

Proposition d'un modèle pour Ordonnancement d'un Système Automatisé de Production Applications des algorithmes génétiques hybrides

Djamila Bouhalouan¹, Nassima Aissani¹, Bouziane Beldjilali²

¹ Département d'informatique, Faculté de Sciences, Université d'Oran Es-Senia,
BP 1524, El-M'Naouer, 31000, Oran, Algérie
{jjilibin,aissani.nassima,bouzianebeldjilali}@yahoo.fr

Résumé. Le principe d'un ordonnancement coopératif selon une modélisation multi-agents résulte du rapprochement des domaines de l'Ordonnancement et de l'Intelligence Artificielle Distribuée.

Le travail que nous présentons dans ce papier a pour objectif l'étude de l'adéquation des métaheuristiques avancées dans un SMA dans le cadre de la résolution d'un problème d'optimisation bien défini qui est l'ordonnancement dans les systèmes Flow Shop Hybride (FSH).

Nous avons implémenté un Algorithme Génétique Hybride et étudié sa contribution dans l'amélioration des performances du système de production, et pour l'évolution génétique des agents, des groupes et des organisations.

Le présent article, explique notre démarche pour réaliser un système d'ordonnancement d'un Flow-Shop basé sur un SMA. Une résolution approchée hybride entre les opérateurs génétiques et une recherche locale a été développée pour l'exploration efficace de l'espace de recherche (Intensification), et pour recentrer la communauté des agents autour d'une solution optimale.

Mots Clés : Ordonnancement, Système de Production, Multi-Agents, Algorithmes Génétiques, Recherche Locale.

1 Introduction

Accroître la productivité en réduisant les coûts est, aujourd'hui, un objectif majeur dans toutes les entreprises, les systèmes de production sont caractérisés par leur dynamique et leur imprévisibilité, les tâches souvent de caractère complexe et sont soumises à des contraintes de temps et d'exigence. Alors, le système de production se donne de nouveaux objectifs à atteindre.

Les modèles basés sur les approches distribuées ou décentralisés offrant des capacités d'adaptation et d'auto-organisation en utilisant, le plus souvent, les systèmes multi-agents ont fait leur preuve et ont donné des meilleurs résultats dans ce domaine.

Les métaheuristiques représentent une stratégie efficace pour la recherche des solutions approchées des problèmes d'optimisation combinatoire. L'hybridation de ces premières apparaît comme une alternative pour accélérer la recherche des solutions approchées. Cette hybridation est une issue qui permet d'intensifier la recherche de solutions et s'assurer de rester autour des zones les plus prometteuses.

L'avènement des systèmes multi-agents qui permettent de distribuer les problèmes sur des ensembles auto-organisés vers une gestion locale, favorisant réactivité et comportements émergent peut faciliter la mise en œuvre du pilotage du système de production. En effet, le but principal d'un SMA et de faire collaborer et coopérer un certain nombre d'agents afin de résoudre un problème, nous nous proposons d'introduire au sein des agents les techniques évolutionnistes afin de leur permettre d'évoluer génétiquement dans le temps et déterminer une meilleure solution d'ordonnement satisfaisant un certain nombre de critères.

Dans un contexte perturbé, le problème d'ordonnement peut être envisagé comme un problème de Réordonnement dont l'objectif est de respecter au mieux les plans pré-calculés par un ordonnancement initial, face à des aléas de production ou autres. Nous abordons cet objectif selon une approche de résolution coopérative faisant interagir l'ensemble des entités composant notre système nous conduisant à proposer des mécanismes de Réordonnement coopératif (réactif).

2 Positionnement du problème

Un problème d'ordonnement consiste à affecter des tâches aux ressources et à décider de leur répartition dans le temps, de manière à optimiser un critère ou à trouver un compromis entre plusieurs critères. La résolution du problème d'ordonnement commence par la modélisation du système de fabrication et son environnement. Alors, nous étions amenés à choisir une modélisation à base d'agents intégrant des techniques évolutionnistes.

