



HAL
open science

Moteur décisionnel reposant sur un modèle de confiance pour des agents autonomes

L Callebert, Domitile Lourdeaux, Jean-Paul Barthès

► To cite this version:

L Callebert, Domitile Lourdeaux, Jean-Paul Barthès. Moteur décisionnel reposant sur un modèle de confiance pour des agents autonomes. Rencontres des jeunes chercheurs en Intelligence Artificielle (RFIA 2016), Jun 2016, Clermont-Ferrand, France. hal-01406106

HAL Id: hal-01406106

<https://hal.science/hal-01406106v1>

Submitted on 30 Nov 2016

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.

Moteur décisionnel reposant sur un modèle de confiance pour des agents autonomes

L. Callebert¹

D. Lourdeaux¹

J.P. Barthès¹

¹ Sorbonne universités, Université de technologie de Compiègne, CNRS, Heudiasyc UMR 7253, CS 60 319, 60 203 Compiègne, France

{lucile.callebert, domitile.lourdeaux, jean-paul.barthes}@hds.utc.fr

Résumé

Dans les environnements virtuels collaboratifs, la prise en compte des facteurs humains tels que la confiance sont indispensables à la modélisation des comportements des personnages virtuels autonomes qui réalisent une activité collective. Nous proposons un moteur décisionnel pour des agents autonomes qui repose sur un modèle de confiance. Trois facteurs de confiance sont considérés : la confiance en l'intégrité, en la bienveillance et en les compétences de chacun des autres agents. Le moteur décisionnel proposé permet aux agents de calculer l'importance de leurs buts personnel et collectif, puis, pour sélectionner une tâche, de calculer l'utilité de chaque tâche. Une évaluation préliminaire montre notamment que la confiance d'un agent en les compétences d'un autre est correctement évaluée par des observateurs en fonction des comportements des agents produits avec notre modèle.

Mots Clef

Moteur décisionnel, modèle de confiance, agents autonomes.

Abstract

In collaborative virtual environments, human factors such as trust are essential to consider for the creation of autonomous virtual characters. We propose a decision-making system for autonomous agents that relies on a trust model. Three factors of trust are considered : the trust in the trustee's integrity, benevolence and abilities. The proposed decision-making system allows agents to compute, taking into account other agents, the importance of their personal and collective goals. Then the utility of the tasks is computed and a task is selected, also taking others into account. We conducted a preliminary evaluation that shows that the trust an agent has in another's abilities is rightly assessed by subjects when observing agents behaviors generated with our model.

Keywords

Decision-making system, trust model, autonomous agents.

1 Introduction

Dans les environnements virtuels collaboratifs pour l'apprentissage, l'apprenant est bien souvent entouré de personnages virtuels avec lesquels il forme une petite équipe et avec lesquels il doit réaliser une tâche collaborative. C'est le cas par exemple dans le projet SECUREVI [Querrec et al., 2003] où l'apprenant et les personnages virtuels forment une équipe de pompiers ; ou dans le projet 3D Virtual Operating Room [Sanselone et al., 2014] où l'apprenant fait partie d'une équipe médicale. Les personnages virtuels dans ce type d'environnement sont souvent autonomes, et leur moteur décisionnel leur permet de choisir dynamiquement quelle action faire pour mener à bien l'activité collective. Pour prendre leurs décisions, les personnages s'appuient généralement sur un modèle d'activité représentant l'activité collective qu'ils ont à réaliser, de manière à ce que leurs comportements soient représentatifs des comportements humains observés en situation réelle. Les actions des personnages sont alors bien souvent déterminées par les rôles qu'ils jouent et qui sont décrits dans le modèle d'activité. Cependant un problème se pose lorsque plusieurs agents jouent le même rôle ou lorsque l'activité collective ne peut pas se découper sous forme de rôles : par exemple, comment trois agents se répartiraient les tâches ménagères pour l'entretien d'une maison ? Ces situations sont aussi rencontrées dans le monde professionnel : dans une équipe de pompiers ayant tous les mêmes qualifications, comment s'organiser pour le sauvetage d'une victime ?

Par ailleurs, les facteurs humains sont souvent oubliés : chaque personnage est supposé être compétent et disposé à agir pour l'intérêt de l'équipe, ce qui n'est malheureusement pas toujours le cas dans la réalité. Dans un contexte collectif, chaque personnage devrait se poser des questions comme *Mon coéquipier va-t-il vraiment agir dans l'intérêt de l'équipe ?* ou *Est-t-il compétent pour réaliser cette tâche ?* avant de prendre une décision. Ces questions reposent sur un facteur essentiel : celui de la confiance. La question *Est-t-il compétent pour réaliser cette tâche ?* se

traduit pas exemple par *Ai-je confiance en la compétence de mon coéquipier*? Nous ne cherchons pas à reproduire les processus cognitifs humains, mais à proposer un moteur décisionnel qui permette de reproduire des comportements humains : nous n'argumentons pas que chaque personne se pose ces questions en réalité, mais que répondre à ces questions permettra aux agents d'adopter des comportements cohérents et semble à ceux des humains. Dans un contexte de formation professionnelle, il serait alors possible de former les apprenants non plus seulement à des compétences techniques mais aussi à des compétences non techniques : savoir prendre en compte les autres et s'adapter aux autres.

Nous proposons dans cet article un moteur décisionnel pour des personnages virtuels formant une petite équipe à laquelle pourra être intégré l'apprenant. Ce moteur décisionnel permet aux personnages autonomes de raisonner sur un modèle de tâches mais aussi sur les relations de confiance qu'ils entretiennent avec les autres pour choisir quelle action faire.

2 Travaux connexes

Nous présentons dans cette section des travaux sur des modèles d'activité utilisés pour générer le comportement d'agents autonomes en environnement virtuel, ainsi que des travaux portant sur des modèles de confiance.

