiver, ce qui signale une satisfaction du besoin, il ajoute les acteurs *RdeR* contenus dans le tampon *A.C.Potentiels* à sa liste d'accointances (*ActeursConnus*), ou les renforce s'ils existent déjà. En résumé : les acteurs *InteroN* et *InteroB* retiennent les configurations spatiales qui ont précédé à court terme leur satisfaction.

[6] On peut penser à différentes techniques d'optimisation pour régler ces valeurs, ainsi que les autres paramètres numériques d'Ève, une dizaine environ : par exemple un algorithme génétique.

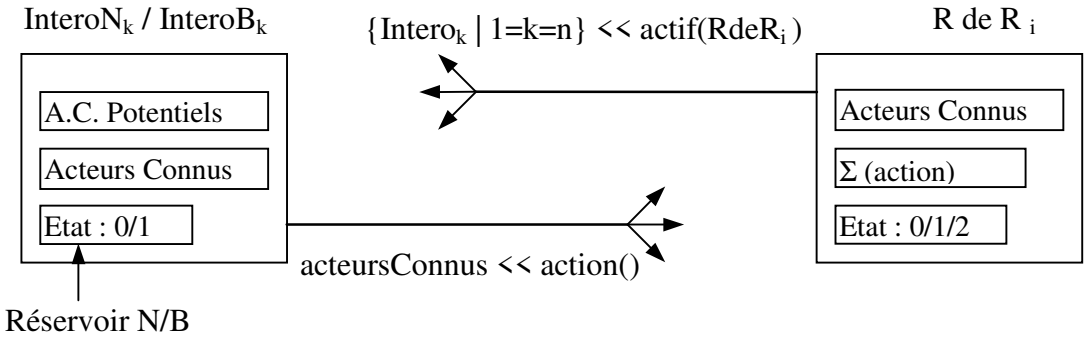

Figure IV-3. Protocole de communication RdeR / Intero.

Inversement, lorsqu'un acteur de type *Intero* est actif, i.e. signale un besoin, il envoie un message '*action()*', ou quantum d'action, aux *RdeR* de sa liste d'accointances (*ActeursConnus*). Chaque acteur *RdeR* transmet alors un message vers les *CelMot*, comme nous allons le voir.

IV.A.3.iii.c Les communications entre RdeR et CelMot

Ce deuxième protocole est semblable au premier. Les cellules motrices ou acteurs *CelMot* sont reliées à un organe moteur qui effectue le déplacement, en sommant les ordres reçus. Chaque *CelMot* peut s'activer sur ordre d'un acteur *RdeR*, ou spontanément avec une faible probabilité.

Lorsqu'une *CelMot* est active, elle le signale aux acteurs *RdeR*, et seuls les *RdeR* actifs vont l'enregistrer parmi leurs accointances (*ActeursConnus*). L'opération est ici immédiate (il n'y a pas de tampon *A.C.Potentiels*) puisqu'il s'agit de noter quelles cellules motrices correspondent à une configuration spatiale, « topo-cinématique », donnée. Il s'agit donc d'un apprentissage du contrôle moteur, parle moyen des effets que les effecteurs induisent sur la perception.

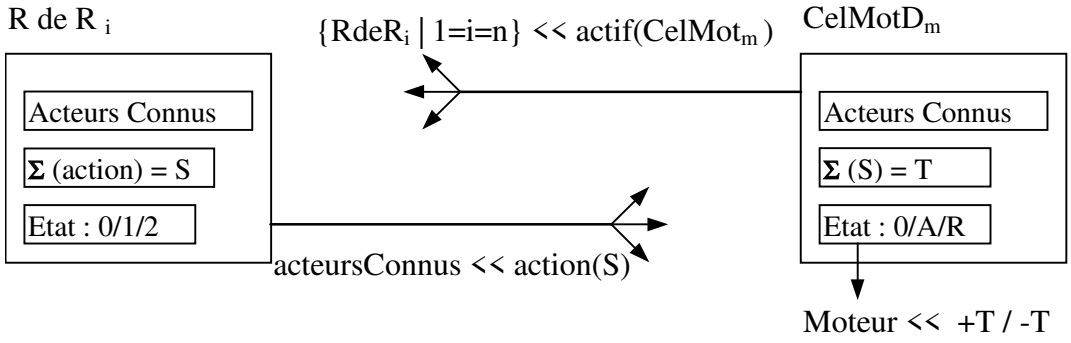

Figure IV-4. Protocole de communication RdeR / CelMot.

En même temps, lorsqu'un acteur *RdeR* activé convenablement par les *RecoPos*, reçoit des messages '*action()*' des cellules *Intero*, il calcule leur somme *S* et transmet aux acteurs *CelMot* connus un message de la forme '*action(S)*'.

Les deux sens du protocole se déroulent en parallèle : nous les avons séparés pour les besoins de l'explication. Ainsi, *l'apprentissage n'est pas séparé de l'exécution* : l'acquisition d'une connaissance (l'établissement de nouveaux liens entre acteurs) est immédiatement suivie de son application (utilisation de ces liens). Le même mécanisme est à l'œuvre pendant toute la « vie » de l'agent, ce qui lui confère, comme pour les systèmes biologiques, des possibilités continues d'apprentissage. Une des qualités de notre modèle est son bon fonctionnement (IV.A.4) malgré sa relative complexité. Les paramètres numériques sont donnés avec leur interface , Figure IV-5.

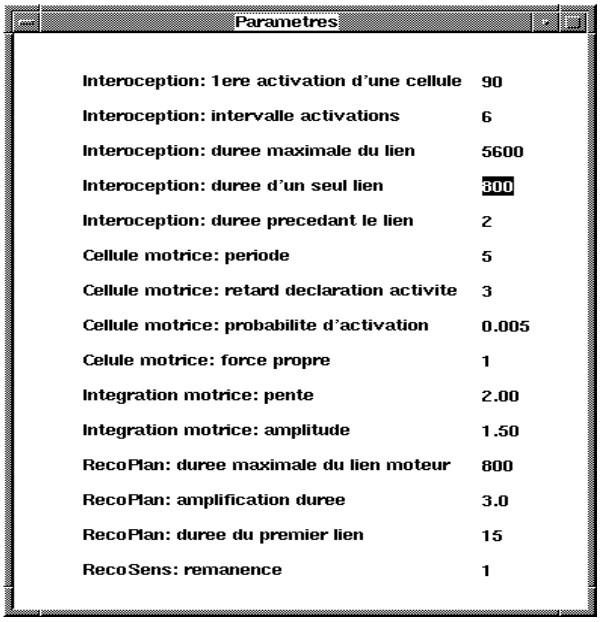

Figure IV-5. Interface de manipulation des paramètres.

IV.A.3.iv Description de haut niveau

IV.A.3.iv.a Représentation du « but » de l'agent

On peut légitimement se demander si la structure précédente, ou du moins ses fonctionnalités, ne peuvent être représentées plus simplement sous la forme d'un système à base de règles. Sans contester en soi la possibilité d'une telle représentation, nous pensons qu'elle est contraire à notre démarche, qui est de mettre en évidence certains principes de la TSGN avec la structure préconisée par son créateur, moyennant des adaptations améliorant sa lisibilité, et non le contraire. Néanmoins, la description du comportement d'Ève à l'aide de règles semble tentante, permettant de mieux comprendre son fonctionnement global. Elle nous permet ici une remarque sur l'« intentionnalité » de l'agent.

D'un point de vue phénoménologique, l'apprentissage consiste à acquérir une règle de la forme "*si <faim> alors <aller sur N>*" à partir d'une expérience où l'on a constaté "*si <être sur N> alors <non faim>*" ; *<faim>* signifie ici un bas niveau dans le réservoir N. Il semble à première vue qu'il suffise d'« inverser » la situation

d'apprentissage (2e règle) pour obtenir la situation d'exécution (1ère règle), et cela semble même s'apparenter au double sens du protocole réentrant.

Cette analogie est en réalité très limitée, car elle présuppose, à la regarder de plus près, une représentation de *<être sur N>* comme but à atteindre, puisque l'inversion plus rigoureuse de l'apprentissage devrait en effet se faire sous la forme :

- *si <faim> alors <but = non faim>* ;
- *si <but = non faim> alors <but = être sur N>* ;
- *si <but = être sur N> alors <aller vers N>*.

Or, dans l'état actuel de l'architecture que nous avons conçue en suivant la TSGN, il n'y a pas de classe d'acteurs capables de représenter un but, i.e. un état futur différent de l'état présent, vers lequel tend l'agent.

IV.A.3.iv.b Modélisation proposée pour l'intentionnalité

Notre agent montre en revanche l'implémentation d'une intentionnalité élémentaire même en l'absence de représentations. En effet, la source première des messages d'activation est à chercher non dans un planificateur pourvu de buts, mais dans les acteurs intéroceptifs (*InteroB*, *InteroN*). Les messages (ou signaux) produits par les acteurs *Intero* sont modulés et redirigés par les acteurs *RdeR* en fonction de l'état perceptif courant, et parviennent aux acteurs *CelMot*, et modifient la position d'Ève dans son environnement.

