Ecole Nationale Supérieure des Mines
SAINT-ETIENNE

N° d'ordre : 426 I

Thèse

présentée par

Guillaume Muller

Pour obtenir le grade de Docteur
de l'École Nationale Supérieure des Mines de Saint-Étienne

Spécialité : Informatique

Utilisation de normes et de réputations pour détecter et sanctionner les contradictions

–

Contribution au contrôle social des interactions dans les systèmes multi-agents ouverts et décentralisés

Soutenue à Saint-Étienne, le 11 décembre 2006

Membres du jury :

Présidente : Salima Hassas	
Rapporteurs :	
Yves Demazeau	Chargé de recherches, CNRS, Institut IMAG, Grenoble
Carles Sierra	Professeur, Universitat Autònoma de Barcelona
Directeurs de thèse :	
Olivier Boissier	Professeur, ÉNS des Mines, Saint-Étienne
Laurent Vercouter	Maître assistant, ÉNS des Mines, Saint-Étienne
Examinateurs :	
Salima Hassas	Professeur, Univ. Claude Bernard, Lyon 1
Andreas Herzig	Directeur de recherches, CNRS, IRIT, Toulouse
Juliette Rouchier	Chargée de recherches, CNRS, GREQAM, Marseille

Spécialités doctorales :	Responsables :
SCIENCES ET GÉNIE DES MATÉRIAUX	**J. DRIVER** Directeur de recherche - Centre SMS
MÉCANIQUE ET INGÉNIERIE	**A. VAUTRIN** Professeur - Centre SMS
GÉNIE DES PROCÉDÉS	**G. THOMAS** Professeur - Centre SPIN
SCIENCES DE LA TERRE	**B. GUY** Maître de recherche
SCIENCES ET GÉNIE DE L'ENVIRONNEMENT	**J. BOURGOIS** Professeur - Centre SITE
MATHÉMATIQUES APPLIQUÉES	**É. TOUBOUL** Ingénieur
INFORMATIQUE	**O. BOISSIER** Professeur - Centre G2I
IMAGE, VISION, SIGNAL	**JC. PINOLI** Professeur - Centre CIS
GÉNIE INDUSTRIEL	**P. BURLAT** Professeur - Centre G2I
MICROÉLECTRONIQUE	**Ph. COLLOT** Professeur - Centre CMP

Enseignants-chercheurs et chercheurs autorisés à diriger des thèses de doctorat (titulaires d'un doctorat d'État ou d'une HDR)

BATTON-HUBERT	Mireille	MA	Sciences & Génie de l'Environnement	SITE
BENABEN	Patrick	PR 2	Sciences & Génie des Matériaux	SMS
BERNACHE-ASSOLANT	Didier	PR 1	Génie des Procédés	CIS
BIGOT	Jean-Pierre	MR	Génie des Procédés	SPIN
BILAL	Essaïd	MR	Sciences de la Terre	SPIN
BOISSIER	Olivier	PR 2	Informatique	G2I
BOUDAREL	Marie-Reine	MA	Sciences de l'inform. & com.	DF
BOURGOIS	Jacques	PR 1	Sciences & Génie de l'Environnement	SITE
BRODHAG	Christian	MR	Sciences & Génie de l'Environnement	SITE
BURLAT	Patrick	PR 2	Génie industriel	G2I
COLLOT	Philippe	PR 1	Microélectronique	CMP
COURNIL	Michel	PR 1	Génie des Procédés	SPIN
DAUZERE-PERES	Stéphane	PR 1	Génie industriel	CMP
DARRIEULAT	Michel	ICM	Sciences & Génie des Matériaux	SMS
DECHOMETS	Roland	PR 2	Sciences & Génie de l'Environnement	SITE
DESRAYAUD	Christophe	MA	Mécanique & Ingénierie	SMS
DELAFOSSE	David	PR 2	Sciences & Génie des Matériaux	SMS
DOLGUI	Alexandre	PR 1	Informatique	G2I
DRAPIER	Sylvain	PR 2	Mécanique & Ingénierie	CIS
DRIVER	Julian	DR	Sciences & Génie des Matériaux	SMS
FOREST	Bernard	PR 1	Sciences & Génie des Matériaux	SMS
FORMISYN	Pascal	PR 1	Sciences & Génie de l'Environnement	SITE
FORTUNIER	Roland	PR 1	Sciences & Génie des Matériaux	CMP
FRACZKIEWICZ	Anna	MR	Sciences & Génie des Matériaux	SMS
GARCIA	Daniel	CR	Génie des Procédés	SPIN
GIRARDOT	Jean-Jacques	MR	Informatique	G2I
GOEURIOT	Dominique	MR	Sciences & Génie des Matériaux	SMS
GOEURIOT	Patrice	MR	Sciences & Génie des Matériaux	SMS
GRAILLOT	Didier	DR	Sciences & Génie de l'Environnement	SITE
GROSSEAU	Philippe	MR	Génie des Procédés	SPIN
GRUY	Frédéric	MR	Génie des Procédés	SPIN
GUILHOT	Bernard	DR	Génie des Procédés	CIS
GUY	Bernard	MR	Sciences de la Terre	SPIN
GUYONNET	René	DR	Génie des Procédés	SPIN
HERRI	Jean-Michel	PR 2	Génie des Procédés	SPIN
KLÖCKER	Helmut	CR	Sciences & Génie des Matériaux	SMS
LAFOREST	Valérie	CR	Sciences & Génie de l'Environnement	SITE
LE COZE	Jean	PR 1	Sciences & Génie des Matériaux	SMS
LI	Jean-Michel	EC (CCI MP)	Microélectronique	CMP
LONDICHE	Henry	MR	Sciences & Génie de l'Environnement	SITE
MOLIMARD	Jérôme	MA	Sciences & Génie des Matériaux	SMS
MONTHEILLET	Frank	DR 1 CNRS	Sciences & Génie des Matériaux	SMS
PERIER-CAMBY	Laurent	MA1	Génie des Procédés	SPIN
PIJOLAT	Christophe	PR 1	Génie des Procédés	SPIN
PIJOLAT	Michèle	PR 1	Génie des Procédés	SPIN
PINOLI	Jean-Charles	PR 1	Image, Vision, Signal	CIS
STOLARZ	Jacques	CR	Sciences & Génie des Matériaux	SMS
SZAFNICKI	Konrad	CR	Sciences de la Terre	SITE
THOMAS	Gérard	PR 1	Génie des Procédés	SPIN
TRAN MINH	Cahn	MR	Génie des Procédés	SPIN
VALDIVIESO	Françoise	CR	Génie des Procédés	SPIN
VALDIVIESO	François	MA	Sciences & Génie des Matériaux	SMS
VAUTRIN	Alain	PR 1	Mécanique & Ingénierie	SMS
VIRICELLE	Jean-Paul	CR	Génie des procédés	SPIN
WOLSKI	Krzysztof	CR	Sciences & Génie des Matériaux	SMS
XIE	Xiaolan	PR 1	Génie industriel	CIS

Glossaire :

PR 1	Professeur 1ère catégorie
PR 2	Professeur 2ème catégorie
MA(MDC)	Maître assistant
DR 1	Directeur de recherche
Ing.	Ingénieur
MR(DR2)	Maître de recherche
CR	Chargé de recherche
EC	Enseignant-chercheur
ICM	Ingénieur en chef des mines

Centres :

SMS	Sciences des Matériaux et des Structures
SPIN	Sciences des Processus Industriels et Naturels
SITE	Sciences Information et Technologies pour l'Environnement
G2I	Génie Industriel et Informatique
CMP	Centre de Microélectronique de Provence
CIS	Centre Ingénierie et Santé

Remerciements

Je tiens à remercier ici tout un ensemble de personnes grâce à qui, directement ou indirectement, j'ai pu mener ce travail à bien.

Je commence par remercier mon jury de thèse pour avoir pris le temps d'évaluer et de critiquer ce travail de manière approfondie et constructive.

Je remercie particulièrement Laurent et Olivier pour m'avoir accompagné et guidé durant ces quatre années. Je leur suis reconnaissant autant du point de vue scientifique que pour le soin apporté à mon intégration dans le milieu de la recherche. Je remercie aussi tous les personnels de l'équipe SMA (ou qui y ont été attachés) avec qui j'ai partagé ces moments inoubliables : Philippe, Franck, Max, Arnaud, Cosmin, Steven, Rahee, Julien, Amandine, Liliane... Ainsi que les membres du service info : JF, Niloo, Dom...

Merci aussi à l'équipe EURISE de l'Université Jean Monnet pour m'avoir accueilli et financé à différentes occasions. En particulier : François Jacquenet, Colin de la Higuera, Rémi, Momo, Thierry, Abdallah, Henri-Maxime...

Je remercie aussi l'ÉSIL et particulièrement le « service info » : G, Fred et Dimitri pour m'avoir rapidement mis à disposition le matériel m'ayant permis de finaliser mes simulations.

Merci aux chercheurs du projet ART-testbed pour les discussions passionnantes que nous avons eues ainsi que pour leur travail sur la plate-forme. En particulier Karen, Jordi et Tomas qui ont tenu bon depuis le début malgré le travail que cela leur a demandé.

Je remercie les différents membres de l'ASEC et de la CJC qui m'ont permis de m'insérer plus « administrativement » dans le milieu de la recherche en me permettant d'en comprendre les rouages.

Merci aux membres d'ALOLISE pour toutes les permanences, install parties, discussions, débats et soirées passionnantes que nous avons partagés : Greg, Kubi...

À titre personnel, je remercie tout particulièrement Natacha, Manu Geoffrey et Rodolphe de l'EMSE pour m'avoir offert les nombreux entraînements sportifs qui m'ont permis de me défouler et de me vider l'esprit aux moments où j'en avais le plus besoin. J'en profite aussi pour remercier Rachell pour les discussions et sorties que nous avons pu avoir ensemble durant ses (trop courts) séjours en France.

Je remercie ensuite les personnes qui m'ont accueillies à Aide-et-Action pendant près d'un an avant que je ne m'engage dans cette thèse. Je pense en particulier à Nançouille, Sylvoune, Isa.

Je dois énormément à Stéphanie Carrière car, sans elle, j'aurais probablement abandonné le milieu de la recherche avant même de l'avoir vraiment abordé. Un grand « merci » donc à Steph', tant pour ça que pour tout le reste !

Je remercie aussi labandedes4 qui m'ont accompagnés tout au long des mes études pour les soirées passionnantes de débats et de résolution de projets. Merci Joe, Meuonf et Régis, sans oublier Gallan.

Merci au différents membres du « royal sa mère » et des Invalides (Raph', Vinz, Deva, Stef, ...) pour m'avoir sorti un peu du milieu de la recherche et pour avoir toujours été là quand j'en avais besoin.

Je remercie mes parents, mes frères, ma sœur pour leur aide morale et financière. Merci aussi à tit Tom pour ses jolis sourires que j'ai eu tous les jours à côté de moi depuis qu'il est né.

Et enfin tous ceux que j'oublie parce que je ne les ai pas vu depuis trop longtemps.

Résumé

Les Systèmes Multi-Agents Ouverts et Décentralisés (SMAOD) sont particulièrement vulnérables à l'introduction d'agents mal conçus ou malveillants. Il est donc nécessaire de contrôler ces systèmes. Dans cette thèse, nous proposons le modèle L.I.A.R., permettant aux agents eux-mêmes de mettre en place un contrôle des interactions des autres agents, à l'aide d'un modèle de réputation. Ce modèle permet d'abord aux agents de représenter les interactions qu'ils perçoivent grâce à des engagements sociaux, ainsi que de modéliser les règles que chaque agent doit respecter à l'aide de normes sociales. En comparant les comportements qu'ils ont observés aux normes dont ils ont connaissance, les agents sont capables d'évaluer leurs pairs et d'estimer les niveaux de réputation qu'ils leur associent. Ensuite, les agents peuvent décider des sanctions à appliquer en s'appuyant sur les niveaux de réputation ainsi estimés. Grâce à l'intégration des deux phases : évaluation des comportements et décision des sanctions à appliquer, le modèle L.I.A.R. permet de mettre en place un contrôle social des interactions entièrement automatisé. Diverses expérimentations ont été menées avec ce modèle dans le cadre d'un réseau pair-à-pair, afin de montrer comment les agents contrôlent les interactions de leurs pairs.

Mot-clefs : Système Multi-Agent, Contrôle Social, Confiance, Réputation, Norme Sociale, Engagement Social.

Abstract

Open and Decentralized Multi-Agent Systems (ODMAS) are particularly vulnerable to the introduction of badly designed or malevolent agents. It is therefore necessary to control such systems. In this thesis, we propose the L.I.A.R. model, which enables agents to control their peers' interactions, thanks to a reputation model. Agents equiped with the L.I.A.R. model can first, represent interactions they perceive with the help of a social commitment model. They can also model the rules that each agent should follow thanks to a model of social norms. By comparing observed behaviours with the norms they know, agents are able to evaluate their peers and to estimate a reputation level to associate to each of them. Agents are then able to make a decision about the sanctions they wish to apply to their peers, based on these levels of reputation. Thanks to the complete integration of both steps: evaluation of the perceived behaviours and decision of the sanctions to apply, the L.I.A.R. model allows the agents to establish a fuly automatic social control of agents' interactions. Various experimentations have been conducted with this model in a peer-to-peer context in order to show how agents where able control their peers' interactions.

Keywords : Multi-Agent System, Social Control, Trust, Reputation, Social Norm, Social Commitment.

Table des matières

Table des figures

Liste des tableaux

Chapitre 1

Introduction

Dans ce chapitre, nous présentons la problématique générale de la thèse. Après avoir présenté les motivations de ce travail de recherche et le contexte scientifique dans lequel il se place – les Systèmes Multi-Agents Ouverts et Décentralisés (SMAOD) –, nous en précisons les objectifs et contributions. Enfin, nous détaillons l'organisation du manuscrit.

