Des écosystèmes artificiels d'aide à la conception: l'exemple du projet CAROSSE[1]

Rémy Foisel* — Alexis Drogoul* — Olivier Cayrol — Mondher Attia** — Nicolas Chauvat*****

**LIP6 - OASIS/MIRIAD*
UPMC/CNRS UMR 7606
Boîte 169, 4 Place Jussieu 75232 Paris CEDEX 05
http://www-poleia.lip6.fr/~*{foisel, drogoul}*
{foisel, drogoul}@poleia-lip6.fr,

***PSA Peugeot-Citroën, DRIA/RPAI/SEE,*
Route de Gisy 78943 Vélizy-Villacoublay
ma22@calva.net

****Information Sciences Institute*
University of Southern California
4676 Admiralty Way
Marina del Rey, CA 90292 Etats-Unis
nico@isi.edu

RÉSUMÉ. Nous présentons dans cet article une approche originale pour l'aide à la conception de systèmes complexes en milieu industriel. Là où la démarche dominante en IAD consiste à aborder la conception sous l'angle de l'analogie avec le travail d'équipe (systèmes multi-experts, SMA cognitifs), nous décrivons les systèmes à concevoir comme autant d'écosystèmes artificiels, peuplés d'agents hétérogènes dont le but est d'œuvrer collectivement à la réalisation des objectifs fixés par le concepteur. Nous argumentons que cette approche permet d'obtenir les qualités de modularité, réutilisabilité et extensibilité qui sont de plus en plus nécessaires dans le domaine industriel, tout en permettant de représenter les différents points de vue et niveaux d'abstraction utilisés lors des processus de conception. L'exemple utilisé est le projet CAROSSE, fruit de la collaboration entre le LIP6, le LORIA et le constructeur automobile PSA, dont le but est d'intégrer aux processus actuels de conception de véhicules des outils d'aide intelligents.

MOTS-CLÉS : aide à la conception, systèmes multi-agents industriels, problèmes d'allocation.

[1] Contrat PREDIT 1998-2000 subventionné par le MENRT.

1. Introduction

Beaucoup de produits industriels de grande consommation, comme les véhicules automobiles (qui constituent l'exemple de cet article), sont conçus comme des ensembles de sous-systèmes discrets et de composants qui interagissent les uns avec les autres au travers d'interfaces variées pour fournir la ou les fonctionnalités requises [PAR 97a]. En terme de conception, si chaque sous-système fonctionne parfaitement de manière autonome, leur mise en relation, et donc les interactions qui en résultent, génère un certain nombre de difficultés (pouvant conduire à des dysfonctionnements) [ATT 98], parmi lesquelles :

- des problèmes de compétition entre les prestations offertes,
- des problèmes de conflit dans l'accès à des ressources communes,
- des problèmes de dépendance d'un composant par rapport à un autre,
- des problèmes de communication dus à leur hétérogénéité.

Parallèlement, le couplage entre certains de ces composants fait apparaître des opportunités de coopération, sous la forme d'une *redondance fonctionnelle*, qu'il peut être intéressant d'exploiter ou utile d'éliminer pour des questions de coût. Réduire les problèmes de compétition, éliminer les conflits, assimiler les contraintes de dépendance, exploiter ou éliminer les redondances, sont les objectifs prioritaires de la conception globale de ces systèmes. Elle est également soumise à des contraintes d'objectif, qui sont une *sûreté maximale de fonctionnement*, un *coût de production minimal*, et une *capacité de fonctionnement en mode dégradé* favorisant la continuité des prestations essentielles au détriment des prestations accessoires.

Ces différents aspects de la conception de produits industriels complexes génèrent ou sont générés par un ensemble de contraintes, exprimées durant le cycle de vie du produit, maintenant bien connues dans les différentes communautés s'intéressant à ce problème (voir par exemple [PAR 97b]). Nous pouvons les résumer dans la liste suivante.

1.1. Problèmes et contraintes de conception

1.1.1. Multi-Expertise

Aucun produit industriel complexe n'est de nos jours conçu, réalisé et vendu par un seul intervenant (ou plusieurs intervenants ayant la même expertise). La multiplicité des compétences nécessaires pour leur conception implique de pouvoir maîtriser et organiser intelligemment, durant la phase d'analyse et de conception, des expertises et points de vue différents. De plus, les contraintes exprimées par les différents intervenants sont susceptibles d'impacter tous les niveaux de conception, à n'importe quel moment du cycle de vie. Il faut donc pour cela disposer d'une méthode de conception qui autorise plusieurs représentations simultanées du même produit, ainsi que des révisions à n'importe quel moment.

1.1.2. *Visions fonctionnelles et matérielles*

Un produit étant conçu pour répondre à une demande particulière, sa première définition va être habituellement fonctionnelle (cf. [PAR 99]), ce qui va entraîner une description descendante, également fonctionnelle, des sous-systèmes qui le composent. A l'inverse, l'architecture matérielle du produit (son instanciation dans le monde réel, c'est-à-dire l'assemblage des différents sous-systèmes matériels), va générer des contraintes ascendantes (disponibilité et caractéristiques des composants existant et de leurs interactions, etc.) tout en devant s'adapter au mieux aux nécessités fonctionnelles. L'interaction entre ces deux démarches nécessite de disposer dans la méthode de conception d'une représentation capable de gérer simultanément des vues fonctionnelles et matérielles, ainsi qu'un mécanisme révisable d'*allocation* (des fonctions sur les composants, ou inversement, qui soit automatique ou semi-automatique) couplé à des techniques de validation avancées.

1.1.3. *Complexité et aide à la conception*

La complexité de ces systèmes se manifeste non pas tant dans le nombre de sous-systèmes impliqués, que dans la multiplicité de leurs interactions. Les rapports de cause à effet entre la modification d'un composant et l'altération subséquente du fonctionnement global ne sont ni toujours discernables a priori, ni toujours compréhensibles par les concepteurs. Cela a pour conséquence, par exemple, la difficulté de détecter le sous-système fautif. Il faut pour cela disposer de méthodes permettant de *représenter* cette complexité (modélisation des composants mais aussi de leurs interactions) et aidant le concepteur dans la recherche d'une solution à un problème donné, y compris en utilisant des méthodes “ intelligentes ” (apprentissage de configurations, de préférences, diagnostic automatique).