Le « but » est donc modélisé ici par *la conjonction des moyens permettant d'y parvenir*, tels qu'ils sont mémorisés lors de l'apprentissage. L'agent ne vise donc pas des buts, mais seulement l'application de moyens adaptés jusqu'à ce que cessent les signaux (les messages) qui sont l'origine de cette application. En forçant un peu le formalisme des règles, Ève observe et apprend :

- *<aller vers N> précède (<faim>→<non faim>)*

et exécute :

- *<faim> déclenche <aller vers N>*.

Dans ces conditions, il faut reconnaître que notre structure élémentaire de contrôle se prête mal à l'exécution d'actions complexes où l'ordre d'application des moyens est essentiel ; tout au plus pourrait-elle y parvenir après un certain nombre d'essais aléatoires.

IV.A.4 Résultats obtenus

Le système décrit a été implémenté en Smalltalk, langage orienté objet, sur une station de travail Sun. Toutes les figures qui suivent sont des copies d'écran durant des exécutions réelles. L'interface permettant de manipuler l'agent est montrée ci-dessous (Figure IV-6).

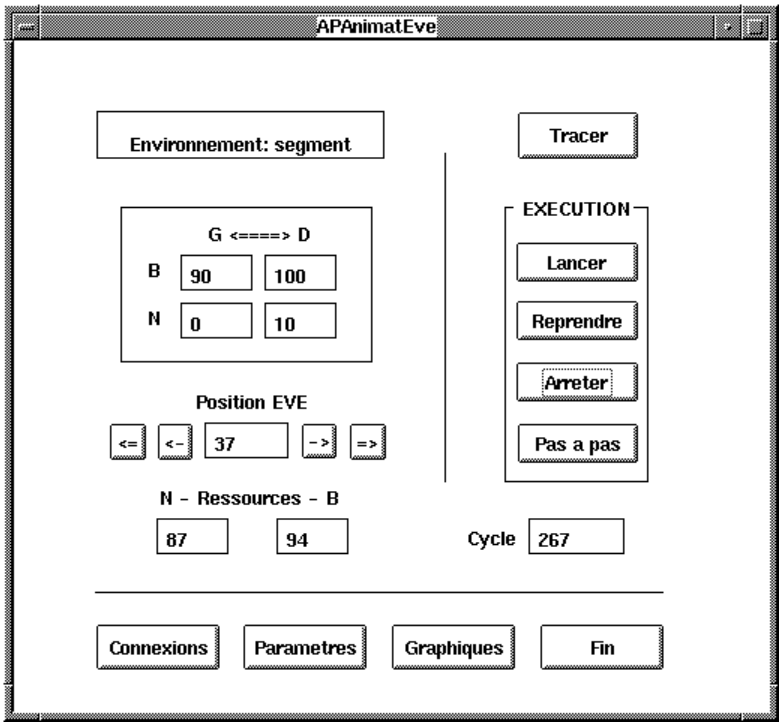

Figure IV-6. Interface d'observation et de manipulation d'une instance.

IV.A.4.i Évolution des connexions entre acteurs

Parmi les aspect distingués dans la structure de contrôle (perception, intéroception, motricité), seules les connexions constituant la perception sont figées ; les deux autres évoluent suivant les protocoles décrits plus haut.

L'acquisition du contrôle moteur (protocole *RdeR / CelMot*, IV.A.3.iii.c) fait l'objet d'un apprentissage autonome : l'activation aléatoire de cellules motrices provoque un déplacement, perçu et catégorisé par la partie perceptive. Comme expliqué, les accointances varient en fonction des activations simultanées des acteurs *RdeR* et *CelMot* : partant de zéro, les liens s'enrichissent, fluctuent et se stabilisent, si l'environnement ne change pas. La figure ci-dessous montre les accointances après 366 cycles d'exécution. On observe bien sûr qu'aucun acteur *RdeR* n'est connecté en même temps à des acteurs *CelMotD* et *CelMotG*.

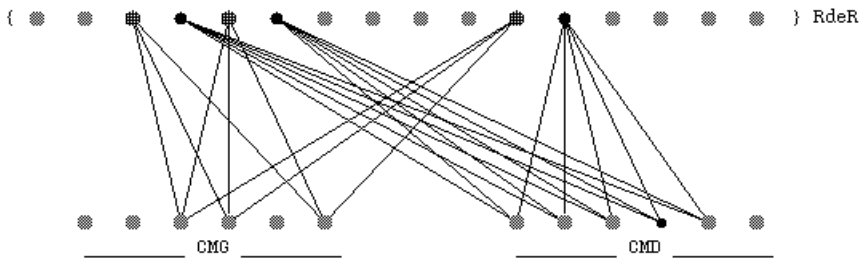

Figure IV-7. Accointances entre RdeR *et* CelMot *(366 cycles).*

La découverte des zones N et B se fait à l'aide d'une intervention extérieure, qui déplace Ève sur N, puis sur B ; l'activité aléatoire des *CelMot* est insuffisante. Ainsi, Ève peut observer que la satisfaction du besoin N, i.e. le remplissage du réservoir N et la désactivation des acteurs *InteroN*, est précédée d'une certaine configuration du champ perceptif – par exemple '*N est à gauche*' et '*Ève va à gauche*'. Cette observation permet l'établissement de nouvelles accointances (visibles sur la Figure IV-8), qui sont immédiatement utilisées.

Nous avons figuré aussi sur les deux schémas l'état d'activation des acteurs (noir par rapport à gris). À l'activation aléatoire des acteurs *CelMot* (Figure IV-7), s'oppose l'activation commandée par le besoin B de *tous* les acteurs *CelMot* connus (Figure IV-8). On constate aussi (sur cette même Figure IV-8) que plusieurs *RdeR* peuvent correspondre à la même configuration perceptive, puisqu'elle peut être décrite (incomplètement) par des combinaisons d'attributs différents.

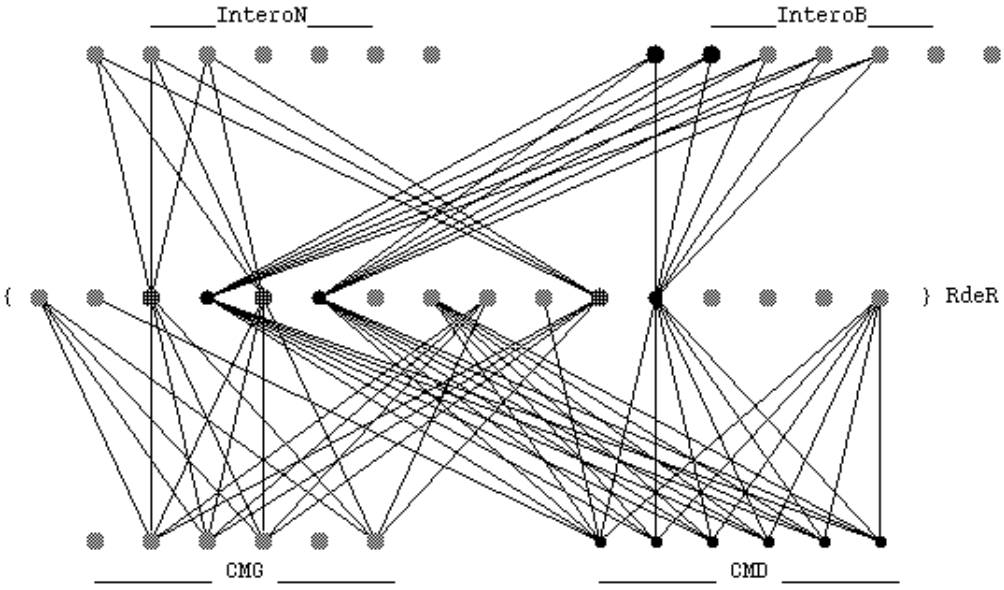

Figure IV-8. Évolution des accointances pour les deux protocoles.

IV.A.4.ii Comportement extérieur

Ève est placée dans l'environnement *segment*, au centre de celui-ci. Sur la Figure IV-9, on peut observer son comportement durant les deux phases décrites plus haut :

- évolution aléatoire autour de la position initiale, pendant que les ressources décroissent – jusqu'à 366 cycles, sur la Figure IV-9 au niveau des traits noirs supérieurs signalant une intervention extérieure ;
- interventions extérieures plaçant Ève successivement sur la zone B (extrémité 100) et la zone N (extrémité 0), avec remplissage des réservoirs. Ces premiers contacts s'avèrent suffisants pour l'apprentissage ;
- mise en œuvre immédiate des accointances établies, ce qui entraîne des allers et retours entre N et B. L'apprentissage étant continu, les succès ultérieurs viennent renforcer ces premières accointances (Figure IV-9b).