1.1 Motivations

Dans cette thèse, nous nous intéressons aux Systèmes Multi-Agents Ouverts et Décentralisés (SMAOD). Ces systèmes sont **ouverts** car capables d'accepter, à tout instant, l'entrée, la sortie ou la modification des caractéristiques des entités qui le constituent. Ils sont **décentralisés**, car il n'existe pas de point central, ni pour le stockage des données, ni pour le contrôle du système [BGG04]. Ainsi, dans un système multi-agent ouvert et décentralisé, n'*importe quel agent* peut entrer, sortir ou se modifier *à tout instant*.

Pour concevoir de tels systèmes et, en particulier, pour garantir que n'importe quel agent pourra y entrer, il est nécessaire de faire le moins d'hypothèses possible sur l'implémentation interne des agents. En conséquence, des agents mal programmés ou programmés avec des intentions malveillantes peuvent s'introduire dans le système. Or, de tels agents peuvent rapidement mettre à mal l'ensemble du système, en particulier si la gestion des tâches collectives, inhérentes aux systèmes décentralisés, n'est plus assurée correctement. Il est donc nécessaire de **contrôler** le comportement des agents présents dans le système.

1.2 Contexte Scientifique

Il existe différentes propositions pour aborder cette problématique : les approches sécuritaires, les approches organisationnelles et les approches sociales.

La première approche consiste à adapter les solutions existantes dans le domaine de la sécurité informatique [SFJ02, BFL96, BFK99]. Il s'agit d'empêcher les agents d'avoir un comportement nuisible en leur imposant des protocoles d'interaction sécurisés et l'implémentation de certaines primitives. Ces contraintes imposées sur la façon dont les agents interagissent et sur leur implémentation interne réduisent l'ouverture du système, ainsi que l'autonomie des agents. Par ailleurs, du fait qu'elles ne s'intéressent qu'aux moyens d'échanger de l'information et non à l'information elle-même, elles ne permettent pas de résoudre certains problèmes, comme la possibilité que des agents mentent. Enfin, la phase initiale d'échange de clefs cryptographiques requise par ces méthodes les rend difficiles à mettre en place dans les systèmes décentralisés.

La deuxième approche du contrôle consiste à définir formellement les organisations dans lesquelles les agents évoluent, en termes de rôles qu'ils peuvent jouer, de hiérarchies entre ces rôles... Des institutions sont mises en place pour contrôler que ces organisations sont bien respectées [BD96, Fox81, Cor83, PCL87, Han03]. Ces institutions sont habituellement implémentées de manière centralisée et doivent généralement disposer de pouvoirs particuliers pour pouvoir sanctionner les agents, par exemple pour les exclure physiquement du système.

La dernière approche envisagée s'appuie sur des modèles de confiance et de réputation [CF98, Rou00, MC01, Sab02, CP02, MD05]. Chaque agent évalue les comportements des autres agents et en déduit le niveau de réputation qu'il leur associe. Il peut alors décider, en fonction de ce niveau, s'il souhaite continuer à interagir en confiance ou non avec eux. Cette approche est particulièrement adaptée aux SMAOD, car elle permet de définir un contrôle *social* des agents, c'est-à-dire un contrôle adaptatif et auto-organisé, mis en place par les agents eux-mêmes [COTZ00]. Cependant, la majeure partie des modèles de réputation actuels nécessitent une déclaration *a priori* du comportement que comptent avoir les agents ou reposent sur une détection triviale des violations, pour calculer automatiquement l'évaluation des agents.

1.3 Objectifs et contributions

L'objectif général de cette thèse est de mettre en place un contrôle des interactions des agents adapté aux SMAOD, où les agents ne disposent pas nécessairement de déclarations *a priori* et explicites du comportement que les autres agents comptent manifester. Nous cherchons donc à déployer un système de contrôle social des interactions des agents s'appuyant sur un modèle de réputation.

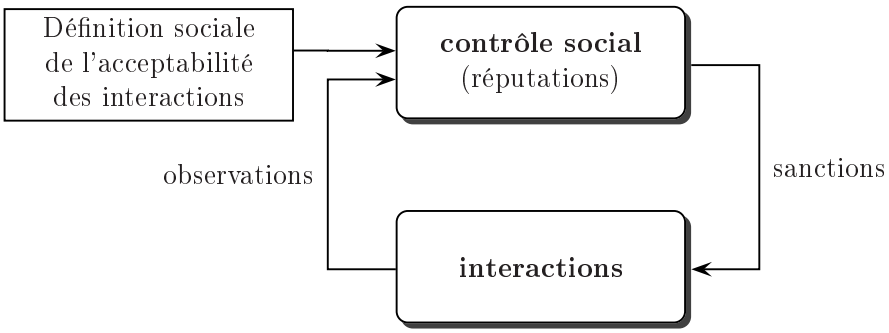


FIG. 1.1 – Contrôle *social* de l'interaction.

La figure 1.1 présente une vision automaticienne du contrôle. Celui-ci prend la forme d'une boucle de rétroaction, caractérisant l'apprentissage des réputations. En s'appuyant sur des observations des interactions et sur des définitions sociales de l'acceptabilité des interactions, le système de contrôle doit pouvoir, d'une part, caractériser les interactions observées et, d'autre part, décider des sanctions à appliquer en conséquence.

Mettre en place un tel contrôle se décompose donc en deux sous-objectifs majeurs :

- **Caractériser une interaction**. Cette phase requiert d'être en mesure de :
  - *Modéliser les interactions*,
  - *Définir l'acceptabilité des interactions*,
  - *Estimer l'acceptabilité d'une interaction observée.*
- **Sanctionner une interaction**. Cette phase correspond à décider des sanctions à appliquer.

La cohérence entre ces quatre points est assurée par le modèle de réputation qui s'appuie sur l'estimation de l'acceptabilité d'une interaction pour

réviser le niveau de la réputation d'un agent et utilise le niveau de réputation ainsi apprécié pour décider des sanctions à appliquer. Pour chacun de ces sous-objectifs, nous étudions les modèles existants, leurs limites dans un cadre ouvert et décentralisé et nous proposons finalement un canevas général, le modèle L.I.A.R., intégrant des modèles adaptés aux SMAOD.

1.4 Organisation du manuscrit

Ce manuscrit est organisé en trois parties. La première partie examine le contrôle social, ainsi que les concepts de confiance et de réputation, à la lumière de notre problématique. La deuxième partie présente le modèle de réputation L.I.A.R., que nous proposons pour le contrôle social des SMAOD. Enfin, la troisième partie aborde la mise en œuvre pratique et l'évaluation de ce modèle.

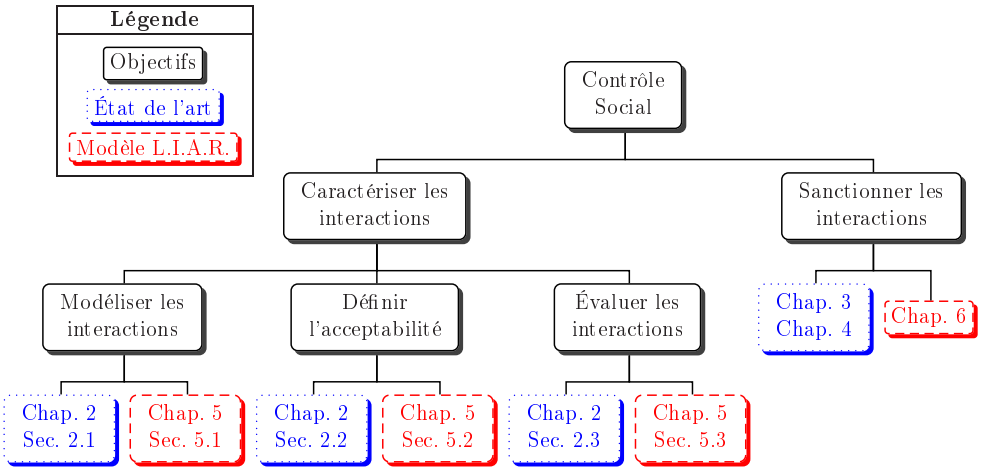


FIG. 1.2 – Organisation du manuscrit.

La figure 1.2 présente l'organisation des deux premières parties de ce manuscrit en regard des objectifs énoncés précédemment.

Le chapitre 2 étudie de manière générale les différentes composantes du contrôle social. Nous commençons par étudier la phase de caractérisation des interactions : dans la section 2.1, nous nous intéressons à la modélisation des interactions des agents puis, dans la section 2.2, aux moyens de définir l'acceptabilité des interactions ; enfin, dans la section 2.3, nous voyons par quels processus évaluer une interaction.

Nous nous intéressons ensuite à la phase de sanction des interactions et étudions alors plus particulièrement les sanctions mises en place grâce aux concepts de confiance et de réputation. Dans le chapitre 3, nous étudions ces deux notions dans les domaines des sciences économiques, humaines et sociales et en tirons une spécification précise. Dans le chapitre 4, nous étudions les modèles computationnels de réputation et montrons que la plupart d'entre eux sont limités par la portée de leur processus d'évaluation des interactions.

Cherchant à combler cette lacune, nous proposons, dans la deuxième partie de ce manuscrit, le modèle L.I.A.R. (« Liar Identification for Agent Reputation »). Le chapitre 5 met l'accent sur la phase de caractérisation des interactions de L.I.A.R. Il décrit un modèle d'engagement social permettant aux agents de modéliser les interactions qu'ils observent (section 5.1), un modèle de normes permettant aux agents de définir l'acceptabilité d'une interaction (section 5.2) et un processus de détection décentralisée de la violation (ou du respect) des normes (section 5.3) permettant aux agents d'établir une évaluation des interactions qu'ils ont observées. Le chapitre 6 s'intéresse à la phase de sanction et présente le modèle de réputation, lequel permet aux agents de décider de continuer ou non à interagir avec les autres agents.

Enfin, la troisième partie (non représentée sur la figure) décrit un scénario d'échange d'information dans un réseau pair-à-pair et les résultats d'expérimentations menées avec le modèle L.I.A.R. dans ce cadre.

Première partie

État de l'art

Chapitre 2

Contrôle social des interactions

Pour mener à bien un contrôle social des interactions dans un SMAOD, les agents doivent tout d'abord être en mesure de *modéliser les interactions qu'ils observent* et de *définir l'acceptabilité de interactions*. Ils peuvent ainsi *évaluer* si les autres agents se comportent correctement ou non et peuvent décider des *sanctions* à leur appliquer en conséquence.

L'objectif de ce chapitre est d'étudier les différents constituants (en italique dans le paragraphe précédent) du contrôle social dans le cadre des systèmes multi-agents ouverts et décentralisés. Nous étudions les solutions proposées dans la littérature sous la contrainte principale de la décentralisation, c'est-à-dire du partage des données et du contrôle entre les agents du système.

Les sections suivantes présentent tout d'abord les modèles d'interaction, qui permettent une observation, par les agents, des interactions. Ensuite, la manière dont l'acceptabilité des interactions peut être définie est exposée. Différents processus par lesquels un agent peut évaluer une interaction sont alors étudiés. Enfin, nous abordons les différents types de sanctions par lesquelles un agent peut influer sur les autres de façon à ce qu'ils modifient leurs interactions pour les rendre acceptables.

2.1 Observation de l'interaction

Dans cette section, nous étudions la littérature concernant la modélisation des interactions. Il s'agit d'examiner les modèles existants dans la perspective de dégager les modèles adaptés à une observation, par les agents eux-mêmes,

de ces interactions. Cette étude, nous permet de poser le premier jalon du contrôle social en déterminant les modèles adaptés à la modélisation du premier type d'entrées nécessaires à la phase de détection du contrôle social.

2.1.1 Motivations

La communication est l'un des principaux moyens dont disposent les agents pour interagir et, plus généralement, agir sur le monde [DvL97, Hab84]. Dans cette section, nous étudions principalement les interactions entre agents effectuées par le biais de la communication.

Dans le cadre d'entités logicielles telles que les agents, la théorie de l'information [SW49], qui ne considère que des échanges de bits, n'est plus suffisante pour décrire les communications. La description des communications des agents doit être plus riche ; nous nous intéressons donc ici à des travaux qui cherchent à leur donner un sens (une sémantique). La théorie des actes de langage [Aus62, Sea69], issue de la recherche en linguistique, perçoit les communications comme des actes et est à la source de la plupart des travaux dans les systèmes multi-agents [CL95]. Ces derniers peuvent être classés selon leur approche : par texte brut, par protocole, cognitiviste, sociale et argumentative.

L'approche par texte brut décrit les communications directement en langue naturelle. Elle n'est toutefois pas assez formelle et trop ambiguë pour être employée directement par des agents logiciels [Sin00].

Dans l'approche par protocole [Smi80, DM90, Cd90], le sens d'un acte de langage est essentiellement défini par l'usage qui en est fait dans des protocoles d'interaction. Cette approche s'intéresse moins au sens de l'acte qu'à sa position dans le dialogue [Sin91]. Elle limite l'ouverture du système en empêchant un agent qui ne connaît pas le protocole d'entrer en interaction avec les autres. D'autre part, le dynamisme du système est entravé par la faible flexibilité des protocoles.

L'approche cognitiviste [CL95, Lab96, FIP02], quant à elle, donne un sens à un acte de langage en s'appuyant sur les états mentaux des agents. Cette approche ne permet donc d'aborder que le sens subjectif [Hab84] d'une communication et, par conséquent, s'oppose à la nature publique des communications [Sin98].

Dans l'approche sociale [Sin91, Sin00], chaque fois qu'un agent émet un acte de langage, un engagement social est créé. Des engagements sociaux peuvent être générés pour donner une sémantique à la communication selon

toutes les dimensions : objective, subjective et pratique [Hab84]. D'autre part, en tant que formalisme externe aux agents, les engagements sociaux sont particulièrement adaptés à l'observation des communications.

L'approche argumentative [Dun94, AC98] s'appuie généralement sur des engagements sociaux et un système d'argumentation (la dialectique [Ham70]) dans le but de modéliser le dialogue dans son ensemble. Bien que pouvant être utilisée pour donner un sens aux communications, cette approche va bien au-delà de la simple sémantique.