2.1 Description de l'activité

Dans les projets SECUREVI [Querrec et al., 2003] et 3D Virtual Operating Room [Sanselone et al., 2014], l'activité des personnages virtuels est décrite respectivement grâce au méta-modèle d'activité HAVE et à des diagrammes Business Process Notation Model. Dans ces deux projets, un rôle est assigné aux agents, qui ont des tâches spécifiques à réaliser et pour lesquelles ils doivent être synchronisés. Or nous souhaitons que les agents autonomes puissent choisir dynamiquement quelle tâche faire en tenant compte des autres et sans être limités à un rôle particulier. De plus ce type de modèle d'activité ne permet pas de spécifier des actions collectives.

Le langage de description de scénario LORA (Language for Object-Relation Application) [Gerbaud et al., 2007] permet de décrire des activités collectives sous forme d'automates hiérarchiques à états finis et dans lesquels il est possible de spécifier des actions collectives. Cependant ce langage est difficile d'accès pour des non-informaticiens ; or nous souhaitons que l'activité puisse être décrite par des experts du domaine ou des ergonomes. Le langage de description d'activité ACTIVITY-DL [Barot, 2014] permet une description hiérarchique des activités sous forme d'arbre de tâches, qui sont logiquement et temporellement ordonnées. Ce langage est inspiré de travaux en ergonomie cognitive, et est donc accessible aux non-informaticiens. Nous choisissons ce méta-modèle pour la description des activités collectives qui seront réalisées par les agents.

2.2 Modèles de Confiance

Un modèle de confiance dyadique permettra aux agents de prendre en compte les caractéristiques individuelles de chacun des autres membres de l'équipe pour prendre une décision. Dans les paragraphes qui suivent, nous utilisons "trustor" pour désigner la personne qui fait confiance et "trustee" pour désigner la personne objet de la confiance du trustor.

[Marsh, 1994] proposent un moteur décisionnel s'appuyant sur un modèle de confiance computationnel permettant à des agents de décider de coopérer avec d'autres. La décision de coopérer est prise lorsque la confiance situationnelle, qui représente combien le trustor fait confiance au trustee dans une situation donnée, est supérieure à un seuil de coopération représentant combien le trustor a besoin de faire confiance au trustee dans cette situation. Il n'y a cependant pas de notion d'équipe dans ce modèle, et bien que nous nous intéressons à des relations de confiance dyadiques, nous pensons que l'équipe en soi et les engagements de personnes vis-à-vis de leur équipe jouent un rôle dans la décision de coopérer ou non.

[Castelfranchi and Falcone, 2001] proposent un moteur décisionnel permettant à des agents de décider de déléguer des tâches. Ce moteur décisionnel s'appuie sur un modèle computationnel de la confiance sociale. Dans ce modèle, le trustor a des croyances sur la motivation et les capacités du trustee à faire une tâche. La croyance liée à la motivation est cependant difficile à évaluer. Elle est par exemple déduite de la profession du trustee (e.g. un médecin est supposé vouloir soigner ses patients) ou d'une relation liant le trustee au trustor (e.g. des amis sont supposés s'entraider). Pour pouvoir l'expliquer, il faudrait donc s'appuyer sur un modèle de relations sociales ou de rôles entretenus par les agents.

[Mayer et al., 1995] proposent un modèle de la confiance interpersonnelle, qui regroupe les facteurs liés à la confiance les plus souvent cités dans la littérature. Trois dimensions de confiance sont identifiées : l'intégrité, la bienveillance et les capacités. Le trustor a confiance en l'intégrité du trustee s'il pense que le trustee tiendra parole et agira honnêtement. Dans notre contexte, l'intégrité peut être interprétée comme le fait de respecter les engagements par rapport à l'équipe et privilégier le but collectif. Avoir confiance en la bienveillance du trustee signifie, pour le trustor, penser que le trustee a de bonnes intentions à son égard, et tiendrait compte de ses intérêts. Enfin le trustor peut avoir confiance ou non dans les capacités du trustee à réaliser une tâche particulière. Ce modèle permet donc de prendre en compte individuellement chaque agent et ne nécessite pas d'être couplé à un autre modèle de relations sociales par exemple, et permet également à l'agent de prendre en compte l'équipe dans sa globalité : il répond donc à notre problématique.

3 Modèle d'activité

3.1 ACTIVITY-DL

ACTIVITY-DL est un méta modèle pour la description des activités humaines. Ce méta modèle est inspiré de travaux en ergonomie. Dans ACTIVITY-DL, l'activité est décrite sous forme de tâches, qui sont décomposées en sous-tâches pour former un arbre. Les tâches composées correspondent à des tâches abstraites, alors que les tâches feuilles correspondent à des actions que les personnages virtuels peuvent réaliser dans l'environnement virtuel. Les sous-tâches d'une tâche composée sont organisées d'une part d'un point de vue temporel, grâce au constructeur de leur tâche mère ; et d'autre part d'un point de vue logique grâce à la contrainte de satisfaction de leur tâche mère. [Barot, 2014] donne une description détaillée de ACTIVITY-DL.

Nous présentons un exemple simple d'activité décrite avec ACTIVITY-DL dans la Figure 1. Cet exemple décrit comment *sauver la victime* pour une équipe de secouristes : il faut d'abord *faire les premiers soins* puis *transporter la victime*. Ces deux sous-tâches doivent être faites (contrainte de satisfaction AND) dans cet ordre (constructeur SEQ-ORD pour "séquentiellement ordonné"). La tâche *faire les premiers soins* est également composée de deux sous-tâches qui sont *poser une attelle* et *prendre la tension*. Ces deux sous-tâches doivent être réalisées (contrainte de satisfaction AND) et peuvent être réalisées en même temps (constructeur PAR pour "parallèle").

