HAL
open science

Planification multi-agent par cycles de conjecture/réfutation

Xavier Clerc, Damien Pellier, Humbert Fiorino

To cite this version:

Xavier Clerc, Damien Pellier, Humbert Fiorino. Planification multi-agent par cycles de conjecture/réfutation. Revue des Sciences et Technologies de l'Information - Série TSI : Technique et Science Informatiques, 2003, 22 (4), pp.129-141. hal-00982578

HAL Id: hal-00982578

https://inria.hal.science/hal-00982578v1

Submitted on 24 Apr 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Planification multi-agent par cycles de conjecture/réfutation

Xavier Clerc — Damien Pellier — Humbert Fiorino

Laboratoire Leibniz (CNRS - INPG - IMAG)
Équipe MAGMA - Bât D
46, Avenue Félix Viallet
F-38031 Grenoble Cedex

{Xavier.Clerc, Damien.Pellier, Humbert.Fiorino}@imag.fr

RÉSUMÉ. *La coordination de systèmes autonomes est une problématique centrale des systèmes multi-agents. On constate une forte dichotomie des modèles proposés entre ceux issus de la planification distribuée et ceux issus des modèles cognitifs. Ces différents modèles se révèlent insatisfaisants ; les premiers par absence de la notion de coopération et les seconds par la difficulté à les rendre opératoires. Dans cet article, nous proposons un modèle de planification distribuée dans lequel la construction du plan est réalisée par un processus coopératif reposant sur le concept de cycles de conjecture/réfutation.*

ABSTRACT. *Coordination of autonomous systems is a central concern of multi-agent systems. There is a strong dichotomy within existing models between those coming from distributed planning and those coming from cognitive models. Both turn out to be unsatistying ; the first ones ingnore the notion of cooperation, and the second ones are difficult to implement. In this paper, we propose a model for distributed planning in which the building of a plan is done by a cooperative dynamic relying on conjecture/refutation cycles.*

MOTS-CLÉS : *planification multi-agent, cycles de conjectures/refutation, jeux d'échecs.*

KEYWORDS: *multi-agent, planning, conjecture/refutation process, chess.*

1re soumission à *JFSMA'03*, le 13 mai 2003.

1. Introduction

Ces dernières années s'est développé un intérêt croissant pour l'utilisation d'approches multi-agents dans le cadre d'applications traitant de problèmes intrinsèquement distribués que ce soit pour de la composition de services en ligne (*e.g.* l'élaboration de voyage [CAM 01]) ou de façon plus générale pour la coordination de systèmes autonomes (*e.g.* la conduite de robots mobiles [ALA 98]).

La coordination est une problématique centrale des systèmes multi-agents dans la mesure où les agents doivent partager un même environnement et sont dans l'obligation de mettre en commun leurs compétences pour réaliser une tâche complexe. Bien qu'il existe des techniques de coordination réactive [AGR 87], nous ne nous intéressons dans cet article qu'aux approches de coordination par des agents cognitifs. Dans ce cadre, la problématique de la coordination a été abordée sous trois angles principaux : la résolution distribuée de problèmes, les modèles cognitifs et la planification distribuée.

L'approche issue de la résolution distribuée de problèmes propose un découpage de la coordination en trois phases distinctes : la décomposition de tâches en sous-tâches [DUR 87], l'allocation [SMI 88] et l'exécution des sous-tâches. Cependant, même si ce découpage théorique permet de bien mettre en exergue les différents niveaux de la coordination nécessaires, l'experience montre que ces trois phases s'entrelacent fortement lors de l'implémentation et qu'il n'est pas toujours aisé de les traiter séparément.

L'approche issue des modèles cognitifs « BDI » propose une formalisation de la coordination au travers des états mentaux des agents [BRA 87, RAO 91, POL 90]. Ces travaux se fondent sur la nécessité pour un groupe d'agents de posséder une intention conjointe [COH 90, LEV 90] de réaliser un but. Ce type d'approche a été validé dans le cadre de l'application STEAM [TAM 97] qui concerne l'aide à la décision pour la conduite d'hélicoptères de combat. D'autres approches cognitives [GRO 93] ne considèrent pas la notion d'intention conjointe comme la clé de voûte de la formalisation mais préfèrent présenter deux types d'intention : l'*intention de* réaliser une action ou un plan et l'*intention qu*'une proposition soit vérifiée. La coordination est réalisée par agrégation des intentions des agents permettant la construction d'un plan multi-agent par raffinements successifs. Ces modèles doivent être considérés comme des spécifications d'agents. La question qui se pose alors est comment passer des ces spécifications d'un haut niveau d'abstraction à leur mise en œuvre.