Dans le cadre de cette thèse, nous nous concentrons sur l'approche sociale. En effet, du fait du caractère externe des engagements sociaux, il s'agit de l'approche la plus adaptée aux systèmes décentralisés.

Diverses spécifications fonctionnelles des engagements sociaux ont été proposées. Dans les sections suivantes, nous étudions ces différentes spécifications d'un point de vue progressif : nous voyons tout d'abord la proposition originelle de [Sin91, Sin00], qui établit une traduction des actes de langage en engagements sociaux, puis les différents enrichissements apportés à ce modèle.

2.1.2 Engagement social pour l'interaction

En s'appuyant sur la logique temporelle, [Sin00] propose le formalisme suivant pour modéliser les engagements sociaux :

Définition 2.1.1 *Un **engagement social** est un quadruplet :*

$$\texttt{SCom}(\texttt{db}, \texttt{cr}, \texttt{wit}, \texttt{cont})$$

- `db` *est le débiteur, l'agent qui a l'engagement social ;*
- `cr` *est le créditeur, l'agent envers qui l'engagement social est pris ;*
- `wit` *est le témoin de l'engagement social, qui pourra sanctionner le débiteur si l'engagement n'est pas tenu ;*
- `cont` *représente le contenu sur lequel le débiteur est engagé.*

[Sin00] et [FC02, FC03] fournissent des tables de conversion entre des actes de langage de différentes classes [Aus62, Sea69, Ven72] en des engagements sociaux. Pour la plupart de ces classes, le débiteur est associé à l'émetteur de l'acte de langage, le créditeur à son récepteur et le contexte est le

système multi-agent dans lequel l'engagement social est pris. L'exemple 2.1.1 illustre l'émission d'un acte de langage de la classe représentative (le `inform`).

Exemple 2.1.1 *Lorsqu'un agent* `x` *informe un agent* `y` *sur un fait* `p` *devant un groupe de témoins* $\mathcal{G}$*, il s'engage auprès de* `y` *sur le fait que* `p` *est vrai, ce qui est caractérisé par la création de l'engagement social suivant :* $\mathbf{C}(\mathtt{x}, \mathtt{y}, \mathcal{G}, \mathtt{p})$.

[Sin00] montre comment son formalisme peut être employé pour vérifier la sémantique des communications selon les trois dimensions [Hab84] : objective (la communication est vraie), subjective (la communication est sincère) et pratique (la communication est justifiée). Cependant, ce formalisme ne prend pas en compte le fait que les engagements sociaux peuvent évoluer.

2.1.3 Engagement social avec cycle de vie

Il est courant qu'au cours d'un dialogue [WK95] un agent revienne sur ses dires ou affine son discours. Les engagements sociaux de cet agent doivent évoluer en conséquence. Il est alors nécessaire de garder une trace de la situation dans laquelle se trouve un engagement social à un instant donné. [FC03, BMCD03, PFCd04] proposent des modèles d'engagement social intégrant la notion d'*état* dans le formalisme. L'évolution de l'engagement social est alors descriptible à l'aide d'un cycle de vie définissant les *transitions* entre ces états.

États

Dans [FC03], l'état peut prendre ses valeurs parmi :
- `empty` : l'engagement social n'a pas encore été créé ;
- `unset` : l'engagement social vient d'être créé ;
- `pending` : l'engagement social est créé mais en attente d'activation ;
- `active` : l'engagement social est actif ;
- `cancelled` : l'engagement social a été annulé ;
- `fulfilled` : l'engagement social a été rempli ;
- `violated` : l'engagement social n'a pas été rempli dans les conditions imparties.

Seuls les trois premiers états ne sont pas communs à tous les modèles présentés ici. Dans [PFCd04] l'état `empty` n'existe pas. Dans [BMCD03] les états

`empty` et `pending` n'existent pas. Chez [PFCd04], l'état `unset` est confondu avec l'état `pending`.

Transitions

Un engagement social évolue au fil du temps. Son évolution est descriptible à l'aide d'un cycle de vie définissant les transitions entre les états.

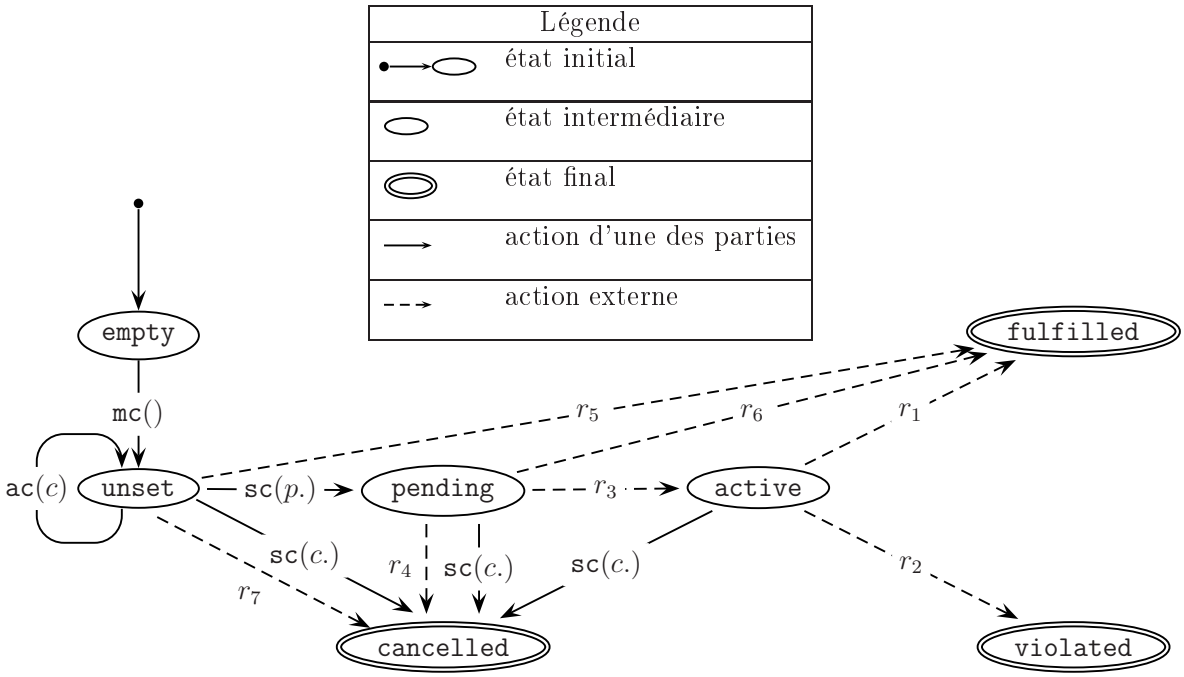


FIG. 2.1 – Cycle de vie d'un engagement social.

[FC03] propose le cycle de vie décrit par le diagramme d'états de la figure 2.1. Deux types d'actions peuvent déclencher le changement d'état d'un engagement social : une action d'une des parties engagées dans le dialogue ou bien l'action du temps. Les traits pleins représentent des actions effectuées sur l'engagement social suite à une action de l'une des parties engagées dans le dialogue. Les traits en pointillés représentent une action externe et sont liés à des règles de transition notées r_i.

Pour refléter les évolutions dues aux actions menées par les parties du dialogue, les engagements sociaux peuvent être modifiés à l'aide de trois fonctions. La fonction `ac`(c) (« add condition ») ajoute la condition c à l'activation de l'engagement social. La fonction `mc`() (« make commitment »)

permet de créer l'engagement social. Enfin, la fonction `sc`(...) (« set commitment ») permet de changer l'état d'un engagement social. `sc`($p.$) passe l'engagement social en état `pending`, `sc`($c.$) passe l'engagement social en état `cancelled`. Cette dernière fonction est la plus utilisée pour gérer le cycle de vie d'un engagement social.

Si le contenu de l'engagement devient vrai, alors l'engagement social passe en état `fulfilled` (règles r_1, r_5 et r_6). Au contraire, s'il devient faux, mais que la condition d'activation est vérifiée, alors l'engagement social est violé (état `violated`, règle r_2). Un engagement social dans l'état `pending` peut soit devenir actif (état `active`) si sa condition d'activation devient vraie (règle r_3), soit être annulé (état `cancelled`) si sa condition d'activation devient fausse (règle r_4). Finalement, la règle r_7 stipule qu'à l'expiration d'un certain délai, un engagement dans l'état `unset` devient `cancelled` s'il n'a pas été mis en attente d'activation (`pending`) ou rempli (`fulfilled`).

Dans [BMCD03, PFCd04], les cycles de vie diffèrent peu de celui présenté ici et principalement du fait des états inexistants. Dans [PFCd04] l'engagement social est directement créé en état `pending` ; avant il n'existe tout simplement pas. Dans [BMCD03] l'engagement social est directement créé en état `active` quand la condition d'activation est vérifiée.

Grâce à l'introduction de la notion d'état, il est possible de gérer l'évolution d'un engagement social. Cependant, cet état ne permet pas, à lui seul, de prendre en compte toute la dynamique d'un dialogue.

2.1.4 Engagement social avec cycle de vie du contenu

Au cours d'un dialogue, un agent pourrait être amené à s'engager à un instant sur un fait qui ne pourra être vérifié que dans le futur. Par exemple, un agent peut s'engager *maintenant* sur le fait qu'il pleuvra *demain*. Un engagement social et son contenu sont donc des éléments temporellement distincts. En conséquence, [BMCD03] propose de modéliser l'engagement social et son contenu comme des entités distinctes, pouvant être dans des états différents et avoir des cycles de vie propres.

En différenciant ainsi le cycle de vie du contenu de l'engagement social de celui de l'engagement social lui-même, des agents peuvent argumenter sur le contenu d'un engagement, *i.e.* faire avancer l'engagement dans son cycle de vie, sans pour autant faire évoluer le contenu et, réciproquement, modifier le contenu d'un engagement social sans forcément modifier l'engagement social lui-même.

[BMCD03] propose un modèle d'engagement social qui tient compte de cette distinction engagement / contenu. Le contenu est alors associé à un état et à un temps de validité, tous deux différents de celui de l'engagement. L'exemple 2.1.2 illustre le formalisme employé.

Exemple 2.1.2 *Un engagement social de contenu* `cont` *est noté comme suit :*

$$\texttt{SCom}(id_n, \texttt{db}, \texttt{cr}, \texttt{S}_{\texttt{SCom}}, \texttt{t}_{\texttt{SCom}}, \texttt{cont}, \texttt{S}_{\texttt{cont}}, \texttt{t}_{\texttt{cont}})$$

Où id_n *est un identifiant unique pour l'engagement social,* `db` *et* `cr` *sont, respectivement, le débiteur et le créditeur.* $\texttt{S}_{\texttt{SCom}}$ *est l'état de l'engagement social,* $\texttt{t}_{\texttt{SCom}}$ *est le temps de validité associé à l'engagement social.* `cont` *est le contenu de l'engagement social,* $\texttt{S}_{\texttt{cont}}$ *est l'état du contenu et, enfin,* $\texttt{t}_{\texttt{cont}}$ *est le temps de validité du contenu.*

La figure 2.2 présente, la reconstitution du cycle de vie du *contenu* d'un engagement à partir des descriptions données dans [BMCD03].

Le contenu de l'engagement social est initialement soumis (`submitted`) par le débiteur (`db` dans la figure). Il peut ensuite être directement accepté ou refusé (`accept` ou `refuse`) par le créditeur (`cr` dans la figure) et arrive alors dans l'un des états terminaux `accepted` ou `refused`. Une argumentation peut aussi avoir lieu : le contenu peut être alternativement remis en cause par le créditeur et défendu par le débiteur (actions `question` et `defend`, états `questioned` et `justified`). Finalement, le débiteur peut modifier le contenu de son engagement (action `change`, état `changed`). Ces modifications sur le contenu de l'engagement social ont lieu suite à des actions communicatives menées par les parties engagées dans le dialogue. Ces actions sont donc liées à la génération de nouveaux engagements sociaux, certains générés par le débiteur, d'autres par le créditeur. La table 2.1 décrit l'ensemble des actions possibles en fonction de l'acteur (débiteur ou créditeur) et de la cible (engagement social ou son contenu).

2.1.5 Engagement social avec sanctions

Afin de s'assurer que les agents respectent leurs engagements sociaux, [PFCd04] propose de punir les agents qui ne respectent pas leurs engagements et de favoriser ceux qui les respectent. Pour ce faire, les auteurs proposent un modèle d'engagement social qui intègre les sanctions (positives ou négatives)

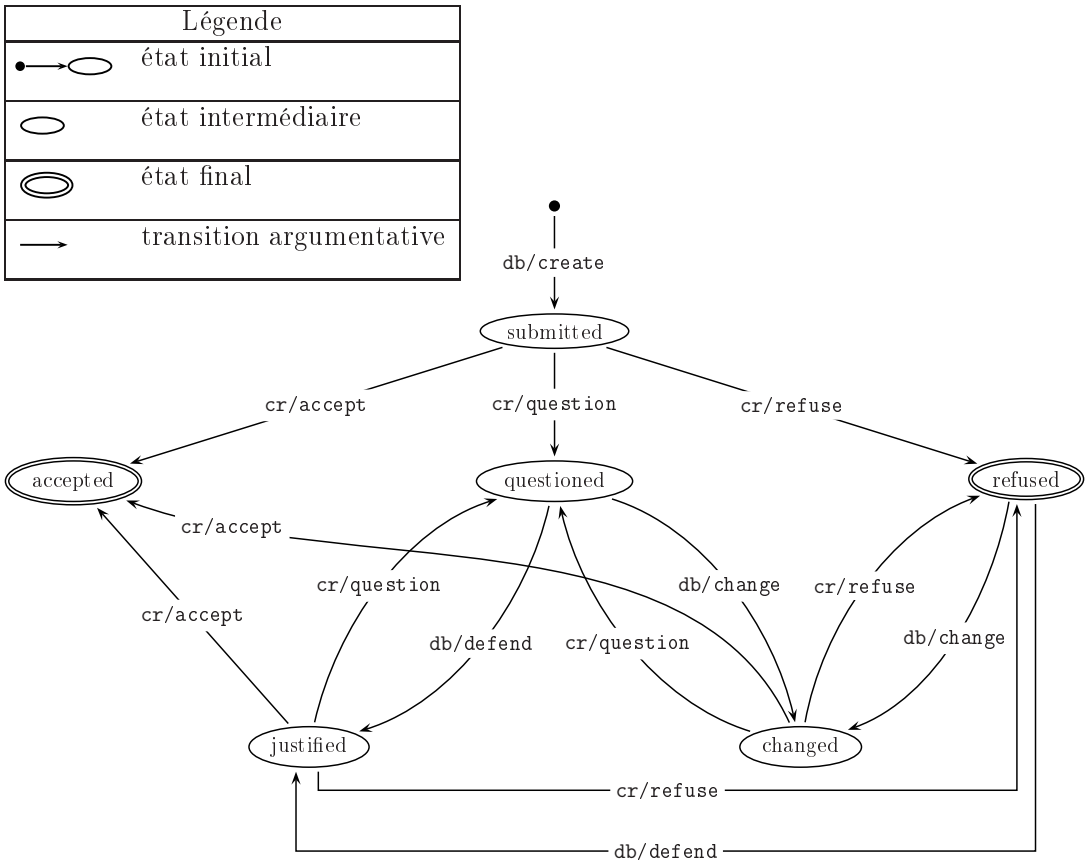


FIG. 2.2 – Cycle de vie du *contenu* d'un engagement social.

	débiteur	créditeur
Engagement social	create, withdraw, violate, fulfill.	—
Contenu	change, defend.	refuse, accept question.