2.1 État de l'art

Deux voies de résolution des problèmes d'ordonnement sont considérées : donner une solution exacte selon une stratégie connue, ou bien donner des heuristiques qui permettent d'obtenir des solutions approchées ayant un écart raisonnable par rapport à la solution optimale ou par rapport à une borne inférieure calculée. Cependant, L'ordonnement de production a fait l'objet d'un très grand nombre de travaux. De nombreuses approches ont été développées à base d'Algorithmes Génétiques (classiques et hybrides). Nous allons faire le point sur ces réalisations.

Des approches à base d'AG ont été développées dans différents axes comme l'optimisation du lancement des produits, les problèmes à machines parallèles, ordonnancement des placements des tâches sur les jobs, etc. ([3], [16], [22], [19], [5] [9] [17], ces recherches concernent l'utilisation d'une approche génétique classique dans le cadre de l'optimisation des problèmes d'ordonnement des systèmes de production de différents types.

Vu les restrictions posées par les approches citées précédemment, d'autres chercheurs ont pensé à hybrider les AGs à d'autres méthodes telle que le Recuit Simulé, les Fouilles de Données, les Règles de Priorité et les méthodes de Recherche Locale...etc. [15][22][9][7][11][1].

Ces réalisations et travaux par lesquels nous nous sommes inspirés pour développer notre approche hybride de résolution (AG/RL) dans le but de l'amélioration de la performance du système paraissent une bonne tentative que nous avons développée au cours de notre travail.

3. Spécification

3.1 Les systèmes multi agents

Un Système Multi-Agents (SMA), est une structure composée d'un environnement et d'un ensemble d'agents artificiels. Ces derniers sont capables d'agir sur l'environnement et, de collaborer entre eux et / ou avec des agents extérieurs. Les SMA empruntent à l'intelligence artificielle distribuée les modes de communication et de concertation entre agents et reprennent les idées d'autonomie et d'émergence du résultat final à partir des interactions individuelles à la vie artificielle. [6].

Les recherches en SMA sont très actives et des systèmes opérationnels ont été déjà développés dans de nombreux domaines comme le diagnostic, l'enseignement, la conception, contrôle de réseaux de communication, etc. [2], [13], [6], [14], [18],[8], [4], [20], etc. Ces recherches concernent l'utilisation de l'approche multi-agents pour structurer des architectures pour le contrôle des processus. Les principales motivations sont de concevoir des applications (non limitées à un domaine spécifique) pour encapsuler des systèmes experts existants en vue de la coopération. Pour notre part nous avons utilisé l'approche multi-agents pour la mise en œuvre du pilotage d'un système de production (Flow Shop Hybride), nous développons dans cet article la structure de pilotage générique basée sur des agents cognitifs et réactifs.

3.2 Architecture Hétérarchique

L'architecture hiérarchique présente quelques inconvénients qui nous poussent à opter pour les architectures hétérarchiques, en effet, l'architecture distribuée supervisée (hétérarchique) constitue un compromis entre les approches distribuée et supervisée. Ce choix est motivé par les avantages qui caractérisent cette approche et qui se résument dans :

- Les caractéristiques de globalisation (en terme d'optimisation ou d'objectif) qui sont celles des approches hiérarchiques : la présence d'un niveau de supervision fournit une vision et une capacité de prise de décision globale, d'où la centralisation du contrôle ;
- Les caractéristiques associées aux processus d'allocation dynamique des structures distribuées qui lui octroient une capacité satisfaisante en terme de réactivité et de

flexibilité, notamment vis-à-vis les perturbations par la distribution des capacités de décision des agents.

Cette structure a été utilisée et a montré son efficacité, citons par exemple les travaux de [21] et [10].

Dans ce papier, nous allons donner et justifier notre choix réalisé pour le problème d'ordonnancement des ateliers Flow-Shop hybride auquel nous allons appliquer les algorithmes génétiques hybrides au sein d'un système multi-agents organisé selon une architecture hétérarchique.

4. Développement de l'approche

Notre contribution à la résolution des problèmes d'ordonnancement évolue selon deux axes.