1.1.4. *Dynamicité*

Les étapes d'analyse et de conception ne peuvent plus se contenter de produire des schémas statiques du système, car celui-ci peut être amené à évoluer en raison de la conjonction d'un certain nombre de causes:

— De nouvelles offres de prestations, qui conduisent à l'évolution des composants actuels et à l'ajout de nouveaux composants,

— De nouvelles normes gouvernementales, qui peuvent obliger l'industriel à adapter les composants existants ou à rajouter de nouveaux composants dans des délais de plus en plus courts, souvent inférieurs aux cycles habituels de conception,

— Des normes différentes selon les pays, qui amènent l'industriel à proposer des combinaisons différentes aux consommateurs de ces pays, la combinatoire des différentes solutions devenant difficile à gérer.

Le produit de la conception, de même que celui de l'analyse, doit donc pouvoir évoluer en fonction des retraits, ajouts ou modifications des composants. En effet, chacune de ces actions aura de manière certaine une incidence sur l'organisation de l'ensemble des sous-systèmes. Il est donc nécessaire de disposer de solutions adaptatives ou capables d'être amendées en peu de temps.

1.1.5. *Modularité, réutilisabilité, extensibilité*

En raison de la dynamicité évoquée plus haut, il est indispensable de disposer de méthodes qui, via une forme quelconque de réification des composants et des fonctions, permettent de les remplacer aisément par d'autres composants ou fonctions, de les réutiliser dans d'autres projets, voire de les supprimer du produit. On est là dans une problématique proche de celles de la conception orientée objets [RUM 91]. A ce titre, il convient de noter la convergence étroite entre les évolutions de l'ingénierie logicielle (où l'on raisonne maintenant en termes de *frameworks* et de *composants*) et de l'ingénierie automobile (où sont mises en avant les notions de *plates-formes* et de *sous-systèmes*).

1.2. *Objectif et plan de l'article*

Développer une méthodologie de conception qui apporte des solutions raisonnables aux problèmes évoqués plus haut et satisfasse à toutes ces contraintes est un défi redoutable tant celles-ci paraissent différentes en nature : là où certaines peuvent être quantifiées ou formalisées, d'autres relèvent d'une appréciation subjective ; certaines contraintes ne s'appliquent qu'à un sous-ensemble de composants alors que d'autres s'expriment sur la globalité du produit ; la plupart proviennent d'expertises ou de points de vue différents (bureaux d'études, services marketing, analyse financières, sous-traitants, etc.) ; toutes ne s'appliquent pas aux mêmes étapes du cycle de vie.

Notre objectif est de montrer qu'une approche multi-agent est à même de résoudre la plupart de ces contraintes et de proposer une vision unifiée des étapes de conception. Contrairement à la plupart des cas d'utilisation connus de SMA en entreprise ([PAR 99]), notre approche est basée sur une décomposition complète des systèmes à concevoir en agents relativement simples (ce mode de décomposition est proche de celui proposé dans [DRO 93]). En cela, elle est radicalement différente de l'approche multi-experts qui prévaut habituellement, et qui consiste à distribuer les connaissances des différents experts participant à la conception sur des agents artificiels [ABD 97], qui devront ensuite coopérer pour proposer des solutions aux concepteurs humains. Dans ce type d'approche, l'analogie utilisée est clairement celle du “ travail en équipe ” et les modes d'interaction entre les agents sont conditionnés, non pas par les données du problème, mais par l'organisation du travail qui prévaut dans l'entreprise [ABC 92 ; ABD 97].

A cette démarche descendante qui aboutit souvent à une décomposition fonctionnelle du travail de conception, et qui réduit d'emblée les chances de pouvoir disposer d'une conception modulaire et évolutive, nous opposons l'utilisation d'une autre analogie possible, curieusement peu exploitée malgré ses liens de parenté évidents avec l'analyse orientée objet : celle de *l'écosystème*, dans laquelle chaque fonction et chaque sous-système du système à concevoir est représenté par un ou plusieurs agents autonomes en interaction constante avec d'autres agents.

Cette décomposition, structurelle et fonctionnelle, permet d'envisager le système comme une communauté d'agents hétérogènes qui œuvrent collectivement à la réalisation d'objectifs communs définis par le concepteur. Comme nous le montrons dans la suite de cet article, cette vision nous permet d'appréhender de façon originale les différents niveaux d'analyse et d'abstraction, de ne pas se couper des objets réels manipulés dans le système par les ingénieurs, tout en appliquant des techniques simples de résolution de problèmes qui servent de support pour l'aide à la conception. En ce sens, elle est le prolongement exact, dans le domaine de la conception industrielle, des idées que nous défendons depuis maintenant plusieurs années, par exemple dans la résolution de problèmes [DRO 91].

Pour cela, nous présentons d'abord en section 2, le projet CAROSSE, pour lequel a été développé ce système de conception multi-agents. Après avoir, en début de section 3, expliqué en quoi consiste la métaphore de l'éco-système, nous présentons son instanciation dans CAROSSE et insistons en particulier sur la souplesse de représentation qu'autorise le système. Un exemple simple de résolution d'un problème d'allocation de fonctions sur les composants électroniques est ensuite présenté en section 4. Nous concluons enfin en section 5 sur les perspectives de recherche qu'offre ce projet, et sur la généralisation possible des techniques employées à d'autres domaines.

2. Le projet CAROSSE (LIP6/PSA/LORIA)

L'exemple illustratif de cet article est celui du projet CAROSSE (**C**onception, **A**nalyse et **R**éalisation de l'**O**rganisation de **S**ous-**S**ystèmes **E**lectroniques **E**mbarqués), fruit de la collaboration entre le LIP6, le LORIA et le groupe PSA (Peugeot-Citroën), qui a pour objectif la création d'un outil d'aide à la conception de l'ensemble du système électronique embarqué dans les véhicules actuels.