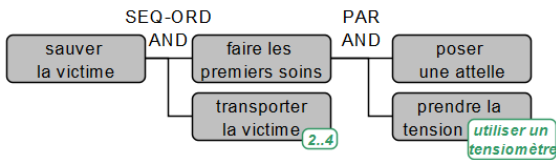


FIGURE 1 – Arbre de tâches ACTIVITY-DL pour sauver une victime

3.2 Description d'activité collective

ACTIVITY-DL était jusqu'à présent utilisé pour décrire l'activité d'un seul opérateur. De manière à pouvoir supporter la description d'une activité collective et le raisonnement des agents, nous avons ajouté les éléments suivants :

Compétence : Pour une tâche correspondant à une action (tâche feuille de l'arbre), il peut être précisé la ou les compétences que l'opérateur doit avoir pour pouvoir réaliser la tâche. Par exemple, la compétence *savoir utiliser un tensiomètre* est nécessaire pour faire la tâche *prendre la tension*.

Nombre d'opérateurs : De manière similaire, il est nécessaire de préciser pour chaque tâche correspondant à une action combien d'opérateurs sont nécessaires. Par exemple, pour *transporter la victime*, il est nécessaire d'être au moins deux, et il peut y avoir jusqu'à quatre personnes qui portent le brancard.

3.3 Propagation des contraintes

Les éléments (compétences et nombre d'opérateurs) spécifiés sur les tâches feuilles doivent être propagés aux tâches composées, de manière à permettre aux agents autonomes de raisonner sur ces dernières. En effet, raisonner sur des tâches abstraites plutôt que sur l'ensemble des tâches feuilles permet d'optimiser le fonctionnement des agents, puisqu'un nombre réduit de tâches est pris en compte (voir section 5.3).

Pour supporter le raisonnement des agents tout au long de la réalisation de l'activité collective, des contraintes sont donc générées à partir des éléments de compétence et de nombre d'opérateurs, et propagées depuis les tâches feuilles aux tâches composées jusqu'à la racine de l'arbre. Ces contraintes sont de deux types :

La contrainte de faisabilité est statique et servira à chaque agent, avant de s'engager dans la réalisation de l'activité collective, à déterminer si l'ensemble de l'équipe peut réaliser cette activité. Pour cela, l'agent utilisera ce qu'il sait de ses propres compétences, du nombre d'agents de l'équipe, et des compétences des autres agents (voir Section 4). La contrainte de faisabilité associée à l'exemple développé dans la Section 3.1 est, textuellement, la suivante : l'équipe doit être composée au minimum de deux agents *ET* au moins l'un d'eux doit *savoir utiliser un tensiomètre*.

La contrainte représentative de l'avancement est dynamique et servira à chaque agent pour déterminer s'il peut, personnellement, prendre part à la réalisation de la tâche. Pour raisonner sur cette contrainte, l'agent ne prendra en compte que ses propres compétences. Dans l'exemple développé dans la Section 3.1, la contrainte représentative de l'avancement à t_0 (i.e. avant que les agents n'aient commencé à réaliser l'activité collective) est *True*. En effet, un agent ne sachant pas utiliser un tensiomètre peut tout de même participer en exécutant l'action *poser une attelle*. Cette contrainte est dite représentative de l'avancement puisqu'elle ne tient compte que des tâches qu'il reste à faire : si la première tâche effectuée est *poser une attelle*, alors la contrainte représentative de l'avancement deviendra *savoir utiliser un tensiomètre*. Les agents ne sachant pas utiliser un tensiomètre ne pourront alors pas, à ce moment, participer à l'activité *sauver la victime*.

4 Modèle d'agent

Pour répondre à notre problématique, nous avons choisi d'opérationnaliser le modèle de confiance de [Mayer et al., 1995] en définissant les dimensions propres des agents et les croyances qu'ils ont sur les autres. Le modèle proposé permettra aux agents de raisonner sur leurs buts et de sélectionner une tâche à réaliser pour atteindre ces buts. Les choix des agents seront influencés par leurs caractéristiques personnelles, ainsi que par leurs croyances sur les autres agents, qui décrivent la confiance qu'ils ont en ces autres agents.

4.1 Buts

Nous souhaitons que les agents puissent avoir deux buts de types différents : un but lié à l'équipe et partagé par les autres membres de l'équipe, et un but qui leur est personnel. Pour l'agent x , le but personnel est noté $\gamma_{x,self}$ alors que le but collectif est noté $\gamma_{x,team}$. Ces buts sont définis par le formateur avant le début de la simulation et sont statiques.

4.2 Dimensions

Nous avons défini les dimensions des agents de manière à ce qu'elles correspondent au modèle de confiance développé par [Mayer et al., 1995].

Intégrité : Chaque agent x a une valeur d'intégrité $i_x \in [0; 1]$ qui indique à quel point cet agent va respecter ses engagements vis-à-vis de l'équipe.

Bienveillance : Chaque agent x a un niveau de bienveillance spécifique à l'égard de chacun des autres membres de l'équipe. La bienveillance de x à l'égard de l'agent y est décrite par la valeur $b_{x,y} \in [0, 1]$.

Capacité : Chaque agent x a un niveau de capacité différent pour chacune des compétences présentes dans la description des activités pouvant être réalisées. Le niveau de capacité de x sur la compétence i est décrit par la valeur $a_{x,i} \in [0, 1]$ (a pour *ability*).

Chacune de ces valeurs doit varier dynamiquement en fonction des comportements des autres agents et des événements se produisant dans l'environnement virtuel. Le mécanisme de mise à jour de ces valeurs ne sera cependant pas développé dans ce papier qui a trait au système décisionnel des agents permettant la sélection d'une action.

4.3 Croyances

L'agent x a des croyances sur chacun des autres agents. Les croyances de x sur y décrivent la confiance que x a en y : nous avons donc défini ces croyances en nous appuyant sur le modèle de confiance de [Mayer et al., 1995]. Les croyances de l'agent x sur l'agent y sont décrites par les valeurs suivantes :

Croyance sur l'intégrité : $i_y^x \in [0; 1]$ représente la confiance qu'a x en l'intégrité de y .