L'approche issue de la planification distribuée s'intéresse aux interactions qui existent entre les plans des agents [MAR 90]. Dans ce type d'approche, l'identification des relations positives (*e.g.* tâches redondantes) ou négatives (*e.g.* conflits de ressources) entre les plans des agents est présentée comme un premier pas vers la compréhension du processus de coordination. La coordination peut s'exprimer comme la résolution des conflits entre les activités des agents. Celle-ci peut être vue comme : un processus de négociation [ZLO 90] au cours duquel les agents doivent parvenir à un consensus ; un processus d'argumentation [KRA 93] où chaque agent essaie de convaincre les autres du bien-fondé de sa proposition ; une synchronisation entre les plans [FAL 96]

ou une fusion de plans [ALA 95]. On peut reprocher à ces approches de ne pas prendre en compte de la notion de travail coopératif : la coordination est implicitement rendue obligatoire par le nécessaire partage de l'environnement mais on considère que les agents possèdent des buts différents. Coopérer c'est « *travailler conjointement avec quelqu'un à* ». La distinction avec la « simple » coordination est donc sensible, notamment dans le cadre de la planification multi-agent. En effet, là où la coordination revêt essentiellement la forme de gestion ou évitement de conflits entre plans établis individuellement, la coopération vise à la construction collective d'un plan.

Dans cet article nous proposons un modèle de planification distribuée visant à la construction coopérative d'un plan fondé sur un processus de conjecture/réfutation. Nous commençons par une description générale du processus de planification et des principes qui le sous-tendent. Ensuite, nous précisons le fonctionnement d'un agent et les opérateurs à sa disposition. Enfin, nous décrivons de manière plus précise la dynamique de l'interaction.

2. Description générale

2.1. *Cadre*

Le domaine qui nous intéresse est celui de la construction conjointe de plans par un groupe d'agents possédant des connaissances et compétences différentes. On considère la distribution de la connaissance et l'impossibilité pour un seul agent de réaliser l'objectif du système comme des propriétés intrinsèques du problème. L'approche retenue consiste à considérer le problème sous l'angle de la planification au sens « classique » : étant donnés un état initial et un objectif, le système multi-agent tente de déterminer un plan qui permette de passer de l'état initial à un état réalisant l'objectif. Le processus proposé ne vise ni à être optimal (*i.e.* on ne cherche pas à déterminer le *meilleur* plan) ni à être complet (*i.e.* on ne garantit pas que le système trouve un plan s'il en existe un). On justifie ces restrictions par le fait que le domaine d'application privilégié est celui pour lequel les techniques classiques de planification sont inadaptées ou insatisfaisantes. Inadaptées du fait d'un espace de recherche trop grand (il faut considérer simultanément les actions de tous les agents) et insatisfaisantes pour des applications dans lesquelles il n'est pas souhaitable de rendre public les connaissances de chaque agent.

2.2. *Principe*

Le processus consiste à construire un plan par le biais d'un raisonnement s'appuyant sur les notions de proposition et contre-proposition. Ce processus s'inspire des thèses de Popper et Lakatos sur la notion de validité d'une preuve. Pour Popper, « *Nous de savons rien, nous pouvons seulement conjecturer* ». Autrement dit, il n'existe pas de preuve *correcte* de manière absolue : toute preuve peut à tout moment être *réfutée* par une expérimentation ou test. Si le test conduit à une réfutation de la

preuve, celle-ci est remise en cause et il faut la *réparer* afin de la rendre plus robuste. La nouvelle version de la preuve peut à son tour être testée et éventuellement réfutée puis réparée. L'obtention d'une preuve est donc un processus itératif de proposition-réfutation-réparation.

L'objectif de cet article est de proposer un modèle de planification distribuée s'inspirant ce ces idées. Le système multi-agent cherche un plan répondant à un objectif. Tour à tour les agents vont prendre la parole pour raffiner, réfuter ou réparer le plan courant. Si le plan réfuté est réparé, il est plus robuste mais reste soumis à de nouvelles tentatives de réfutation. S'il est réfuté et ne peut être réparé, les agents l'abandonnent et poursuivent leur recherche sur un autre plan. Si un plan n'est pas réfuté, il est considéré comme acceptable et constitue la solution du système à l'objectif proposé.
Ces cycles que l'on peut qualifier de conjecture/réfutation vont permettre la convergence vers un plan réalisant le but du groupe d'agents et qui ne peut plus être réfuté (étant données les connaissances courantes des agents).