Concevoir un tel système qui regroupe plusieurs fonctions distinctes et fait appel à de nombreux composants matériels hétérogènes, consiste à effectuer les définitions successives de son architecture fonctionnelle, de son architecture physique et de son architecture organique (ou opérationnelle). L'architecture fonctionnelle comprend différents composants fonctionnels (source, puit ou entité transformant des informations) reliés par des flux d'information, l'architecture physique contient les composants matériels (plates-formes d'exécution à base de microcontrôleurs) reliés par des média de communication (bus de communication, etc.) et enfin l'architecture organique est composée de l'architecture fonctionnelle et de l'architecture physique reliées entre elles par des choix de placement exprimant la répartition des composants fonctionnels sur les composants matériels (par exemple, des fonctions sur les supports d'exécution).

Ce travail de conception fait intervenir successivement ou simultanément différents experts se focalisant sur une ou plusieurs des étapes du processus de conception :

— la *conception des fonctions véhicule*, qui consiste à concevoir les architectures fonctionnelles des différentes sous-parties (appelées *fonctions* *véhicule*) pouvant être présentes dans un véhicule (par exemple l'ABR[2] ou la climatisation).

— la *construction de l'architecture fonctionnelle*, à savoir l'assemblage des fonctions véhicule pour obtenir l'architecture fonctionnelle complète du système à embarquer avec comme objectif l'élimination des dysfonctionnements éventuels résultant des interactions entre fonctions.

— la *construction de l'architecture organique*, i.e. la mise en correspondance de l'architecture fonctionnelle avec l'architecture physique hôte du système en limitant les coûts issus des choix de placement et d'architecture physique (câblage, etc.).

L'outil envisagé doit fournir un cadre méthodologique pour l'organisation du travail de ces différents experts et leur fournir des fonctionnalités spécifiques qui puissent les aider efficacement dans leurs tâches respectives. Ces fonctionnalités ont plusieurs buts :

— permettre une conception modulaire et réutilisable des fonctions véhicule d'un point de vue fonctionnel ou organique (ce dernier point de vue correspondant ici à des contraintes de placement exprimées par les experts lors de la conception des fonctions véhicule),

— permettre une réutilisation et un assemblage aisé de ces fonctions véhicule pour bâtir des architectures fonctionnelles ou organiques,

— aider l'utilisateur à visualiser et à maintenir la cohérence et la complétude des fonctions véhicule ou des architectures fonctionnelles pendant leur conception,

— aider au placement d'une architecture fonctionnelle sur une architecture physique (afin de construire une architecture organique complète),

— et aider à la construction des modes dégradés du système électronique.

3. Proposition

Comme indiqué dans l'introduction, notre approche est basée sur une décomposition à la fois structurelle et fonctionnelle du système à concevoir, et non sur une distribution des expertises invoquées durant le processus de conception. Cette démarche, qui possède des similarités avec à la fois l'approche orientée objet [RUM 91] et l'approche « collective » de la Vie Artificielle [DRO 93] est basée sur la constatation que l'intérêt principal d'une approche multi-agents (surtout dans le domaine de la conception) dépend de la capacité des populations d'agents à s'organiser et à s'adapter dynamiquement aux circonstances sans devoir être entièrement contrôlées par le concepteur ou l'opérateur. Cette constatation, bien que partagée par la grande majorité des chercheurs du domaine de l'IAD, n'est en fait jamais véritablement traduite dans les faits : concevoir un système multi-agents composé d'agents complexes capables d'appliquer les expertises extraites d'experts humains, et de raisonner sur leur coordination avec les autres agents, loin de faciliter

[2] Anti-Blocage des Roues.

l'aide à la conception, ne fait que réintroduire les problèmes classiques de la conception de systèmes complexes dans un système artificiel.

Comme nous l'avons déjà évoqué dans l'introduction, contrairement, à cette démarche descendante, nous proposons l'utilisation d'une autre analogie facilitant une conception modulaire et évolutive : celle de *l'écosystème*, dans laquelle chaque composant et chaque sous-système du système à concevoir est représenté par un ou plusieurs agents autonomes en interaction constante avec d'autres agents. Cette démarche, qui possède des similarités avec à la fois l'approche orientée objet [RUM 91] et l'approche « collective » de la Vie Artificielle [DRO 93] est basée sur la constatation que l'intérêt principal d'une approche multi-agent dépend de la capacité des populations d'agents à s'organiser et à s'adapter dynamiquement aux circonstances sans devoir être entièrement contrôlées par le concepteur ou l'opérateur.

Certains systèmes collectifs « naturels », comme les populations d'insectes ou d'autres animaux, montrent qu'une autre voie est possible pour aider à l'ingénierie de systèmes complexes. Nous avons déjà argumenté [DRO 97] sur le parallèle qui pouvait être tracé entre l'approche biologique et l'ingénierie, suivant en cela les théories défendues par [DEN 95], pour qui l'évolution naturelle est en elle-même un processus de conception de systèmes incroyablement complexes, mené au jour le jour par des agents très simples. Dans cette approche, souvent qualifiée d'émergentiste, c'est le comportement global et l'auto-organisation de l'ensemble des agents qui importe, et pas celui des agents individuels, qui n'ont qu'une représentation limitée des objectifs à atteindre. La figure 1 montre ce que cette approche peut avoir comme conséquences en résolution de problèmes [DRO 91].