Croyances sur la bienveillance : $b_{y,z}^x \in [0; 1]$ représente la confiance qu'a x en la bienveillance de y vis-à-vis de z , z pouvant représenter n'importe quel agent de l'équipe hormis y lui-même.

Croyances sur les capacités : $a_{y,i}^x \in [0; 1]$ représente la confiance de x en la capacité de y à réaliser une tâche nécessitant la compétence i .

De même que pour les dimensions des agents, ces croyances sont dynamiques mais le mécanisme de mise à jour des croyances ne sera pas détaillé dans ce papier.

5 Moteur décisionnel

Nous proposons ici un nouveau moteur décisionnel pour des agents autonomes qui forment une équipe. Ce moteur décisionnel est fortement lié au modèle de confiance de

[Mayer et al., 1995] mais n'en est pas une opérationnalisation directe. En effet, bien que stipulant que les relations de confiance ont une forte influence sur la prise de décision d'une personne, les auteurs ne développent pas quels sont exactement les processus décisionnels en jeu. Nous proposons donc un moteur décisionnel inspiré de ces travaux qui réponde à notre problématique particulière comme décrite en introduction.

5.1 Fonctionnement général

Les agents suivent un cycle *perception, décision, action* : en fonction de l'état du monde qu'ils perçoivent, les agents pourront décider d'une tâche à accomplir puis agir en réalisant cette tâche.

Dans la phase de *décision*, les agents doivent dans un premier temps choisir sur lequel de leurs deux buts (individuel et collectif) concentrer leurs efforts avant de se lancer dans toute activité. Cette distinction est complexe et ne doit pas se faire de manière unilatérale et systématique (e.g. tous les agents privilégieraient le but collectif), car cela ne serait pas représentatif des comportements observés en réalité. Comment les agents doivent-ils alors choisir entre un but personnel et un but collectif ? Comment la perception des autres influence-t-elle l'importance qu'un agent accorde à ses buts ? La perception des autres est en effet très importante : un agent perdra sûrement sa motivation à réaliser la tâche collective s'il pense que les autres privilégieront leur objectif personnel. Nous proposons pour répondre à cette problématique un mécanisme de calcul de l'importance d'un but développé dans la Section 5.2.

Dans un second temps, les agents doivent identifier quelles tâches il serait le plus utile d'accomplir par rapport à leurs buts : les agents vont calculer l'utilité de chaque tâche. Encore une fois ce processus est complexe : les agents ne doivent pas seulement prendre en compte la contribution des tâches à leurs buts, mais aussi leur capacité à effectuer les tâches. Le mécanisme de calcul de l'utilité des tâches que nous proposons est développé dans la Section 5.3.

Enfin, pour sélectionner une tâche, il est nécessaire pour les agents de prendre en compte les autres : quelles sont leurs capacités et leurs envies ? Dans notre exemple, soit une équipe de deux agents x et y qui accordent de l'importance au but *sauver la victime*. Si x a une valeur de capacité $c_{x,j} = 1$ sur la compétence $j = \textit{savoir utiliser un tensiometre}$ et y a une valeur de capacité $c_{y,j} = 0$ sur cette même compétence, alors pour que les tâches soient réparties de manière optimale, x doit prendre en compte l'incapacité de y à utiliser un tensiometre, et réaliser la tâche *prendre la tension* pour laisser à y le soin de *poser l'attelle*. Nous expliquons dans la Section 5.3 le mécanisme qui permet aux agents de sélectionner une tâche.

5.2 Importance d'un but

L'importance que l'agent x accorde à l'un de ses buts γ est notée $Imp_x(\gamma)$ et est comprise dans l'intervalle $[0; 1]$. Si les deux buts sont identiques ou réaliser un but contribue à la réalisation de l'autre, alors il n'y a pas à choisir de

privilegier l'un des deux, et leur importance est 1. Dans le cas contraire, calculer l'importance qu'il accorde à chacun de ses buts permettra à l'agent x de privilegier la réalisation de l'un d'eux.

Dans un premier temps, seules les caractéristiques propres à l'agent x sont prises en compte pour le calcul de l'importance d'un but. Selon le modèle de confiance de [Mayer et al., 1995], le respect de ses engagements par une personne est liée à son intégrité (honnêteté). Nous considérons donc qu'une personne plutôt intègre privilegiera le but collectif sur lequel elle est engagée, alors qu'une personne peu intègre privilegiera son but personnel. Nous définissons donc $Imp_x(\gamma_{x,team}) = i_x$ et $Imp_x(\gamma_{x,self}) = 1 - i_x$. Pour tous les autres buts (e.g. but personnel de l'agent y par exemple), l'importance pour x est initialement nulle.

Dans un second temps, l'agent x prend en compte les autres : il calcule alors $Imp_y^x(\gamma) \in [0; 1]$, valeur qu'il imagine être l'importance du but γ pour un autre agent y . Pour le but collectif de y , x calcule $Imp_y^x(\gamma_{y,team}) = i_y^x$; pour le but personnel de y , x calcule $Imp_y^x(\gamma_{y,self}) = 1 - i_y^x$. Pour moduler l'importance qu'il accorde à ses buts en fonction des autres, l'agent x considère ensuite sa bienveillance à l'égard des autres. En effet, si x est bienveillant à l'égard de y , alors il aura envie d'aider y et l'importance pour x d'un but important pour y augmentera. x calcule donc $d_{x,y}(\gamma)$ sa désidérabilité par rapport à y de voir le but γ accompli avec la Formule 1. $d_{x,y}(\gamma)$ est proportionnel à $Imp_y^x(\gamma)$ et est positif si x apprécie y , et négatif dans le cas contraire.