La Figure IV-9b représente un extrait plus long de l'exécution, sans intervention extérieure autre que celle initiale. Ève continue indéfiniment ses allers et retours entre N et B, de façon non déterministe à cause de l'interaction des nombreux acteurs, et aussi à cause de la baisse de ses ressources à des vitesses différentes. Les accointances varient selon le mécanisme de rappel/oubli décrit (IV.A.3.iii), et les cellules motrices continuent à présenter une faible activité aléatoire. La Figure IV-9b représente plus de 10000 cycles, et nous avons arrêté cette exécution à 55000 cycles.

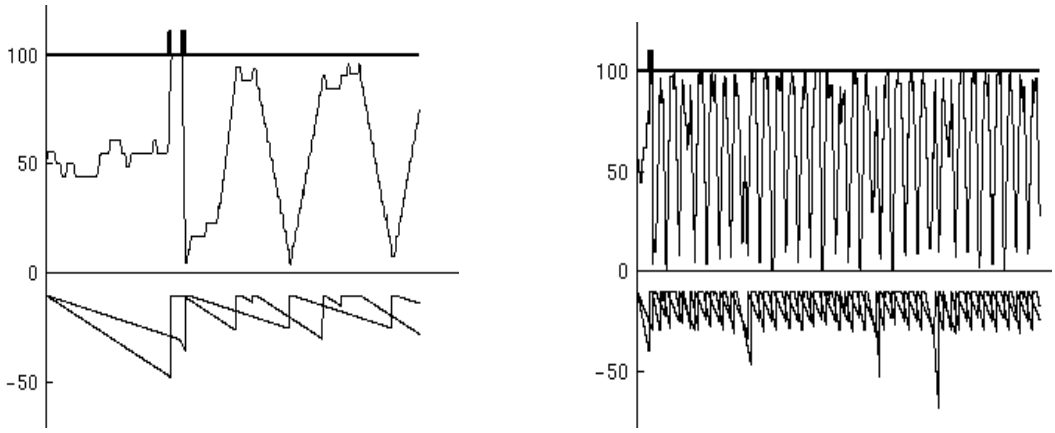

Figure IV-9 : (a) 1118 cycles (b) 10125 cycles. Position (courbe supérieure) et ressources (courbe inférieure) de l'agent au cours du temps ; interventions extérieures (traits verticaux tout en haut).

IV.A.4.iii Études statistiques

Une interface de notre système (Figure IV-10) permet de manipuler aisément et notamment de sauvegarder différentes instances de l'agent, et de les placer dans différents environnements.

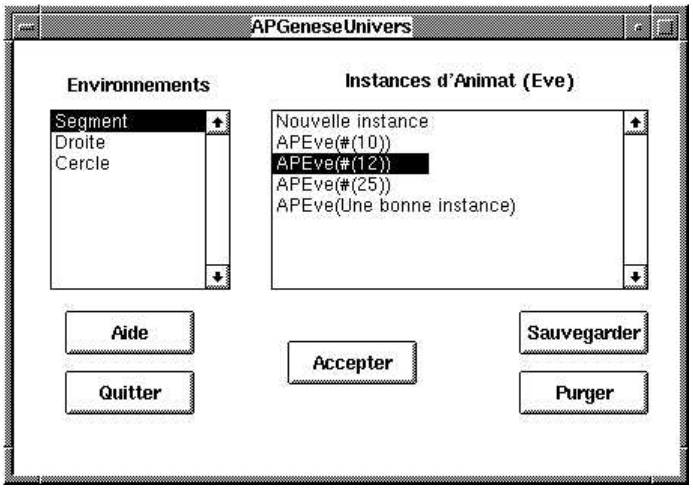

Figure IV-10. Interface de manipulation des instances et des environnements.

Nous évaluons d'abord le rôle des mouvements aléatoires dans la survie de l'agent. Au début, l'une des deux zones N ou B peut en effet être découverte par exploration aléatoire, mais pour moins de 20% des instances d'Ève testées. L'amplitude de ces mouvements aléatoires est donc faible. De surcroît, lorsqu'une des zones N ou B est découverte, celle-ci devient « trop attractive » pour que l'*autre* zone puisse être découverte aussi par hasard : aucune instance observée n'y parvient. La durée de vie observée sans apprentissage est par conséquent toujours inférieure à 1000 cycles.

Afin d'étudier la durée de survie de l'agent, i.e. la stabilité de la structure de contrôle, nous avons adopté le procédé suivant : les intervention extérieures sont permises seulement jusqu'à 1000 cycles. On ne retient que les instances qui sont « vivantes » (réservoirs non vides) à 4000 cycles, ce qui permet d'éliminer les biais dus à l'apprentissage des zones N et B. On note enfin la durée de vie observée.

Dans une série d'expériences, sur 12 entités instruites, 11 étaient vivantes à 4000 cycles : l'apprentissage est donc fiable. La durée moyenne de vie a été de 1 550 000 cycles ($1{,}55 \cdot 10^6$). En particulier, 5 instances ont dépassé 10^6 cycles, et ont été arrêtées de l'extérieur – l'une après $6 \cdot 10^6$ cycles. Étant donné l'impossibilité d'une preuve mathématique, ces résultats apportent une confirmation très encourageante de la stabilité du système.

IV.A.4.iv Mise en évidence de deux difficultés

IV.A.4.iv.a Saturation des accointances en l'absence d'inhibition

Même si l'activité aléatoire des acteurs *CelMot* est faible, il arrive que des *CelMot* opposés soient actifs en même temps – par exemple deux *CelMotD* et un *CelMotG*. Dans ce cas, il y a un déplacement résultant non nul, perçu par le système visuel et catégorisé par les acteurs *RdeR*, et il y a aussi enregistrement des trois cellules motrices parmi les accointances des *RdeR* ainsi activés. Le problème, c'est que deux seulement des *CelMot* étaient des « causes » du mouvement, alors que les trois sont perçus comme nécessaires. Lorsque les *RdeR* reçoivent et transmettent des activations

depuis *Intero*, les accointances *CelMot* erronées ne sont pas oubliées avec le temps, puisqu'elles reçoivent (à tort) des activations. Il y a alors une saturation progressive des listes d'accointances pour chaque *RdeR*, ce qui conduit à un ralentissement assez visible de la commande motrice : le système activera deux *CelMotD* et un *CelMotG* pour aller à droite.

La solution que nous adoptons consiste à inhiber les acteurs *CelMot* actifs dont le sens est contraire au mouvement effectif, ce qui empêche leur enregistrement comme accointances. Cela revient à introduire des capacités proprioceptives, i.e. donner la possibilité à l'organe lui-même d'être sensible au sens dans lequel il est utilisé. Cette solution dépasse les deux principes de la TSGN exploités (catégorisation et réentrance), montrant ainsi la nécessité de spécifications supplémentaires. L'inhibition est aussi présente, sous différentes formes, dans *Darwin III*.

IV.A.4.iv.b La temporalité

L'appréhension de la temporalité à travers les deux seuls principes de la TSGN utilisés semble difficile, comme le montre aussi Edelman (1989:112-118). Un élément de solution serait l'introduction de nouveaux répertoires permettant d'introduire des retards, et de comparer ainsi les signaux présents et les signaux passés. Cela répondrait au problème des enchaînements d'actions permettant de satisfaire un but ; mais nous reviendrons au Chapitre III sur la gestion de séquences d'actions.

IV.A.4.v Des capacités de généralisation limitées

IV.A.4.v.a Description

Une propriété importante de *Darwin II* sont ses facultés de généralisation dans la reconnaissance des formes (Reeke et Edelman 1987), qui restent toutefois assez limitées. , Notre agent possède aussi ces propriétés, bien qu'il soit pourvu d'un système de catégorisation plus rudimentaire ; l'agent présente l'avantage d'une meilleure mise en évidence, par le comportement moteur.

La translation des zones N et B ne pose pas de problème, pourvu que N reste à gauche de B (Figure IV-2). La translation modifie l'état perceptif de l'agent, et provoque un changement adéquat de l'activation motrice. En revanche, lorsque N et B sont permutés, les connexions apprises ne sont plus pertinentes, et un nouvel apprentissage est nécessaire.

Lorsque les extrémités du segment sont « ouvertes », l'agent ne quitte pas spontanément l'intérieur du segment N–B. Si une intervention externe déplace à l'extérieur une instance instruite sur l'intérieur du segment, la décision correcte qui entraîne le retour sur l'intérieur n'est pas assurée. En effet, la configuration étant nouvelle, la structure de contrôle utilise les acteurs *RdeR* associés aux descripteurs spatiaux encore valides, qui ne sont pas toujours pertinents – cela dépend du réservoir qu'il est le plus urgent de remplir. Une instruction convenable, i.e. à l'intérieur *et* à l'extérieur du segment, permet bien sûr à Ève de revenir toujours à l'intérieur du segment.

IV.A.4.v.b Exemple expérimental

Après apprentissage sur le segment fermé, l'agent a été déplacé manuellement à gauche de N (à l'extérieur donc du segment), et laissé évoluer. On constate, sur la Figure IV-11, son retour vers N, indiqué (1). Ensuite, une tentative de maintenir l'agent à droite de B échoue (2), ainsi que trois autres tentatives ultérieures (en B, N et N) : on constate à chaque fois son retour plus ou moins rapide (en fonction des besoins) vers les ressources. L'évolution libre est semblable aux précédentes : l'agent n'a aucune « raison » de quitter l'intérieur du segment N–B.