Le premier axe concerne l'amélioration d'une des méthodes de résolution approchées de problèmes d'ordonnancement. Notre contribution concerne la modélisation et l'étude de la structure de ces problèmes afin de proposer et mettre en œuvre des méthodes de résolutions des problèmes d'optimisation qualifiées d'hybrides, nous avons implémenté au niveau de l'agent Superviseur une approche génétique hybride et nous avons testé réellement le fruit qu'a donné cette hybridation. L'ordonnancement initial est calculé par l'agent Superviseur dès le lancement des centres de travail la première fois.

Motivation de l'hybridation des AGs avec une méthode de RL

Les AGs et la RL présentent quelques inconvénients qui peuvent être palliés en les hybridant dans une seule architecture. Les AGs sont des algorithmes de recherche globale qui possèdent un parallélisme intrinsèque Néanmoins les AGs ne possèdent pas une preuve de convergence vers l'optimum global, et souffrent aussi du mécanisme de sélection centralisé et global qui rend une population vite homogène et peut conduire de ce fait à une convergence prématurée. La RL, et malgré son caractère séquentiel, peut être combinée à l'AG (recherche globale) pour l'aider à *intensifier* sa recherche dans les zones les plus prometteuses (autour d'une bonne solution), dans le but de l'amélioration de la performance de l'AG.

L'idée principale de cette technique est de rendre plus agressif un algorithme génétique par l'ajout d'une recherche locale en plus de la mutation. Cette recherche sera appliquée à tout nouvel individu obtenu au cours de la recherche [13].

Le second axe concerne l'étude et la résolution de nouveaux problèmes d'ordonnancement issus de la prise en compte des imprévus (aléas de production). En ordonnancement, ces aléas concernent la présence de tâches imprévues (nouveau ordre de fabrication) à réaliser, la disponibilité incertaine des ressources. Pour traiter ces aléas, notre approche consiste à ne pas attribuer la fonction d'ordonnancement à un agent unique comme dans le premier axe (Agent superviseur), mais émerge d'un processus de négociation inter agents, ceci permet de diminuer la charge de l'agent

superviseur. Cette approche suppose que des données sont connues au moment où l'ordonnancement initial est calculé par un premier algorithme, (approche génétique hybride) dit prédictif. Les aléas (ou modification des données) surviennent alors que l'ordonnancement ainsi pré-calculé est partiellement réalisé. Il devient alors nécessaire de déterminer rapidement un nouvel ordonnancement compatible avec les nouvelles données avec une seconde méthode, dit dynamique (réactif). Cette méthode de résolution est un processus de collaboration et négociation des agents interagissant pour résoudre la situation troublée. La qualité de la solution (de la résolution) mesure alors la capacité du processus de négociation à faire face de manière satisfaisante aux aléas.

Par conséquent, nous nous proposons de mettre en évidence la relation qu'il peut y avoir entre les S.M.A. et les A.G. afin de définir une nouvelle propriété des S.M.A. : la notion D'AGENTS GENETIQUES.

4.1 Modélisation

Nous considérons un système de production composé d'un ensemble d'agents coopératifs sous le contrôle d'un superviseur. La figure ci-dessous présente notre architecture retenue où nous distinguons l'agent Superviseur et plusieurs autres agents simples consacrés chacun à un centre de travail (Agent Etage), chacun étant constitué de ses propres ressources (machines, stock, convoyeurs...).

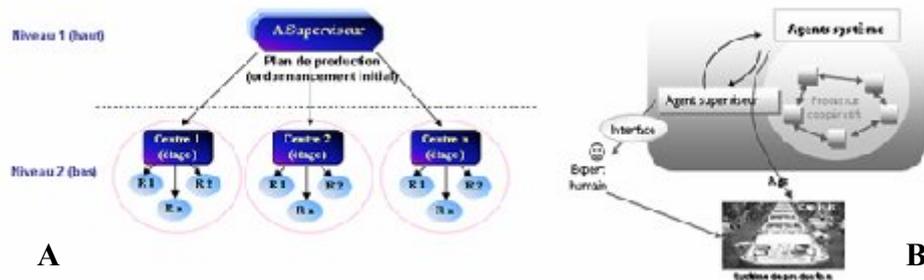


Fig.2. A : Modélisation de notre structure de pilotage par un SMA, B : Architecture globale du système

