

COOPERATIONS RECHERCHE / ENTREPRISES POUR LA CONCEPTION DE LOGICIELS D'ORDONNANCEMENT : SIPAPLUS, IO, TAPAS

Christian Bérard*, Bernard Grabot** et Philippe Nguyen***

Résumé. - L'objectif de cet article est de décrire différents types de coopération entreprises/laboratoires de recherche dans le cadre du développement de logiciels d'ordonnancement. Les produits développés s'appuient tous sur quelques constatations issues de situations réelles de l'industrie : (a) l'optimalité des solutions d'ordonnancement par rapport à un seul critère n'a pas beaucoup d'importance devant la variabilité de l'environnement, il est plus important (b) de chercher la robustesse des solutions d'ordonnancement, (c) d'en fournir diverses évaluations au décideur et, surtout, (d) de lui donner le moyen d'entrer dans le processus d'élaboration des solutions d'ordonnancement. Les collaborations ayant mené au développement de trois logiciels découlant de ces constatations sont décrites et analysées. Le transfert d'un travail de recherche à une entreprise industrielle est d'abord abordé avec SipaPlus. Le développement conjoint d'un produit par des chercheurs et des développeurs industriels est ensuite abordé avec IO. La dernière expérience décrite est celle d'un développement universitaire intégré dans un produit industriel (TAPAS). Ce dernier projet montre comment certaines idées, abandonnées dans les logiciels précédents, peuvent être reprises dans un cadre différent.

Mots-clés : ordonnancement, coopération recherche/industrie, transfert.

1. Introduction

En ordonnancement comme dans d'autres domaines, le partenariat Recherche-Entreprise pour le développement de produits peut recouvrir des réalités très différentes et utiliser des modalités variées. Nous proposons dans cet article trois exemples apparemment

* LAP/GRAI, Université de Bordeaux I, Talence.

** LGP/ENIT, Tarbes.

*** CESIUM, Bordeaux.

semblables (développement de logiciels d'ordonnancement) mais qui abordent le problème de la collaboration Recherche-Entreprise sous trois angles différents.

Le cas du passage d'un logiciel de recherche (APIS) à un logiciel industriel (SIPA (GPSA)) est un exemple relativement classique de transfert de résultat de recherche. Le développement du logiciel Io (CESIUM) est par contre le fruit d'une collaboration dans laquelle chercheurs et industriels ont été impliqués dès le début du projet. Dans le cas de TAPAS, il s'agit davantage d'une collaboration par "complémentarité" : contrairement à ce qui se passe dans les autres projets, chaque partenaire apporte une contribution originale complète qui doit s'interfacer avec celles de ses partenaires.

La problématique de l'ordonnancement d'atelier est tout d'abord rappelée, et l'intérêt des approches coopératives dans ce domaine est souligné. Les caractéristiques principales des trois logiciels faisant l'objet de cet article sont présentées dans la troisième partie. La quatrième partie présente les types de coopération ayant permis le développement de ces produits, ainsi que le lien entre les fonctionnalités des produits et les types de coopération qui les ont vu naître. Quelques leçons seront dégagées de ces expériences.

2. Rappel de la problématique

Après avoir rappelé la problématique générale de l'aide à la décision en ordonnancement, nous montrerons comment les logiciels ont évolué de l'interactivité vers la coopération.

2.1 *Problématique générale*

Au moment où les travaux évoqués ici ont commencé (vers 1980), l'ordonnancement informatisé était considéré comme un calcul complexe dont les hommes de terrain étaient exclus. L'apparition des micro-ordinateurs, des écrans graphiques et aujourd'hui des réseaux a rendu l'ordinateur accessible à un plus grand nombre d'acteurs de l'entreprise. De plus, l'accroissement de la rapidité des processeurs a permis d'augmenter les volumes de données traitées. Tous ces facteurs ont favorisé le développement des possibilités d'interaction entre les décideurs et le calcul des ordonnancements.

Dans ce contexte, la problématique générale peut être résumée ainsi : "Comment concevoir des logiciels d'aide à l'ordonnancement ?". Cette problématique implique plusieurs domaines de compétence : Productique, Informatique, Ergonomie. De plus, elle oblige à réaliser des prototypes afin de disposer d'un terrain expérimental réel et de valider les principes proposés.

2.2 *Ordonnancement et interactivité*

Pour beaucoup d'entreprises manufacturières, l'évolution de la demande des clients s'est faite vers des produits de plus en plus particularisés, en petites quantités. Une réponse efficace à ce genre de besoin a été rendue possible par une augmentation de la flexibilité des ateliers, qui s'est traduite par une augmentation des degrés de liberté à gérer. Ces degrés de liberté concernent aussi bien le choix des ressources que les gammes de fabrication. D'autre

part, la nécessité croissante d'optimiser l'utilisation des ressources de production a rendu indispensable la prise en compte de contraintes technologiques très spécifiques dans les logiciels d'ordonnancement. Les spécificités d'un atelier donné pouvant difficilement être prises en compte dans des logiciels voulant rester génériques, des logiciels interactifs, dont Sipaplus fut un pionnier dès 1984, sont alors apparus. Le principe de base est de permettre une amélioration manuelle d'une solution d'ordonnancement faisable obtenue suivant des méthodes génériques. La démarche d'utilisation d'un tel logiciel peut alors être la suivante :

- définition des hypothèses d'ordonnancement,
- proposition d'un ordonnancement par le logiciel,
- analyse de l'ordonnancement par le gestionnaire d'atelier afin d'évaluer si cet ordonnancement est compatible avec les objectifs de production courants, et satisfait les contraintes technologiques,
- utilisation des degrés de liberté gérés par le logiciel pour modifier les hypothèses d'ordonnancement,
- création d'un nouvel ordonnancement.

Cette alternance entre décision humaine et production d'un ordonnancement par le logiciel a déjà été proposée dans (Farhoodi, 90). Cette approche a depuis été étendue à une production d'ordonnancement résultant d'une réelle coopération entre utilisateur et logiciel.