À titre d'illustration, voici un exemple de dialogue illustrant ce principe : Albert, Basile et Coralie se trouvent à Grenoble et veulent aller à Los Angeles rendre visite à Drew (plan initial réduit au seul but initial).

– Albert : « *Allons à Lyon prendre un avion à destination de Los Angeles puis prenons un taxi pour aller chez Drew.* ». (1) c'est un raffinement du plan initial en 3 sous-buts ;

– Basile : « *Prenons donc le TGV 8913 de 16h qui permet d'arriver à Lyon à 17h.* ». (2) Nouveau raffinement du plan en substituant une action à un objectif ;

– Coralie : « *Malheureusement, il n'y a aucun avion au départ de Lyon ayant Los Angeles pour destination.* ». (3) Il s'agit d'une réfutation du plan ;

– Basile : « *Dans ce cas, allons à Paris prendre un avion pour Los Angeles puis prenons le taxi jusque chez Drew.* ». (4) Nouveau raffinement du plan initial – le raffinement proposé par Albert est abandonné ;

– Coralie : « *Le vol 421 à destination de Los Angeles part de Paris à 15h.* ». (5) Raffinement du plan ;

– Basile : « *Pour avoir ce vol, prenons le TGV 1289 qui permet d'arriver à Paris à 15h.* ». (6) Raffinement du plan ;

– Albert : « *Mais une fois à Los Angeles, nous ne pourrons pas prendre le taxi : nous n'aurons pas de dollars.* ». (7) Réfutation du plan ;

– Coralie : « *Il suffit de passer à la banque avant de prendre le TGV.* » (8) Réparation du plan.

Ce dialogue montre comment les raffinements permettent de proposer une réalisation pour un sous-but (étapes 1, 4, 5 et 6) et comment les réfutations permettent soit d'invalider un plan (étape 3) soit de le remanier (étape 8). À la suite de ce dialogue, le plan-solution obtenu est le suivant : (i) aller à la banque effectuer le change, (ii) prendre le TGV 1289, (iii) prendre le vol 421 et (iv) prendre le taxi.

2.3. *Hypothèses sur l'environnement*

Selon la typologie des environnements de Russell et Norvig [RUS 95], on suppose que les agents raisonnent dans un environnement *discret* (le nombre d'états que peut prendre l'environnement est fini, de même que le nombre des actions exécutables), *déterministe* (le résultat d'une action est parfaitement connu), *dynamique* (des processus autres que les agents peuvent modifier l'environnement) et *accessible* (chaque agent peut obtenir une information complète, précise et à-jour sur l'environnement). L'accessibilité est une hypothèse forte qui est ici justifiée par le fait que le point qui nous intéresse est la construction collaborative de plans et non l'élaboration d'un processus permettant à un groupe d'agents d'évaluer l'état de l'environnement.

On suppose de plus que chaque agent est capable de modéliser l'évolution de l'environnement. Précisément, on suppose que l'environnement « réagit » aux actions des agents et que chaque agent est capable de déterminer la réaction la plus probable de l'environnement. Ce modèle est utilisé comme source de réfutations aux plans proposés (d'une manière générale, la contradiction est apportée par des connaissances de l'agent sur le domaine).

2.4. *Application*

Dans un premier temps, afin de mettre au point et valider notre approche, l'application que nous avons choisie est le jeu d'échecs car il se modélise aisément mais est suffisamment complexe pour permettre une expérimentation non-triviale du modèle proposé ; il satisfait aux contraintes évoquées en 2.3 et il nous a semblé intéressant de l'aborder sous la forme d'un système multi-agent. Mais il est clair que notre objectif est de mettre au point des outils pour des domaines comme la composition de services en ligne.

Dans ce qui suit, chaque pièce blanche est un agent. Le problème que les agents cherchent à résoudre est la construction d'un plan aboutissant au mat en n coups à partir d'une position donnée. Les réactions de l'environnement correspondent aux coups pouvant être joués par les noirs.

3. Structure de l'agent

Un agent est composé de cinq modules :

1) *base de connaissances* : état initial de l'environnement, but à atteindre et capacité à déterminer les réactions probables de l'environnement ;

2) *raffinements* : associe à un but, un plan permettant de l'atteindre ;

3) *réfutation* : associe à un plan, un ensemble (potentiellement vide) de réfutations, ici fondé sur un modèle de l'environnement ;

4) *réparation* : associe à un plan dont un élément est marqué réfuté un nouveau plan enrichi d'actions et/ou buts;

5) *stratégie* : détermine quelle action l'agent veut effectuer ; détermine en particulier si l'agent demande la parole et quel plan il désire modifier.