4.1.2 Prototypage de Résolution et Interaction entre Agents

Pour notre problème d'ordonnancement de production, les agents développés ont pour but de déterminer une meilleure solution (ordonnancement) satisfaisant un nombre plus ou moins important de critères. Toutefois, un modèle de Résolution est implémenté au niveau des agents, fait référence à un processus collaboratif et coopératif faisant interagir ces derniers évoluant au court du temps, concourant et négociant au cours d'un processus de la production pour garantir le bon déroulement de celle-ci, ou pour résoudre une situation troublée (perturbée) dans le cas d'avènement des événements inattendus. Dans la plupart des cas, une perturbation détectée au 2eme niveau décisionnel (machine) sera traitée au mieux au niveau

quasiment plus haut (le centre (étage) concerné) qui fait intervenir ses ressources, sous la supervision de l'agent du 1^{er} niveau, le superviseur.

Notre système multi-agents est conçu pour assurer le contrôle de l'activité de production dans un FSH, le système est distribué aussi bien physiquement (vue la répartition du production sur plusieurs centres de travail (étages)), que fonctionnellement (il est constitué d'un ensemble d'agents cognitifs et réactifs) qui coopèrent pour accomplir des fonctions d'ordonnancement et de contrôle. Les agents utilisent une stratégie d'ordonnancement hybride (préventive [statique 1]/réactive [dynamique 2]), basée sur l'utilisation respectivement de l'approche génétique hybride au niveau de l'agent Superviseur pour le 1er cas, et un protocole de coopération émergeant de l'interaction entre agents pour le 2eme cas.

4.1.4 Optimisation

4.1.4.1 Sélection et usage des algorithmes génétiques hybrides

De manière générale, les agents sont des entités distribuées communiquant pour collaborer à un même but commun : trouver une meilleure solution à un problème d'ordonnancement de type Flow Shop Hybride. Ainsi, l'utilisation de la notion d'évolution, par l'introduction d'algorithmes génétiques dans le but de simuler un processus darwinien est un moyen de fournir des solutions plus adaptées pour atteindre ce but.

L'utilisation des AG dans l'agent superviseur ne confère pas une transmission d'informations génétiques à un descendant. Mais elle fait référence à un travail où l'agent augmente ses performances en fonction de la qualité de l'ordonnancement calculé utilisant les AG et en fonction des informations qu'il reçoit lors des communications avec les autres agents. A ce stade là, on peut dire que par l'introduction d'une génétique d'agent, nous avons montré la possibilité pour qu'un système tende vers une bonne solution au sein d'une organisation d'agent.

En générale l'extension progressive AG → AGH est relativement simple par ce que nous allons garder tous les principes d'un AG et nous allons simplement ajouter une recherche locale dans l'algorithme, le critère à minimiser dans notre cas étant le Cmax.

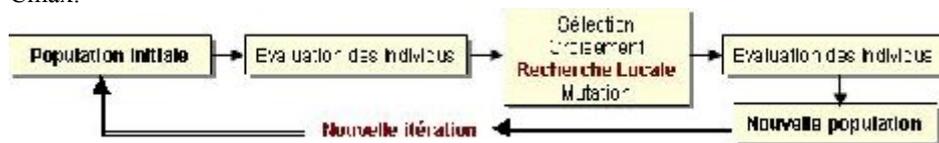


Fig.4. Organigramme de l'algorithme génétique hybride

A chaque itération, deux solutions $P1$ et $P2$ sont choisies aléatoirement dans la population, ces deux solutions-parents subissent un croisement. La solution (enfant) obtenue C est évaluée optimalement. Vient après l'inclusion d'une procédure de recherche locale comme agent d'intensification, dans le but d'améliorer significativement les résultats, et on applique une mutation. La recherche locale pour

notre cas est appliquée à tout nouvel enfant C avec une probabilité fixée P_m . L'itération courante de l'algorithme se termine en remplaçant la solution courante, l'individu le plus mauvais (i.e., le plus coûteux) de la population, par l'enfant C à condition qu'il vérifie la condition de diversité.