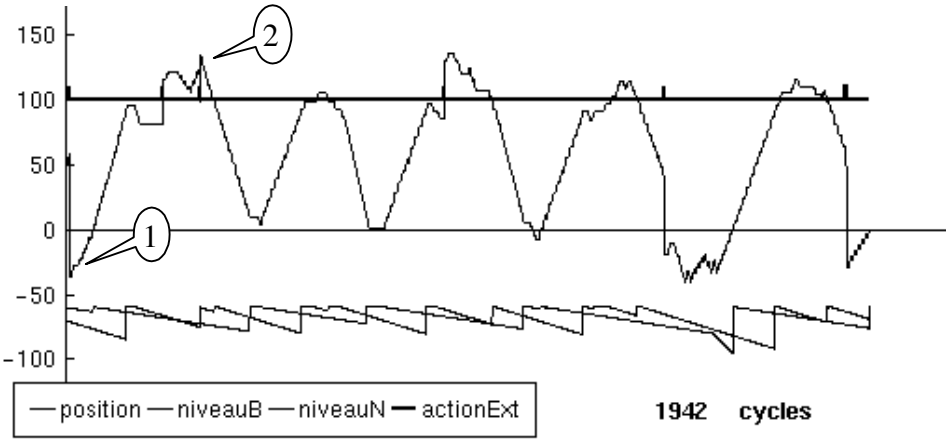

Figure IV-11. Maintien de l'agent à l'intérieur du segment, après la phase d'apprentissage (non représentée). Les interventions sont indiquées par des traits sur la ligne supérieure.

Ces résultats s'expliquent par la structure interne : les configurations spatiales sont décrites par des conjonctions de descripteurs, i.e. des messages reçus par les acteurs *RdeR* en provenance des *RecoPosN*, *RecoPosB* et *RecoSens*. L'agent pourra catégoriser une configuration nouvelle en utilisant les conjonctions de descripteurs déjà reliés aux valeurs. Par exemple, l'agent aura appris à l'intérieur du segment à associer un *InteroN* avec le *RdeR* correspondant à <'*N à gauche*', '*je vais à gauche*'>, et avec des *CelMotG*. À l'extérieur du segment, à droite de B, cette association continue à donner de bons résultats en cas de « faim » (besoin N), car elle dirige l'agent vers la gauche ; en revanche à gauche de N l'association est inopérante, le *RdeR* n'étant plus actif. C'est à une autre association d'intervenir.

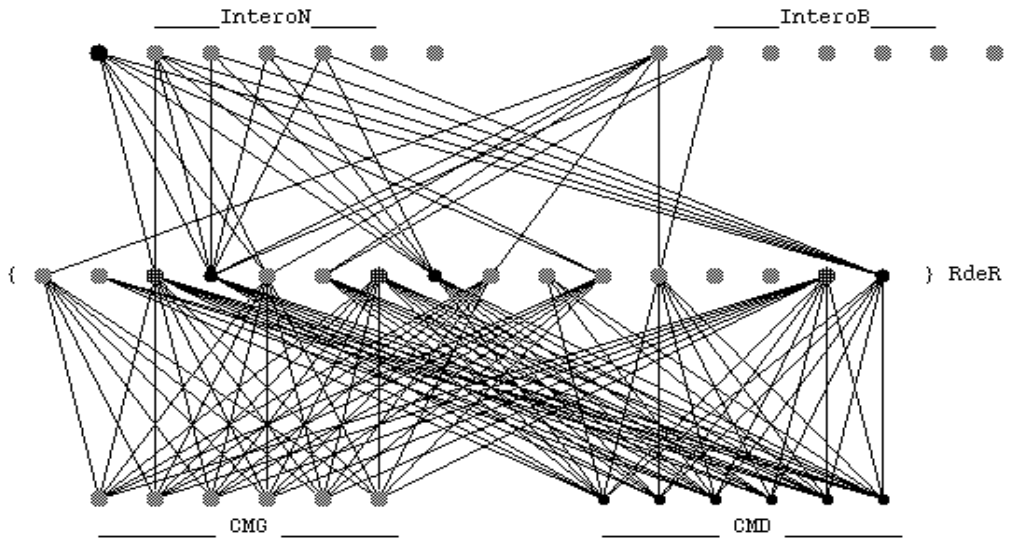

Figure IV-12. Accointances acquises sur la droite (comparer à la Figure IV-8).

L'apprentissage continu offre l'avantage de n'utiliser ces généralisations limitées que lorsque la situation est vraiment nouvelle ; ensuite, une fois celle-ci apprise et les accointances modifiées, on revient à un fonctionnement plus fiable. Il est en ce sens parlant de comparer la Figure IV-12 avec la Figure IV-8, pour constater que le nombre d'acteurs *RdeR* possédant des accointances a augmenté, car les configurations spatiales possibles sont plus variées sur la droite que sur le segment[7].

IV.A.5 Conclusion

La structure de contrôle de cet agent visait à étudier l'application de deux principes fondateurs de la TSGN, la catégorisation et une forme de réentrance, à travers une modélisation originale à base d'acteurs et de messages. Le fonctionnement satisfaisant de l'agent, malgré la simplicité du modèle, paraît donc encourageant.

La modélisation des « groupes neuronaux » de la TSGN à l'aide d'acteurs est aisée, puisqu'on peut leur faire accomplir des fonctions plus élaborées que celles des « unités intégratives », sans toutefois manipuler des équations numériques complexes, comme dans *Darwin II* et surtout *Darwin III*. Les acteurs permettent un expression relativement aisée et compréhensible des protocoles réentrants.

Cette modélisation est donc une étape, certes préliminaire, vers la construction de structures complexes et extensibles mettant en œuvre les visées d'explication cognitive de la TSGN. Nous nous tournons maintenant vers l'exploration d'un aspect plus visiblement lié à la modélisation des facultés langagières, mais qui utilise des « agents » fondés sur des techniques informatiques « symboliques » plus classiques.

[7] Contrairement à la Figure IV-8, dans la situation de la Figure IV-12, la « faim » (besoin N) n'entraîne plus un déplacement vers la gauche, mais vers la droite, c'est-à-dire dans la direction adaptée à la position actuelle d'Ève.

IV.B Étude de l'émergence de conventions syntaxiques dans une population d'agents

IV.B.1 Introduction

La seconde expérience présentée ici étudie l'établissement de conventions syntaxiques parmi des agents engagés dans des dialogues portant sur des situations observées en commun. Les agents sont supposés pourvus des *facultés de catégorisation* nécessaires à la « compréhension » des situations, ainsi que des *principes grammaticaux*. Font en revanche l'objet de conventions les *paramètres* de cette grammaire, notamment l'ordre des mots, et le lexique utilisé.

Le principal intérêt de cette expérience est la mise en évidence d'un code de communication entre agents fondé sur une grammaire TAG (Grammaire d'Arbres Adjoints) simplifiée – à savoir un formalisme syntaxique reconnu. La lexicalisation de cette grammaire permet une bonne intégration avec le lexique, nécessaire puisque les conventions touchant ce dernier conditionnent les conventions syntaxiques. L'étude expérimentale des protocoles collectifs d'apprentissage, est aussi un intérêt de cette simulation. Comme dans la réalisation précédente, il n'y a pas pour les agents de séparation entre la phase d'apprentissage et celle d'utilisation des connaissances.

Par rapport aux modélisation existantes (II.D.2), la présente étude cherche à la fois à explorer un formalisme grammatical reconnu, et à mieux étudier son utilisation en rapport avec un environnement donné. Ainsi, notre code de communication apparaît comme plus proche d'une langue que les simulations de MacLennan (Crumpton 1994, MacLennan 1992), Dyer (Werner et Dyer 1992) ou Batali (1998). Ces études utilisent des codes de communication plus simples et plus formels, et se concentrent sur l'étude de leur dynamique collective ; la notion de syntaxe en est absente ou discutable. Hashimoto et Ikegami se sont penchés plus particulièrement sur l'organisation des « mots » dans un message, mais la sémantique est absente de leur code, qui n'est lié à aucun environnement. Tout autre est le cas des expériences de Luc Steels, que nous avons décrites en détail (II.D.2.iv) : l'utilisation de robots réels permet un ancrage assez réaliste du code de communication. En revanche, les recherches sur la syntaxe semblent se limiter à un classement de mots en catégories « syntaxiques » émergentes, qui apparemment tiennent plus de la sémantique, et à une formalisation très particulière de la correspondance syntaxe/sémantique.

IV.B.2 Principes de la simulation

Les agents s'engagent, par deux, dans des dialogues ; l'agent émetteur utilise sa grammaire TAG pour produire un message linéaire à partir de la structure conceptuelle représentant la situation observée dans l'environnement. La complexité des situations est peu à peu augmentée, ce qui permet aux agents d'établir des conventions d'abord lexicales, puis syntaxiques.