Dans notre application, pour proposer un raffinement, chaque agent-pièce connaît (i) les coups qu'il peut effectuer (ii) les coups qu'il pourrait effectuer si une condition donnée était observée sur l'environnement (*e.g.* ♙ $a1d4$ est possible si les cases $b2$ et $c3$ sont vides et $d4$ n'est pas occupée par une pièce amie). Le premier cas fournit des raffinements du type $but \rightarrow \langle action \rangle$ alors que le second fournit des raffinements du type $but \rightarrow \langle but' \,;\, action \rangle$.

En ce qui concerne les réfutations, lorsqu'un raffinement est proposé, chaque pièce utilise son modèle de l'environnement pour déterminer si le nouveau plan est, d'une part, exécutable et, d'autre part, permet d'atteindre l'objectif. Si l'un des deux tests est négatif, l'agent peut prendre la parole pour proposer une réfutation.

Les réparations sont des heuristiques qui proposent une modification d'un plan réfuté par l'ajout d'actions ou de buts. Les réparations sont donc similaires aux raffinements, à ceci près qu'elles s'appliquent à un plan réfuté.
Une réfutation peut à son tour être réfutée et si un but a été ajouté par la réparation, les agents tentent de le raffiner. Ainsi, les agents s'engagent dans un processus itératif de raffinements, réfutations et réparations.

La stratégie des agents est d'explorer en profondeur d'abord, c'est-à-dire en appliquant les raffinements sur le même plan jusqu'à ce que cela ne soit plus possible ou que la longueur du plan dépasse une borne fixée (par les données du problèmes). Si un plan a été découvert, on tente de le réfuter et le cas échéant de le réparer. Quand tous les *recours* ont été épuisés, on passe à la branche suivante. On pourrait contraindre un agent à n'appliquer un raffinement que s'il ne peut le réfuter ; cependant, cette contrainte n'a pas été retenue dans la mesure où un tiers pourrait connaître une réparation à la réfutation.

4. Définition et modification du plan

4.1. *Plan*

Un plan est une séquence d'éléments pouvant être soit des actions soit des buts à atteindre. Un plan est dit *complet* s'il ne contient que des actions et *partiel* s'il contient au moins un but.

Dans notre application, une action est un mouvement d'une pièce (*e.g.* ♘ $e2f4$ – le cavalier en $e2$ se déplace en $f4$) et un but est une modification au sens des plans de Pitrat [PIT 77]. Ainsi, un but consiste en un état à atteindre concernant l'occupation d'une case de l'échiquier. Les buts possibles sont :

– case (non-) vide (*e.g.* $f5\ vide$) ;

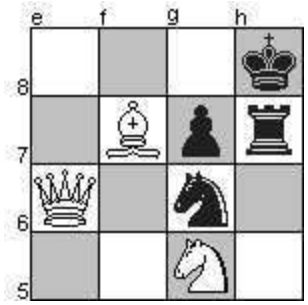

Figure 1. *But initial : mat en 2 coups.*

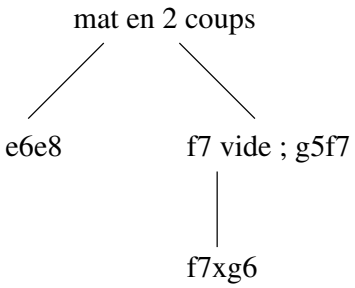

Figure 2. *Exemple d'arbre de recherche.*

– case (non-) occupée par une pièce amie, en précisant éventuellement le type de pièce (*e.g.* $a3B\♘$ – amener un cavalier $b$lanc en $a3$) ;

– case (non-) occupée par une pièce ennemie, en précisant éventuellement le type de pièce (*e.g.* $b6\neg N\♕$ – la case $b6$ ne doit pas contenir la reine $n$oire) ;

– mettre le roi adverse en échec.

Exemple de plan partiel : $\langle f5\ vide$; $\♕\ f1f8\rangle$ – rendre la case $f5$ vide puis déplacer la reine de $f1$ à $f8$.

4.2. *Arbre de recherche*

L'état courant du raisonnement (*i.e.* l'ensemble des conjectures développées par les agents) est consigné dans une structure de données partagée accessible de manière exclusive (*i.e.* type blackboard) et les agents y accédent de manière opportuniste. Précisément, il s'agit d'un arbre qui garde la trace des raffinements effectués et précise quels éléments ont été réfutés. Chaque nœud est étiqueté par un plan, l'étiquette de la racine étant le but initial que le groupe d'agents cherche à atteindre. De chaque but, part un ensemble d'arcs ; chacun des fils d'un but représente un plan proposé pour la réalisation de ce but (*i.e.* un raffinement, *cf.* 4.4) . Ainsi l'arbre de recherche est comparable à l'arbre de dérivation d'un mot dans une grammaire dont les terminaux sont les actions et les non-terminaux les buts.