4.2.5.1 La recherche locale

L'inclusion d'une recherche locale dans une métaheuristique permet d'intensifier la recherche autour d'une solution donnée. Une recherche locale nécessite la définition d'un voisinage $N(x)$ pour toute solution x . En pratique, $N(x)$ contient un petit ensemble de solutions dérivées de x par des transformations simples appelées mouvements. La procédure cherche une solution x' meilleure que x dans le voisinage $N(x)$ de la solution actuelle x . Elle stoppe si x' n'est pas trouvée : x est alors localement optimal dans son voisinage. Sinon, x est remplacé par x' et on itère le processus.

Les systèmes de voisinage classique pour l'ordonnement dans un problème de Flow Shop Hybride sont la permutation et l'insertion. Nous proposons d'utiliser ce type de système de voisinage pour un ordonnancement σ en entrée.

Le problème traité est celui de l'ordonnement des N pièces en entrée du système et de l'affectation de ces pièces sur les machines. On recherche un ordonnancement et une affectation qui optimisent un critère de performance. Le critère à optimiser (minimiser) est le C_{max} . Un codage spécifique de solutions est attribué à cette classe des FSH. Le génome de chaque solution est composé de deux chromosomes. L'un détermine l'affectation des jobs sur les machines et l'autre, l'ordre des jobs sur ces machines.

5 Expérimentations numériques

5.1 L'environnement expérimental

Les expérimentations ont été réalisées sur un PC standard, équipé d'un Pentium IV 3GHz, 512 Mo de RAM, fonctionnant sous Windows XP version 2002 SP2.

Le modèle du système multi-agents est implémenté sur la plate forme SMA **JADE** réalisée en Java, les agents que nous avons créés sont inspirés du modèle Agent Management Reference proposé par FIPA. Ce modèle établit les règles normatives qui permettent à une société d'agents d'inter-opérer.

Afin de mieux cerner la portée de l'application des Algorithmes génétiques hybrides aux problèmes d'ordonnement de la production et la nécessité d'améliorer les Algorithmes génétiques classiques, il convient de faire des expérimentation sur l'évolution du système dans ce champs, nous passerons après à une simulation globale du système réalisé en prenant en compte les aléas.

Pour mieux étudier le problème d'ordonnement du FSH nous allons considérer deux problèmes du FSH et pour chacun nous varions le nombre de pièces à

ordonnancer. Le premier est FH2 (P3, P2)||Cmax et le deuxième FH3 (P4, P2, P3) ||Cmax (Figure 5). Le FH2 contient deux étages le 1^{er} est constitué de trois machines identiques et parallèles et le 2^{ème} contient deux machines identiques et parallèles, le FH3 contient trois étages, le 1^{er} étant composé de quatre machines identiques et parallèles, le 2^{ème}, deux machines identiques et parallèles et le 3^{ème} est composé de trois machines identiques et parallèles.

5.2 Application de l'AGH sur le FSH1 : FH2 (P3, P2) ||Cmax et FSH2 : FH3 (P4, P2, P3) ||Cmax

Pour la simulation nous changeons pour chaque problème (nombre de pièces) la méthode de recherche locale utilisée et nous prenons la moyenne des dix essais. Les résultats du Cmax sont donnés par le graphe illustré par la figure 6 suivante :