IV.B.2.i Les agents

L'unique tâche des agents est de participer à des dialogues, où ils sont alternativement émetteurs ou récepteurs d'un message. Nous supposons les agents déjà capables de catégoriser leur environnement : ceux-ci manipulent directement une représentation conceptuelle des situations de l'environnement. Ce mécanisme de catégorisation est celui qui a été expérimenté précédemment (IV.A). Toutefois, afin de ne pas compliquer cette deuxième expérience, nous n'avons pas réalisé l'articulation avec la première, nous fournissons directement aux agents une représentation conceptuelle des situations. L'application du mécanisme précédent demanderait aussi une analyse de la catégorisation des *relations* entre objets, alors que ceux-ci et leurs propriétés semblent poser moins de problèmes.

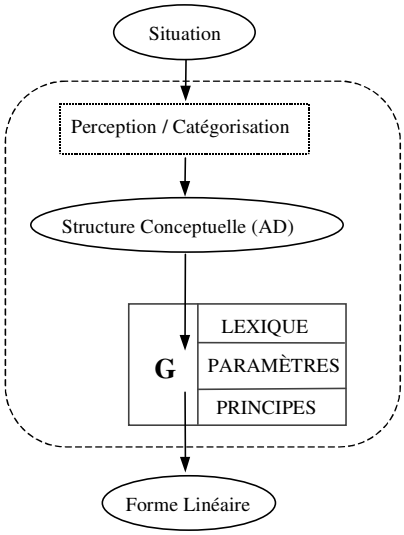

Figure IV-13. Configuration d'émission d'un agent.

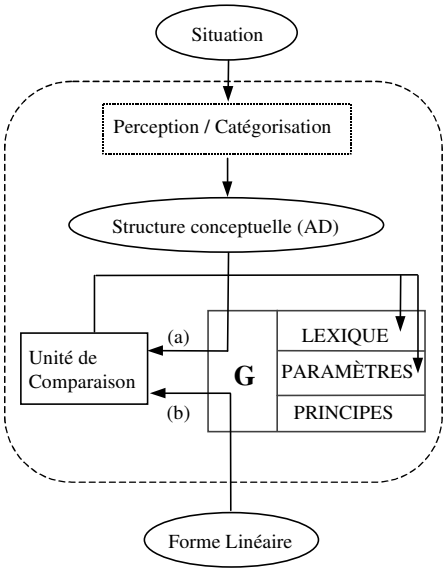

Figure IV-14. Configuration de réception d'un agent.

Lorsqu'il est l'émetteur, un agent utilise ses connaissances grammaticales (IV.B.3) pour produire à partir de la structure ou représentation conceptuelle une chaîne de mots qui décrit la situation, totalement ou en partie. L'agent récepteur décode cette chaîne à l'aide de ses propres connaissances grammaticales, et la compare avec la structure conceptuelle de la situation (IV.B.4.i). Si la chaîne est interprétable par le récepteur (celui-ci ayant le droit de « deviner » quelques éléments), celui-ci remet à jour ses connaissances (IV.B.4.iii et IV.B.4.iv). La structure d'un agent et les données sont représentées sur la Figure IV-13 en configuration d'émission, et sur la Figure IV-14 en réception[1].

IV.B.2.ii L'environnement

Les situations sont construites directement, par le programme de contrôle, en termes conceptuels, et données aux agents. Leur représentation utilise les arbres de dérivation du formalisme TAG (cf. en particulier le IV.B.3.iv ci-dessous). Les entités du monde sont regroupés en classes, auxquelles correspondent des arbres élémentaires TAG ; parmi les entités, nous classons aussi les relations et caractéristiques. Les entités autorisées sont[2] :

- objets: *cube*, *sphere*, *cylinder* ;
- caractéristiques des objets: *red*, *big*, *heavy* ;
- relations de position (type *r1*) entre objets : *on*, *in_front_of*, *left_of* ;
- relations de comparaison (type *r2*) : *greater_than*, *taller_than*, *heavier_than*.

Ce nombre réduit convient à une première approche, mais devrait être étendu pour de futures expériences ; on devrait aussi varier plus l'*arité* (le nombre d'arguments) de chaque relation. Toutefois, la complexité des situations est déjà illimitée, puisque les relations sont des constructeurs de nouveau objets, qui peuvent être sujets de nouvelles relations, et ainsi de suite (générativité). Par exemple, un cube rouge peut être à côté d'un cube bleu, et un lourd cylindre posé sur *eux*. Les restrictions apportées à ce genre de combinaisons sont données dans la Table IV-1 (page 108).

IV.B.3 Compétence linguistique : grammaire TAG

Nous avons déjà rencontré la grammaire TAG, ou grammaire d'arbres adjoints dans la présentation de Aravind K. Joshi à IJCAI-97 (cf. I.B.3 et (Joshi 1998)). Il

[1] Remarquons que nous utilisons le terme « agent » et non « animat », puisque ces entités n'ont ni effecteurs, ni besoins simulés. De plus, les « représentations » qu'ils manipulent s'approchent plutôt des représentations cognitives, ce qui influence notre choix du terme (Ferber 1995:32-34).

[2] Nous allons donner des mots anglais aux concepts (objets, caractéristiques et relations) intervenant dans la simulation. Outre que cela permet au lecteur français de les situer immédiatement comme éléments de simulation, nous avons fait ce choix dans le programme pour une meilleure diffusion dans la communauté de recherche majoritairement anglophone de ce domaine.

s'agissait d'un exemple de méthode formelle pour le TAL, à la fois puissante d'un point de vue computationnel, et pertinente du point de vue de la linguistique descriptive. Nous avons décidé de l'utiliser pour la représentation des connaissances grammaticales de nos agents, même si pour l'heure nous ne pouvons pas en proposer une implémentation pertinente d'un point de vue neurobiologique.

IV.B.3.i Une version simplifiée du formalisme TAG

La grammaire TAG, proposée par l'équipe d'Aravind K. Joshi a été choisie ici pour plusieurs raisons[3]. D'abord, il s'agit d'une grammaire lexicalisée – du moins dans la version utilisée ici – c'est-à-dire que chaque règle syntaxique est portée par un item lexical, ce qui facilite l'amorçage de la syntaxe à partir du lexique. Ces règles sont représentées par des *arbres élémentaires*, dont l'item lexical occupe une feuille. Les autres sont libres pour les combinaisons d'arbres, qui permettent de construire l'*arbre syntaxique* d'une phrase, ou *arbre dérivé*.

Autre intérêt de TAG, il a été montré (Shieber et Schabes 1990) que dans le formalisme TAG on peut établir une relation simple entre la représentation syntaxique d'une phrase et sa représentation conceptuelle (la représentation de son sens). La combinaison des arbres élémentaires (un pour chaque item lexical) conduit à l'arbre syntaxique (AS), alors que l'arbre qui représente comment ces arbres élémentaires ont été combinés fournit une représentation conceptuelle, semblable à un graphe de dépendances. Ce deuxième arbre est appelé *arbre dérivé* (AD)[4]. Lorsque l'on parcourt de droite à gauche les feuilles de l'arbre syntaxique, on aboutit bien entendu à la *forme linéaire* de la phrase (FL).

Certains travaux sur des analyseurs TAG pour le français et l'anglais (Abeillé et Schabes 1989) ont montré que les arbres de dérivation n'avaient pas toute la puissance nécessaire pour représenter des relations conceptuelles plus complexes, comme par exemple la portée exacte des modifieurs ou quantifieurs. Toutefois, pour l'utilisation très simple que nous en ferons ici, nous emploierons ces arbres de dérivation pour représenter conceptuellement des situations de l'environnement.

IV.B.3.ii Arbres élémentaires

Le formalisme TAG introduit deux règles de combinaison des arbres, la substitution et l'adjonction. Corrélativement, il y a deux sortes d'arbres élémentaires : les *arbres initiaux*, munis de nœuds à substitution, et les *arbres auxiliaires*, munis de nœuds d'adjonction. Ces arbres, correspondant en général aux modifieurs, pourront être adjoints

[3] Pour un aperçu en français de ce formalisme nous renvoyons au chapitre correspondant dans l'ouvrage d'Anne Abeillé (Abeillé 1993), qui fournit aussi de nombreuses références. Un des travaux précurseurs est (Joshi, Levy et Takahashi 1975), et parmi les contributions plus récentes on peut citer (Abeillé, Bishop, Cote, Joshi et Schabes 1989, Abeillé et Schabes 1989, Schabes, Abeillé et Joshi 1988, Shieber et Schabes 1990).

[4] Nous préférons l'appellation arbre syntaxique (AS) à la place de arbre de dérivation (« AD » ?), pour mieux le distinguer de l'arbre dérivé (AD) dans l'abréviation.

à des arbres existant déjà. Des exemples d'arbres élémentaires pour chaque classe d'entités sont donnés sur la Figure IV-15 – *o*, *r1* et *r2* sont des arbres initiaux, alors que *c* est un arbre auxiliaire. Chaque arbre élémentaire possède une et une seule feuille lexicale, figurée avec les conventions d'un agent quelconque sur la Figure IV-15.