La figure 2 montre un arbre de recherche pour la résolution du but *mat en (au plus) 2 coups* sur la position de la figure 1. Trois raffinements ont été appliqués ; deux pour répondre au but initial et un troisième pour raffiner un sous-but introduit par un agent (ici le cavalier blanc).

4.3. *Extraction d'un plan-solution*

L'objectif du système multi-agent est de trouver un plan complet qui permette d'atteindre le but initial ; on appelle un tel plan « *plan-solution* ». Il est possible d'extraire de l'arbre de recherche un plan-solution lorsqu'il existe un sous-arbre vérifiant les propriétés suivantes :

1) tous les buts de tous les nœuds de ce sous-arbre ont un fils ;

2) aucun arc de ce sous-arbre n'est marqué *réfuté* ;

3) ce sous-arbre contient la racine.

Dès lors, le plan-solution est la projection des actions du sous-arbre, qui constitue un plan complet.

De l'arbre de la figure 2, un agent peut extraire le sous-arbre correspondant à la branche droite qu'il projette pour obtenir le plan-solution $\langle f7 \times g6 \,;\, g5f7 \rangle$.

4.4. *Raffinement*

Raffiner un plan consiste à remplacer un but par un sous-plan. Dans l'exemple de la figure 2, les agents ont effectué 3 raffinements :

– *mat en 2 coups* est remplacé par le plan $\langle e6e8 \rangle$;

– *mat en 2 coups* est remplacé par le plan $\langle f7\ vide \,;\, g5f7 \rangle$;

– *f7 vide* est remplacé par le plan $\langle f7 \times g6 \rangle$.

La seule restriction s'appliquant aux raffinements est l'interdiction pour le sous-plan de contenir un but se trouvant sur la branche reliant le nœud courant à la racine. Un raffinement peut remplacer n'importe quel but du plan si ce dernier en contient plusieurs. En cela, la résolution peut être qualifiée d'opportuniste.
Cependant, on distingue deux types de buts : les buts *globaux* et les buts *locaux*. Un but est global si sa réalisation est susceptible d'affecter la manière dont les buts futurs seront résolus. Les agents ne doivent raisonner sur un but que si tous les buts globaux qui le précèdent ont été résolus ; ils peuvent alors déterminer l'état de l'environnement dans lequel ils doivent résoudre le but suivant.

Au jeu d'échecs tous les buts sont globaux dans la mesure où il est bien connu que tout coup, même le plus insignifiant, peut avoir une incidence sur le cours de la partie. Ainsi, dans notre application, les raffinements sont toujours appliqués de gauche à droite. À l'opposé, dans le domaine de l'exemple introductif, la plupart des objectifs situés en France (*e.g.* « se rendre à Paris ») n'affectent pas la réflexion sur les objectifs situés aux États-Unis (*e.g.* « prendre le taxi à Los Angeles »). Typiquement, que l'on se soit rendu à Paris en train ou en voiture n'a pas d'incidence sur les buts et actions concernant le séjour à Los Angeles.

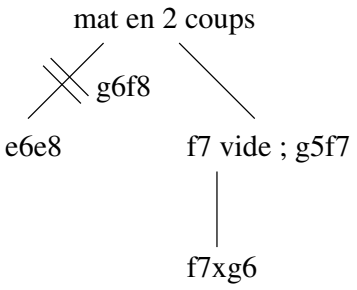

Figure 3. *Exemple de réfutation.*

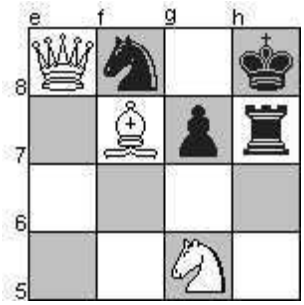

Figure 4. *Position après réfutation.*

Les raffinements sont semblables aux règles de réécriture d'une grammaire qui à un non-terminal associent un mot composé de terminaux et de non-terminaux. Un raffinement associe à un but un plan composé d'actions et de buts. Dans notre application, les raffinements peuvent prendre la forme suivante :

– si le but est de « vider une case c » et « je suis sur la case c » alors raffiner le but avec un « mouvement vers une case c' » ;

– si le but est de « vider une case c occupée par un ennemi e » alors :

- « menacer e » ; ou
- « prendre une pièce soutenue par e » ; ou
- « se placer dans une case c' telle que e peut capturer en c' ».