$$d_{x,y}(\gamma) = Imp_y^x(\gamma) \times (b_{x,y} - 0.5) \quad (1)$$

Lorsque x a calculé, pour chaque autre agent y_i et pour chaque but γ , sa désidérabilité de voir y_i réaliser son but γ , alors x met à jour l'importance qu'il accorde à ses buts. x calcule donc $v_x(\gamma)$, le facteur qui modulera l'importance de son but γ , de manière à ce que ce facteur soit représentatif de l'ensemble des désidérabilités $d_{x,y_i}(\gamma)$ avec la Formule 2, où n est le nombre total d'agents. Finalement, $Imp_x(\gamma)$ est modulée par le facteur $v_x(\gamma)$, tout en restant comprise dans l'intervalle $[0; 1]$, comme défini dans la Formule 3.

$$v_x(\gamma) = \sum_{i=1}^{n-1} d_{x,y_i}(\gamma) \quad (2)$$

$$\begin{cases} Imp_x(\gamma) \leftarrow 1 \text{ si } Imp_x(\gamma) + v_x(\gamma) > 1 \\ Imp_x(\gamma) \leftarrow 0 \text{ si } Imp_x(\gamma) + v_x(\gamma) < 0 \\ Imp_x(\gamma) \leftarrow Imp_x(\gamma) + v_x(\gamma) \text{ sinon} \end{cases} \quad (3)$$

5.3 Raisonnement sur les tâches

Les agents cherchent à réaliser les buts qui ont le plus d'importance à leurs yeux. Pour cela ils raisonnent sur les arbres de tâches décrits avec ACTIVITY-DL. Lorsqu'ils ont sélectionné une tâche feuille, ils exécutent l'action correspondante dans l'environnement virtuel. Ils cherchent ensuite une nouvelle tâche à effectuer, et ce jusqu'à accomplir leurs buts. Nous détaillons les processus décisionnels permettant la sélection d'une tâche dans les paragraphes suivants.

Utilité d'une tâche. Les agents cherchent à déterminer quelle tâche il serait le plus judicieux de réaliser en fonction de l'importance qu'ils accordent à leurs buts. Pour cela, chaque agent x calcule l'utilité des tâches, en commençant par les tâches racines des arbres. Deux facteurs sont à prendre en compte pour ce calcul. D'une part pour une tâche τ , x doit se demander si elle contribue à un but auquel il accorde de l'importance. Si τ contribue effectivement à un but γ important pour x , alors l'utilité de τ est proportionnelle à l'importance accordée par x à γ (voir Formule 4). Sinon, l'utilité de τ pour x est nulle.

x raisonne d'autre part sur les compétences requises par τ et sur ses valeurs de capacités sur ses compétences. Les compétences requises par τ sont listées dans la contrainte représentative de l'avancement associée à τ . x calcule alors $a_x(\tau)$ sa capacité à réaliser τ . L'utilité de τ sera proportionnelle à la capacité de x à réaliser τ , de manière à ce que x privilegie les tâches pour lesquelles il est doué.

L'utilité $U_x(\tau) \in [0; 1]$ de la tâche τ pour l'agent x est donc donnée par la Formule 4.

$$U_x(\tau) = fImp_x(\gamma) \times a_{x,j} \quad (4)$$

Sélection d'une tâche. Une fois que les agents ont calculé l'utilité de chacune des tâches, il est nécessaire qu'ils prennent en compte les autres membres de l'équipe, comme expliqué dans la Section 5.1. La prise en compte des autres se fait tout d'abord par le calcul de l'utilité des tâches pour chacun des autres agents. L'agent x calcule donc, pour chaque autre agent y et pour chaque tâche τ , $U_y^x(\tau)$ ce qu'il pense être l'utilité de τ pour y . L'agent x applique le même processus que pour lui-même en utilisant $a_y^x(\tau)$ et $Imp_y^x(\gamma)$ i.e. respectivement ce qu'il croit être la capacité de y à réaliser la tâche τ et ce qu'il croit être l'importance pour y du but γ auquel contribue τ .

Enfin, x génère toutes les répartitions possibles de tâches entre les agents de l'équipe. Pour générer ces répartitions, x prend en compte le nombre d'agents nécessaires à la réalisation de chaque tâche. Cette information est exprimée dans la contrainte représentative de l'avancement liée à la tâche. x calcule également l'utilité associée à chaque répartition de tâches comme une moyenne des utilités de chaque tâche de la répartition.

Finalement, l'agent x sélectionne la tâche correspondant à la répartition ayant la plus grande utilité.

Il est important de noter qu'en sélectionnant une tâche correspondant à une répartition que l'agent x estime la meilleure, x n'ordonne pas aux autres agents de réaliser les tâches correspondant à cette répartition. En effet, par ce processus, l'agent x imagine quelle serait, selon lui, la meilleure répartition, et ce compte tenu de ce qu'il pense être les envies et les capacités des autres membres de l'équipe. Si l'agent x connaît bien les autres membres de l'équipe, i.e. si les croyances qu'a x sur les autres agents correspondent à la réalité, alors ce que feront les autres agents correspondra sûrement à ce qu'a imaginé x . Cependant, si les croyances de x sont fausses, ses prédictions sur

les choix des autres agents le seront aussi. Cela peut mener à deux situations problématiques :

- Un trop grand nombre d’agents décident de réaliser la même action. Dans ce cas, il faudrait ajouter un processus de négociation entre les agents de manière à trouver un accord. Nous n’avons pas intégré pour le moment un tel processus, et lorsque ce cas de figure se produit, nous appliquons le principe "premier arrivé, premier servi".
- Trop peu d’agent décident de réaliser la même action. Dans ce cas, les agents attendent d’être suffisamment nombreux. Si l’attente est trop longue, les agents abandonnent cette action et le processus décisionnel est relancé.