2.3 Ordonnancement et coopération

La coopération en ordonnancement est comprise ici comme la mise au point conjointe d'un ordonnancement par le gestionnaire d'atelier et le logiciel. Une condition nécessaire à la coopération est de permettre aux acteurs impliqués d'agir sur les mêmes objets. Cette condition est souvent interprétée en ordonnancement comme la possibilité pour l'utilisateur de modifier directement le planning (Bérard et al. 97), (Lopez et al. 95). Cette possibilité n'est à notre avis pas suffisante pour parler de coopération, celle-ci nécessite :

- de proposer à l'utilisateur de nouveaux modes de visualisation de l'ordonnancement existant, de manière à lui permettre de diagnostiquer rapidement les améliorations à apporter,
- de le soulager de la gestion de contraintes temporelles difficiles à manier, comme le respect de durées minimales ou maximales entre opérations, ou la synchronisation de plusieurs opérations,
- de permettre d'élaborer manuellement tout ou partie du planning, en bénéficiant d'un environnement ergonomique dans ce but et en pouvant être immédiatement relayé par le logiciel. L'ordonnancement pourra dans ce cas être réalisé en plusieurs passes, effectuées soit manuellement par le gestionnaire d'atelier, soit automatiquement par le logiciel. Le placement des opérations de production pourra ainsi être parfaitement maîtrisé par l'utilisateur,
- de proposer des outils efficaces d'évaluation des performances, pour comparer deux ordonnancements possibles en fonction des objectifs de production courants.

Les logiciels qui sont brièvement présentés dans la partie suivante illustrent bien le passage de l'interactivité à la coopération en ordonnancement.

3. Les logiciels

3.1 *Apis (GRAI)*

APIS a été défini dans (Bérard, 83). Le principe d'utilisation d'APIS est dès l'origine une démarche coopérative dans la résolution de l'ordonnancement. Chaque machine est modélisée par un module de pilotage, de manière à ce qu'ajouter une machine revienne à ajouter un module sans remettre en cause le reste du modèle. L'accès rapide aux données, et leur traitement dans les meilleurs délais, conditions sine qua non d'une réelle coopération homme-logiciel, sont assurés par un réseau de données situé en mémoire vive de l'ordinateur. Le support de la coopération homme-logiciel est un diagramme de Gantt interactif permettant aussi bien de visualiser un résultat que d'entrer une action de l'opérateur (déplacement d'opération, augmentation de la capacité d'une ressource, etc.).

3.2 *Sipaplus (GPSA)*

Sipa, version industrielle d'Apis, est un logiciel d'aide à la décision interactif pour la planification à court terme (Seguin et al., 91). Dans sa version actuelle (Sipaplus), il se compose de trois modules principaux : un plan de charge, un module d'ordonnancement et un module de suivi.

Le module d'ordonnancement était seul l'objet du transfert entre le Laboratoire GRAI et la société GPSA. Le module de suivi été immédiatement rajouté pour permettre une utilisation industrielle, tandis que le plan de charge a été développé plus tardivement par GPSA pour satisfaire une demande importante. En effet, le plan de charge permet de traiter de problèmes d'adéquation charge/capacité qui peuvent plus difficilement être abordés efficacement au niveau de l'ordonnancement. La figure 1 montre un écran typique du plan de charge : les profils de charge induits sur les différents centres de charge par le plan de production sont visualisés, conformément à la vocation d'interactivité du logiciel, diverses actions sont possibles pour visualiser la situation (trace d'un ordre de fabrication sur ses postes de charge successifs, pourcentage de la charge induite par l'ordre de fabrication...), puis pour tenter de résoudre les problèmes de surcharge détectés à ce niveau (transferts de charge, augmentation ponctuelle de la capacité, sous-traitance...).

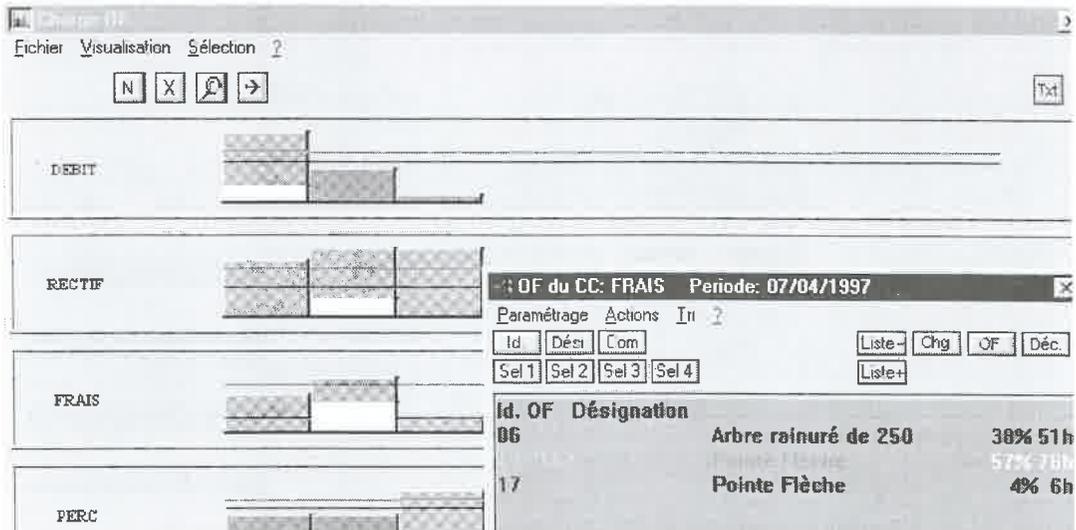


Figure 1 : Le plan de charge de Sipaplus

Lorsque l'adéquation charge/capacité semble correcte, le module ordonnancement peut réaliser un planning plus fin, à capacité finie. Une fonctionnalité importante du module d'ordonnancement de Sipaplus est sa capacité de gestion multi-ressource. Une seule ressource est en fait associée à chaque phase de la gamme, mais le logiciel permet de "forcer" plusieurs phases à se dérouler en parallèle. Il suffit alors d'associer les diverses ressources à gérer à des opérations se déroulant en parallèle pour arriver à coordonner l'utilisation de plusieurs ressources de fabrication lors d'une opération. Lorsqu'un ordonnancement a été effectué, il est visualisé au moyen d'un diagramme de Gantt (voir figure 2). L'interactivité du logiciel se traduit une nouvelle fois par la possibilité de diagnostiquer la situation (visualisation du retard d'un ordre de fabrication, visualisation d'un ordre de fabrication unique...) puis par un nombre important d'actions correctrices possibles au niveau du Gantt. On peut par exemple citer :

- le basculement d'une gamme principale vers une gamme de remplacement,
- le déplacement manuel d'une opération, si cela n'enfreint pas les contraintes décrites dans les données techniques,
- la modification de la capacité des ressources, par modification directe de leur calendrier. On voit ainsi sur la figure 2 une partie de la fenêtre de gestion du calendrier hebdomadaire d'une ressource qui peut être ouverte à partir du Gantt, et permet d'augmenter ponctuellement les heures d'ouverture de la machine.