4.5. *Réfutation*

On suppose que l'application d'un raffinement est toujours fondée et que la source de réfutation des plans est le modèle de l'environnement. Ainsi, les agents ne discutent pas des conditions d'application du raffinement mais s'appliquent à réparer un plan pour lequel un agent a fourni une réfutation. En utilisant leurs connaissances, les agents sont capables de réfuter un plan, c'est-à-dire montrer qu'il n'atteint pas le but fixé.
Il existe deux causes de réfutation :

– une action ne sera pas possible après réaction estimée de l'environnement (*e.g.* déplacement d'une tour de $a1$ vers $a8$ si l'environnement place une pièce en $a3$) ;

– les actions d'un sous-plan sont exécutables mais leur exécution ne permet pas d'atteindre le but fixé (*e.g.* pour faire échec au roi en $g8$, on place la dame en $c8$ mais l'environnement réagit en interposant un cavalier en $f8$).

Dans la position de la figure 1, après ♕ $e6e8$, une réaction telle que ♘ $g6f8$ constitue une réfutation du plan $\langle e6e8 \rangle$ dont le but est de mater le roi noir. Une telle réaction est appelée « *réaction défavorable* ». Un agent qui détermine (en utilisant sa base de connaissances) que ♘ $g6f8$ est la réaction probable de l'environnement

marque l'arbre comme montré à la figure 3. La position qui en résulte est montrée à la figure 4.

4.6. *Réparation*

Quand un arc a été marqué réfuté, les agents tentent de *réparer* le plan, c'est-à-dire de le modifier pour que le sous-plan remis en question par la réfutation atteigne son but. Pour cela les agents peuvent s'engager dans deux types de réparation :

– réparations *a posteriori* : les agents analysent l'état de l'environnement pour déterminer s'il est possible d'atteindre le but en tirant profit des dernières modifications de l'environnement dues à la réaction (*e.g.* prise de la pièce qui effectue le coup de réfutation) ;

– réparations *a priori* : les agents copient la branche réfutée et modifient le plan d'un nœud en ajoutant un but dont l'objectif est de rendre la *réaction défavorable* impossible (*e.g.* placer une pièce sur la trajectoire de la *réaction défavorable*). Ensuite, la société d'agents appliquera des raffinements pour trouver un sous-plan exécutant ce but.

Dans notre application, les réparations *a posteriori* consistent à déterminer tous les coups rendus possibles par la réaction défavorable ; c'est-à-dire dans notre exemple $\{$coups possibles après $e6e8$ et $g6f8\} - \{$coups possibles dans la position initiale$\}$. Les réparations *a priori* consistent à rendre la réaction défavorable impossible en ajoutant un but rendant ♘ $g6f8$ impossible.

Dans notre exemple, la reine en $e8$ peut proposer deux réparations :

– *a posteriori* : le coup ♕ $e8 \times f8$ (prise du cavalier noir par la dame blanche) qui permet d'atteindre le but initial *mat en 2 coups*, l'arbre de recherche est modifié comme indiqué à la figure 6 ;

– *a priori* : demander aux agents de rendre impossible le déplacement ♘ $g6f8$, ce que le fou blanc permettra en proposant le raffinement $f7g6$. L'arbre de recherche est alors modifié comme indiqué à la figure 5.

5. Dynamique de l'interaction

À chaque instant, l'arbre de recherche ne peut être modifié que par un et un seul agent. On met donc en place un mécanisme qui garantit un accès exclusif à l'arbre de recherche. Les agents qui n'ont pas accès à l'arbre peuvent demander la parole en précisant une priorité indicatrice de l'intérêt présumé de la modification qu'ils souhaitent effectuer.

Un agent qui a la parole et effectue un raffinement peut, si le plan qu'il ajoute à l'arbre est partiel, *diriger* temporairement l'interaction en émettant un *appel d'offre* (*cf. contract net* [SMI 88]) correspondant à la réalisation d'un but du plan partiel ajouté. Les agents répondent à l'émetteur de l'appel d'offre qui à son tour est capable