Processus récursif. Nous avons précisé au début du paragraphe concernant le calcul de l’utilité d’une tâche que l’agent x raisonne dans un premier temps sur les tâches correspondant aux racines des arbres de tâches. Lorsque x sélectionne une tâche τ , celle-ci ne correspond donc pas forcément à une action mais peut être une tâche abstraite. Le processus de sélection d’une tâche est un processus récursif, et à chaque étape de ce processus, seules les tâches pertinentes et seuls les agents pertinents pour le raisonnement de x sont pris en compte. Nous présentons dans la Figure 2 une illustration des ensembles de tâches et d’agents sur lequel(s) x raisonne aux étapes n et $n + 1$ de son raisonnement récursif, et nous expliquons dans les paragraphes suivants comment l’agent x constitue ces ensembles.

Lorsque x sélectionne une tâche abstraite τ_k , deux cas sont à distinguer :

Cas 1 : τ_k a un constructeur SEQ-ORD : ses sous-tâches doivent être effectuées séquentiellement dans l’ordre donné. x n’a alors pas de choix à faire quant à la sous-tâche de τ_k à effectuer. On note τ_{kk} cette sous-tâche.

Cas 2 : τ_k a un constructeur PAR : ses sous-tâches peuvent être effectuées parallèlement. x doit alors choisir laquelle des sous-tâches de τ_k effectuer.

Nous nous intéressons au *cas 2*, lorsque x doit alors choisir l’une des sous-tâches de τ_k . Soit $T(k)$ l’ensemble des sous-tâches de τ_k : à l’appel récursif suivant, x raisonne sur l’ensemble de tâches $T(k)$. En partant des racines des arbres et par récursivité x raisonne donc sur des tâches de plus en plus concrètes, jusqu’à sélectionner une tâche correspondant à une action.

De même, l’agent x ne considère pas, à chaque étape de son raisonnement récursif, tous les autres agents. En effet, si x pense que y ne va pas se consacrer à la tâche *sauver la victime*, alors il est inutile d’imaginer se répartir les tâches *poser une attelle et prendre la tension* avec y . On désigne par $T_{x,n}$ et $A_{x,n}$ respectivement les ensembles de tâches et d’agents sur lesquels x raisonne à l’étape n de son raisonnement récursif. Lorsque x sélectionne la tâche τ_k à l’issue l’étape n , alors à l’étape $n + 1$:

- $T_{x,n+1} = T(k)$ si τ_k a un constructeur PAR.
- $T_{x,n+1} = \tau_{kk}$ si τ_k a un constructeur SEQ-ORD.
- $A_{x,n+1}$ est l’ensemble des agents, qui d’après les projections de x , choisirons aussi τ_k à l’issue l’étape n . On

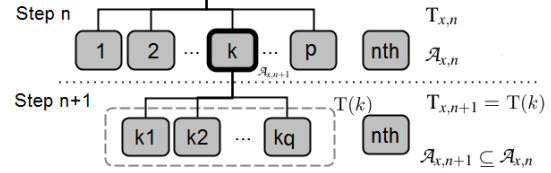


FIGURE 2 – Ensembles de tâches $T_{x,n}$, $T_{x,n+1}$ et ensembles d’agents $A_{x,n}$, $A_{x,n+1}$ sur lesquels raisonne l’agent x aux étapes n et $n + 1$ de son raisonnement récursif dans le cas du constructeur PAR sur τ_k

a donc $A_{x,n+1} \subseteq A_{x,n}$.

Cette récursivité permet une grande simplification des processus décisionnels des agents : cela leur évite d’avoir à raisonner sur un arbre entier et sur l’ensemble des agents, ce qui limite grandement le nombre de calculs à effectuer par chaque agent.

6 Résultats préliminaires

Nous présentons dans cette section des résultats préliminaires obtenus, d’une part sur le fonctionnement d’agents dotés du modèle décisionnel proposé, et d’autre part sur la perception et l’interprétation du comportement des agents par des observateurs humains.

6.1 Tests de fonctionnement

Nous avons implémenté le modèle d’agent et le moteur décisionnel proposés en Python. Chaque agent fonctionne dans un thread différent, ce qui permet aux agents de fonctionner parallèlement.

Nous avons dans un premier temps testé le bon fonctionnement des agents utilisant le moteur décisionnel proposé en faisant varier le nombre d’agents considérés et la taille de l’arbre de tâches considéré. Pour cela nous avons généré aléatoirement des arbres abstraits de tâches comportant d’une dizaine à plus d’un millier de tâches, et observé, avec différents nombres d’agents, si les agents pouvaient mener à bien l’activité décrite dans l’arbre. Nous avons pu constater que les agents fonctionnent quand :

- Les agents sont peu nombreux dans l’environnement : cinq agents peuvent raisonner en temps interactif sur des arbres de tâches de taille conséquente (allant jusqu’à 1000 tâches environ).
- Les agents sont un peu plus nombreux, mais les arbres d’activité sont de taille réduite : sept agents peuvent raisonner en temps interactif sur des arbres ayant une vingtaine de tâches.

Au delà de sept agents, le moteur décisionnel de chaque agent ne fonctionne plus en temps interactif. Cependant cela correspond à nos objectifs : proposer un moteur décisionnel pour un agent autonome faisant partie d’une équipe de travail de petite taille et dont l’agent connaît tous les membres.

6.2 Évaluation Préliminaire

Nous avons également conduit une évaluation préliminaire portant sur la perception des comportements des agents par des observateurs humains.

Efficacité des agents. Nous avons d'abord voulu vérifier avec cette évaluation préliminaire que le modèle proposé permet de produire des agents aux comportements considérés comme appropriés en équipe aussi bien que des agents dont les comportements font d'eux des mauvais équipiers. L'intégrité étant le principal facteur d'influence sur l'importance qu'un agent accorde à son but collectif, nous avons fait les hypothèses suivantes :

H1 : une équipe d'agents intègres sera perçue comme efficace par les observateurs, alors qu'une équipe d'agents non-intègres sera perçue comme inefficace.

H2 : un agent intègre sera jugé par les observateurs comme participant au bon fonctionnement de l'équipe, ce qui ne sera pas le cas pour un agent non intègre.