TAB. 2.1 – Actions possibles sur un engagement social ou son contenu.

encourues par les différentes parties en fonction de l'état de l'engagement social.

Pour pouvoir mettre en place ce système de sanction, les auteurs redé-

finissent certaines transitions du cycle de vie de l'engagement social et en définissent de nouvelles. Les transitions redéfinies sont les suivantes :

- Au moment de la **création** d'un engagement social, les sanctions doivent être définies. Elle peuvent l'être soit statiquement, soit par après négociation entre les parties.
- L'**annulation** d'un engagement social correspond au rejet d'un engagement actif précédemment accepté. Il est donc possible que certaines sanctions soient appliquées à cette occasion. En général, la décision d'appliquer ou non les sanctions est prise par l'agent qui n'a pas annulé l'engagement social.

Les transitions nouvellement définies sont les suivantes :

- L'**élimination** d'un engagement social **violé** a lieu quand les agents impliqués (i) s'accordent pour reconnaître que l'engagement social est violé, (ii) appliquent les sanctions et (iii) éliminent l'engagement social. Sur la figure 2.1, page 13, ceci se traduit par l'ajout d'une transition de l'état `violated` à l'état `cancelled`.
- De la même manière, l'**élimination** d'un engagement social **rempli** a lieu quand les agents impliqués (i) s'accordent pour reconnaître que l'engagement social est rempli, (ii) appliquent les sanctions et (iii) éliminent l'engagement social. Sur la figure 2.1, page 13, ceci correspond à l'ajout d'une transition de l'état `fulfilled` à l'état `cancelled`.

Dans cette section, nous avons étudié la littérature permettant aux agents de modéliser les interactions qu'ils perçoivent. Nous avons étudiés les modèles d'engagement social d'un point de vue progressif, depuis la proposition originelle la plus simple, jusqu'aux modèles actuels les plus riches, intégrant le cycle de vie du contenu et des sanctions.

2.2 Normes sociales

Dans cette section, nous étudions, dans le cadre des systèmes multi-agents ouverts et décentralisés, la littérature concernant la définition de l'*acceptabilité* d'interactions. Il s'agit d'examiner les modèles existants pour représenter les interactions acceptables ou non, dans la perspective de dégager les modèles adaptés à une utilisation par les agents eux-mêmes. À l'issue de cette étude, nous aurons étudié l'ensemble des entrées nécessaires à la phase de détection du processus de contrôle de la figure 1.1, page 3.

Dans cette section, nous nous intéressons, plus particulièrement, aux tra-

vaux sur les normes menés dans le domaine des systèmes multi-agents, des sciences juridiques et des sciences sociales. Cette section est organisée comme suit : dans un premier temps, nous dressons un bref panorama des différents types de normes afin de cadrer notre travail. Ensuite, nous nous intéressons aux modèles computationnels qui permettent aux agents de raisonner sur des normes.

2.2.1 Normes de bon comportement

Les normes permettent de définir les comportements acceptables pour un agent. Cependant, le terme de « norme » est employé dans de très nombreux domaines : philosophie, psychologie, sociologie, sciences de l'information, sciences économiques, sciences juridiques... avec des définitions sensiblement différentes. Par exemple, en philosophie une norme est une référence pour juger quelque chose [Col01, Dig02, VS03]. En psychologie, il s'agit d'un modèle [GDT04]. En sociologie, une norme est un comportement standard partagé par les membres d'un groupe social [Bri06, Dig02, VS03]. En économie, c'est un modèle qui doit exister ou être suivi ou une moyenne de ce qui existe actuellement dans un certain contexte [Dig02, VS03]. En science de l'information, il s'agit généralement d'un ensemble de règles fonctionnelles relatives à des produits, des activités ou des résultats, établies par des spécialistes et consignées dans un document produit par un organisme reconnu [GDT04].

[LLd02] distingue les normes individuelles et les normes sociales. Les normes individuelles, parmi lesquelles se trouvent les buts qu'un agent se fixe [Tuo95], sont privées à l'agent. Elles permettent donc pas de définir l'acceptabilité des comportements d'un agent d'un point de vue externe à celui-ci. Nous nous intéressons donc ici aux normes *sociales*.

De manière générale, les normes sociales apparaissent dans la littérature comme des concepts légaux. En utilisant le principe de la séparation des pouvoirs [Fra], il nous est possible de définir la grille d'analyse suivante :

- quelles sont les parties impliquées ;
- qui exerce le pouvoir législatif : qui crée la norme, quelle en est la source ;
- qui exerce le pouvoir judiciaire : qui décide qu'une norme a été violée ;
- qui exerce le pouvoir exécutif : qui sanctionne les violations ;
- quel type de sanction est appliqué, par exemple en cas de violation.

Ces critères ne sont pas totalement indépendants puisque, par exemple, le

type de sanction appliqué peut dépendre de l'entité qui l'applique. Cependant, ils permettent de structurer un peu plus les fondements des typologies classiques de normes existant dans la littérature.

Il existe deux types de normes sociales [von93]. [DMSC00] nomme les premières obligations et les secondes normes sociales. Dans [LLd02] les premières sont nommées obligations et les secondes codes sociaux. Enfin, [Tuo95, TBT95] les nomme r-normes et s-normes. C'est cette dernière terminologie que nous employons ici.

R-normes

Les r-normes sont définies comme étant des règles, pouvant prendre la forme de lois ou de chartes (donc explicites) auxquelles des individus doivent se conformer [TBT95, von93, DMSC00]. Les r-normes sont proposées par un groupe d'agents. Elles sont entérinées par une autorité qui se charge aussi de les faire respecter [TBT95, von93]. Le concept de r-norme repose donc fortement sur l'*acceptation* de la norme (ou, plus généralement, de l'autorité) par le groupe d'agents soumis à cette règle. Les r-normes ne peuvent être modifiées que si l'autorité le décide. Ces règles entraînent toujours des sanctions, les r-sanctions. Ces sanctions sont *explicites* et *connues à l'avance* [DMSC00]. L'exemple 2.2.1 illustre un cas de r-norme issu de codes de l'état français.

Exemple 2.2.1 *L'article L.413-1 du code de la route français (complété par le code pénal, article 132-11) sanctionne un excès de vitesse supérieur ou égal à 50 km/h (en cas de récidive dans les trois ans) d'une amende maximum de* 3750 €*, d'une suspension de permis de trois ans, d'un retrait de six points et jusqu'à trois mois de prison, accompagné d'une interdiction de conduire certains véhicules terrestres à moteur, pour une durée de cinq ans au plus.*

En conclusion, il est possible de caractériser les r-normes en fonction de la grille d'analyse définie en début de section comme suit : les r-normes lient un agent individuel à une autorité. Elles sont proposées par des agents, mais écrites et publiées par une autorité. Cette autorité est d'ailleurs aussi en charge d'en détecter et d'en sanctionner les violations selon les barèmes prévus et donnés à l'avance.

S-normes

Les s-normes définissent des règles de bon comportement [DMSC00]. Elles émergent de comportements individuels dans la société [von93]. Elles représentent un comportement global de la société [TBT95]. Ce sont des sortes de standards de fait. Elles sont automatiquement modifiées en fonction des changements de comportements des individus. Elles sont fondées sur des croyances ou des acceptations mutuelles. Elles ne sont pas obligatoirement disponibles sous forme écrite et peuvent être de deux types : celles appliquées à toute la communauté et celles appliquées uniquement à un groupe d'individus en fonction du rôle qu'il joue au sein de la société [TBT95]. N'étant pas écrites, elles doivent être enseignées aux nouveaux arrivants dans un groupe. Finalement, elles ne sont pas forcément sanctionnées. Si elles le sont, alors les s-sanctions ne sont pas connues *a priori* et, par conséquent, sont implicites. L'exemple 2.2.2 illustre un cas de s-norme dans la société française.

Exemple 2.2.2 *Une bonne éducation veut que, quand quelqu'un sort d'un magasin, il tienne la porte à toute personne qui le suit. Cela constitue une norme sociale qui n'est pas punie explicitement en cas de violation. Une personne à qui la porte n'a pas été tenue peut se sentir lésée et sanctionner la personne qui n'a pas tenu la porte par le moyen qu'elle veut (par une baisse d'estime, de réputation, etc.) et avec l'ampleur qu'elle souhaite.*

En conclusion, il est possible de caractériser les s-normes en fonction de la grille d'analyse définie en début de section comme suit : les s-normes lient un agent à son groupe social. Elles sont définies par les membres du groupe et sont transmises par celui-ci aux nouveaux venus. Tout agent est susceptible d'en détecter la violation et de punir l'un des membres du groupe comme il l'entend.

Comparaison des r-normes et s-normes

Pour [DMSC00], les r-normes agissent en restreignant l'autonomie des agents et les s-normes en rendant la coordination plus efficace. Pour sa part, [von93] distingue les r-normes et les s-normes par le fait qu'il s'agit pour l'une d'un ordre normatif *imposé* et pour l'autre d'une *émergence* due aux comportements d'une catégorie d'individus. Selon la grille définie en début de section, les r-normes et les s-normes divergent donc principalement sur des

critères liés aux sanctions : sur l'entité en charge d'appliquer des sanctions et sur le type de sanction mis en place (toujours appliquées ou non, connues *a priori* ou non). En général, dans les r-normes, une autorité a été autorisée au préalable à sanctionner les agents. Dans le cas des s-normes, c'est généralement l'agent victime de la violation qui décide si et comment sanctionner le violateur.

Transitions et inconsistances

Soulignons finalement que les frontières entre les différents types de normes sont assez floues : il est possible que certaines normes changent de type [TBT95] : par exemple, une s-norme peut devenir une r-norme si elle est explicitée et sanctionnée par une autorité. Il est aussi possible que des normes soient inconsistantes. [TBT95] définit une hiérarchisation des différents types de normes qui peut servir pour régler les conflits entre celles-ci.

2.2.2 Formalismes de normes

Dans cette section, nous étudions les formalismes existants pour représenter les normes sociales. Il s'agit d'étudier les formalismes proposés dans la littérature qui permettent de modéliser une norme sociale, de manière à pouvoir raisonner dessus, en particulier pour en détecter les violations.

Il existe deux manières d'implémenter les normes dans les systèmes multi-agents [JS93] : par enrégimentation ou non. Dans le premier cas, la norme est vue comme la spécification d'un bon comportement qui *doit* être respecté par les agents et est donc codée en dur dans les agents [ST95, MT95, BGM98, Bom99]. Ceux-ci sont alors enrégimentés et il n'est pas nécessaire de prévoir comment les sanctionner, puisqu'ils ne peuvent pas violer les normes sociales [BvdT05, vRW05]. Dans ce cas, le formalisme utilisé pour représenter les normes sociales est généralement la logique déontique.

Il est aussi possible de considérer les normes sociales comme des indications de comportements à suivre [CDJT00, BL00, DMSC00, VSD03, LLd04]. Les agents doivent alors disposer d'un formalisme plus souple leur permettant de décider (i) d'*accepter* (adopter) les normes sociales, c'est-à-dire prendre connaissance de leur existence et (ii) de décider de les *respecter*, c'est-à-dire d'agir sans les violer [LLd02]. Les agents sont ainsi capables de différents degrés d'autonomie, raisonnant *a priori* sur les conséquences éventuelles de leurs actes [Dig99, DMSC00, LLd02]. Il est alors nécessaire de prévoir des

moyens de contrôle (détection et sanction des violations). Les formalismes utilisés dans ce cas sont plutôt de type descriptif.

Logiques déontiques

La logique déontique est particulièrement étudiée dans les sciences juridiques [Val95]. Le but de la définition d'une telle logique est de disposer d'un formalisme permettant de décrire les lois et les faits de manière non ambiguë ainsi que de disposer d'un moteur d'inférence. Grâce à ce moteur d'inférence, il est possible de raisonner sur le respect et / ou la cohérence des lois.

La logique déontique définit des opérateurs d'obligation (O), de permission (P) et d'interdiction (F) [von51]. L'exemple 2.2.3 illustre l'utilisation de la logique déontique pour définir une obligation.

Exemple 2.2.3 *Si α, est un fait, alors il est possible de définir l'obligation que ce fait soit vérifié par : $O\alpha$.*

Les différents opérateurs de la logique déontique peuvent généralement se définir les uns par rapport aux autres. Par exemple, les opérateurs F et P peuvent se définir en fonction de O de la façon suivante : $F\alpha \equiv O\neg\alpha$, ce qui signifie que l'interdiction de α est équivalente à l'obligation de sa négation. De même, $P\alpha \equiv \neg O\neg\alpha$, exprime que α est permis s'il n'est pas interdit.