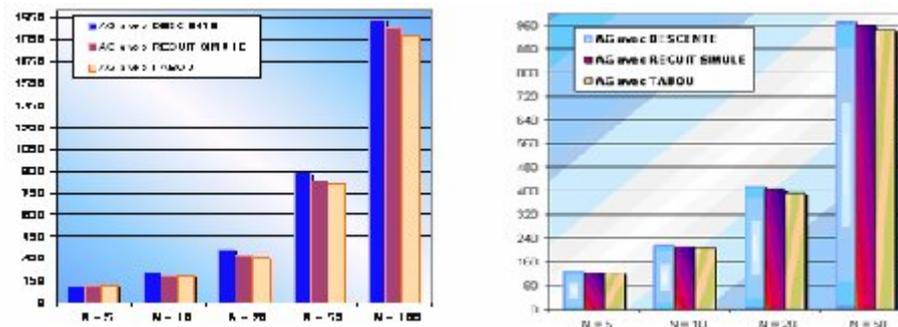


Fig. 6. Résultat de la variation du Cmax pour les FH2, FH3

Discussion

D'après les graphes ci-dessus obtenus pour les jeux de tests sur les deux types du FSH et pour les différents nombres de pièces nous constatons une différence entre les résultats obtenus par les trois méthodes de recherche locale. Pour le nombre de pièces $N = 5$ la différence est pratiquement nulle mais elle devient importante lorsque ce nombre dépasse 10 pièces à ordonnancer. Donc nous pouvons utiliser la descente pour des problèmes de petite taille et pour des grands problèmes nous conseillons d'utiliser soit le Recuit simulé soit la recherche tabou qui ont prouvé leur efficacité pour des grands problèmes.

5.3 Comparaison entre l'AG +(descente/Recuit simulé/recherche taboue) et l'AG

Nous avons pris le premier FSH qui est FH2 (P3, P2)||Cmax, à chaque itération de simulation nous appliquons les deux algorithmes : l'AGH avec la recherche locale RL, l'AG et nous prenons les résultats donnés et ceci pour les différents nombre de pièces à ordonnancer (5,10, 20, 50, 100). Nous fixons la recherche locale de l'AGH par l'une des trois méthodes (descente/ recuit simulé/ recherche taboue) et nous

appliquons les deux algorithmes : AG avec une des méthode de la RL, Algorithme Génétique (AG), les résultats sont donnés par le graphe illustré par la figure 7 suivante :

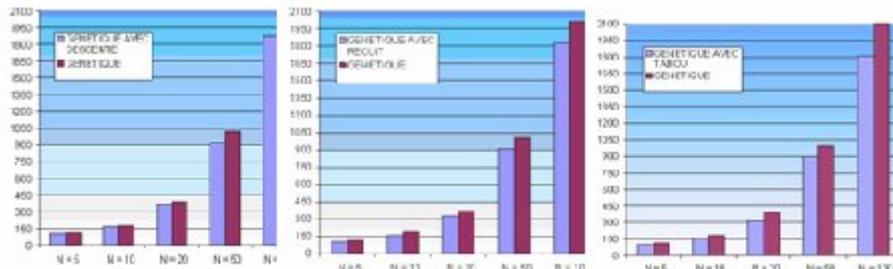


Fig. 7. Résultat du comparaison de l'AG et l'AG + (Descente/Recuit simulé/Recherche taboue)

Discussion

Nous constatons que les résultats obtenus l'AGH avec la Recherche taboue sont nettement mieux par rapport aux autres et cela explique la puissance de la RT.

De ces constatations, nous pouvons dire que l'hybridation des algorithmes génétiques avec des méthodes de recherche locale améliore clairement les performances des premiers en terme de solution.

Nous passons maintenant à présenter quelques exemples de scénarios de processus décisionnel développé ainsi que les résultats obtenus par la maquette informatique réalisée.

Rappelons que notre structure de pilotage est composée d'un agent Superviseur, Etage, Machine, Pièce, Perturbateur, Convoyeur et Stock.

Soit un Flow Shop Hybride à deux étages avec des stocks de capacité illimitée. Le 1^{er} étage est composé de trois machines identiques en parallèle et le second de quatre machines identiques en parallèle :

Calcul d'un ordonnancement prédictif

Cette phase déclenche le comportement qui concerne la recherche de la meilleure machine pouvant exécuter les tâches, et ceci en faisant appel au module génétique implémenté dans l'agent Superviseur (comme a été montré) pour déterminer les meilleurs ordonnancements et affectations des pièces aux différentes machines.

Calcul d'un Réordonnement- réactif

Ce plan contrôle le bon déroulement du processus en cours. Il est activé dès la détection d'une perturbation, l'ordonnement émerge d'un processus coopératif des gents interagissant pour résoudre la situation troublée et préserver dans la mesure du possible une partie de l'ordonnement pré-calculé.