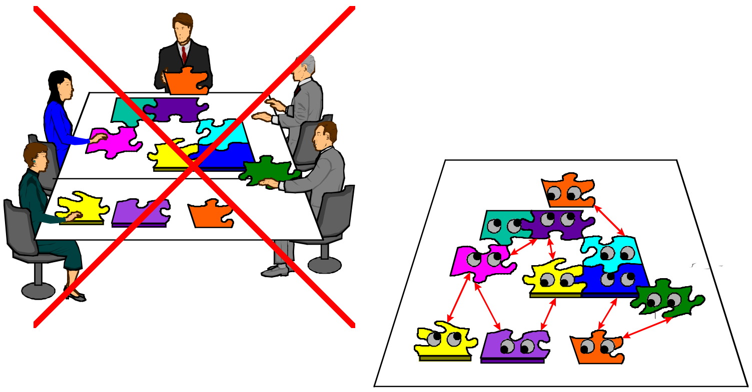

Figure 1. *Approches possibles de la résolution distribuée de problèmes*

Dans le domaine de l'aide à la conception, qui nous intéresse ici, l'idée principale est de proposer au concepteur exactement ce que la simulation multi-agents propose déjà au simulateur :

— la possibilité de représenter la complexité d'un système, non pas sous la forme d'un modèle complexe, mais sous celle de l'interaction orientée d'une multitude de modèles simples,

— la possibilité de représenter simultanément plusieurs niveaux d'abstraction en utilisant le même formalisme, ainsi que de modéliser différents points de vue et expertises sur le système (cf. en simulation [SER 98]),
— la possibilité d'intervenir à ces différents niveaux et d'effectuer des modifications locales ou globales immédiatement répercutées dans les parties du système concernées (à ce titre, une modification locale - suppression ou ajout d'agents, par exemple - n'impactera que ce qui est nécessaire),
— la possibilité, par simulation, de voir dynamiquement émerger des solutions de conception possibles, de pouvoir agir sur ces solutions, et de pouvoir assister aux réorganisations engendrées par cette action (comme si le système possédait des qualités homéostatiques).

A ce titre, il s'agit donc bien de proposer au concepteur un *laboratoire virtuel de conception* lui permettant de mener à bien l'étude d'un *écosystème artificiel*. Si l'on se réfère à la liste, détaillée dans l'introduction, des contraintes pesant sur la conception industrielle, on voit bien ce qu'un tel outil peut apporter, et il n'est pas déraisonnable de penser que ce genre d'approche prépare dans le domaine de la conception industrielle une révolution comparable à celle réalisée par l'approche orientée objet dans le domaine de la conception logicielle.

Cependant, il y a loin de la coupe aux lèvres, et les quelques projets entamés ici et là (cf. une liste plus ou moins exhaustive dans [PAR 99]) avec la même démarche n'en sont encore qu'à leur démarrage. Il est en effet difficile, dans ce domaine encore balbutiant, de pouvoir, d'une part, trouver des partenaires industriels désireux de s'aventurer dans ce qui apparaît souvent comme une approche peu réaliste, et, d'autre part, de pouvoir mener des expérimentations longues et délicates (surtout quand elles doivent aboutir à la conception d'un produit réel de grande consommation). A ce titre, le projet CAROSSE, collaboration (pour sa partie multi-agents) entre le LIP6 et le groupe PSA, est porteur à nos yeux de beaucoup d'espoirs. Afin d'illustrer cette opinion, nous allons détailler maintenant la façon dont notre démarche s'est trouvée instanciée dans ce projet.

3.1. Représentation de la complexité

La complexité d'un projet comme CAROSSE se mesure à l'un des multiples aspects à prendre en compte dans sa réalisation, donc des différentes espèces d'agents qui peuplent ce projet. Nous distinguons ici trois aspects importants pour la conception de systèmes électronique embarqués :

— Le premier aspect consiste à disposer d'une approche par composants des concepts et éléments manipulés lors de la conception, afin de permettre une conception modulaire, la réutilisation partielle ou totale du produit des conceptions antérieures, ainsi que l'extensibilité et l'incrémentalité des systèmes conçus [LHU 98]. La plus grande partie des agents seront issus de cette décomposition.
— Le second aspect est de permettre une vision systémique [LAP 92 ; LEM 90] de la conception selon différents niveaux de détails du système conçus, afin

de permettre aux utilisateurs de travailler simultanément selon ces différents niveaux d'abstractions ou points de vue.
— Enfin, le dernier aspect est la prise en compte des spécificités du travail des concepteurs devant utiliser l'outil, en leur permettant d'adapter la visualisation et l'aide qu'il leur fournit.

Nous allons détailler ces différents aspects dans les sections suivantes en montrant comment l'approche multi-agents aborde leur représentation.

3.1.1. Approche par composants

La conception d'un système électronique peut être interprétée comme l'assemblage de composants abstraits ou concrets afin de définir l'architecture fonctionnelle ou l'architecture organique du système [CHA 98]. Dans le cadre de CAROSSE, nous distinguons différents types de composants :

— Les *composants élémentaires* qui permettent de définir le niveau le plus détaillé des fonctions véhicule ou des architectures. Nous distinguons :
- Les *composants élémentaires fonctionnels*, tels que les fonctions élémentaires, les capteurs, les actionneurs et les flux d'informations, qui permettent de définir le niveau le plus détaillé d'une architecture fonctionnelle (les capteurs et actionneurs sont considérés ici comme des sources et des puits d'information).
- Les composants *élémentaires physiques*, tels que les bus de communication, les microcontrôleurs, et les connections, qui permettent de définir le niveau le plus détaillé d'une architecture physique hôte du système.

— Les *composants abstraits* qui permettent de fournir une vision plus abstraite de l'ensemble des composants élémentaires. Nous distinguons :
- Le*s fonctions véhicule* qui permettent de disposer d'une modularité et d'une réutilisation aisée des grandes catégories de fonctions intervenant dans les véhicules pour la conception des systèmes. Ces composants sont décrits comme une conjonction de composants élémentaires fonctionnels, et peuvent être classés dans un arbre par similarité ou différence des services qu'ils fournissent.
- Les *groupes matériels* qui sont décrits comme la conjonction de composants physiques. Ce composant est associé à une fonction véhicule (figure 2), pour exprimer les contraintes industrielles d'allocation liées à cette partie d'architecture fonctionnelle.
- Les *architectures physiques* qui sont décrits comme la conjonction de groupes matériels. Ce composant correspond à l'architecture physique hôte du système, et est associé à un projet, comme l'illustre la figure 2.
- Enfin, les *projets* qui sont le regroupement d'un ensemble de fonctions véhicule (constituant une architecture fonctionnelle ou organique).