Dans le formalisme TAG d'origine, les nœuds et feuilles des arbres (sauf les items lexicaux) sont munis d'étiquettes qui restreignent la catégorie grammaticale des arbres pouvant être substitués ou adjoints aux nœuds respectifs. Ici, pour ne pas compliquer le processus d'acquisition étudié, nous avons supposé que tous les arbres appartiennent à la catégorie *syntaxique* « objet » : un objet (*o*) muni d'une caractéristique (*c*) est un objet, et deux objets en relation (*r1* ou *r2*) forment un autre objet (plus complexe). Les étiquettes *C*, *R1* et *R2* sur la Figure IV-15 servent seulement à caractériser les items lexicaux.

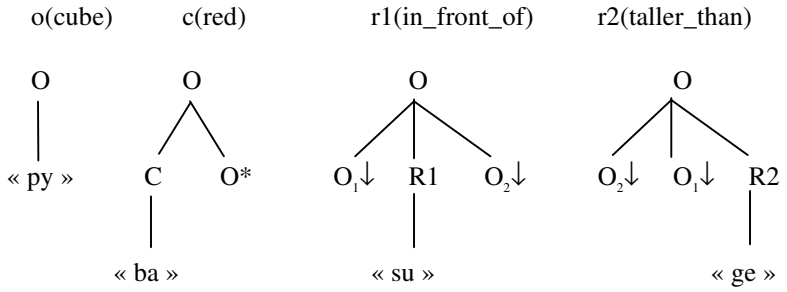

Figure IV-15. Arbres élémentaires avec leurs noms (l'ordre des branches et les items lexicaux sont des exemples).

L'item lexical est appelé « nœud pied » ; les autres feuilles forment les « nœuds feuille » de deux types : nœuds à substitution (marqués par ↓) et nœuds à adjonction (marqués par *). Schématiquement, la substitution remplit un argument obligatoire, ajoutant un arbre à un nœud ↓, alors que l'adjonction exprime une propriété facultative et s'insère sur n'importe quel nœud de même étiquette que le nœud *. Un exemple d'arbre de dérivation, d'arbre syntaxique (ou dérivé) et de forme linéaire pour une même situation est donné Figure IV-16.

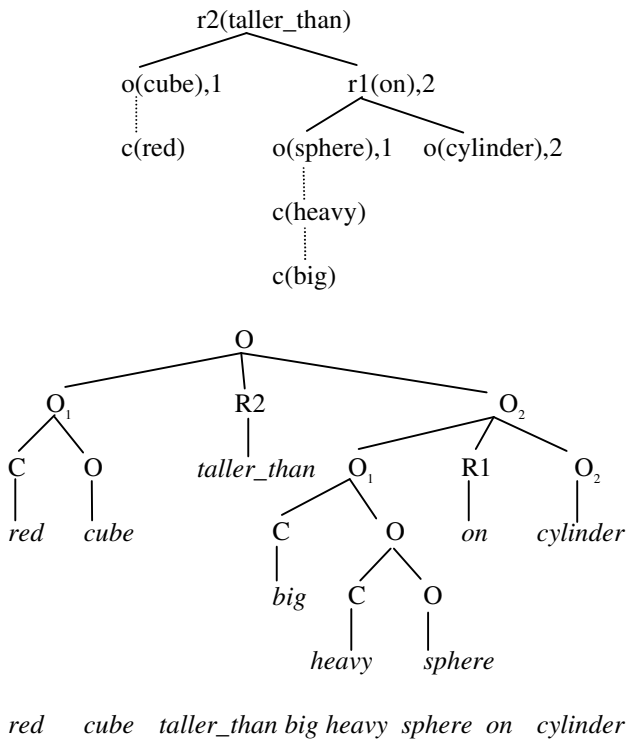

Figure IV-16. Arbre de dérivation (AD), arbre syntaxique (AS) et forme linéaire (FL) pour une situation donnée (les items lexicaux son « traduits »).

IV.B.3.iii Variation pour chaque agent : principes et paramètres

IV.B.3.iii.a Les principes

Les agents connaissent dès le début de notre expérience le nombre et type de branches pour les arbres de chaque classe d'entités ; seul l'ordre de ces branches est indéterminé, ainsi que les items lexicaux. Pour situer ces contraintes par rapport aux considérations sur la partie innée du langage (cf. les discussions du II.A.1.iii et II.B.3.ii), on peut dire que la structure même des arbres pourrait être d'une certaine façon inhérente à la connectivité de certaines aires cérébrales dédiées au langage. D'autre part, le nombre et le type des branches, qui sont analogues à une structure d'arguments (en particulier pour les verbes et les adjectifs), peuvent provenir des contraintes cognitives perçues dans l'environnement : par exemple, le concept de « plus haut » comprend forcément deux objets, celui de « rouge » caractérise un objet, etc.

IV.B.3.iii.b Les paramètres grammaticaux

L'ordre des branches est particulier à chaque agent. Il y a ainsi, pour les arbres des *caractéristiques c* deux possibilités, à savoir O* à gauche ou à droite, ce qui correspond dans la forme linéaire à un adjectif antéposé, respectivement postposé. Il y a six possibilités pour les arbres associés à *r1* (deux arguments plus l'item lexical), et six pour *r2*, cela correspondant bien sûr aux permutations du triplet <*nomDeRelation*,

premierArgument, *secondArgument*>. Cette variation simule donc la variation de l'ordre sujet/verbe/objet entre les langues.

IV.B.3.iii.c Le lexique

Les items lexicaux ou « mots » correspondant aux feuilles lexicales de chaque arbre élémentaire sont aussi particuliers à chaque agent. Initialement, les agents ne connaissent aucun mot, et ils sont autorisés à en inventer, utilisant des couples <*consonne*, *voyelle*> aléatoires. Les agents sont aussi autorisés à « deviner » un certain nombre de mots dans un message reçu, afin que des conventions puissent se répandre.

Un agent peut gérer plusieurs arbres élémentaires pour chaque classe d'entités, ainsi que plusieurs items lexicaux pour chaque entité. L'agent peut donc comprendre plusieurs « dialectes » ou plutôt « idiolectes ». Toutefois, chaque agent aura des arbres et items lexicaux préférés (IV.B.4.iii), qu'il utilisera pour la génération de ses messages ; ce système conduit peu à peu à l'émergence d'un dialecte dominant dans la population (IV.B.5).

IV.B.3.iv Les arbres de dérivation

La représentation conceptuelle des situations est fournie, nous l'avons dit, par les arbres de dérivation (AD), puisque ceux-ci peuvent exprimer, en TAG, les dépendances sémantiques. Chaque arbre élémentaire est représenté dans l'arbre de dérivation seulement par son nom, par exemple *o*(*cube*), *c*(*red*), *r1*(*taller_than*). Par convention, la substitution d'un arbre élémentaire *a1* sur *a2* est représentée par un trait en pointillé, et l'adjonction par un trait plein – cf. (Figure IV-16) et (Abeillé 1993:210 sq.).

Le module de contrôle de la simulation génère directement des arbres de dérivation afin de représenter des situations choisies au hasard dans l'environnement. Toutefois, étant donné les étiquettes syntaxiques associées aux feuilles des arbres élémentaires (essentiellement des « O »), les combinaisons autorisées par les principes de TAG ne correspondent pas toujours à des situations conceptuelles acceptables pour les situations. Par exemple, il serait grammaticalement possible d'attribuer une couleur à l'objet composé « cube sur sphère », mais cela n'est pas acceptable ici.

Afin de restreindre ces combinaisons, on pourrait particulariser le jeu d'étiquettes, ou leur ajouter des traits. Nous allons plutôt contraindre directement la construction de l'arbre de dérivation, en imposant des contraintes au module de génération. Ainsi, nous distinguons pour chaque nœud de l'arbre de dérivation ses arguments substitués et ses arguments adjoints, et posons des contraintes pour chacun d'entre eux (Table IV-1).

Nom de la famille d'arbres	Arguments substitués	Arguments adjoints
objet (o)	Ø	c
caractéristique (c)	Ø	c
relation_1 (r1)	o | r1 | r2 , o | r1| r2	Ø
relation_2 (r2)	o | r1 | r2 , o | r1| r2	Ø

Table IV-1. Restrictions sur la génération automatique d'arbres de dérivation : les arguments substitués sont obligatoires, les arguments adjoints sont optionnels.

IV.B.4 L'interaction entre agents

IV.B.4.i Principe des dialogues

Le seul but de nos agents est de participer à des dialogues, en essayant tour à tour de produire ou de comprendre des messages décrivant la situation qui leur est présentée. Les participants à chaque dialogue sont choisis au hasard. Le principe de ces dialogues est analogue à celui que propose Luc Steels notamment (II.D.2.iv), qui s'est inspiré de la notion de « jeu de langage » décrite par Wittgenstein.