Nous avons pour cela présenté à douze observateurs trois déclinaisons d'un même scénario où trois agents x, y et z doivent ranger leur bureau avant que leur chef n'arrive. La tâche collective *ranger le bureau* est décomposée en plusieurs sous-tâches : *accrocher le tableau, laver le sol, monter le bureau, cacher la machine à café et réparer l'ordinateur*. Les agents ont également la possibilité de *boire du café*, tâche qui correspond à un but personnel. Dans ce scénario, aucune compétence spécifique n'est requise. Dans une équipe où les relations entre agents sont neutres ($\forall i$ et $\forall j, b_{i,j} = 0.5$) nous avons fait varier l'intégrité des agents. Les trois déclinaisons du scénario sont les suivantes :

Variante 1 : tous les agents sont intègres (i.e. $i_x = i_y = i_z = 1$); et participent donc tous à la tâche collective.

Variante 2 : un seul agent x n'est pas intègre (i.e. $i_x = 0, i_y = i_z = 1$) et boit du café alors que les deux autres rangent le bureau.

Variante 3 : tous les agents ne sont pas intègres (i.e. $i_x = i_y = i_z = 0$) et boivent du café.

Pour chaque variante, la répartition des tâches entre les agents et leur enchaînement étaient présentés aux observateurs sous forme de graphe. Puis deux affirmations étaient présentées aux observateurs, qui devaient indiquer à quel degré ils étaient d'accord avec ces affirmations sur une échelle de Likert à cinq niveaux (de 1 = *pas du tout d'accord* à 5 = *tout à fait d'accord*). La première affirmation concerne l'efficacité de l'équipe (A1 : *les agents forment une équipe efficace*); et la seconde concerne le comportement d'un agent particulier dans l'équipe (A2 : *l'agent x contribue au bon fonctionnement de l'équipe*). Les résultats obtenus sont présentés dans le Tableau 1.

Pour tester nos hypothèses H1 et H2 nous avons appliqué deux tests ANOVA à un facteur sur (i) l'efficacité perçue de l'équipe (affirmation A1) et (ii) sur la participation au de l'agent x au bon fonctionnement de l'équipe (affirmation A2). Le facteur intra-sujet testé est dans les deux cas (i) et (ii) l'intégrité de l'ensemble des agents de

Variante	(i) A1	(ii) A2
1	m = 4.92, $\sigma = 0.29$	m = 4.83, $\sigma = 0.39$
2	m = 3.17, $\sigma = 1.19$	m = 1.00, $\sigma = 0.00$
3	m = 1.00, $\sigma = 0.00$	m = 1.17, $\sigma = 0.58$

TABLE 1 – Moyennes m et écarts types σ obtenus pour chacune des affirmations et chacune des variantes de rangement de bureau

Variantes	(i) A1	(ii) A2
1 et 2	t = 6.05	t = 23.36
1 et 3	t = 13.53	t = 22.34
2 et 3	t = 7.49	t = 1.0167

TABLE 2 – Valeurs t obtenues par les tests de Bonferroni sur chaque combinaison de variantes et sur (i) l'affirmation A1 et (ii) l'affirmation A2.

l'équipe. Les résultats obtenus ((i) $F(2,33) = 62, p < 0.05$ et (ii) $F(2,33) = 349, p < 0.05$) montrent que les sujets ont une appréciation significativement différente de (i) l'efficacité de l'équipe et (ii) la participation de x au bon fonctionnement de l'équipe dans les trois variantes du scénario proposées. Pour vérifier que les sujets distinguent bien les variations des comportements des agents dans chacune de ses variantes, nous avons ensuite appliqué plusieurs comparaisons paire par paire avec le test de Bonferroni dont les résultats sont présentés dans le Tableau 2.

Les différences dans l'évaluation (i) du comportement de l'équipe par les sujets sont donc fortement significatives pour toutes les variantes proposées. Les différences dans l'appréciation (ii) de comportement de l'agent x sont significatives dans les variantes 1 et 2, et 1 et 3; mais non significatives dans les variantes 2 et 3, ce qui était attendu puisque l'agent x a la même intégrité et se comporte de manière identique dans les deux variantes. Nos deux hypothèses H1 et H2 sont donc supportées. On peut néanmoins constater que l'efficacité perçue de l'équipe diminue significativement dès que l'un des agents n'est pas intègre.

Confiance. Nous avons ensuite cherché à vérifier que les observateurs pouvaient percevoir les relations de confiance entretenues entre les agents de par les comportements des agents. Nous nous sommes intéressés à la confiance en les compétences des autres, paramètre qui est essentiel dans le travail d'équipe. Notre formulons l'hypothèse suivante :

H3 : les sujets seront capables d'identifier le niveau de confiance d'un agent en les compétences d'un autre de par le comportement des agents.

Pour vérifier cette hypothèse, nous avons présenté un second scénario portant sur le sauvetage d'une victime par une équipe médicale formée par trois agents devant réaliser les tâches $\tau_1 = \text{préparer l'ambulance}$, $\tau_2 = \text{faire les premiers soins}$ et $\tau_3 = \text{rassembler et transmettre les informations}$ sur la victime au centre médical. Leurs compétences sont les suivantes :

– Un agent ambulancier (agent a) est compétent pour faire τ_1 , mais pas pour τ_2 et τ_3 , i.e. : $a_{a,\tau_1} = 1, a_{a,\tau_2} = a_{a,\tau_3} = 0$.

Variante	(iii) A3	(iv) A4
1	$m = 4.92, \sigma = 0.29$	$m = 4.92, \sigma = 0.29$
2	$m = 1.5, \sigma = 0.91$	$m = 3.08, \sigma = 1.73$

TABLE 3 – Moyennes m et écarts types σ obtenus pour chacune des affirmations et chacune des variantes du scénario de sauvetage de victime

- Un agent infirmier (agent i) est compétent pour la tâche τ_2 , mais pas τ_1 ni τ_3 , i.e : $a_{i,\tau_2} = 1, a_{i,\tau_1} = a_{i,\tau_3} = 0$.
- Un agent médecin (agent m) est compétent pour toutes les tâches, i.e : $a_{m,\tau_1} = a_{m,\tau_2} = a_{m,\tau_3} = 1$.