Cette classification en terme de composants des éléments manipulés pour la conception du système va servir de base à une décomposition orientée agent. En associant à ces composants des comportements spécifiques, nous leur donnons la possibilité de gérer de manière décentralisée les connaissances qui les caractérisent et le savoir-faire nécessaire à l'aide à la conception (cf. section 1). Cette *agentification* permet ainsi de simplifier la complexité des connaissances et compétences à répartir, de disposer d'un cadre réactif pour l'adaptation des solutions et de faciliter la réutilisation des composants concernés dans différents projets de conception.

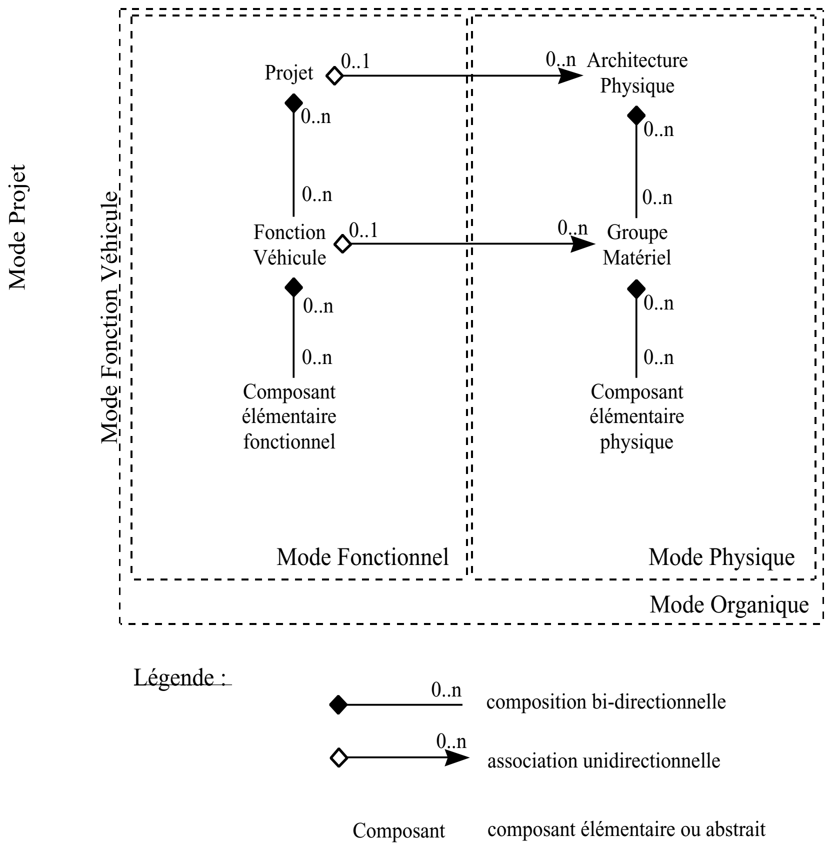

Figure 2. *Diagramme de composition et d'association des composants.*

3.1.2. Vision systémique

La conception d'un système embarqué ou d'une partie d'un système embarqué requiert différents niveaux de représentation plus ou moins détaillés [CHA 98].

L'utilisateur doit disposer de différentes vues de son travail afin de travailler à partir de composants abstraits ou élémentaires (conception descendante, ascendante ou mixte) et le système d'aide à la conception doit lui permettre de visualiser aisément l'impact des modifications effectuées à un niveau sur l'autre. Nous distinguons donc deux niveaux de travail simultanés pour CAROSSE :

— le niveau des composants élémentaires qui est l'espace de travail le plus détaillé que propose l'outil d'aide à la conception (voir la partie inférieure de la figure 3),
— le niveau des regroupements qui est une abstraction du premier niveau à partir de composants abstraits (voir la partie supérieure de la figure 3).

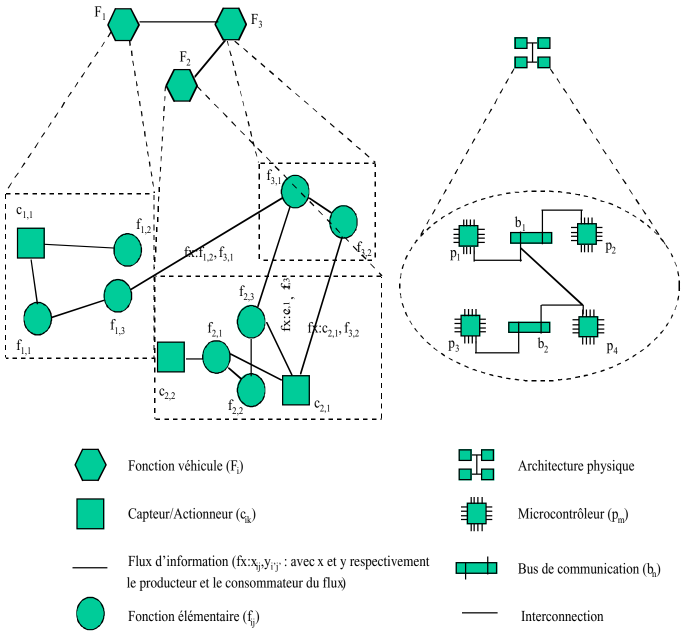

Figure 3. *Vision systémique d'un projet.*

Ces deux niveaux de visualisation correspondent aussi à un environnement d'interaction avec l'utilisateur où les différents agents, que nous avons identifés, sont représentés [RIC 97 ; FOI 98a]. La mise à jour des modifications effectuées dans un des deux niveaux est alors répercutée sur l'autre grâce aux composants abstraits correspondant à des *groupes* [FOI 98b]. L'action des agents permet de façonner l'environnement, qui, en retour (ou sous l'impulsion de l'utilisateur), façonne les agents.

3.1.3. Point de vue multi-experts

Comme nous l'avons évoqué lors de l'introduction de cette partie les différentes spécificités des experts ou métiers impliqués doivent être prise en compte [ABD 97]. Pour cela, nous avons adopté une classification en termes de modes de travail pour adapter les visualisations aux deux niveaux de travail (cf figure 3).