Néanmoins, les systèmes juridiques actuels sont très complexes et définissent parfois des règles inconsistantes. Dans de telles situations, un moteur d'inférence utilisant la logique déontique standard [von51, von68, von81] se trouverait bloqué. D'autre part, la logique déontique standard elle-même, de part l'axiomatisation de ses opérateurs, contient quelques paradoxes. L'exemple 2.2.4 illustre l'un des douze paradoxes identifiés dans [MDW94].

Exemple 2.2.4 *Le « paradoxe du pénitent » est lié à l'axiomatisation de l'opérateur F. En effet, celle-ci autorise l'écriture : $F(\varphi) \rightarrow F(\varphi \wedge \psi)$. Si l'on considère que φ représente le fait de tuer quelqu'un et que ψ est le fait d'aller en prison, alors le fait d'interdire de tuer quelqu'un implique le fait d'interdire à la fois de tuer quelqu'un et d'aller en prison !*

Du fait de la présence de tels paradoxes, de nouvelles axiomatisations des opérateurs ont été proposées, particulièrement avec les logiques déontiques annulables (« defeasible deontic logics ») [Jon93, vdT94, MDW94].

Cependant, les différentes propositions de logiques déontiques annulables ne résolvent que quelques paradoxes à la fois et restent indécidables.

Formalismes descriptifs

Partant de l'observation que les approches utilisant la logique déontique ne sont, de toutes les façons, que déclaratives [AGVSD05, VS03] (elles expriment ce qui est acceptable) et n'ont pas de sémantique opérationnelle (elles ne disent pas comment atteindre l'acceptable), des approches plus descriptives ont été proposées.

Ainsi, [VSDD05, GCNRA05, KN05, dSdL05] proposent des formalismes descriptifs au pouvoir expressif assez proche de celui des logiques déontiques les plus riches. Ceux-ci intègrent la plupart des éléments suivants :

- un **type**, parmi : obligation, permission ou interdiction ;
- des **sanctions**, principalement en cas de violation ;
- un **objet** : ce sur quoi porte la norme, parfois accompagné de conditions ;
- un **sujet** : l'agent soumis à la norme sociale ;
- une **condition simple** d'activation de la norme (ex. : délai).

Du fait de leur caractère descriptif, ces approches permettent cependant une représentation des normes sociales encore plus riche. Ainsi, [LLd04, LL04, Sal02] étendent la liste d'éléments précédente avec :

- la prise en compte de conditions plus fines à la fois pour l'activation de la norme et sur son objet ;
- la possibilité d'avoir comme sujet un rôle ou un ensemble d'agents ;
- la notion de **bénéficiaire** : un agent envers qui le sujet est engagé à respecter la norme sociale ;
- un **identifiant** [Sal02].

Finalement, [Str02, GBKD05] ajoutent encore d'autres éléments, aboutissant ainsi à des formalisme proches de la description des normes telles qu'elle est fournie en section 2.2.1 :

- la prise en compte de l'**autorité** en charge de faire appliquer la norme sociale. Ce champ remplace le simple bénéficiaire. L'autorité est alors décrite de façon générique. Tout agent représentant cette autorité étant un mesure de sanctionner un agent qui viole la norme sociale ;
- l'**auteur** de la norme sociale [Str02], c'est-à-dire l'entité exerçant le pouvoir législatif ;

– une **priorité**, servant en cas de conflit entre les normes pour décider quelles sont les normes sociales qu'il est plus important de respecter [GBKD05].

Politiques Sociales

[VFC05] et [Sin99] proposent de décrire les normes par des engagements sociaux. Les différents constituants des engagements sociaux sont présents dans le modèle : le débiteur est ici l'agent soumis à la norme sociale, le créditeur est l'agent chargé de la faire respecter et le contenu est celui de la norme. La norme peut être dans différents états (`inactive`, `active`...). Le contenu peut faire référence à des engagements sociaux représentant des interactions. De ce fait, les engagements sociaux instanciant des normes sont d'un niveau d'abstraction supérieur à ceux qui représentent les interactions. [Sin99] les différencie en les nommant « politiques sociales ». Dans [VFC05], les agents évoluent dans une institution sujette à différents événements (action du temps, changement de l'un de ses constituants, action d'un agent). Ces événements déclenchent les normes, ce qui génère les politiques sociales.

Ces formalismes ne sont que descriptifs et les agents sont autonomes. De ce fait, même si les agents sont en possession de la description des normes en vigueur dans le système, rien ne les oblige à les respecter. Il est donc nécessaire de mettre en place des structures de contrôle qui gèrent, en particulier, la détection et la sanction des violations de ces normes.

2.3 Détection et sanction des violations

Dans cette section, nous nous intéressons aux structures de contrôle qu'il est possible de mettre en place pour faire en sorte que les normes soient respectées par les agents.

Dans la littérature, il existe deux formes d'un tel contrôle : une forme interne et une forme externe. La forme interne correspond à la capacité d'un agent à identifier par lui-même le comportement correct à avoir en fonction de normes sociales définies de manière externe. Dans ce cas, le contrôle repose sur les capacités cognitives des agents pour comprendre et évaluer leur propre comportement normatif [Dig99, LLd02, CDJT00, BL00, KN05, CB05]. La forme externe s'applique quand une structure institutionnelle s'est vue conférer le pouvoir d'agir (éventuellement physiquement) sur un agent pour l'in-

fluencer, de façon à lui faire adopter un comportement correspondant à la norme. Le comportement de l'agent est alors interprété et évalué par d'autres entités (autorité, groupe social...) en fonction de normes sociales gouvernant le système ou le groupe auquel appartient l'agent [PANS98, GCNRA05, GBKD05]. Dans le cadre de cette thèse, nous nous intéressons à nue approche différente de l'instauration d'institutions centralisées. Nous cherchons à mettre en place un contrôle lui aussi externe, mais appliqué par le groupe social.

Ce contrôle ne peut s'exercer qu'à travers la détection de la violation ou du respect des normes et de la sanction positive ou négative correspondante (figure 1.1, page 3). Dans cette section, nous présentons d'abord une définition formelle de la violation. Nous montrons ensuite les différentes approches qui sont proposées dans la littérature pour mettre en place la détection de violations dans des systèmes informatiques. Enfin, nous concluons sur les différentes manières de sanctionner les comportements des agents.

2.3.1 Définition de la violation

En s'appuyant sur la définition de la fraude de [Sim95], [FTL98] définit formellement la violation d'une obligation à l'aide de la logique déontique standard [von51, Che80] et de la logique de l'action [Elg93, SCJ97]. L'opérateur O de la logique déontique standard est utilisé conjointement aux opérateurs E et H de la logique de l'action. Ceux-ci sont définis de la façon suivante : $E_{\mathtt{i}}A$ signifie « l'agent `i` a amené l'état du monde A » et $H_{\mathtt{i}}A$ signifie « l'agent `i` a essayé d'amener l'état du monde A », mais sans contrainte sur le succès de l'opération. Notons que $E_{\mathtt{i}}A \rightarrow H_{\mathtt{i}}A$, c'est-à-dire qu'un agent qui a réussi à amener l'état du monde A a forcément essayé d'amener l'état du monde A, la réciproque n'étant pas vraie.

La violation d'une obligation, est alors définie par la définition 2.3.1 suivante :

Définition 2.3.1 *La **violation** se définit de manière générique comme la non réalisation de ce qui est obligatoire :*

$$O\varphi \wedge \neg\varphi$$

En remplaçant le fait φ dans la définition 2.3.1 par les expressions représentant des actions ($E_{\mathtt{i}}A$ et $H_{\mathtt{i}}A$), [FTL98] obtient trois formules valides

représentant la violation d'une obligation de mener une action :

$$OE_{\mathtt{i}}A \wedge \neg E_{\mathtt{i}}A \tag{2.1}$$

$$OE_{\mathtt{i}}A \wedge \neg H_{\mathtt{i}}A \tag{2.2}$$

$$OE_{\mathtt{i}}A \wedge H_{\mathtt{i}}A \wedge \neg E_{\mathtt{i}}A \tag{2.3}$$

La formule 2.1 exprime le fait que l'agent `i` avait l'obligation d'amener l'état du monde A mais qu'il ne l'a pas fait. La formule 2.2 représente le fait que l'agent `i` avait l'obligation d'amener l'état du monde A, mais qu'il n'a même pas essayé de le faire. Finalement, la formule 2.3 signifie que l'agent `i`, qui avait l'obligation d'amener l'état du monde A a essayé de le faire, mais n'a pas réussi. Cette formule exprime donc une violation faible : dans ce cas, l'agent a montré de la bonne volonté, mais s'est avéré incompétent.

2.3.2 Détection de violation

Dans cette section, nous étudions les systèmes de détection de violation. C'est à dire des systèmes capables de comparer les comportements tels qu'ils ont été effectués par les agents (représentés, par exemple, grâce à un modèle des interactions des agents) aux comportements tels qu'ils devraient être (par exemple, décrits par des normes).

Violation de transactions téléphoniques ou bancaires

Les approches les plus courantes dans le domaine de la détection de violations consistent à détecter des fraudes bancaires ou téléphoniques [AHTW97, Jen97, CLPS02]. Des comportements réels de clients sont étiquetés en fonction de leur acceptabilité. Ces comportements étiquetés sont alors fournis à un algorithme qui les apprend et les stocke sous une forme donnée : base de règles, réseaux neuronaux, etc. Le système consiste ensuite à comparer ces comportements de référence aux comportements courants des utilisateurs, contenus dans une base de données de transactions.

Violation de protocole

[LS04] proposent de détecter des violations dans un protocole de communication entre agents. Pour ce faire, les auteurs proposent de définir l'état global du système à un instant t à partir d'une combinaison des états des

agents et de l'environnement à cet instant. Ces états globaux, représentés par des automates d'états finis, sont ensuite interprétés sous forme de formules de la logique propositionnelle. Le protocole, quant à lui, est représenté sous forme de formules déontiques. Il est alors possible de déterminer si les agents violent les règles de transition d'état définies par le protocole. En étudiant plus en détails les violations, il est même possible de définir des protocoles de récupération.

Analyse critique

Si les définitions formelles de la violation de [FTL98] ont l'avantage d'en donner une définition non ambiguë, elles restent cependant limitées. D'une part, la dimension temporelle n'est pas prise en compte dans ces travaux, d'autre part, la logique déontique étant indécidable, il n'est pas possible de garantir que, si un agent utilisait un tel formalisme, il pourrait détecter toute violation en un temps fini.

Les violations détectées dans les systèmes bancaires ou téléphoniques et dans les protocoles ne sont pas adaptées aux systèmes ouverts et décentralisés puisque la détection est menée de façon centralisée et hors-ligne.

2.3.3 Sanctions

Selon [PFCd04] les différents types de sanction que l'on peut appliquer à un agent peuvent être catégorisées selon trois axes :

- La **direction** : la sanction peut-être *positive*, pour encourager un comportement désiré ou *négative* pour décourager des comportements violant les normes du système.
- Le **type** : la sanction peut être *automatique*, c'est le cas par exemple de l'automobiliste qui conduit à l'envers sur l'autoroute et qui va faire un séjour à l'hôpital suite à son comportement. Les sanctions non automatiques peuvent être de trois types :
  - *matérielles*, par exemple un acte de violence envers le violateur, une action de réparation ou une compensation financière demandée au violateur...
  - *sociales*, par exemple une baisse ou une hausse de réputation, de confiance ou de crédibilité du violateur.
  - *psychologiques*, par exemple en terme de culpabilité, honte...

L'*horizon temporel* des sanctions permet de savoir si une sanction perdure au cours du temps ou bien est instantanée. Par exemple, les sanctions en terme de réputation peuvent perdurer plus longtemps qu'une compensation financière ponctuelle.

– Le **style** : dans le cadre des systèmes multi-agents, les sanctions peuvent être *explicites*, c'est-à-dire discutées par les parties impliquées dans l'engagement social et connues publiquement ou bien *implicites*, c'est-à-dire décidées unilatéralement. De plus, les sanctions peuvent être décidées *a priori*, c'est-à-dire être connues à l'avance par les parties ou *a posteriori*, c'est-à-dire connues seulement au moment de leur application, ce qui ne permet pas à l'agent sanctionné de raisonner à l'avance sur les pénalités qu'il encoure en cas de violation.

Il existe donc une grande variétés de sanctions, qui ne sont pas toutes applicables dans les mêmes situations.

2.4 Synthèse

Dans ce chapitre, nous nous sommes intéressés à la problématique du contrôle des interactions des agents. Ce contrôle consiste à comparer des interactions d'agents telles qu'elles ont pu être observés aux définitions des interactions acceptables et à décider des sanctions à appliquer. Dans le cadre de cette thèse, nous nous intéressons plus particulièrement à la déclinaison sociale de ce contrôle, c'est-à-dire à sa réalisation par les agents eux-mêmes. Pour ce faire, chaque agent doit pouvoir : (1) modéliser les interactions qu'il perçoit, (2) définir l'acceptabilité des interactions et (3) décider des sanctions à appliquer.

Nous nous sommes tout d'abord intéressés aux modèles permettant de modéliser les interactions des agents. Nous avons retenu l'approche sociale comme la meilleure candidate dans un cadre ouvert et décentralisé, du fait de sa large portée, de son caractère publique et du fait qu'elle nécessite un minimum d'hypothèses sur l'implémentation interne des agents. Nous nous sommes alors intéressés aux modèles permettant aux agents de manipuler ces engagements sociaux. Nous avons vu que le modèle originel de [Sin91] a été enrichi de la capacité à prendre en compte l'évolution temporelle d'un engagement social et de son contenu, ainsi que de la possibilité d'adjoindre des sanctions en fonction de l'aboutissement de cette évolution.