Le module de contrôle compose une situation, i.e. un arbre de dérivation, et la fournit aux agents émetteur et récepteur choisis. L'émetteur engendre alors un arbre syntaxique et une forme linéaire en utilisant ses paramètres et lexique préférés – ou des nouveaux s'il n'en a pas – et envoie la forme linéaire au récepteur. Dans certains cas, l'émetteur peut n'encoder dans le message qu'une partie de la situation. Cette situation simule le dialogue humain à propos d'un élément du contexte visible à chacun des interlocuteurs. Sa mise en relief par rapport au reste du contexte peut être « exacte » (pointage précis) ou « approximative » (pointage flou) – ce dernier cas étant simulé par l'encodage, par l'agent récepteur, d'une partie seulement de la situation partagée.

Le récepteur reçoit le message, et doit utiliser sa connaissance de la situation partagée, ainsi que sa propre grammaire, pour décoder le message, i.e. l'apparier avec la situation perçue. En toute rigueur, on peut imaginer deux méthodes opposées pour réaliser cette opération :

d) conversion de l'arbre de dérivation de la situation en une ou plusieurs formes linéaires et comparaison avec la forme linéaire reçue (chemin (a) sur la Figure IV-14), ou bien,

e) analyse syntaxique de la forme linéaire reçue, puis analyse « sémantique » et construction d'un ou plusieurs arbres dérivés, à comparer avec celui de la situation (chemin (b) sur la Figure IV-14).

Le second procédé s'avère long et inefficace, puisqu'il suppose une « inversion » des connaissances grammaticales et l'application d'un algorithme d'analyse. D'autre part, ni

le premier ni le second ne permettent à l'agent de modifier dynamiquement ses connaissances lexicales, par exemple pour « deviner » un nouveau mot. Un algorithme hybride, intermédiaire entre (a) et (b), est utilisé pour permettre cet apprentissage ; l'algorithme effectue directement la comparaison sur la forme linéaire, en s'arrêtant au premier mot non interprétable.

IV.B.4.ii Évolution des connaissances de chaque agent

Les connaissances linguistiques de chaque agent dépendent des dialogues auxquels il a pris part. Pour chaque entité, les mots connus sont rangés selon une échelle de préférences, établie à l'aide des scores des mots. Le score d'un mot attaché à un arbre élémentaire est le nombre de fois où il a été utilisé pour le concept respectif dans les n derniers dialogues. Un agent peut aussi deviner un ou plusieurs (g) mots à chaque dialogue. Les paramètres n et g sont réglables.

Chaque agent possède aussi un ensemble d'arbres élémentaires pour chaque classe de concepts (*o*, *c*, *r1*, *r2*), qui diffèrent par l'ordre de leurs branches. Ces ensembles sont aussi classés par ordre de préférence, les scores des arbres qui permettent de comprendre et d'exprimer plus de phrases étant augmentés. Ces ensembles sont renouvelés de temps en temps par de nouvelles permutations des branches.

IV.B.4.iii Succès d'un dialogue

La compréhension d'un message signifie son appariement avec tout ou partie de la situation partagée. Pour ce faire, le récepteur étudie toutes les interprétations possibles, à l'aide de toutes ses connaissances grammaticales et lexicales. Pour chaque interprétation, une seule association <*entité*, *mot*> ou <*classe d'entités, arbre*> peut être utilisée. Le récepteur considère d'abord toutes les interprétations qui ont utilisé les arbres élémentaires et items lexicaux *préférés* (sur l'échelle décrite ci-dessus). Il y a trois possibilités :

- une seule interprétation a été construite en utilisant les arbres et les mots préférés. Le dialogue est appelé un *succès total* ;
- plusieurs interprétations sont possibles avec les arbres et les mots préférés, ou bien aucune (dans ce cas toutes les interprétations possibles utilisent au moins un arbre ou un mot non préféré). Le dialogue est appelé un *succès partiel* ;
- aucune interprétation n'a pu être construite : le récepteur ne possédait pas les arbres nécessaires, ou bien avait trop de mots à deviner. Le dialogue est appelé un *échec*.

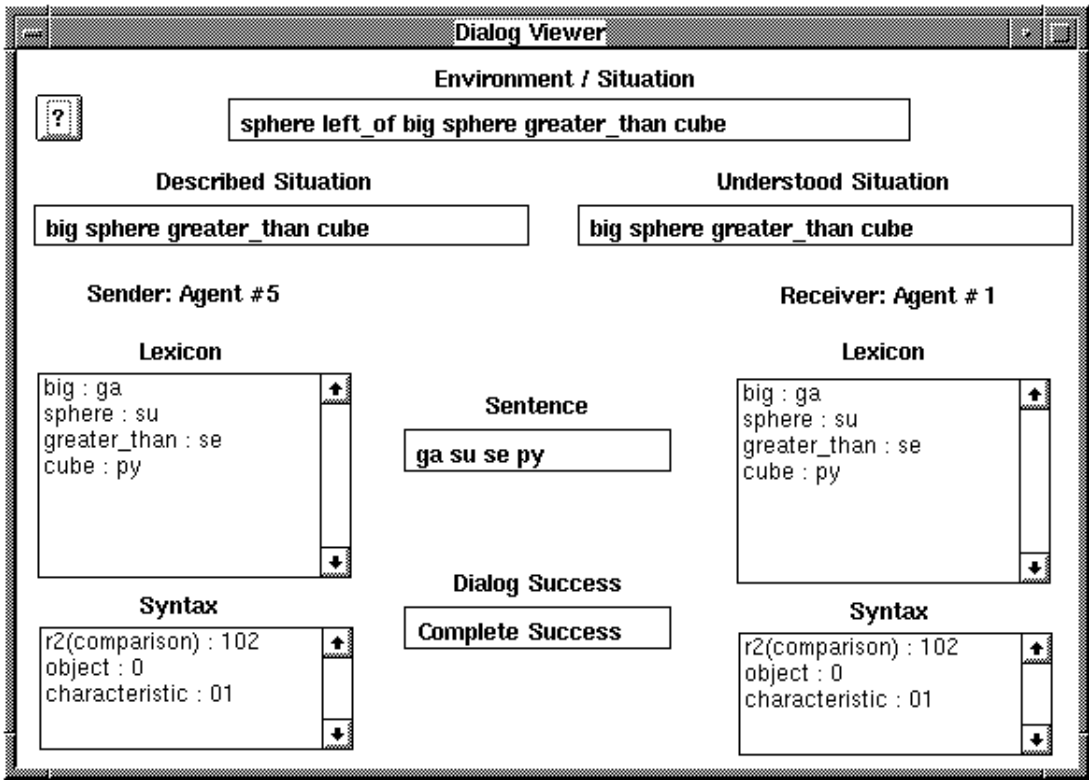

Figure IV-17. Exemple de dialogue réussi (succès total).

Un exemple de dialogue est donné sur la Figure IV-17, dans l'interface correspondante[5]. La situation est représentée en haut de la fenêtre ; il s'agit de '*left_of*(*sphere*, *greater_than*(*sphere*(*big*), *cube*)'. L'agent n°5 choisit d'en décrire une partie, et produit le message '*ga su se py*' en utilisant ses mots et ses arbres préférés, affichés à gauche. L'agent n°1 a réussi à décoder le message à l'aide de ses propres mots et arbres préférés, et à l'apparier à un fragment unique de la situation partagée. C'est pourquoi le dialogue est un *succès total*.

IV.B.4.iv Mise à jour de la grammaire et du lexique

Après chaque dialogue (émission plus réception), les deux protagonistes modifient leurs scores respectifs pour les mots et les arbres, en fonction du succès du dialogue.

- *Succès total* : l'émetteur et le récepteur augmentent le score des mots et des arbres utilisés pour produire le message, respectivement le comprendre. Un point est affecté à chaque couple <entité, mot> et <classe d'entités, arbre> utilisés ; les scores des autres couples demeurent inchangés.

[5] Le système est programmé en Smalltalk. Pour les raisons citées (note 2), cette description est donnée en anglais et avec les conventions grammaticales de cette langue (adjectif antéposé, préposition au milieu). La convention d'affichage des arbres est simple : '0' désigne la branche de l'item lexical, puis '1' et éventuellement '2' désignent les autres arguments, dans l'ordre.

- *Succès partiel* : l'émetteur diminue les scores des mots et arbres utilisés, puisque ceux-ci ne suivent probablement la « tendance dominante » du dialecte, ou bien sont ambigus. En revanche, le récepteur augmente les scores de tous les mots et de tous les arbres lui ayant permis de comprendre le message, essayant ainsi de se rapprocher des connaissances de l'émetteur.
- *Échec* : l'émetteur diminue les scores des mots et arbres utilisés, le récepteur ne change aucun score.

Par ailleurs, les mots qui ont un score nul depuis suffisamment longtemps sont supprimés, ce qui permet d'éliminer les mots utilisés au début mais non renforcés collectivement.