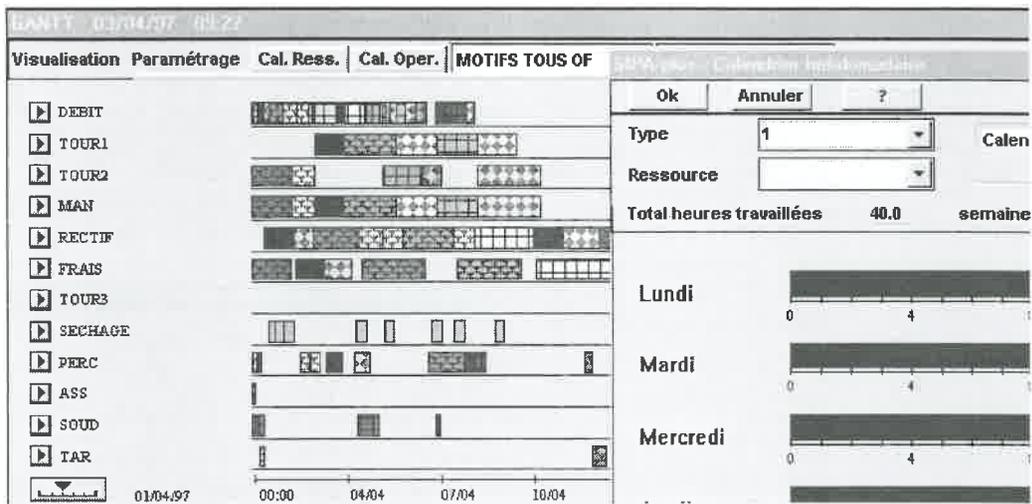


Figure 2 : Diagramme de Gantt de l'ordonnancement de Sipaplus

Sipaplus peut être considéré comme un succès, puisque plus de cent vingt licences ont été vendues à ce jour, ce qui en fait un des logiciels spécialisés en ordonnancement les plus vendus en France.

Sipaplus a repris la majorité des concepts d'Apis, hormis deux notions :

- la notion d'activité, permettant la modélisation du système physique : une activité (exemple : fraisage) peut être réalisée par différents couples (machine, opérateur). L'implantation de cette notion oblige à réécrire les gammes en termes d'activité : cette obligation a semblé inacceptable d'un point de vue commercial. Elle permet pourtant de décrire simplement des degrés de liberté dont la gestion apparaît encore aujourd'hui comme fondamentale, en particulier pour réagir à des aléas.
- la manipulation directe des objets planifiés (ordre de fabrication complet par exemple) dans le cadre d'améliorations manuelles de l'ordonnancement. Cette notion a semblé d'une implantation trop complexe à l'époque.

3.3 IO (CESIUM)

Le développement de Io a été lancé par la société CESIUM en 1995. Io est maintenant commercialisé depuis 1997. Sur le plan de la modélisation de l'atelier, Io utilise la notion de poste de charge, ce qui permet de redonner des degrés de liberté à l'affectation des ressources aux opérations d'une gamme. De plus, de nombreuses fonctionnalités ont été introduites pour permettre une utilisation du logiciel sur un mode coopératif :

- de nombreux modes de visualisation de l'ordonnancement sont proposés à l'utilisateur afin de faciliter sa compréhension de la situation réelle de l'atelier. La figure 3 montre ainsi un mode de visualisation où les ordres de fabrication (et non les ressources) sont proposés dans la partie gauche de la fenêtre : les pavés du Gantt montrent alors l'utilisation des différentes ressources par l'ordre de fabrication. Dans ce mode de visualisation, des opérations peuvent être superposées (du fait du chevauchement par exemple). Un mode "développé" a donc été implanté, visible pour le deuxième et le troisième ordre de fabrication de

la figure 3 : ce mode permet de contrôler facilement le trajet de l'ordre de fabrication dans l'atelier.

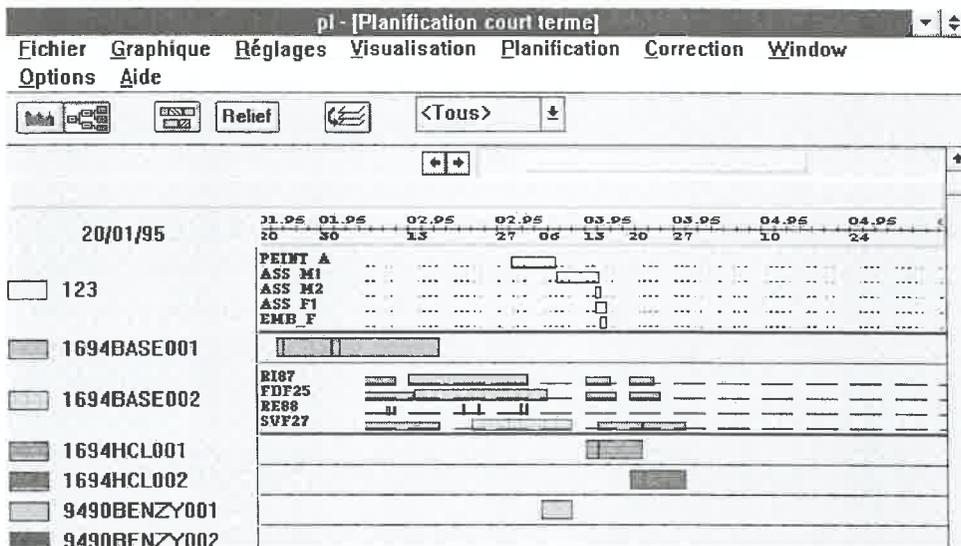


Figure 3 : Gantt par ordre de fabrication dans IO

Un diagramme de Gantt par ressource est lui aussi affichable (cf. figure 4). Des modes de visualisation supplémentaires ont toutefois été ajoutés. On voit ainsi sur la figure 4 que l'utilisation du troisième poste de travail (ASS_M2) est visualisée sous forme d'un plan de charge, ce qui donne des indications complémentaires par rapport au Gantt. De plus, une visualisation de type "en-cours" est elle aussi possible, comme sur le poste GRIG_AG de la figure 4. On peut ainsi voir comment une file d'attente devant une machine est progressivement résorbée.