Dans ce scénario, tous les agents sont intègres et ont des relations de bienveillance neutres. Les croyances des agents à propos de l'intégrité et la bienveillance des autres correspondent à la réalité (i.e. $\forall x, \forall y, \forall z, i_x = 1, b_{x,y} = 0.5, i_y^x = 1$ et $b_{y,z}^x = 0.5$). Seule la confiance qu'a le médecin en les compétences de ses équipiers varie :

Variante 1 : le médecin a confiance en les compétences de l'ambulancier et de l'infirmier (i.e. $a_{a,\tau_1}^m = a_{i,\tau_1}^m = 1$) et les laisse respectivement *préparer l'ambulance* et *faire les premiers soins* pendant qu'il s'occupe de *rassembler et transmettre les informations*.

Variante 2 : le médecin n'a pas confiance en les compétences de ses équipiers (i.e. $a_{a,\tau_1}^m = a_{i,\tau_1}^m = 0$), et commence par aider l'ambulancier pour *préparer l'ambulance*, puis aide l'infirmier à *faire les premiers soins* avant de *rassembler et transmettre les informations*.

Nous avons appliqué le même protocole expérimental que pour le premier scénario, et présenté aux sujets des affirmations portant sur la confiance du médecin (iii) en l'ambulancier (A3 : *Le médecin a confiance en l'ambulancier*) et (iv) en l'infirmier (A4 : *le médecin a confiance en l'infirmier*). Les résultats sont décrits dans le Tableau 3.

Nous avons appliqué des tests de Student sur (iii) la confiance perçue du médecin en l'ambulancier et (iv) la confiance perçue du médecin en l'infirmier. Les résultats ((iii) $p < 0.0001$ et (iv) $p = 0.004$) montrent que les différences d'appréciation entre les deux variantes sont significatives à $p < 0.01$. Notre hypothèse H3 est donc supportée. D'après les commentaires des participants, le fait que dans la seconde variante, la confiance du médecin en l'infirmier soit mieux jugée par les observateurs est dû au fait que le médecin aide d'abord l'ambulancier et ensuite l'infirmier.

Ces évaluations préliminaires nous ont permis de mettre en place un protocole qui pourra être utilisé pour une évaluation approfondie. Les scénarios utilisés sont très simples mais les résultats sont encourageants : nous arrivons à produire des "bons" et des "mauvais" comportements d'équipe, et les sujets sont capables de percevoir des variations de confiance d'un agent en les compétences d'un autre de par les comportements adoptés par les agents.

7 Conclusions

Nous avons proposé un modèle d'agent s'appuyant sur le modèle de confiance de [Mayer et al., 1995]. En s'inspirant de ce modèle, nous avons défini un moteur décision-

nel qui permet aux agents de tenir compte de la confiance qu'ils ont en chacun des autres agents tout en raisonnant sur un arbre de tâches spécifié en ACTIVITY-DL et décrivant une activité collective. Dans un premier temps les agents calculent l'importance qu'ils accordent aux buts ; puis raisonnent récursivement sur un arbre de tâches ACTIVITY-DL pour calculer l'utilité des tâches et en choisir une. Nous avons implémenté le modèle proposé et vérifié son bon fonctionnement pour des petites équipes d'agents et sur des arbres de tâches de taille conséquente. Nous avons conduit une évaluation préliminaire qui nous a permis de vérifier que les sujets apprécient différemment les comportements des agents produits avec notre modèle ; notamment, les sujets évaluent différemment la confiance d'un agent en les compétences d'un autre selon le comportement du premier.

Remerciements

Ces travaux sont menés dans le cadre d'une thèse financée par la Direction Générale de l'Armement (DGA) et le Labex MS2T, lui même financé par le gouvernement français au travers du programme Investissements d'avenir géré par l'Agence Nationale de la Recherche (ANR - référence ANR-11-IDEX-0004-02). Les auteurs ont égal bénéficié du soutien du projet VICTEAMS (ANR-14-CE24-0027), financé par l'ANR et la DGA.

Références

- [Barot, 2014] Barot, C. (2014). *Scénarisation d environnements virtuels. Vers un équilibre entre contrôle, cohérence et adaptabilité*. PhD thesis, Université de Technologie de Compiègne.
- [Castelfranchi and Falcone, 2001] Castelfranchi, C. and Falcone, R. (2001). *Social Trust : A Cognitive Approach*. J. Pitt. London : Wiley.
- [Gerbaud et al., 2007] Gerbaud, S., Mollet, N., and Araldi, B. (2007). *Virtual environments for training : from individual learning to collaboration with humans*. In *Technologies for E-Learning and Digital Entertainment*, pages 116–127. Springer.
- [Marsh, 1994] Marsh, S. P. (1994). *Formalising Trust as Computational Concept*.
- [Mayer et al., 1995] Mayer, R. C., Davis, J. H., and Schoorman, F. D. (1995). *An Integrative Model of Organizational Trust*. *The Academy of Management Review*, 20(3) :709.
- [Querrec et al., 2003] Querrec, R., Buche, C., Maffre, E., and Chevaillier, P. (2003). *Securevi : virtual environments for fire-fighting training*. In *5th virtual reality international conference (VRIC03)*, pages 169–175.
- [Sanselone et al., 2014] Sanselone, M., Sanchez, S., Sanza, C., Panzoli, D., and Duthen, Y. (2014). *Constrained control of non-playing characters using Monte Carlo Tree Search*. In *Computational Intelligence and Games (CIG), 2014 IEEE Conference on*, pages 1–8. IEEE.