Tout d'abord, deux modes *principaux d'utilisation* de l'outil ont été identifiés : le mode de conception de fonction véhicule (visant à concevoir une sous-partie (un service) pour un ou plusieurs systèmes, voir figure 2), et le mode de conception de projet (qui vise à concevoir une architecture de système embarqué, voir figure 2) qui correspond au métier des architectes (fonctionnel ou organique). Ces deux modes peuvent être combinés à *trois modes de visualisation* possibles : le mode de conception fonctionnelle, le mode de conception physique et le mode de conception organique du système embarqué. La figure 2 présente ces modes et leur espace respectif par rapport aux agents identifiés.

Cette classification permet alors d'adapter les espaces de visualisation des composants manipulés à la spécificité de la tâche des différents métiers impliqués, mais aussi d'adapter les techniques de résolution que les agents mettent en œuvre pour cette tâche. Nous parlerons dans la suite de l'article de *modes de résolution* pour nous référer aux six combinaisons possibles de ces modes de travail (l'une d'elle est par exemple : le mode projet avec le mode fonctionnel) qui forment des points de vue de résolution.

3.2. Aide à la conception

3.2.1. Différents problèmes à résoudre

L'aide à la conception telle que nous l'envisageons ne consiste pas uniquement à représenter et à visualiser le système conçu. Comme nous l'avons signalé dans la section consacrée à la problématique de CAROSSE, d'autres fonctionnalités doivent être envisagées pour aider les utilisateurs :

— maintenir la cohérence et la complétude des fonctions véhicule ou des projets au cours de leur construction,
— aider au placement d'une architecture fonctionnelle sur une architecture physique afin de construire l'architecture organique,
— aider à la construction des modes dégradés du système électronique.

Ces fonctionnalités correspondent à plusieurs problématiques d'aide pour lesquelles plusieurs techniques sont envisageables. Par exemple, diverses stratégies de placement peuvent être définies et nous envisageons de fournir aux utilisateurs la possibilité de paramétrer le mode de résolution utilisé pour choisir, parmi ces stratégies d'aide, celle qui leur semble la mieux adaptée.

3.2.2. Diverses techniques de résolution à répartir

Nous proposons d'associer aux composants visualisés des comportements spécifiques d'aide à la conception en fonction du mode de résolution actif. Ces comportements permettent d'exprimer de manière décentralisée l'exploitation des caractéristiques des composants selon des stratégies spécifiques aux modes de résolution possibles. Chaque composant est alors un agent qui participe à la résolution des problématiques identifiées (cf section 3.2.1). En raison des différents modes de résolution dans lesquels les agents vont pouvoir tenir un rôle, nous proposons d'adopter une architecture d'agent, composée d'un module principal (réagissant aux actions de l'utilisateur via l'interface) activant l'un des six autres modules secondaires spécifiques à chacun des modes de résolution possibles (qui pourront être adapter selon les préférences de l'utilisateur via le paramétrage évoqué dans la section précédente) lors d'une phase de résolution du système.

Nous avons déjà commencé (voir section 4) à définir un premier ensemble de ces modules formant une stratégie de résolution de problème, inspiré de l'éco-résolution [DRO 91], pour l'aide à l'allocation du fonctionnel sur le physique. Cette première étape a déjà fourni des résultats et des enseignements remarquables, mais nous devons réfléchir à l'incorporation d'autres techniques, peut-être plus efficaces ou plus facilement paramétrables, afin de fournir une gamme variée de techniques d'aide au placement. Un autre intérêt de cette approche est de permettre, à terme, la comparaison entre diverses techniques de résolution, parmi lesquelles nous pouvons d'ores et déjà citer :

— L'optimisation distribuée de contraintes, évoquée pour son intérêt dans le domaine de la conception industrielle par [PAR 97b], et dont on trouve une bonne illustration dans [GHE 99].
— L'apprentissage automatique, sur lequel nous avons déjà réalisé une première étude [DRO 98], et qui pourrait être utilisé à différents niveaux, soit pour assurer la capitalisation des solutions précédemment retenues, soit pour l'exploration de paramétrages des systèmes de RdP employés,
— L'évolution artificielle, soit en termes d'optimisation des architectures existantes (par algorithmes génétiques), soit en termes de proposition d'architectures organiques à partir du regroupement dynamique et évolutif (sanctionné par le concepteur) de fonctions et de composants.

Implémenter l'ensemble de ces techniques est un travail de longue haleine. Le projet CAROSSE n'ayant démarré qu'en 1998, il est normal que nous n'en ayons pas apprécié toutes les difficultés. Mais le travail préliminaire présenté ci-dessous nous a convaincu de la pertinence de notre approche.

4. Exemple: Allocation de fonctions sur composants

Nous illustrons ici un exemple de stratégie de résolution pour le placement des composants élémentaires fonctionnels sur une architecture physique hôte. Nous adoptons la notation présenté dans la figure 3 pour expliciter la stratégie et nous illustrons le déroulement de celle-ci dans la figure 4 à partir de l'ensemble des composants présentés dans la phase 1 de cette figure.

Le placement consiste à allouer les f_{ij}, c_{ik} et fx:x_{ij}, $y_{i'j'}$ sur des supports physiques p_m, b_n et i_o, en prenant en compte les contraintes exprimées par les concepteurs du système électronique. Le système multi-agents peut alors être vue comme un ensemble d'agents liés par des dépendances : *fonctionnelles* (entre les flux d'informations et les fonctions élémentaires, etc.), *structurelles* (entre bus et microcontrôleurs, etc.) et de placement (des flux sur les bus ou les fils, des fonctions élémentaires sur des microcontrôleurs, etc.).

Les dépendances fonctionnelles et structurelles sont présentées dans la phase 2 de la figure 4 par des flèches grisées entre les agents concernés. Toutes les dépendances n'ont pas été représentées afin d'éviter de surcharger le schéma.

Les dépendances de placement sont représentées par le même type de flèches en pointillé. On peut ainsi constater sur la figure que les concepteurs ont contraint le placement de la fonction élémentaire $f_{1,2}$ sur le microcontrôleur p_1, et l'allocation des flux fx:c_{21}, f_{32}, fx:c_{21}, f_{23}, fx:c_{21}, f_{21} et du capteur $c_{2,1}$ sur un fil.