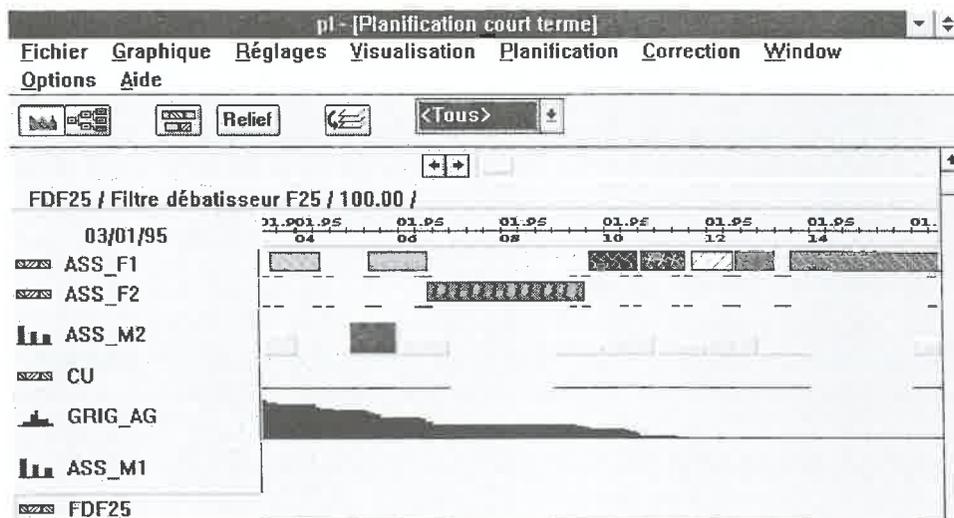


Figure 4 : Gantt par ressource dans IO

IO dispose d'outils classiques permettant une amélioration de l'ordonnancement à partir du Gantt, mais ce qui fait son originalité principale est certainement sa capacité à

réaliser des planifications partielles. A partir d'un planning vide, un nombre quelconque d'ordres de fabrication peuvent être sélectionnés, puis planifiés, les uns après les autres ou simultanément. A tout instant, l'ordonnancement peut être terminé automatiquement par un placement de l'ensemble des ordres de fabrication restants par le logiciel. Cette approche permet de remettre l'utilisateur au cœur de la planification, par un meilleur contrôle du planning généré. L'utilisateur planifie les sous-ensembles d'ordres de fabrication avec des règles qui peuvent être différentes pour chaque sous-ensemble, et une correction manuelle est possible entre deux planifications.

De plus, l'extension de la définition de contraintes classiques de succession entre opérations (chevauchement et tolérance) permet de positionner très précisément deux opérations l'une par rapport à l'autre (Grabot et al., 97a). Des règles plus précises, (par exemple, regroupement sur les outils), peuvent aussi être incorporées. Cette possibilité explique que parmi les premiers clients de IO figurent plusieurs sociétés utilisant des process continus ou semi-continus, pour lesquels les logiciels classiques d'ordonnancement ne permettent pas toujours de satisfaire des contraintes technologiques très strictes (temps maximum entre deux opérations par exemple). Une vingtaine de licences du logiciel ont été vendues depuis sa mise sur le marché. La dernière évolution du produit incorpore un module charge qui permet de gérer finement l'adéquation charge/capacité pour avoir un planning plus précis à court terme, et une gestion de charge à moyen terme.

3.4 TAPAS

TAPAS (*The Almost Perfect Approach to Scheduling*) est davantage une plate-forme de développement de produits d'ordonnancement qu'un produit en tant que tel. Son développement a été initialisé dans le cadre d'un partenariat entre le Laboratoire Génie de Production de l'Ecole Nationale d'Ingénieurs de Tarbes (ENIT), l'ESII (Ecole d'Ingénieurs de San Sebastian) et la société FATRONIK (Espagne). Cette étude est en partie soutenue par la Communauté de Travail des Pyrénées (Pays Basque espagnol et Région Midi-Pyrénées).

FATRONIK commercialise des cellules flexibles destinées à l'enseignement (échelle 1/2) mais fonctionnant réellement, et utilisant des commandes numériques du marché. Ces cellules sont faites de machines hétérogènes (transporteurs, machines d'usinage, stockages) pouvant présenter des particularités difficiles à modéliser pour un logiciel d'ordonnancement courant (convoyeurs de type "carrousel", stockages mobiles, palettes etc.). De plus, les cellules flexibles ont des zones de stockage à capacité finie, et des fractionnements ou regroupements de lots sont possibles.

Le développement du système de pilotage de ces cellules prend typiquement de quatre à cinq mois et une collaboration a été lancée entre FATRONIK et l'ESII afin de concevoir une boîte à outils permettant une assistance au développement des pilotes des machines utilisées (Aguirre et al., 98). Le but du projet TAPAS a été de compléter le système de pilotage par des fonctionnalités d'ordonnancement très modulaires afin de proposer le logiciel le plus adapté à un problème donné (Grabot et al., 97b). Les caractéristiques principales de TAPAS sont les suivantes :

- prise en compte des ressources particulières impliquées dans les cellules flexibles (stockages à capacité finie, transporteurs, palettes, etc.),

- manipulation directe des objets planifiés (planifications/déplanifications partielles des ordres de fabrication),
- gestion des tailles de lot au niveau de chaque opération décrite dans la gamme,
- possibilité de décrire des gammes sous forme de réseau, incluant assemblages/désassemblages, mais aussi degrés de liberté pour effectuer des opérations,
- génération du logiciel à partir d'un choix parmi des modules génériques, puis de leur particularisation.