Nous nous sommes ensuite intéressés aux normes, afin de permettre aux

agents de décrire ce qu'est une interaction acceptable ou non. Nous avons réduit notre étude aux normes sociales, puisqu'elles seules peuvent être utilisées par des agents pour caractériser les interactions d'autres agents. Nous avons vu qu'il existait différents types de normes sociales : les r-normes et les s-normes, ainsi que deux grandes classes de formalismes pour les représenter : les formalismes issus de la logique déontique et les formalismes plus descriptifs.

Enfin, nous avons étudié les propositions existantes pour permettre aux agents de décider des sanctions à appliquer, c'est-à-dire de mettre en place le contrôle lui-même. Nous avons vu qu'il existe deux manières d'implanter le contrôle : interne ou externe, ainsi que de nombreux types de sanctions et de façons de les appliquer. Nous avons vu que les sanctions de type social étaient plus adaptées aux Systèmes Multi-Agents Ouverts et Décentralisés car elles ne nécessitent pas de pouvoirs particuliers pour être appliquées.

Chapitre 3

Confiance et Réputation

Dans ce chapitre, nous nous intéressons aux notions de confiance et de réputation comme moyen de sanctionner socialement les interactions des agents. Dès que l'on aborde ces concepts dans la littérature, on s'aperçoit qu'il n'existe pas de définition unique et communément admise. Bien au contraire, il existe toute une variété de définitions, qui dépendent principalement du domaine d'étude et du propre point de vue de l'auteur. Dans ce chapitre, nous nous intéressons particulièrement aux domaines qui ont, les premiers, cherché à définir les concepts de réputation et de confiance : il s'agit principalement des sciences humaines, sociales et économiques.

L'objectif du présent chapitre n'est pas de mener une étude exhaustive de l'ensemble des définitions proposées puisque celles-ci sont, à l'heure actuelle, toujours sujettes à débats [JoM, JAS]. Nous étudions ici ces différentes définitions afin, d'une part, de circonscrire notre domaine de recherche et, d'autre part, d'en extraire une caractérisation précise des concepts étudiés. Nous obtenons ainsi un ensemble de propriétés caractérisant les réputations, à partir duquel nous déduisons une grille d'analyse. Cette grille est alors utilisée, dans le chapitre 4, pour présenter les modèles computationnels de réputation.

Ce chapitre est organisé comme suit : constatant que les définitions de la confiance sont fort variées, nous utilisons le travail de [MC01] pour les catégoriser et restreignons notre champ d'investigation à la seule classe de confiance dite *interpersonnelle*. Nous caractérisons ensuite plus spécifiquement cette classe de confiance et établissons les premiers liens entre les concepts de confiance et de réputation. En cherchant, dans un deuxième temps, à approfondir le concept de réputation, nous étendons les travaux de [MHM02] et montrons qu'il en existe différents types. Finalement, nous restreignons

encore notre champ d'investigation aux seules *réputation fondée sur les interactions* et discutons brièvement leurs caractéristiques.

3.1 Panorama

Dans la section psychologie du Grand Dictionnaire Terminologique [GDT04] la confiance est définie comme le « sentiment ferme de pouvoir s'en remettre au comportement ou au jugement de quelqu'un ou de soi-même ». Dans le Merriam-Webster [Mer04] la confiance est définie comme « le fait de pouvoir s'appuyer, de manière assurée, sur le caractère, les capacités, la force, ou la vérité de quelqu'un ou de quelque chose ». Ces deux définitions illustrent à quel point, même à un niveau très abstrait, deux définitions peuvent diverger. Par exemple, la définition du [GDT04] n'applique la confiance qu'à des êtres humains alors que la définition du [Mer04] autorise à l'appliquer aux objets.

Dans cette section, nous menons une étude plus approfondie des définitions de la confiance et de la réputation, principalement issues des sciences humaines, sociales et économiques. Dans une première partie de cette section, nous allons illustrer le flou qui règne autour de la définition du concept de confiance. Nous montrons ensuite qu'il existe différentes classes de confiance, puis nous précisons le fonctionnement de la classe de confiance dite interpersonnelle et les relations entre cette classe et la réputation.

3.1.1 Confiance

Le concept de confiance a été étudié dans de très nombreux domaines de recherche : la biologie évolutionnaire [Bat00, PD92], la sociologie [Luh79, Luh00, CP02, CF98, Qué01], la psychologie sociale [Deu62], les sciences économiques [Das90, Rou00, Del01a, Del00, Sha87, Fuk95], l'histoire [Gam00a, Pag00], la philosophie [Lag92, Her88]... Étant donné que ces domaines de recherche sont souvent liés à des perceptions du monde et à des méthodes de travail spécifiques, des définitions très différentes ont été données dans ces différents domaines. Certaines définitions sont propres au point de vue du domaine des auteurs, d'autres sont encore plus spécifiques au point de vue de l'auteur lui-même. En particulier, les psychologues perçoivent la confiance comme un trait personnel alors que les sociologues considèrent la confiance comme une structure sociale. Les économistes, quant à eux, la perçoivent généralement comme une intention de comportement ou comme un mécanisme

de choix. Ainsi, le concept de confiance a été défini comme des croyances sur divers attributs d'une personne [CF98], comme une attente sur le comportement de l'autre [Das90]... Dans le domaine du management, d'autres chercheurs [LB95] ont défini le concept de confiance à travers ses processus de construction, par exemple fondés sur la dissuasion, fondés sur la connaissance, ou fondés sur l'identification.

D'une telle multitude de définitions si différentes, il résulte un « pot pourri déroutant » [Sha87] (ou « fatras conceptuel » [Bar83]). Pour les nouveaux chercheurs qui abordent le domaine de la confiance, cette multitude de définitions engendre plus de difficultés qu'elle n'apporte d'aide.

De ce fait, de nombreux auteurs (en particulier [FBKS03, FBSS04, CBSS05]) ne définissent plus le concept de confiance sur lequel ils s'appuient et laissent au lecteur le soin d'utiliser sa propre définition de la confiance ou de deviner la définition employée. D'autres auteurs [MC02] surtout dans le domaine des sciences humaines et sociales, se refusent même désormais à définir ce concept. Dans cette thèse, nous avons pris le parti de fixer les définitions afin de lever toute ambiguïté sur les termes employés.

Dans la section suivante, nous présentons une classification des différentes définitions de la confiance. Cette classification permet alors de cibler plus précisément le domaine de recherche dans lequel cette thèse se situe et de poser les premières définitions générales des concepts de confiance et de réputation.

3.1.2 Classes de confiance

[MC01] proposent de différencier les grandes classes de confiance suivantes : la confiance *dispositionnelle*, la confiance *institutionnelle* et la confiance *interpersonnelle*, auxquelles [Das90] et [Qué01] ajoutent la confiance *de groupe*. D'autres auteurs [CS05] identifient aussi la confiance *dans les objets, les lieux, les événements ou les activités* comme des classes de confiance à part entière. Nous caractérisons dans cette section ces différentes classes de confiance et étudions succinctement leurs relations.

Confiance dispositionnelle

La confiance dispositionnelle est la confiance générale dans l'environnement. C'est une tendance générale qu'a un individu à faire confiance ; c'est un état mental de l'individu, qui n'est ni ciblé envers un individu particulier, ni lié à un contexte particulier. Cette classe de confiance n'est pas remise

en cause quotidiennement. Par exemple, un individu peut avoir tendance à toujours faire confiance.

Confiance institutionnelle

La confiance institutionnelle est la confiance dans les institutions. Elle est liée au fait que celles-ci sont garantes des engagements des individus qu'elles gouvernent. Ce type de confiance n'est pas remis en cause fréquemment. Il s'agit de ce que Luhmann et Gambetta appellent « confidence » dans [Gam00b]. Selon [MC01], cette classe de confiance peut aussi être liée à une situation donnée. Par exemple, au Mexique ou au Cameroun, les institutions officielles sont si corrompues que les individus ne leur font pas confiance [Rou00].

Confiance interpersonnelle

La confiance interpersonnelle est la confiance dans un autre individu. Il s'agit de la confiance qu'un individu associe spécifiquement à un autre individu. Elle se fonde sur des interactions dont le premier individu a connaissance et qui impliquent le deuxième individu. Elle est donc remise en cause assez fréquemment, lors de chaque nouvelle interaction. Ce type de confiance est par exemple utilisé par un acheteur lorsqu'il achète à son vendeur habituel.

Confiance de groupe

La confiance de groupe [Qué01] est la confiance qu'un individu accorde à un certain groupe de personnes. Une catégorie de personne est un groupe de personnes ayant un attribut commun [Wor05]. Il est donc clair que la confiance attribuée à une catégorie de personnes est un cas particulier de la confiance d'un groupe. Par exemple, certaines personnes ont peu confiance dans la catégorie de personnes que constitue les garagistes.

Confiance dans un objet

La confiance dans un objet [Qué01, CF98] diffère des autres classes présentées ci-dessus par le fait que la cible de la confiance est un objet et non un individu ou un ensemble d'individus. Cela constitue une différence particulièrement importante.

En effet, [Qué01] fait remarquer qu'un facteur de l'efficacité de la confiance repose sur le principe de réciprocité [CP02, Ors98, CFP03, MHM02]. Si un individu modélise la confiance d'un autre individu pour se protéger de ses tromperies, alors, par réciprocité, cet individu peut facilement s'imaginer que l'autre agit de la même façon envers lui. Ce principe de réciprocité crée une incitation pour les individus à bien se comporter les uns envers les autres. Or, dans le cadre de la confiance en un objet, cette réciprocité n'existe pas. De ce fait et contrairement à la définition de [HDM03], il semble que la notion de confiance dans un objet corresponde plus exactement à une notion de qualité de l'objet, plus qu'à une véritable confiance [Qué01]. Dans le reste de cette thèse, nous ne considérons donc pas ce concept comme une confiance. Pour les mêmes raisons, nous ignorons les confiances relatives à des lieux, des événements ou des activités [CS05].

Liens entre les différentes classes

	Institutionnelle	Interpersonnelle	de Groupe
Dispositionnelle	[MC01]	[MC01]	–
Institutionnelle		[Rou00], [MC01], [CF98], [Bro00], [TBT95], [CP02]	[CF98]
Interpersonnelle			[Das90]

TAB. 3.1 – Liens entre les classes de confiance.

La table 3.1, recense les travaux qui étudient les relations entre ces différentes classes de confiance. Dans cette table, nous ne nous intéressons ni au sens de la relation, ni à la relation d'une classe avec elle-même (d'où les cases grisées, marquant l'absence de certaines lignes et colonnes). Les paragraphes ci-dessous décrivent brièvement les relations ligne par ligne.

La confiance dispositionnelle influe directement sur la confiance institutionnelle car ce qu'un individu pense de l'environnement et des autres au sens général déteint sur ce que cet individu pense des institutions dans lesquelles ces autres ont accepté d'être enrôlés [MC01].

Dans le cas de situations nouvelles et inconnues, la confiance dispositionnelle influe directement sur la confiance interpersonnelle [MC01]. En effet, dans ces situations, un individu ne peut s'appuyer sur des interactions qui

impliquent le partenaire potentiel pour décider s'il doit agir en confiance avec lui. L'individu tendra donc à s'appuyer sur sa disposition générale à faire confiance pour prendre sa décision.

La confiance institutionnelle influe sur la confiance interpersonnelle : [Rou00] remarque que « la stabilité des institutions facilite l'apparition de la confiance » interpersonnelle. Cette idée se retrouve également chez [MC01]. En effet, les institutions permettent d'obtenir certaines garanties sur un interlocuteur. Plus les institutions sont stables, reconnues et en mesure de sanctionner les mauvais comportements, plus elles facilitent la confiance interpersonnelle. Cette relation est d'ailleurs réciproque : [CF98], remarquent qu'il ne peut exister d'institution sans confiance interpersonnelle. En effet, si une majorité d'individus dépendant de l'institution n'a pas confiance dans les membres représentant individuellement cette institution, alors l'institution est inefficace.

Un cas particulier de cette double relation entre la confiance institutionnelle et la confiance interpersonnelle est celui des contrats commerciaux. En effet, plus on a confiance dans le partenaire commercial moins on a besoin de spécifier formellement le contrat [Bro00, CP02]. Dans les sociétés reposant fortement sur la confiance, par exemple au Japon [TBT95], les contrats sont très peu formels. Moins on a confiance dans le partenaire, plus on aura tendance à spécifier précisément les termes du contrat et à s'appuyer sur la confiance que l'on a dans les institutions. Celles-ci garantissent alors les comportements du partenaire et l'application des sanctions s'il se comporte mal.

Il existe aussi un lien non négligeable entre la confiance de groupe et la confiance institutionnelle. En effet, il existe des groupes de personnes chargées de représenter des institutions. La confiance que l'on a dans ces représentants influe fortement sur la confiance que l'on a dans l'institution elle-même [CF98].

Finalement, [Das90] relie la confiance de groupe à la confiance interpersonnelle. En effet, la confiance acquise par un groupe est fortement liée aux comportements des individus constituant ce groupe. Par exemple, si la confiance de groupe qu'a un individu dans les garagistes est faible, c'est peut-être parce que l'ensemble des garagistes avec lesquels il a mené des interactions se sont mal comportés.

Nous avons identifié dans cette section quatre classes de confiance et éliminé les objets, les lieux, les événements et les activités comme cibles possibles de la confiance. Dans le cadre de cette thèse, nous nous concentrons

sur la confiance interpersonnelle, c'est-à-dire sur la confiance qu'un individu accorde à un autre.

3.1.3 Confiance interpersonnelle

Dans cette section, nous cherchons à caractériser plus précisément la classe de confiance interpersonnelle. Ceci nous permet d'établir les premiers liens entre confiance et réputation et de poser les premières définitions très générales de ces concepts.

[Qué01] estime qu'il y a bien souvent une confusion entre les croyances de confiance (le fait d'« avoir confiance ») et l'action en confiance (le fait d'« agir en confiance »). [MC01] distingue un niveau intermédiaire : les *intentions* de confiance : un individu se construit des croyances de confiance en fonction des interactions qu'il a eues avec un autre individu ; plus tard, lorsqu'il se trouve dans une situation où il doit décider d'agir avec ce partenaire, ses croyances de confiance lui permettent de **raisonner**, c'est-à-dire de déterminer s'il a l'intention ou non de faire confiance à ce partenaire [Dem04] ; les deux partenaires peuvent alors effectivement agir en confiance : se **faire confiance**. La figure 3.1 illustre cette décomposition.