A partir des liens de dépendances, les agents commencent la résolution en créant dynamiquement de nouveaux agents (entourés en pointillé sur la figure 4). Ces nouveaux agents peuvent être de deux types :

— Les connections filaires ou fils (i_o) qui sont des agents créés à partir des contraintes de placement imposées par les concepteurs sur le cheminement de certains flux d'informations (fx:x_{ij}, $y_{i'j'}$). Un cas particulier de ce type de création est illustré dans la phase 2 de la figure 4 pour les flux fx:c_{21}, f_{32}, fx:c_{21}, f_{23} et fx:c_{21}, f_{21} qui ont pour contrainte d'être affectés sur un fil à la sortie du capteur $c_{2,1}$ qui devra lui-même être connecté au microcontrôleur p_2 (cette connectivité étant une dépendance structurelle).

— Les fonctions de servitude qui sont créées pour compléter l'architecture fonctionnel après placement des composants fonctionnels. Le cas du flux fx:c_{21}, f_{23} illustre ce propos puisqu'il doit cheminer du microcontrôleur p_2 (ayant pour hôte contraint le fil i_3 connecté au capteur c_{21}) au microcontrôleur p_3 (ayant pour hôte contraint la fonction élémentaire f_{23}) par le bus b_1 et b_2. La fonction de servitude fs_l est alors créée et allouée sur le seul microcontrôleur (p_4) pouvant assurer le transit d'informations entre les deux bus.

La résolution du problème de l'allocation consiste alors à détecter les conflits éventuels (pour les problèmes sur-contraints), ou à demander à l'utilisateur de supprimer certaines contraintes ou d'attribuer plus de ressources aux composants physiques (ou éventuellement d'ajouter des composants physiques à cette architecture), ceci afin de fournir une proposition de placement correspondant à une architecture organique, comme le montre la phase 3 de la figure 4. Cette architecture peut ensuite être améliorée manuellement par l'utilisateur.

Les algorithmes de résolution utilisés par les agents sont très proches de ceux définis dans les systèmes d'éco-résolution [DRO 91 ; DRO 95]. Chaque agent possède une fonction d'évaluation locale de sa satisfaction, et interagit (agression, fuite) avec des accointances spécifiques (selon ses dépendances, et ses buts) pour la maximiser. Une description plus détaillée de ces comportements est disponible dans [CHA 98].

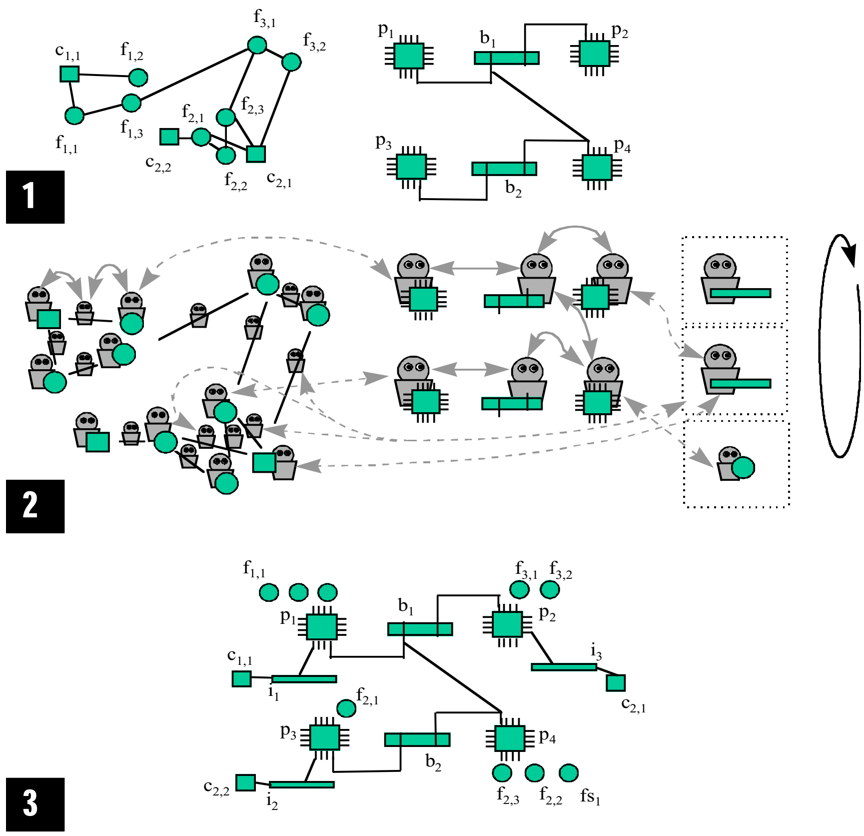

Figure 4. *Exemple de placement de composants fonctionnels sur des composants physiques.*

5. Conclusion

Cet article nous a permis de présenter une approche multi-agents originale de l'aide à la conception de systèmes complexes en environnement industriel. Le point principal repose sur une décomposition intrinsèque du système à concevoir en un ensemble d'agents hétérogènes, représentant fonctions ou composants matériels, et formant collectivement un écosystème artificiel dont l'évolution est guidée par le concepteur. Cette approche offre de nombreux avantages par rapport à l'approche traditionnelle adoptée en IAD [ABC 92], en ce qu'elle permet, tout en restant la plus proche possible des objets réels manipulés par les concepteurs, de distribuer efficacement et localement les contraintes de conception, et surtout de travailler simultanément à différents niveaux d'abstraction ou selon plusieurs points de vue.

Plusieurs questions restent cependant en suspens, et c'est sur celles-ci que nous souhaitons poursuivre notre effort dans les années à venir :

— Comment facilement intégrer ce type d'outil dans le cycle global de conception ? Doit-il le couvrir intégralement, au risque de devenir par trop généraliste, ou simplement être utilisé comme système d'aide à la conception ponctuel au sein d'un processus déjà existant ? Seules les campagnes d'expérimentation prévues avec les ingénieurs de PSA nous permettront de répondre à cette double question.