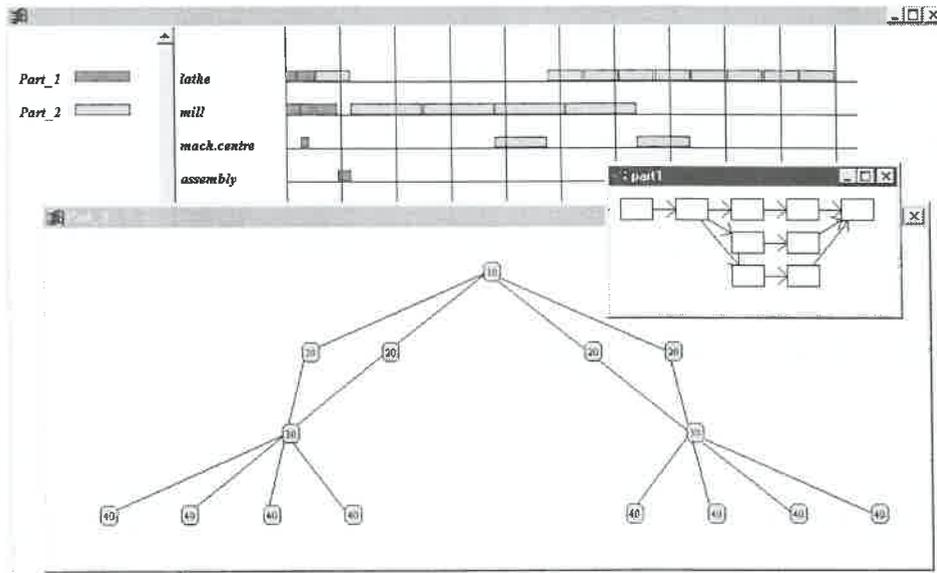


Figure 5 : Exemple d'interface de TAPAS

Sur la figure 5 sont illustrées certaines des originalités de TAPAS. L'ordre de fabrication "part_1" est composé de trois pièces. La première et la troisième sont usinées sur le tour, puis sur la fraiseuse. La seconde utilise la fraiseuse, puis le centre d'usinage. Leur assemblage est ensuite réalisé sur un poste d'assemblage. La gamme réseau qui décrit l'usinage de ces pièces, puis leur assemblage, est visualisée dans la petite fenêtre au centre de la figure 5. Les gammes peuvent ainsi être visualisées, mais aussi décrites graphiquement. L'ordre de fabrication "part_2" illustre les effets de variations de tailles des lots en usinage; il est composé de huit pièces. La première opération, qui utilise le tour, n'a pas de "granularité". Les huit pièces sont donc effectuées dans le même lot, et une seule opération est visible sur le Gantt. La deuxième opération a une granularité de 2 : les 8 pièces sont traitées en 4 lots. La troisième opération, sur le centre d'usinage, a une granularité de 4 : 2 lots peuvent être distingués sur le diagramme de Gantt. Enfin, la quatrième opération a une granularité de 1 : l'usinage des huit pièces se déroule en 8 opérations distinctes. Pour visualiser l'évolution des tailles de lots, un graphe, visible sur la figure 5, est construit et peut être affiché à tout instant.

TAPAS produit un fichier décrivant l'ordonnancement au format STEP 213 (ISO, 94), et est directement relié au système de pilotage des ressources physiques avec un mode de communication conforme au standard CORBA : c'est en fait un serveur MMS (Manufacturing

Message Specification, ISO, 88) comme les autres modules du système de pilotage, et le lancement peut ainsi demander un nouvel ordonnancement et l'interpréter pour lancer les opérations de fabrication tandis que le module de suivi renvoie l'état de l'atelier sous forme de message MMS. Des expériences à distance de contrôle d'une cellule de fabrication au moyen du couple TAPAS/MCSTools par l'intermédiaire du réseau Internet ont d'ailleurs été réalisées.

Du point de vue du développement, TAPAS a été développé sous UNIX à partir de bibliothèques graphiques permettant sa portabilité aussi bien dans l'environnement PC (Windows 95 ou Windows NT) que dans l'environnement Macintosh (au prix de quelques efforts supplémentaires...). Plusieurs expériences d'implantation réelle de TAPAS et de MCSTools sont en cours : sur la cellule flexible de l'ESII à San Sebastian, sur une cellule flexible à l'Université Panaméricaine de Mexico, et sur une cellule de démonstration de FATRONIK. TAPAS ne peut pourtant pas être considéré pour l'instant comme un produit industriel, mais plutôt comme un démonstrateur avancé.

Les divers types de collaboration Recherche/Entreprise ayant permis les développements évoqués sont précisés dans la troisième partie.

4. Types de coopération et leçons

Dans chacun de ces cas, il est intéressant d'analyser le rôle des acteurs humains et celui des "institutions". Loin de prétendre à l'exhaustivité, cette partie dégage sur ces exemples quelques avantages et quelques inconvénients ainsi que l'apport de chacun dans le projet.

4.1 *Apis-Sipa : le transfert*

Le principe de base d'un transfert Recherche/industrie est qu'un établissement de recherche n'a ni la vocation ni la capacité technique à concevoir un produit immédiatement commercialisable. Le transfert d'Apis du Laboratoire GRAI à la société GRAI Productique est un transfert contractuel entre deux institutions. Un contrat de cession et de royalties a été signé entre les deux parties. Les deux institutions ont donc eu des rôles successifs, avec toutefois une période importante de chevauchement lors du passage de la version laboratoire à la version industrielle du produit. La société GPSA a ainsi repris et re-développé un logiciel déjà arrivé à une certaine maturité, au moins en ce qui concerne ses fonctionnalités principales.

Les institutions ne sont rien sans les hommes qui les animent. Et, dans ce cas, le caractère "obstiné" du chercheur et la "fougue" du directeur du laboratoire ont été nécessaires pour convaincre les industriels que ce nouveau type de logiciel d'ordonnancement était utile. La collaboration étroite entre les deux structures a permis de poursuivre le travail à partir de l'expérimentation du logiciel en grandeur réelle (projet Esprit 418 (Milin, 87)). Ces échanges n'ont pas toujours abouti à de nouveaux transferts mais ils ont permis aux deux organismes de capitaliser des savoir-faire importants.

Dans le cas d'un transfert (Bérard, 95), le premier travail du laboratoire de recherche est de convaincre la direction de l'entreprise que le logiciel est économiquement viable. Puis, il

est nécessaire de motiver le personnel de l'entreprise, de l'"accrocher" au futur produit. Il s'agit presque d'un transfert "affectif" qui va conduire le personnel de la société à s'approprier le logiciel comme le sien et qui doit faire accepter au chercheur de laisser partir son "enfant" pour qu'il se développe... ou, pire, pour qu'il change ! Enfin, il faut former le personnel de l'entreprise afin qu'il comprenne bien les principes fondamentaux du produit et prenne conscience que pour que ce résultat de recherche reste novateur, il doit continuer à collaborer avec des structures de recherche.

Les conditions pour qu'un tel type de collaboration fonctionne nous paraissent être les suivantes :

1. *Le prototype universitaire correspond à un vrai créneau industriel.* Cette condition suppose une bonne connaissance des produits disponibles sur le marché de la part des chercheurs, mais aussi une certaine connaissance des besoins potentiels des utilisateurs qui est beaucoup plus difficile à acquérir pour des universitaires.

2. *Les chercheurs acceptent de voir leur produit changer.* Un développement universitaire n'est jamais un produit. Les chercheurs doivent donc être capables d'accepter que leur produit soit modifié pour devenir plus accessible à ses utilisateurs potentiels, pour être maintenu plus facilement, ou pour accroître ses performances (temps de calcul par exemple).