IV.B.5 Résultats

L'étude expérimentale a permis d'étudier les propriétés attendues du modèle. Dans la mesure où les conventions linguistiques évoluent dynamiquement, et que les dialogues ont lieu au hasard, il n'existe pas de méthode pour prédire de façon déterministe les résultats de la simulation. Seule l'étude expérimentale fournit des résultats. Les différences entre expérimentations sont fréquentes, c'est pourquoi nous les avons répétées plusieurs fois, et effectué des moyennes.

IV.B.5.i Apprentissage incrémental

Le procédé expérimental typique pour voir s'établir des conventions linguistiques consiste à proposer aux agents des situations dont la complexité augmente graduellement. Par exemple, pour un groupe de 5 agents, nous leur présentons d'abord seulement des objets : 500 dialogues sont en moyenne nécessaires pour voir s'établir des noms uniques pour les objets[6]. Des entités de type « caractéristique » sont ensuite ajoutées aux objets, et environ 1200 dialogues sont nécessaires pour que des mots et arbres uniques soient établis. Puis, environ 9000 dialogues sont nécessaires pour que des conventions sur *r1* s'établissent, et ensuite à peu près le même nombre pour des conventions sur *r2*[7]. La complexité d'une situation, i.e. le nombre d'entités qui la constituent, est elle aussi progressivement augmentée lorsqu'on introduit *r1* et *r2*.

La durée de l'apprentissage incrémental augmente avec la taille de la population. Avec 10 agents, environ 1000 dialogues sont nécessaires pour fixer les conventions pour

[6] Ce chiffre est déjà assez élevé, et montre que pour l'apprentissage lexical il vaut sans doute mieux imposer des contraintes d'acquisition plus fortes ; par exemple, un agent « instructeur » transmettrait des conventions lexicales aux autres agents. On perdrait toutefois l'homogénéité de la population, et la simultanéité de l'apprentissage et de l'utilisation – ce qui devrait être cognitivement justifié.

[7] La convergence est aussi ralentie par une capacité des agents non utilisée ici : ceux-ci tiennent aussi compte des classes d'entités autorisées comme arguments de *c*, *r1*, et *r2*, bien qu'ici toutes soient autorisées. Cette capacité pourra permettre l'utilisation d'étiquettes variées pour les feuilles des arbres élémentaires.

les objets *o*, puis environ 2800 pour *c*, environ 50 000 pour *r1* et de même pour *r2*. On peut remarquer que ces chiffres ne semblent pas dépendre linéairement du nombre d'agents.

Un phénomène intéressant est la polysémie des mots, à savoir l'usage d'un même mot pour deux entités différentes. Celle-ci n'est pas due, en général, à la génération aléatoire de deux mots nouveaux identiques, mais apparaît lorsqu'un agent récepteur « devine » de façon erronée un mot (puisqu'il est autorisé à en deviner). La polysémie est irrattrapable sans une « condition d'informativité » supplémentaire, car ce phénomène conduit à des succès plus fréquents pour les dialogues. On constate alors un début d'évolution vers une langue à un seul mot, qui désigne toutes les entités. Toutefois, il s'agit d'un phénomène plutôt rare : nous l'avons observé, pour un seul mot, environ une fois toutes les dix expériences, souvent entre des entités *o* et *r1*, ou *o* et *r2*.

IV.B.5.ii Échec de l'apprentissage non incrémental

Nous avons aussi fourni dès le début des situations faisant intervenir des entités appartenant aux quatre catégories (*o*, *c*, *r1* et *r2*), et cela sans restreindre non plus leur complexité – l'arbre de dérivation pouvant aller jusqu'à, par exemple, quatre niveaux. Dans ce cas, les agents ne parviennent pas à établir un langage différencié. Le résultat le plus fréquent est un langage à un seul mot, totalement polysème ; dans ce cas, l'ordre des mots, et donc les arbres, n'ont plus aucune importance.

IV.B.5.iii Description d'un fragment de situation

Le protocole d'apprentissage incrémental suppose que les agents émetteurs décrivent l'intégralité de la situation partagée. Une fois les conventions linguistiques établies, on peut permettre aux émetteurs de ne décrire qu'une partie de la situation, comme évoqué plus haut (cf. IV.B.4.i). Dans ce cas, alors que la capacité d'évolution des agents n'est bien sûr pas changée, les conventions linguistiques restent stables dans la population : le récepteur parvient donc à identifier correctement, en moyenne, le fragment de situation encodé par l'émetteur.

IV.B.5.iv Ajout de nouveaux agents

Lorsqu'un nouvel agent est ajouté à une population déjà stabilisée, cet agent acquiert rapidement les conventions linguistiques – que la situation soit décrite totalement ou partiellement par les émetteurs. Il n'y a pas besoin d'apprentissage incrémental dans ce cas, puisque le nouvel agent commence par comprendre seulement les descriptions courtes (*o*, puis *o*+*c*), et parvient peu à peu à comprendre les descriptions plus longues. En revanche, lorsqu'une nouvelle population est ajoutée à une population évoluée de même taille, l'apprentissage incrémental devient nécessaire. L'ajout d'un nouvel agent peut simuler de façon le processus d'acquisition du langage par un enfant, situé nécessairement dans une communauté de locuteurs déjà adultes.

IV.B.5.v Mélange de deux populations

Deux populations stabilisées de façon indépendante, avec des conventions différentes, peuvent être mélangées. Les résultats obtenus montrent que ni l'apprentissage incrémental, ni *a fortiori* celui non incrémental ne parviennent à unifier les conventions linguistiques. Ce résultat relativement surprenant s'explique, à nos yeux, par une définition trop indulgente des modifications linguistiques en cas de *succès partiel* : les mots et les arbres non préférés ne sont pas éliminés assez rapidement .

IV.B.6 Conclusion sur la syntaxe

Notre expérience d'émergence de conventions syntaxiques a montré dans quelles conditions celles-ci pouvaient s'établir de façon stable dans une population d'agents. Le modèle proposé modélise le lien entre syntaxe et sémantique grâce à l'utilisation d'un formalisme linguistique connu et éprouvé, la grammaire TAG. Le scénario d'émergence des conventions commence au niveau du lexique (stade à un mot), et construit peu à peu des messages plus complexes (deux mots, trois, etc.). Ainsi, l'ajout d'un agent non instruit à une population stabilisée reproduit dans une certaine mesure les phases de l'acquisition du langage : cet agent ne comprend d'abord que les messages à un mot, puis deux, etc.

D'un point de vue méthodologique, cette expérience montre bien la complémentarité de la « pragmatique simulée » par rapport à l'approche symbolique. En effet, la grammaire TAG est en soi une représentation formelle de la langue (nous avons vu toutefois au I.B.3 que l'on pouvait lui adjoindre une composante statistique). Cette grammaire est utilisée ici dans le contexte d'agents confrontés à des situations de l'environnement et à une « tâche ». Elle contribue alors à la modélisation d'un code de communication simulant certaines propriétés des langues : le caractère conventionnel et décentralisé, l'aisance de l'acquisition, la synonymie, la fluctuation lexicale. Toutefois, une de nos perspectives est la conception d'une représentation de la grammaire TAG de type connexionniste ou à acteurs, comme nous allons le préciser.

IV.C Perspectives intégratives

Les deux réalisations présentées ici rappellent les deux pôles que nous avions indiqués lors de notre synthèse des travaux existants. Il y avait d'une part, des agents plutôt autonomes et situés (II.C), et de l'autre des simulations plutôt formelles ou symboliques des échanges linguistiques (II.D).Le meilleur compromis étaient de loin les expériences de Luc Steels ; toutefois, ses propositions sur la syntaxe s'avèrent pour l'instant fondées sur une grammaire formelle qui ne semble pas utilisée ailleurs en linguistique. Il est par conséquent nécessaire d'intégrer ces deux tendances, c'est-à-dire, dans notre cas, réunir l'agent edelmanien et les agents symboliques utilisant la grammaire TAG.

Pour ce faire, nous visons deux objectifs prioritaires. Pour l'intégration de TAG dans une structure d'agent à base d'acteurs (ou, le cas échéant, d'un réseau connexionniste) il faut pouvoir exprimer les arbres et les différentes opérations en termes subsymboliques, ce qui est une tâche théorique sérieuse. Il s'agit toutefois d'une direction très prometteuse, car elle permettrait d'apporter des éléments de réponse au débat sur l'éventuelle représentation, dans le cerveau, de règles grammaticales.

Le second objectif consiste bien sûr à munir les agents « syntaxiques » de capacités de catégorisation inspirées de celles de notre premier agent autonome. Il s'agit de la capacité à détecter des objets ainsi que leurs propriétés et leurs positions dans un environnement simulé, et les représenter en termes d'arbres de dérivation (en accord avec leur à acteurs ou connexionniste). Cela nécessite dans un premier temps la constitution de l'environnement, et la définition d'une tâche pour laquelle la communication soit avantageuse. En outre, la faculté de reconnaître des entités, et user de « noms communs » pour les désigner, fait l'objet d'une étude détaillée dans la seconde partie de cette thèse, et d'un modèle d'implémentation dans un agent poursuivant l'inspiration edelmanienne.