— Si jamais cet outil arrivait (même de façon parallèle au cycle traditionnel) à couvrir tout le cycle, la tentation est grande de projeter les agents ayant servi à la conception dans le système réel lui-même, afin de les utiliser pour réaliser du diagnostic *on-line* ou, pourquoi pas, se réorganiser pour proposer des solutions en temps réel au problème des modes dégradés (qui est résolu actuellement de façon *off-line*). Vus les récents développements en ce sens dans l'industrie automobile (par exemple, le SRI travaille dans le cadre d'une collaboration industrielle à l'intégration de communautés d'agents assistants aux futurs véhicules), il peut être intéressant de se pencher sur ce problème, même s'il dépasse le cadre actuel de CAROSSE.

— Enfin, il est clair que, malgré la spécificité de la terminologie et de certains concepts, la conception de l'architecture électronique embarquée dans un véhicule présente de nombreux points communs avec toutes les formes de conceptions de systèmes industriels complexes. Un de nos objectifs prioritaires va donc être de généraliser l'architecture multi-agents employée à d'autres problèmes de conception.

6. Bibliographie

[ABC 92] Abchiche, N., Collinot, A., and David, J.-M. "Modelling Cooperation Reasoning." *AAAI'92 Workshop on Cooperation among Heterogeneous Intelligent Systems*, San Jose (Etats-Unis). 1992.

[ABD 97] Abdalla, H.. "Concurrent Engineering for Global Manufacturing. " In Proceeding of 14th International Conference on Production Research, Osaka, Japan, 4-8 August 1997.

[ATT 98] Attia, M., Cayrol, O., Collinot, A., and Drogoul, A. "Methodology to design and optimise electronic system architecture for vehicle motion functions." *IEEE-SMC Cesa'98*, Hammamet, Tunisia.

[COL 98] Collinot, A., and Drogoul, A.. "La méthode de conception multi-agent Cassiopée : application à la robotique collective." Revue d'Intelligence Artificielle (Hermès), 1998.

[CHA 98] Chauvat, N.. “Etude d’une approche multi-agents pour l’aide à l’optimisation de l’électronique d’un véhicule. ” Rapport de DEA IARFA, Université Paris 6, Septembre 1998.

[DEN 95] Dennett, D. "Darwin's Dangerous Idea", Touchstone, Simon & Schuster, New-York, 1995.

[DRO 91] Drogoul, A., and Dubreuil, C. “Eco-Problem-Solving Model: Results of the N-Puzzle.” Decentralized A.I. 3, E. Werner and Y. Demazeau, eds., Elsevier North Holland, Amsterdam, 283-295. 1991.

[DRO 93] Drogoul, A. De la Simulation multi-agent à la résolution collective de problèmes, Thèse de 3ème Cycle, Université Pierre et Marie Curie, 1993.

[DRO 95] Drogoul, A. “When Ants Play Chess (or Can Strategies Emerge from Tactical Behaviors ?).” From reaction to cognition, C. Castelfranchi and J. P. Müller, eds., Springer-Verlag, Berlin-Heidelberg, 13-27. 1995.

[DRO 97] Drogoul, A., Collinot, A. “Entre réductionnisme méthodologique et stratégie intentionnelle, l'éthologie, un modèle alternatif pour l'IAD ?” *JFIADSMA'97*, 307-322. 1997.

[DRO 98] Drogoul, A., and Zucker, J.-D. “Methodological Issues for Designing Multi-Agent Systems with Machine Learning Techniques: Capitalizing Experiences from the RoboCup Challenge.” Rapport technique *041*, LIP6 - University of Paris 6, Paris. 1998.

[FOI 98a] Foisel, R.. Modèle de réorganisation de systèmes multi-agents : Une approche descriptive et opérationnelle, Thèse de doctorat de l'université Henri Poincaré, Nancy 1, Novembre 1998.

[FOI 98b] Foisel, R.. “Construire des systèmes multi-agents à partir de schémas d'interactions. ” Actes des 6èmes Journées Francophones IAD et SMA. Hermès, Pont-à-Mousson, Novembre 1998

[GHE 99] Ghedira, K.. Une Approche Distribuée d'Optimisation sous Contraintes. Présentation au Séminaire OASIS/LIP6. Paris, Mars 1999.

[LAP 92] Lapierre, J.-W. L'Analyse de Systèmes; L'application aux Sciences Sociales, Syros, Paris. 1992.

[LEM 90] Le Moigne, J.-L. *La Modélisation des Systèmes Complexes*, Editions Dunod, Paris. 1990.

[LHU 98] Lhuillier, M.. Une approche à base de composants logiciels pour la conception d’agents : Principes et mise en œuvre à travers la plate-forme Maleva, Thèse de doctorat de l'université Paris 6, Février 1998.

[PAR 97a] Parunak, H.V.D.. "Go to the Ant": Engineering Principles from Natural Agent Systems (1/97) Annals of Operations Research 75 (1997) 69-101.

[PAR 97b] Parunak, H.V.D., Ward, A., Fleischer, M., Sauter, J., and Chang, T.-C.. Distributed Component-Centered Design as Agent-Based Distributed Constraint

Optimization. In Proceedings of AAAI Workshop on Constraints and Agents, pages 93-99, 1997.

[PAR 99] Parunak, H.V.D.. Industrial and Practical Applications of DAI. In G. Weiss, editor, Multiagent Systems. MIT Press, 1999.

[RIC 97] Rich, C. and Sidner, C.. "COLLAGEN : When agents collaborate with people." In Proceedings of the International Conference on Autonomous Agents, 1997.

[RUM 91] Rumbaugh, J., Blaha, M., Eddy, F., Premerlani, W., and Lorensen, W. Object Oriented Modeling and Design, Prentice Hall. 1991.

[SER 98] Servat, D., Perrier, E., Treuil, J.-P., and Drogoul, A. "Towards Virtual Experiment Laboratories: How Multi-Agent Simulations Can Cope with Multiple Scales of Analysis and Viewpoints." Actes de *Virtual Worlds'98*, Paris, France. 1998.