3. *Les développeurs ont bien compris les originalités du prototype.* Un prototype industriel est toujours simplifié pour être industrialisé; il est ainsi fréquent de voir un logiciel universitaire perdre certaines de ses fonctionnalités pour gagner au niveau fiabilité ou interfaces lors de son passage au stade industriel. Cette simplification doit toutefois respecter ce qui fait la nouveauté et l'intérêt de l'approche proposée. En fait, on peut constater que les partenaires industriels ont souvent tendance à vouloir satisfaire un besoin existant, tandis que le chercheur voudra aller au delà des besoins exprimés pour modifier les manières de travailler. Dans le cadre de Sipaplus, l'interactivité avait été clairement identifiée comme un créneau du logiciel, et a donc été complètement implantée malgré la lourdeur des développements impliqués (interfaces graphiques actives, etc.). Par contre, toute fonctionnalité impliquant un changement des habitudes des clients a été refusée par le partenaire industriel, dans un souci de prudence.

4. *Le dernier mot est au vendeur.* Les discussions sont nombreuses dans le cadre d'un transfert, et il est naturel que chacun défende son point de vue avec acharnement. Lors de blocages éventuels, il semble toutefois de bon sens de laisser le dernier mot aux industriels, qui ont le contact avec le client et dont l'avenir professionnel dépend largement de celui de leur produit. Les chercheurs, eux-mêmes fonctionnaires, auraient beau jeu de demander à leurs partenaires de prendre des risques qu'eux-mêmes n'assument pas totalement...

4.2 *Io : la collaboration*

Le cas du développement de Io est très différent du précédent puisque chercheurs et industriels ont travaillé ensemble dès le début du projet (phase de spécification du logiciel) sans implication des laboratoires universitaires en tant que tels. Les personnes qui ont participé à la conception de Io avaient toutes une expérience importante de l'aide à la décision pour l'ordonnancement. Cette expérience avait été acquise pour certains sur Sipaplus, mais aussi sur d'autres logiciels du marché et par des développements de recherche

variés. Le travail de spécification a été conduit par la société Césium, mais il s'est nourri d'une série de brainstormings entre les auteurs de ce papier et la PDG de Césium. Au cours de ces brainstormings, chaque individu, selon son domaine de compétence (ordonnancement industriel, évaluation de performances, algorithmes, structures de données, interfaces Homme-Machine), a alimenté la synthèse des besoins industriels et des solutions possibles pour les satisfaire. La qualité du logiciel obtenu est liée à celle du dialogue qui a permis de sélectionner les meilleures idées, de les assembler et de les concrétiser dans un logiciel. Des rôles typiques semblent avoir été alternativement joués par les différents acteurs de ces brainstormings, avec quelques retournements spectaculaires :

- le Responsable Produit, qui s'inquiète non seulement des fonctionnalités qui seront proposées au client, mais aussi des versions suivantes du logiciel (gestion de l'évolution du produit), en faisant le tri entre nécessaire, intéressant et superflu,
- le Développeur, inquiet de la faisabilité des idées échangées et de leurs conséquences sur le temps de traitement du logiciel,
- le "Chercheur Fou", désireux de tester des idées de recherche,
- le Représentant du Client, qui a en tête le produit qu'il aimerait qu'on lui propose s'il était un utilisateur.

Il est à noter que la société Césium valide en final le logiciel à l'aide de deux sites de test industriel, ce qui permet de rester dans des problématiques réalistes.

Ce travail continue actuellement avec des extensions des fonctionnalités du logiciel, mais n'implique pas officiellement les deux structures de recherche : il n'a en effet pas de cadre contractuel. L'intérêt des chercheurs est alors centré sur la possibilité de valider des idées théoriques et sur le retour d'expérience permis par un cadre industriel, avec les possibilités de publication scientifique qui en découlent. L'intérêt de l'entreprise est de diversifier sa connaissance des besoins du marché et des logiciels d'ordonnancement existants, tout en étant sollicitée par des idées nouvelles issues de la recherche.

Dans la collaboration pour Io, le succès dépend d'abord d'une ferme conviction de l'entreprise de travailler avec des personnes physiques plus qu'avec un laboratoire particulier. D'autre part, il requiert une forte implication personnelle des chercheurs. Dans ce cas particulier, la collaboration est facilitée par le fait que les acteurs se sont connus dans le cadre du transfert précédent et qu'ils ont la même "culture".

Dans le cadre d'une "collaboration" comme celle qui a donné naissance à IO, les points suivants nous paraissent être importants :

1. *Les chercheurs doivent avoir l'impression d'être écoutés.* Chacun comprend que toutes les idées ne peuvent pas être intégrées (au moins immédiatement) à un logiciel qui doit concilier sa première implantation sur le marché avec sa pérennité à moyen et à long terme. Néanmoins, les chercheurs doivent sentir un cadre réceptif au test et à la validation de leurs idées de recherche. Dans le cas contraire, leur participation n'a naturellement plus lieu d'être.

2. *Le but du projet (création d'un produit) doit être clair pour chacun.* En complément de l'idée précédente, il doit être clair que tout aspect innovant n'a d'autre but que de conférer un

"plus" concurrentiel au produit. Le projet faisant l'objet de la collaboration ne doit pas pas être ressenti comme un acte de recherche en soi.

3. *Les impératifs de confidentialité ne doivent pas empêcher les publications.* Il est vital pour la collaboration de concilier les impératifs de publication des chercheurs avec les impératifs de confidentialité de l'entreprise. Par expérience, un terrain d'entente peut toujours être trouvé dans la mesure où rien de ce qui est visible dans le logiciel n'est en fait confidentiel : c'est en général le savoir-faire attaché à l'implantation des concepts dans le logiciel qui doit être protégé.

4. *Une nouvelle fois, le dernier mot doit être laissé aux industriels...* Cela est encore plus vrai dans le cas de la collaboration que dans le cas du transfert : lors d'un transfert, la connaissance originelle appartient au chercheur, qui a toujours tendance à croire que l'industriel rejette une idée parce qu'il ne l'a pas comprise. Lors d'une collaboration, la mise au point des idées est conjointe et la paternité d'un concept implanté dans le logiciel est pratiquement impossible à retrouver. En cas de désaccord, il est pourtant impossible de ne pas laisser aux personnes qui engagent leur avenir professionnel la maîtrise finale du produit.