Point de vue général de la résolution

La figure ci-dessous décrit les grandes étapes par lesquelles passent les agents. Cette figure permet d'identifier les points d'entrée et de sortie des flux de communication entre les agents ainsi que les actions effectuées par ces mêmes agents pour traiter les perturbations. En voici un extrait de ce processus interactif :

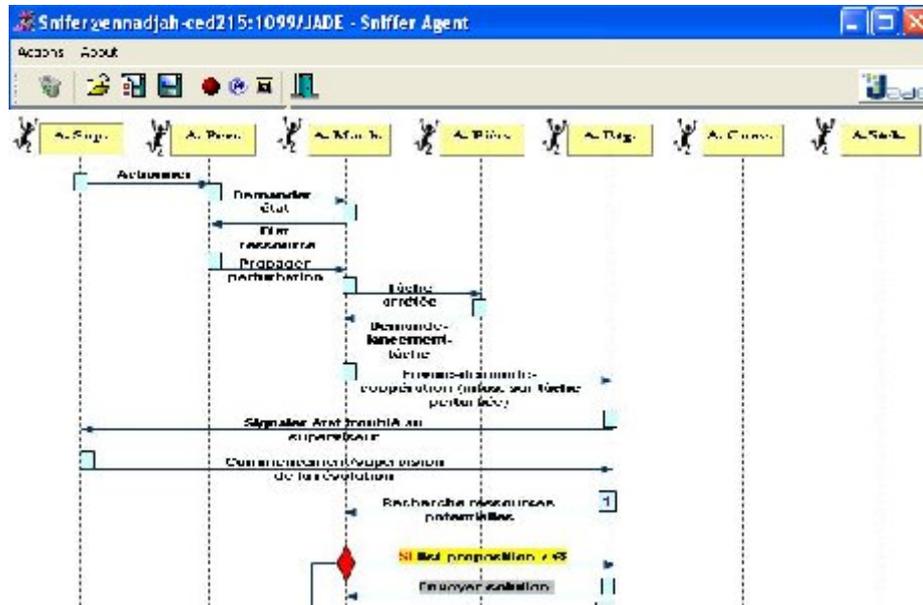


Fig. 8. Exemple de comportements des agents lors de l'interaction

7. Conclusion et perspectives

Notre objectif était d'élaborer une modélisation multi-agents GENETIQUE d'un FS de telle façon que les décisions prises par le système soient le résultat d'un travail de groupe d'agents dont l'organisation est en développement continu. Nous avons utilisés les métaheuristiques AGs et RL qui sont deux métaheuristiques avec des propriétés complémentaires qui peuvent être combinées pour remédier aux inconvénients de leur utilisation individuelle. Ainsi, le fait d'avoir introduit une génétique d'agents dans notre système nous a permis de montrer l'importance d'une évolution génétique des agents pour atteindre une solution optimale dont il faut trouver un ordonnancement des jobs et une affectation de ces jobs aux ressources de façon à optimiser un critère de performance.

Nous pensons à l'utilisation des systèmes distribués qui offrent des techniques de calculs performantes telles que les grilles de calcul. Ou encore l'utilisation du concept du parallélisme des AGH, et nous pensons également à l'utilisation de l'apprentissage par renforcement pour une optimisation en temps d'apprentissage et en qualité.