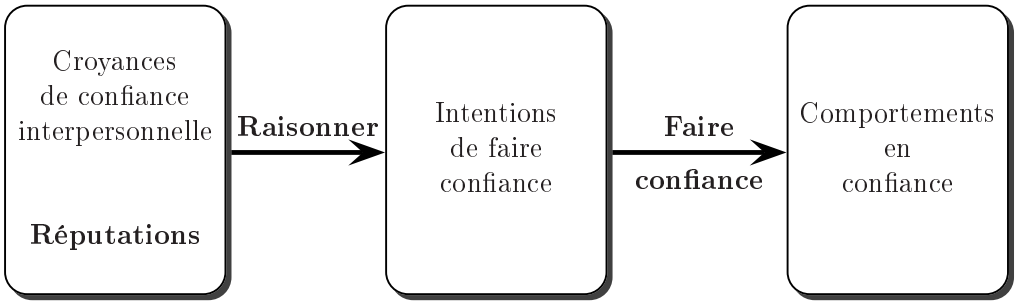


FIG. 3.1 – Processus de décision de faire confiance et réputations.

À partir de cette décomposition, il nous est possible de poser les définitions suivantes[1] :

[1]En posant de telles définitions, nous adoptons la vision anglophone, majoritairement présente dans le domaine au niveau international. Nous divergeons cependant sensiblement de la conception intuitive en langue française qui n'associe au terme réputation que les

Définition 3.1.1 *Le terme* ***réputation*** *est utilisé dans cette thèse pour faire référence aux croyances de la classe de confiance interpersonnelle, c'est-à-dire aux croyances sur lesquelles un agent s'appuie pour prendre sa décision d'agir en confiance avec un autre agent.*

Définition 3.1.2 *L'expression* ***décision de faire confiance*** *est utilisée dans cette thèse pour faire référence au processus de décision de la confiance interpersonnelle, qui aboutit à un comportement en confiance.*

En cherchant à caractériser le concept de confiance, nous avons pu cibler de plus en plus précisément notre domaine de recherche. Nous avons aussi établi les premiers liens entre les concepts de confiance et de réputation et pu poser des définitions simples et générales, mais non ambiguës de ces concepts. De la même façon que nous nous sommes intéressés dans cette section au concept de confiance, nous étudions maintenant le concept de réputation.

3.2 Réputations d'un agent

Dans cette section, nous procédons de la même manière que dans la section précédente : en partant du concept général que nous cherchons à caractériser, la réputation, nous précisons petit à petit notre domaine de recherche et aboutissons à une caractérisation précise du concept étudié.

[CS05] spécifient la réputation d'un point de vue fonctionnel. Les auteurs proposent dans ces travaux une ontologie, dont la partie spécifiquement dédiée au concept de réputation sert de trame à cette section. Cette partie de l'ontologie que nous avons retenue est schématisée dans la figure 3.2. Elle identifie différents *types* de réputation en fonction des classes de confiance auxquelles elles font référence et des *rôles* que les agents jouent au cours des différents *processus* qui manipulent des réputations.

Dans cette section, nous suivons la description en sens inverse : nous commençons par identifier l'ensemble des processus dans lesquels la réputation intervient. Ensuite, nous étudions les rôles que les agents peuvent jouer au cours de ces processus, tels qu'ils ont été définis par [CP02]. Enfin, nous présentons les différents types de réputation établis par [MHM02] et étudions

croyances partagées par plusieurs agents et issues du commérage.

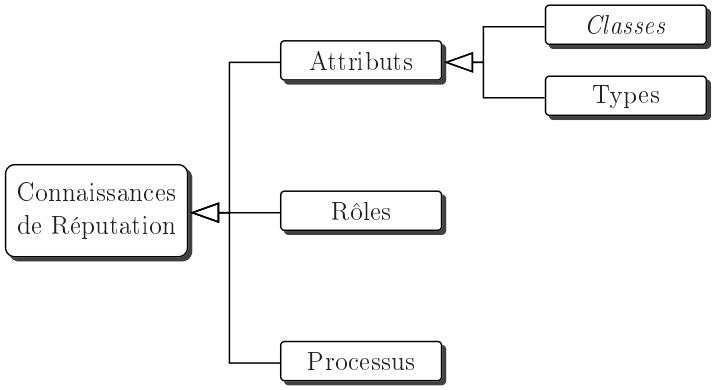


FIG. 3.2 – Ontologie des connaissances de réputation.

comment certains de ces types peuvent être redéfinis à partir des rôles que jouent les agents. Finalement, nous restreignons encore notre domaine de recherche aux seules réputations fondées sur les interactions et les caractérisons à l'aide d'un ensemble de propriétés que nous discutons brièvement.

3.2.1 Processus de gestion et d'utilisation des réputations

Les réputations sont impliqués dans différents processus qui participent à la création, la gestion et l'utilisation des réputations.(partie inférieure de la figure 3.2). En section 3.1.3, page 37, nous avons utilisé le processus de décision de faire confiance afin de distinguer les expressions « réputations » et « faire confiance ». L'ontologie fonctionnelle de la réputation [CS05] identifie de plus les processus d'évaluation, de punition et de propagation. [AH00] ajoute le processus d'initialisation et [Dem04] celui de raisonnement.

Nous obtenons ainsi un ensemble de six processus faisant intervenir les réputations :

- **initialisation** [AH00] : définir le degré de réputation d'un agent en l'absence d'information sur celui-ci.
- **révision** : maintenir l'adéquation entre l'évaluation des comportements d'un autre agent et le degré de sa réputation. Ce processus se décompose en deux autres processus [CS05] :

  - **évaluation** [CS05] : évaluer un comportement d'agent à un instant donné. Ce processus consiste à construire une évaluation à partir d'un ensemble d'observations (*cf.* définitions 3.2.2 et 3.2.3, ci-dessous). Les processus de détection de violations étudiés en section 2.3, page 24 sont des exemples de processus d'évaluation.
  - **punition** [MC01] : suivre l'évolution temporelle des comportements, c'est-à-dire agréger des évaluations pour former un degré de réputation.
- **raisonnement** [Dem04] : tirer les conséquences des niveaux de réputations dans un contexte donné.
- **décision** [MC01] : prendre des décisions d'agir en confiance ou non.
- **propagation** [CP02] : définir quand, pourquoi, à qui et comment diffuser des recommandations (voir définition 3.2.4, ci-dessous).

La détermination de cet ensemble de processus fait apparaître les quatre notions suivantes : un agent qui fait confiance doit être capable d'*observer* et d'*évaluer* des *interactions* (définitions 3.2.1, 3.2.2, et 3.2.3), ainsi que d'envoyer ou recevoir des *recommandations* (définition 3.2.4).

Définition 3.2.1 *Une **interaction** est une action menée en commun par plusieurs agents.*

Définition 3.2.2 *Une **observation** est une interaction telle que perçue par un agent.*

Définition 3.2.3 *Une **évaluation** est un jugement que porte un agent sur un autre à partir d'observations.*

Définition 3.2.4 *Une **recommandation** est une interaction particulière : il s'agit de la communication par un agent d'une information ayant rapport avec la réputation (observation ou évaluation).*

Finalement, [AH00] souligne particulièrement la différence entre les processus d'initialisation et de révision. Le processus d'initialisation fait référence au cas d'ignorance, c'est-à-dire à l'absence d'interaction, alors que le processus de révision implique l'existence d'un certain nombre d'interactions.

3.2.2 Rôles des agents dans la gestion des réputations

Au cours des différents processus identifiés précédemment, les agents peuvent jouer différents rôles (partie médiane de la figure 3.2). Les rôles identifiés par [CP02] sont repris dans l'ontologie fonctionnelle de la réputation de [CS05] :

- Une *cible* est un agent qui est observé et dont la réputation est évaluée ;
- Un *évaluateur* est un agent qui observe la cible et évalue sa réputation ;
- Un *propagateur* est un agent en position de transmettre une information de réputation, qu'il l'ait évaluée lui-même ou non et qu'il la partage ou non ;
- Un *bénéficiaire* est un agent qui profite de l'évaluation de la réputation de la cible.

3.2.3 Types de réputation

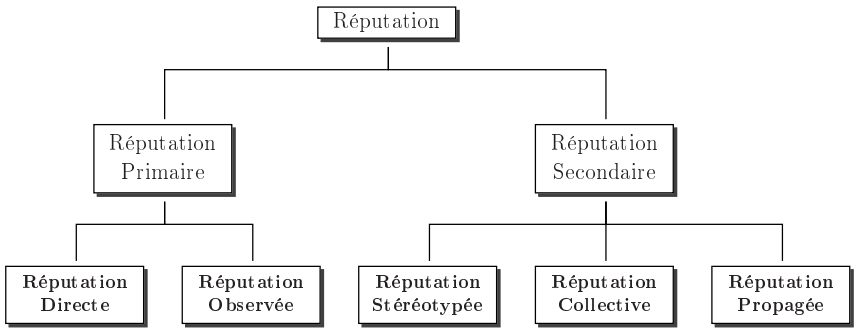


FIG. 3.3 – Typologie des réputations.

La typologie utilisée dans l'ontologie fonctionnelle de la réputation [CS05] (partie supérieure de la figure 3.2) est celle définie par [MHM02] et représentée dans la figure 3.3. Le type de la réputation y est défini en fonction du type et de la source de l'information qui servent à la construire.

En fonction du *type de l'information* qui sert à construire la réputation, il est possible de partager les réputations en deux grands types :

Réputation Primaire Réputation fondée sur des interactions directes entre l'*évaluateur* et la *cible* : l'individu source de l'information est connu. Dans le cadre du commerce électronique, une interaction directe peut, par exemple, consister en une transaction commerciale entre un vendeur et un acheteur.

Réputation Secondaire Réputation fondée sur de l'information ne provenant pas d'une interaction directe entre l'évaluateur et la cible.

Ces deux types de réputation se découpent chacun en sous-types, selon la *source de l'information*. Les deux sous-types composant la Réputation Primaire sont les suivants :

Réputation Directe Cette forme de réputation est estimée par le bénéficiaire après qu'il a effectué lui-même une interaction directe avec la cible. Lors d'une transaction commerciale entre un acheteur et un vendeur, il peut s'agir, par exemple, de la réputation qu'associera le vendeur à l'acheteur.

Réputation Observée Cette forme de réputation est estimée à partir de l'information fournie par des individus entrés en interaction directe avec la cible. L'individu qui reçoit la valeur, n'est pas entré directement en interaction avec la cible. Les individus qui fournissent l'information, en revanche, sont entrés directement en interaction avec la cible. Par exemple, il peut s'agir de la réputation qu'un second acheteur aurait suite à une recommandation de l'acheteur de l'exemple ci-dessus.

Les trois sous-types composant la Réputation Secondaire sont les suivants :

Réputation Stéréotypée Cette forme de réputation est fondée sur des *a priori* sur la cible. Elle n'implique donc pas l'existence d'interactions (directes ou non) avec la cible. Par exemple, certaines personnes estiment qu'une personne propre et bien habillée a plus de chance d'être honnête qu'une personne sale et mal habillée. À partir d'un tel *a priori* un individu pourra ainsi associer une réputation élevée à un vendeur propre et bien habillé (même s'il est inconnu).

Réputation Collective Un individu peut hériter de la réputation du groupe auquel il appartient. Par exemple, un nouveau vendeur dans un certain magasin pourra profiter de la réputation acquise par son groupe de collègues auprès d'un acheteur donné.

Réputation Propagée Cette forme de réputation est la réputation qui est construite à partir d'informations propagées mais pour laquelle l'évaluateur n'a pas mené une interaction directe avec la cible. Il peut s'agir de la réputation d'un vendeur qu'un acheteur aura acquise grâce à de l'information de seconde main, venant d'acheteurs ayant évalué des interactions auxquelles ils n'ont pas participé.

En section 3.1.2, page 33, nous remarquons que les classes de confiance ne sont pas indépendantes les unes des autres. Les liens entre la classe de confiance individuelle et les autres classes se retrouvent ici dans les réputations secondaires : la réputation stéréotypée est en fait une croyance issue de la classe de confiance dispositionnelle et la réputation collective est une croyance issue de la classe de confiance de groupe.

Nous considérons dans la suite uniquement les réputations liées à des interactions, c'est-à-dire les deux réputations primaires et la réputation propagée. Nous employons dans la suite l'expression « Réputation fondée sur les Interactions » pour désigner celles-ci.

3.2.4 Réputations fondées sur les Interactions et rôles

Dans cette section, nous nous intéressons plus particulièrement aux réputations fondées sur les interactions et les explicitons en fonction des rôles que jouent les agents au cours des différents processus. Les figures 3.4 à 3.6 indiquent les rôles que jouent les agents et les types de réputations que les bénéficiaires construisent.

La figure 3.4 présente le cas de la réputation directe. Alice et Bertrand interagissent directement, par exemple en menant une transaction commerciale. Si Alice est à la fois évaluateur, bénéficiaire et propagateur dans le processus d'évaluation de la réputation de Bertrand, alors il s'agit de réputation directe.

La figure 3.5 illustre le cas de la réputation observée. Si Alice est l'évaluateur et le propagateur mais que Charles est le bénéficiaire, alors il s'agit de réputation observée, puisque l'information transmise est une information de première main, de type « observation d'une rencontre directe ».

Finalement, la figure 3.6 présente le cas de la réputation propagée. Si Alice et Bertrand interagissent que Bertrand est la cible et que Charles est évaluateur et propagateur dans une chaîne de longueur quelconque d'agents propagateurs (les différents agents de la chaîne sont symbolisés avec des icônes plus petites dans la figure 3.6) et que Daniel est le bénéficiaire, alors il s'agit de Réputation Propagée car, pour Daniel (le bénéficiaire) la source de la recommandation n'est pas une interaction directe entre l'évaluateur et la cible.