4.3 Tapas : la complémentarité

Dans le cas du développement de TAPAS, la partie "lancement" et "pilotage des ressources" avait déjà été développée par l'ESII en collaboration avec FATRONIK lorsque le projet tripartite avait débuté. Par rapport à l'ordonnancement, FATRONIK pouvait exprimer des besoins-clients élémentaires, mais n'avait pas de connaissance approfondie sur les possibilités réelles d'un logiciel d'ordonnancement actuel. La connaissance de la problématique "ordonnancement" était donc seulement présente du côté universitaire. Il est évident qu'une grande partie du caractère innovant du projet vient de l'intégration de produits très différents dans une offre unique (ordonnancement, lancement, contrôle) avec une philosophie commune d'intégration de produits modulaires au moyen d'une communication assurée par les standards informatiques les plus actuels.

Dans le cadre du développement de TAPAS, les intervenants du laboratoire ont joué les quatre rôles décrits pour le projet IO. L'équipe de chercheurs s'est pour cela étoffée de développeurs ayant une bonne expérience de développement dans un milieu industriel. Il est clair que le projet n'a été fructueux que grâce aux expériences qui avaient été menées auparavant, qui ont permis de canaliser les innovations implantées dans le produit. Le problème principal a été en fait d'analyser soigneusement les besoins de FATRONIK pour permettre leur satisfaction au moyen de concepts plus génériques, afin de ne pas spécialiser inutilement le logiciel.

L'entreprise impliquée se rapproche davantage, dans le cas évoqué, d'un client final que d'un vendeur de produits d'ordonnancement comme dans les exemples précédents. Tout peut alors sembler permis au laboratoire développeur, puisqu'il détient seul des compétences qui devraient être distribuées. Il faut alors garder un équilibre difficile entre "se faire plaisir" (c'est-à-dire faire une action de recherche, pouvant être inapplicable en pratique à très court terme) et avoir un rôle de société de service pur, qui ne peut pas être satisfaisant pour un chercheur. La confiance entre les partenaires est alors primordiale, puisque les possibilités de

critiquer le travail des autres sont limitées par les compétences techniques de chacun. Débuter par une collaboration de ce type paraît alors dangereux.

Un point important à signaler est que l'implication des institutions (ENIT, ESII) a facilité le montage financier du projet par l'intermédiaire d'aides régionales à la recherche, de part et d'autre des Pyrénées (Gouvernement Basque et Région Midi-Pyrénées). Grâce au cadre de l'Union Européenne, il est d'ailleurs intéressant de constater que la présence de partenaires étrangers semble plutôt constituer un avantage pour accéder à des sources de financement publiques.

Les points-clefs de ce type de collaboration sont à notre avis les suivants :

1. *Chaque partenaire doit avoir atteint un niveau de maturité important dans sa spécialité.* Le manque de compétence commune empêche en effet de bénéficier du regard critique du partenaire sur sa partie du travail.

2. *Les partenaires doivent avoir pleinement confiance les uns dans les autres.* C'est le corollaire du point précédent : il est évident que l'on ne peut se donner pleinement dans le projet si on se pose des questions sur la capacité des autres partenaires à réaliser leur part de travail.

3. *La cible du produit doit être parfaitement connue des partenaires.* Chaque module du logiciel va être en contact avec le client. Il est donc important que les besoins de ce client relatifs aux différents aspects du projet soient connus des différents intervenants. Dans le cas de TAPAS, chaque partenaire avait déjà une bonne connaissance des besoins situés au niveau du pilotage d'atelier et des cellules flexibles de fabrication.

5. Les leçons

Soeren KIERKEGAARD disait, à propos de l'aide et du conseil aux autres : "Si vous devez aider quelqu'un, vous devez avant tout comprendre ce qu'il comprend et voir la situation telle qu'il la voit". De même, chercheurs et industriels doivent comprendre quels sont les savoir-faire (**S** dans la figure 6) et les objectifs (**O**) de leurs partenaires.

Pour schématiser, le savoir-faire peut concerner la conception du produit **Sc** (savoir-faire de conception) et/ou la fabrication du produit **Sf** (savoir-faire de fabrication). Chaque partenaire doit bien identifier la répartition de ces savoir-faire. Quant aux objectifs **O**, ils peuvent être classés en trois catégories : **Op**, **Of**, **Os**, relatifs respectivement à la validation (preuve), aux finances et au savoir amenés par le projet.

Objectifs	Labo	Entreprises
Op : Preuve	Publications prouvant une faisabilité, satisfaction d'amener ses recherches jusqu'à l'application,	Image de marque
Of : Finance	Royalties, moyen de travailler,	Bénéfices, conquête d'un marché
Os : Savoir	Connaître un terrain expérimental, avoir une expérience pratique	Accroître son savoir-faire à court terme, à long terme, collaboration avec un seul ou plusieurs labos

Table 1 : Structure des objectifs de coopération

Le choix de coopérer est motivé par l'atteinte de ces objectifs. Un partenaire est pleinement satisfait quand les trois types d'objectifs sont atteints : **Op** lui apporte la satisfaction d'avoir eu raison, **Of** lui confirme que cela va jusqu'à une reconnaissance financière, et **Os** que cette expérience lui confère un savoir.

Les contextes des trois expériences évoquées sont très différents. Dans le cadre du transfert, il s'agit pour l'entreprise d'industrialiser un prototype innovant pour le vendre (**Of**). L'objectif de l'entreprise est clair dans la mesure où le produit final a une certaine "visibilité" (un prototype avancé existe déjà). Pour le laboratoire, le transfert d'idées a pour objectif de valider un travail (**Op**); par contre, le transfert de prototype n'a pas d'intérêt pour le chercheur, sauf s'il se prolonge par un retour d'expérience (**Os**).

Il n'en va pas de même dans le deuxième deux cas (collaboration), dont le résultat est beaucoup plus incertain. Comme cela a été évoqué, le point-clef du succès du transfert est à notre avis l'équilibre à trouver entre le respect du caractère innovant du prototype et les impératifs de développement industriel. La collaboration vise à la création en commun d'un produit par des individus aux cultures très différentes. Le résultat est tellement incertain, et dépend tellement des individus engagés, qu'il se prête mal, à notre avis, à une contractualisation des relations entre les intervenants. L'équilibre se maintient si chaque partenaire continue à apprendre (**Os**).

La complémentarité a pour base un regroupement pour réaliser une offre complète. Les éléments constitutifs du produit final existent en général déjà, même à un état embryonnaire, puisqu'une expertise est présente chez les intervenants. On peut donc avoir une idée du produit final, dont la viabilité dépendra en partie des capacités d'intégration des produits et de la bonne compréhension des fonctionnalités désirées pour chaque module. Il s'agit donc avant tout de créer des ponts fiables entre les constituants du produit final : les standards de communication actuels facilitent grandement cette tâche. Chaque partenaire sait que le produit final lui permettra de prouver (**Op**) que son savoir-faire ou sa théorie est utile.