Références

1. Arroub, M., Brahimi, N., Daste, D., Essafi, I., Guéret-Jussien, C., Kadrou, Y., Lahlou, C., Najid, N., problèmes critiques d'optimisation et d'aide à la décision dans les systèmes de production : Problèmes d'ordonnancement à une machine, Institut de Recherche en Communications et en Cybernétique de Nantes. (2007)
2. Burg, B., Arlabose, F., ARCHON, une plate forme industrielle pour l'IAD, 2eme journées francophones intelligence artificielle distribuée et système multi-agents, Voiron, p.223--234 mai (1994).
3. Baudet P., Azzaro-Pantel C., Domenech S., Pibouleau L., A discrete-event simulation model for batch chemical plant scheduling, ADEDOPS Workshop, Imperial College, Londres, 10-11 Avril (1995).
4. Coudert, T., Archimed, B., Grabot, B., un système multi-agents pour la coopération production/maintenance. MOSIM'99, Annency-France, 6--8 octobre, (1999).
5. Eom, D. H., Shin, H. J., Kwun, I. H., Shim, J. K., et Kim, S. S. Scheduling jobs on parallel machines with sequence-dependant family set-up times. The international journal of advanced manufacturing technology, 19 : 926–932. (2002).
6. Ferber, J, les systèmes multi-agents (vers une intelligence collective) InterEdition, Paris, (1995).
7. Fowler, J.W., Horng, S. M., et Cochran, J. K. A hybridized genetic algorithm to solve parallel machine scheduling problems with sequence dependent set-up. International journal of industrial engineering, 10(3) : 232–243. (2003).
8. Gauardères, E., Application de l'approche multi-agents au pilotage d'une cellule flexible de production, 3rd International Industrial Engineerinf Conference, Proceeding Vol 3, Montréal, Québec, p. 1695—1704, mai (1999).
9. Harrath, Y .,: Algorithmes génétiques et fouille de données pour un ordonnancement réactif dans un atelier de type job-shop.(2002).
10. Kouiss K., Pierreval H.et Merbaki N., « *Toward the use of a multi-agent approach to the dynamic scheduling of flexible manufacturing systems* », Références bibliographiques Modélisation pour la simulation d'un système d'aide au pilotage industriel p.222 International Conference on Industrial Engineering and Production Management (IEPM'95), Marrakech, Maroc, pp. 118--125, avril , (1995).
11. Mahdi, A., Utilisation de métaheuristiques hybrides pour la résolution de problèmes d'agencement d'atelier, de découpe 2D et d'ordonnancement, (2007).
12. Portmann. M-C, 01-04, Antony Vignier: Algorithmes génétiques en ordonnancement.
13. Ramos, C., An architecture and a negociation protocol for the dynamic scheduling systems, IEEE, p 1050—4729, (1994).
14. Roy, D., Vernadat, F., Reactive Shop-Floor Control with multi-agent System. IFAC/IFIP Int Conf. on management and control of production and logistics, Brazil, p. 426—431, (1997)
15. Sauer, W., Weigert, G., and Hampel, D. An open optimization system for controlling of manufacturing processes. In 7th International Conference FAIM .97, Flexible Automation & Intelligent Manufacturing, Middlesbrough, England, June 1997, p. 261-268. (1997).
16. Serifoglu, F. et Ulusoy, G. Parallel machine scheduling with earliness and tardiness penalties. Computers and Operations Research. (1999).
17. Sevaux.M : Métaheuristiques : Stratégies pour l'optimisation de la production de biens et de services. Habilitation à diriger des recherches. Laboratoire d'Automatique, de Mécanique d'informatique Industrielles et Humaines du CNRS (UMR CNRS 8530), (2004).
18. Shen ,W., Norrie, D.H., An agent based approach for dynamic manufacturing scheduling. In Workshop at autonomous agents.(1998).

19. Thomas.H, : Utilisation d'algorithmes génétiques lors de la génération de séquences d'usinage. (2005).
20. Tranvouez, E., Espinasse B., Ferrarini, A., Réqolution coopérative et distribuée des problèmes : une application multi-agents au réordonnancement d'atelier, 3rd International Industrial Engineering confernece, Proceeding Vol.3, p. 1543--1552, Mai (1999).
21. Trentesaux D., « *Conception d'un système de pilotage distribué, supervisé et multicritère pour les systèmes automatisés de production* », Thèse de doctorat en Automatique-Productique, INP Grenoble, janvier (1996).
22. Weigert, G., Hampel, D., and Sauer, W. Scheduling of FMS - Application of an open optimization system. 8th International Conference FAIM.98, Flexible Automation & Intelligent Manufacturing, Portland/Oregon, USA, proceedings pp. 771-780. (1998).