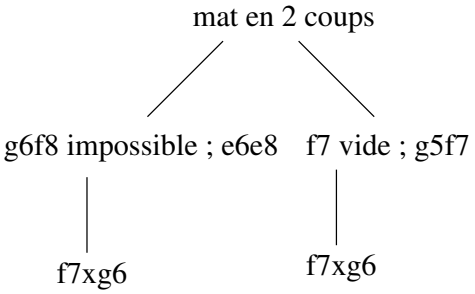

Figure 5. *Exemple de réparation a priori.*

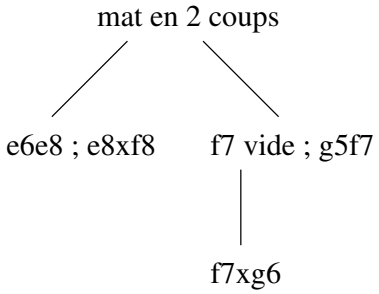

Figure 6. *Exemple de réparation a posteriori.*

de choisir un des raffinements proposés et de l'appliquer à l'arbre. Ce système d'appel d'offre qui supplée momentanément le passage de parole permet à l'agent qui propose un nouveau sous-but d'évaluer lui-même les réponses proposées par les autres agents. Ce mécanisme repose sur la supposition que l'agent qui propose –par raffinement– un nouveau but est souvent le plus à même de juger de la qualité des raffinements à ce nouveau but.

L'objectif étant l'obtention d'*un* plan solution et non du plan optimal, on effectue une recherche que l'on qualifie d'*en profondeur d'abord*. Cela signfie que les agents travaillent sur la même branche afin de pouvoir extraire au plus tôt un plan solution. Sur cette branche, les agents effectuent itérativement les opérations suivantes : (i) raffinement, (ii) réfutation et (iii) réparation (qui produit des buts devant être à leur tour raffinés).
Lorsqu'il n'est plus possible d'effectuer ce cycle (parce qu'il n'y a pas de raffinement à un but ou pas de réparation à une réfutation), ou qu'une autre branche semble plus prometteuse les agents changent de branche à explorer.

Lorsqu'un agent applique à l'arbre un raffinement qui ajoute un plan complet, il doit vérifier si les conditions d'extraction d'un sous-arbre sont vérifiées (*cf.* 4.3). Si un tel sous-arbre existe, l'agent qui possède le jeton lance un appel d'offre pour tenter de réfuter un élément du sous-arbre. Si aucun agent (y compris l'émetteur de l'appel d'offre) ne possède de réfutation, il est possible de projeter le sous-arbre afin d'obtenir un plan-solution et l'activité des agents se termine.
Une autre cause de terminaison est le constat d'échec qui se caractérise par le fait que (i) il n'existe pas de sous-arbre exhibant les propriétés décrites en 4.3 et (ii) aucun agent ne peut appliquer un raffinement ou une réparation à l'arbre.
Enfin, pour assurer la terminaison de l'interaction, il est nécessaire de se donner un critère bornant l' *effort de recherche*. Il peut s'agir, par exemple, d'une borne sur la durée, le nombre de nœuds explorés ou la longueur d'un plan.

Dans notre application, on explore une branche jusqu'à ce qu'il ne soit plus possible d'effectuer le cycle. La condition d'arrêt est déterminée à partir du problème à résoudre : « mat en n coups » fournit une borne à la longueur du plan à trouver.

6. Conclusion

Le modèle de la « *métaphore de la communauté scientifique* » présentée par Kornfeld et Hewitt dans [KOR 88] s'inspire également des thèses de Popper et Lakatos et les applique à la résolution de problèmes par des systèmes experts. Contrairement au modèle présenté ici, chaque entité y possède un rôle prédéfini (proposant ou opposant) et la stratégie de résolution consiste à explorer un grand nombre de solutions candidates en parallèle, les ressources étant allouées par des *sponsors*. Cependant, un tel schéma de résolution n'est pas adapté à un domaine dans lequel les agents cherchent à minimiser la connaissance partagée.

Le processus de planification proposé dans cet article a été testé dans le cadre du jeu d'échecs. Un programme, développé en Java, permet la résolution de problèmes du type « *mat en n coup* ». Il a été utilisé avec succès sur des positions de mats en 2 à 4 coups[1].

Le travail en cours sur le modèle consiste à d'identifier des motifs indépendants du domaine d'application dans l'expression des raffinements et réparations. On s'intéresse également à la généralisation du processus de réfutation –ici fondé sur un modèle de l'environnement– et des stratégies (notamment de prise de parole) des agents pouvant guider la recherche d'une solution ou assurer une convergence plus rapide. En effet, contrairement à d'autres approches qui découplent la phase de planification (chaque agent planifie sans tenir compte des autres) et de résolution des conflits, il nous semble intéressant de s'appuyer au plus tôt sur ces conflits pour itérativement construire un plan solution notamment pour des applications dans lesquelles les agents partagent un but commun.

7. Bibliographie

[AGR 87] AGRE P. E., CHAPMAN D., « Pengi: An Implementation of a Theory of Activity », *Proc. of AAAI-87*, Seattle, WA, 1987, p. 268-272.