Le tableau 2 tente de présenter une synthèse des cas de coopérations Recherche/Entreprises pour la conception de logiciels d'ordonnement.

Types de coopération	Objectif principal		Savoir-faire		Exemples	Durée
	Labo	Entreprise	Labo	Entreprise		
Transfert d'idées	Op	Os	Sc	Sf,	achat d'une étude ou d'une maquette de laboratoire, seule l'idée du produit Sc est définie	Ponctuel
Transfert prototype	Os si cela dure	Of	Sc+Sf		Le labo a testé la faisabilité et défini le produit complètement (sa fabrication Sf est définie)	Plus long du fait du transfert de Sf
Collaboration	Os	Os	Sc+Sf	Sc+Sf	On ne sait pas qui a la paternité des idées car les acteurs des deux structures ont dès le début des idées sur la conception et sur la fabrication du produit	Très long
Complémentarité	Op	Op	Sc'+Sf'	Sc''+Sf'' avec Sc'' ≠ Sc' et Sf'' ≠ Sf'	Chaque structure a des compétences distinctes de l'autre. Le produit final résulte de l'assemblage de sous-produits	Souvent long

Table 2 : Synthèse des différentes coopérations¹

Enfin, dans tous les cas, la proximité géographique des partenaires nous semble un facteur déterminant du succès de l'entreprise. Tous les partenaires évoqués sont ainsi situés dans un triangle de 200 km de côté, entre Toulouse, Bordeaux et San Sebastian. Bien que, théoriquement, les outils actuels de communication comme les messageries électroniques, Internet, téléphone, fax, etc., permettent de travailler à distance, des problèmes pratiques semblant insolubles se sont presque toujours aplanis après des rencontres physiques, quelquefois de très courte durée. De plus, les moyens de communication "électroniques" sont souvent établis d'individu à individu, et ne se prêtent pas encore au travail de groupe. Il est à ce titre intéressant de constater qu'une réunion avec le partenaire est souvent l'occasion de constater que l'on a perdu de vue l'état du travail de son voisin de bureau !

¹ Sc : Savoir-faire de conception, Sf : Savoir-faire de fabrication, Os : Objectif de savoir-faire, Of : objectif savoir fabriquer.

6. Conclusion

Dans le domaine de l'aide à la décision, un travail de recherche bénéficie grandement de la possibilité de disposer d'un terrain expérimental proche. Celui-ci permet d'évaluer la pertinence des idées émises sur des problèmes réels, mais aussi de se positionner par rapport à une concurrence commerciale. Cela est encore plus vrai dans le cas de l'ordonnancement des systèmes de production car la modélisation complexe et la variabilité de l'environnement font apparaître des problèmes de décision non visibles sur des maquettes de laboratoire.

Un avantage de ce caractère applicatif est qu'il facilite la communication entre les équipes de recherche travaillant sur le même terrain expérimental. Il permet à chaque chercheur d'envisager l'interfaçage (ou la comparaison) avec les travaux des autres. Par contre, un piège serait d'abandonner toute idée non directement exploitable en termes de produit, en privilégiant trop le court terme comme le font beaucoup d'industriels. Pour éviter cela, la présence de personnes des entreprises s'assurant de la faisabilité pratique du projet permet au chercheur de se décharger de ce rôle l'amenant souvent à s'auto-censurer, et lui permet ainsi de défendre plus efficacement ses idées.

Certaines idées peuvent demander plusieurs années avant d'être réellement implantables : la modularité de TAPAS comme sa connexion avec un module de pilotage de ressources, permises en grande partie par l'approche objet utilisée pour son développement et par la maturité actuelle des standards d'échange, peuvent ainsi se rapprocher de la notion de Module de Pilotage déjà présente dans APIS mais jugée trop complexe et abandonnée à l'époque. La ténacité des chercheurs (ou leur entêtement...) peut ainsi permettre de ne pas oublier trop rapidement certaines idées lorsqu'elles arrivent de manière prématurée.

7. Bibliographie

- Aguirre Suarez, O., Ajuria Foronda, J.L., Martin Abreu, F., (1998), "Standard Based Framework for the Development of Manufacturing Control Systems" *International Journal of Computer Integrated Manufacturing*, vol. 11, n°5, pp. 401-415.
- Bérard, Ch., (1983), "Contribution à la conception de structures logicielles pour le pilotage d'atelier", Thèse de Docteur-Ingénieur, Université de Bordeaux I.
- Bérard, Ch., (1995), "APIS/SIPA, Principes et application industrielle", *RAPA*, Hermès, vol.8, n°5, 735-752.
- Bérard, Ch., Deschamps, J-C., Farthouat, P., (1997), "A cooperative approach to schedule manufacturing systems", 7th Mini EURO Conference, Bruges, Belgique, 24-27 Mars.
- Farhoodi F., (1990), "A knowledge approach to dynamic job-shop scheduling", *Int. Journal of Computer Integrated Manufacturing*, vol.3, n°2, pp. 84-95.
- Grabot, B., Bérard, Ch., Nguyen, Ph., (1996), "IO: a practical approach for co-operative scheduling", Workshop on Production Planning and Control, Mons, Belgique, 9-11 septembre.
- Grabot, B., Bérard, Ch., Nguyen, Ph., (1997a), "Généralisation des contraintes pour l'ordonnancement coopératif", *MOSIM'97 : Systèmes de Production et de Logistique*, 5-6 juin, Rouen, France, Hermès, pp. 501, 508.

Grabot B., Geneste L., Moutarlier Ph., Martin F., Bueno R., (1997b), "Software reuse for Production Activity Control: an experiment", 6th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA'97, Los Angeles, USA, 9-12 septembre.

ISO TC184, (1988), *Manufacturing Message Specification*, ISO 9506.

ISO TC184, (1994), *Product Data Representation and Exchange (STEP)*, ISO 10303.

Lopez, P., Haudot, L., Sicard, M., Esquirol, P., (1995), "Constraint-based approach to design a DSS for scheduling", 3rd International Conference on The Practical Application of Prolog (PAP'95), Paris, France, 4-7 avril, pp. 405-422.

Milin, A.M., (1987), "Amélioration des solutions d'ordonnancement pour l'aide au pilotage d'atelier", Thèse d'Automatique, Université de Bordeaux I.

Seguin, L.T.N., Bérard, Ch., Dupeux, A., (1991), "Manuel d'utilisation de SIPA+", Laboratoire GRAI, Université de Bordeaux I.