[ALA 95] ALAMI R., ROBERT F., INGRAND F., SUZUKI S., « Multi-robot Cooperation through Incremental Plan-Merging », *International Conference on Robotics and Automation*, Washington D.C., 1995, IEEE, IEEE Computer Society Press, p. 2573–2579.

[ALA 98] ALAMI R., FLEURY S., HERB M., INGRAND F., ROBERT. F., « Multi robot cooperation in the Martha project », *IEEE Robotics and Automation Magazine*, IEEE, IEEE Computer Society Press, 1998.

1. Le module de réfutation des agents utilise le moteur de Crafty (http://www.limunltd.com/crafty/).

[BRA 87] BRATMAN M. E., *Intention, Plans, and Practical Reason*, Harvard University Press, Cambridge, MA, 1987.

[CAM 01] CAMACHO D., MOLINA J., BORRAJO D., ALER R., « MAPWEB: Cooperation between Planning Agents and Web Agents », 2001.

[COH 90] COHEN P. R., LEVESQUE H. J., « Intention is choice with commitment », *Artificial Intelligence*, vol. 42, n° 2-3, 1990, p. 213–261, Elsevier Science Publishers Ltd.

[DUR 87] DURFEE E. H., LESSER V. R., « Using Partial Global Plans to Coordinate Distributed Problem Solvers », *Proc. of the 10th IJCAI*, Milan, Italy, 1987, p. 875-883.

[FAL 96] FALLAH-SEGHROUCHNI A. E., HADDAD S., « A Coordination Algorithm for Multi-Agent Planning », VAN HOE R., Ed., *Agents breaking away — Proceedings of the Seventh European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-96)*, vol. 1038 de *Lecture Notes on Artificial Intelligence*, Berlin, janvier 1996, Springer Verlag.

[GRO 93] GROSZ B. J., KRAUS S., « Collaborative Plans for Group Activities », *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93)*, San Mateo, CA, août 1993, Morgan Kaufmann Publishers.

[KOR 88] KORNFELD W. A., HEWITT C. E., « The Scientific Community Metaphor », BOND A. H., GASSER L., Eds., *Readings in Distributed Artificial Intelligence*, p. 311-320, Kaufmann, San Mateo, CA, 1988.

[KRA 93] KRAUS S., NIRKHE M., SYCARA K. P., « Reaching agreements through argumentation: a logical model (Preliminary report) », *Proceedings of the 12th International Workshop on Distributed Artificial Intelligence*, Hidden Valley, Pennsylvania, 1993, p. 233–247.

[LEV 90] LEVESQUE H. J., COHEN P. R., NUNES J. H. T., « On Acting Together », *Proc. of AAAI-90*, Boston, MA, 1990, p. 94-99.

[MAR 90] VON MARTIAL F., « Interactions among Autonomous Planning Agents », DEMAZEAU Y., MÜLLER J.-P., Eds., *Decentralized A.I. : Proc. of the First European Workshop on Modelling Autonomous Agents in a Multi-Agent World, Cambridge, England*, p. 105-119, North-Holland, Amsterdam, 1990.

[PIT 77] PITRAT J., « A Chess Combination Program Which Uses Plans », *Artificial Intelligence*, vol. 8, 1977, p. 275-321.

[POL 90] POLLACK M., « Plans as Complex Mental Attitudes », COHEN P. R., MORGAN J., POLLACK M., Eds., *Intentions in Communication*, p. 77–103, MIT Press, Cambridge, Massachusetts, 1990.

[RAO 91] RAO A. S., GEORGEFF M. P., « Modeling Rational Agents within a BDI-Architecture », ALLEN J., FIKES R., SANDEWALL E., Eds., *Principles of Knowledge Representation and Reasoning: Proc. of the Second International Conference (KR'91)*, p. 473-484, Kaufmann, San Mateo, CA, 1991.

[RUS 95] RUSSELL S. J., NORVIG P., *Artificial intelligence: a modern approach*, Prentice-Hall, Inc., 1995.

[SMI 88] SMITH R. G., DAVIS R., « Frameworks for cooperation in distributed problem solving », , 1988, p. 61–70, Morgan Kaufmann Publishers Inc.

[TAM 97] TAMBE M., « Towards Flexible Teamwork », *Journal of Artificial Intelligence Research*, vol. 7, 1997, p. 83–124.

[ZLO 90] ZLOTKIN G., ROSENSCHEIN J. S., « Negotiation and Conflict Resolution in Non-Cooperative Domains », *Proc. of AAAI-90*, Boston, MA, 1990, p. 100-105.