Les différents travaux regroupés dans l'ontologie fonctionnelle de la réputation [CS05] nous permettent de circonscrire encore notre domaine d'étude aux seules réputations fondées sur les interactions qui sont décrites ci-dessus.

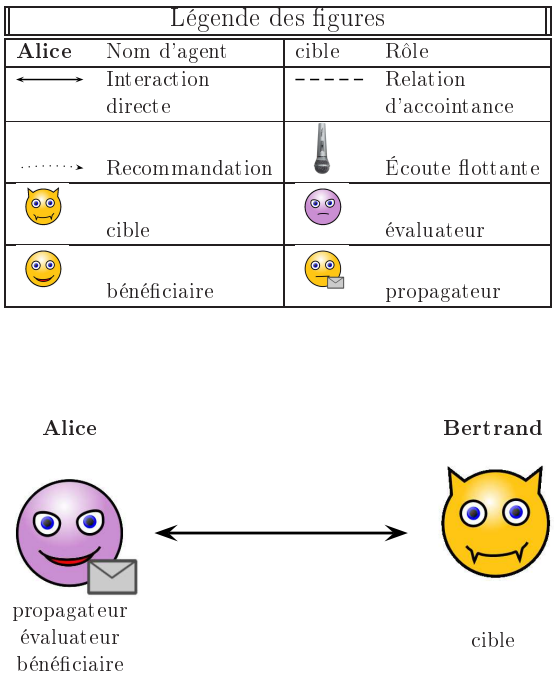


FIG. 3.4 – Réputation directe.

3.2.5 Propriétés des réputations fondées sur les interactions

Les réputations fondées sur les interactions constituent l'objet d'étude principal de cette thèse. Cette section présente l'aboutissement de notre étude de la littérature sur les concepts de confiance et de réputation. Il s'agit d'une caractérisation précise des réputations fondées sur les interactions. Cette caractérisation prend la forme d'un ensemble de propriétés, chacune discutée brièvement. À travers cette spécification précise des réputations fondées sur les interactions, nous détenons tous les éléments nécessaires à la définition d'une grille d'analyse applicable aux modèles computationnels de réputation.

Le terme de réputation (même employé seul) dans cette section fait toujours référence à une réputation fondée sur les interactions.

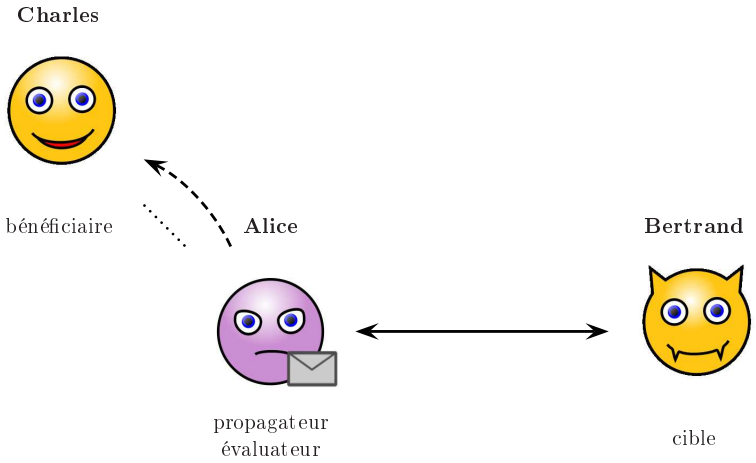


FIG. 3.5 – Réputation observée.

Subjectivité et asymétrie

Les agents n'ont pas la capacité d'observer l'ensemble des événements qui se déroulent dans le système auquel ils appartiennent. Ils ne peuvent donc former que des croyances partielles sur celui-ci. Ainsi, deux agents distincts peuvent interpréter différemment une même observation. En conséquence, les Réputations fondées sur les Interactions sont fortement subjectives [Gam00a, CF98, Sab02, Abd04, BF03, Del01b, ENT+02].

Selon [BDS00] cette propriété de subjectivité est la source de leur asymétrie [GS00, Rou00, SCCD94, Qué01] : un agent `x` peut associer une forte réputation à un autre agent `y` sans que celui-ci ne lui associe, en retour, une forte réputation. Cette asymétrie est d'une importance particulière dans l'utilisation qui est faite des Réputations fondées sur les Interactions. En effet, c'est grâce au fait que ces réputations sont fortement subjectives et asymétriques qu'elles peuvent être utilisées comme sanction sociale. Les Réputations fondées sur les Interactions qui sanctionnent le comportement d'un agent sont maintenues par d'autres agents. Ces autres agents possèdent un contrôle total sur l'évolution de leurs croyances. Le sanctionné ne peut donc échapper à sa sanction (ce n'est pas le cas pour les sanctions matérielles, section 2.3.3, page 27). Toutefois, certains travaux considèrent la réputation symétrique [AD01, Lam01, BBK94].

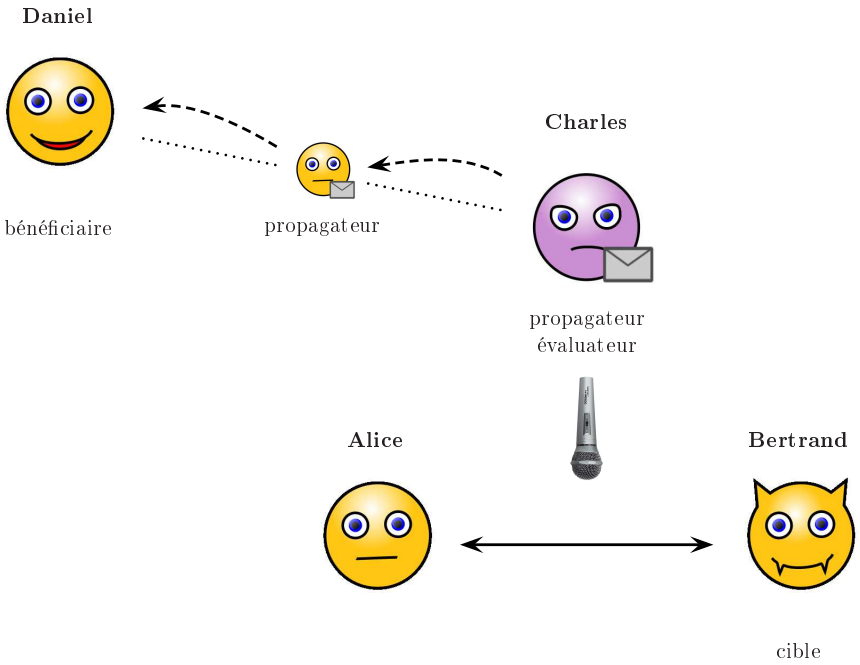


FIG. 3.6 – Réputation propagée.

Multi-facette

Certains modèles (ex. : [ZMM99, CMD03]) associent une Réputation fondée sur les Interactions unique et générale à un agent. Cependant, un agent comporte diverses facettes sur lesquelles un autre agent peut l'évaluer [Sab02, WV03, MD05]. Différentes Réputations fondées sur les Interactions peuvent donc être associées à un même agent selon la facette considérée. Il est possible d'avoir à la fois une forte réputation dans un agent en considérant l'une de ses facettes, et une faible réputation pour ce même agent, mais en considérant une autre de ses facettes. L'exemple 3.2.1 illustre le principe des réputations multi-facettes.

Exemple 3.2.1 *Un agent* x *peut associer une forte réputation à un agent* y ***pour sa capacité à fournir de bonnes informations météorologiques****, mais une faible réputation* ***pour sa capacité à fournir de bonnes informations culinaires****.*

Ces Réputations fondées sur les Interactions multi-facettes ne sont pas forcément indépendantes les unes des autres puisque les différentes facettes d'un même agent peuvent être elles-mêmes liées [Sab02, WV03, MD05].

En fonction des conditions dans lesquelles un agent va prendre une décision de faire confiance, c'est-à-dire en fonction du contexte du processus de décision, il ne fera pas nécessairement intervenir les réputations selon toutes les facettes.

Multi-dimension

Un agent peut aussi différencier les Réputations fondées sur les Interactions qu'il maintient en fonction de la façon dont il les a construites. En effet, il existe plusieurs dimensions selon lesquelles chaque facette d'un agent peut être jugée. Différentes réputations peuvent alors être associées à un même agent selon la dimension considérée.

[CF98] estiment qu'on ne peut faire confiance à un agent pour remplir une tâche que si, au minimum, (1) il est compétent pour faire cette tâche et (2) il est sincère quand il dit qu'il va nous aider. Ils perçoivent donc deux dimensions selon lesquelles un agent peut être évalué : la *compétence* et la *sincérité*.

Pour leur part, [MC01] considèrent que, pour pouvoir faire confiance à un agent, il faut que celui-ci dispose des capacités et du pouvoir de faire ce qui est demandé, qu'il soit possible de prévoir comment il va se comporter, qu'il ait l'intention de nous aider et qu'il soit sincère et honnête. Ils perçoivent donc exactement quatre dimensions : la *compétence*, la *prévisibilité*, la *bonne volonté* et l'*intégrité*.

Il est donc possible qu'un agent associe une forte réputation à un agent en considérant une dimension et une faible réputation pour ce même agent, mais en considérant une autre dimension. L'exemple 3.2.2 illustre une telle situation.

Exemple 3.2.2 *Un concurrent d'une course nautique peut associer une forte réputation à un autre concurrent quant à sa* ***compétence*** *à fournir de bonnes information météorologiques (puisqu'il s'agit d'une condition nécessaire pour gagner la course), mais une très faible réputation quant à son* ***intégrité*** *pour fournir de telles informations (puisqu'il préférera mentir à ses concurrents pour garder l'avantage).*

Suivant le contexte, les réputations selon les différentes dimensions n'entrent pas nécessairement toutes en compte au moment de la prise de décision.

Dynamisme

En tant que croyances [MC01, Qué01, CF98, SN85, SHT73, Bro00, GS00], les Réputations fondées sur les Interactions doivent pouvoir être (et d'ailleurs sont) remises en cause régulièrement. Elles évoluent donc avec le temps en fonction des comportements de la cible. Elles sont dynamiques [Rou01, GS00, CP99, Mar94, RS03].

Graduation

Il n'existe pas d'unité dans laquelle mesurer la réputation [Das90]. Cependant, un agent `x` peut associer une Réputation fondée sur les Interactions plus forte à un agent `y` qu'à un agent `z`. La réputation demeure donc susceptible de degré [AH00, Rou00, MKN98, Qué01].

[Gra03, Del03, Mar94] discutent la représentation de ces degrés sous forme quantitative ou qualitative, discrète ou continue. [Mar94] regrette que certains modèles, en particulier les modèles probabilistes, distinguent des degrés de bonne réputation mais réduisent souvent la défiance à un unique degré.

Évolution

La propriété de dynamisme a pour conséquence une évolution possible de la réputation dans son domaine de graduation. Nous étudions dans cette section les différentes formes d'évolution des réputations.

Si l'augmentation du nombre d'interactions augmente la familiarité entre deux individus [Rou00], elle n'implique pas forcément l'augmentation de la réputation, puisque certaines interactions peuvent se conclure positivement et

d'autres négativement. D'un point de vue général, l'évolution des réputations n'est donc pas monotone [AH00].

En revanche, en différenciant les interactions positives et négatives, il est possible de considérer que la réputation évolue de façon monotone [JT02] : en effet, plus le nombre d'interactions à l'issue positive grandit, plus la réputation est forte. Réciproquement, plus le nombre d'interactions négatives est grand, plus la réputation est faible.

Il est intéressant de noter que, si les auteurs parlent bien dans les deux cas de monotonie au sens mathématique du terme, ils ne l'appliquent pas, en revanche, aux mêmes données. D'un côté l'ensemble des interactions est retenu, de l'autre il est séparé en deux sous-ensembles : interactions positives et interactions négatives. Pour distinguer ces deux concepts, nous posons les définitions 3.2.5 et 3.2.6 suivantes :

Définition 3.2.5 *La **monotonie non différenciée** fait référence à l'évolution monotone des réputations dans le cas où toutes les interactions sont utilisées.*

Définition 3.2.6 *La **monotonie différenciée** fait référence à l'évolution monotone des réputations dans la situation où les interactions positives et négatives sont différenciées.*

[Gam00a, BZL03, MD05] estiment que la réputation est fragile, c'est-à-dire qu'une forte réputation est plus facile à perdre qu'à conserver. Cette propriété peut être considérée de deux points de vue : du point de vue temporel ou du point de vue de l'interaction. Du point de vue temporel, [MD05] définit la fragilité par le fait qu'il faut plus de temps à un agent pour se construire une forte réputation que pour la détruire. Du point de vue de l'interaction [Gam00a] traduit la fragilité par un déséquilibre entre le poids (important) d'une interaction à l'issue négative pour un individu jouissant d'une forte réputation et le poids (moins important) qu'aurait la même interaction pour un individu qui a déjà une faible réputation. La fragilité de la réputation s'explique par le fait qu'il est relativement facile de donner une preuve d'un comportement malfaisant, mais pas d'un comportement bienfaisant.

Finalement, [Fab96, MD05] évoquent le processus d'érosion des réputations : en l'absence de nouvelles interactions, celles-ci s'affaibliraient d'elles-mêmes avec le temps.

Transitivité

Au sens mathématique, la transitivité d'une relation $\rightarrow$ est définie par : $x \rightarrow y \wedge y \rightarrow z \Rightarrow x \rightarrow z$.

Dans le cadre des concepts étudiés ici, il est possible de considérer deux types de transitivité : la transitivité de la décision de faire confiance et la transitivité des réputations.

La transitivité de la décision correspond au fait qu'un agent `a` va décider de faire confiance à un autre agent `b` parce qu'un troisième agent `c` lui a dit qu'il faisait confiance à l'agent `b` et parce que l'agent `a` fait lui-même confiance à l'agent `c`. C'est dans le but de générer des modèles de réputation ayant cette propriété de transitivité de la décision que certains auteurs considèrent que la réputation est transitive [OP05, Abd97, ZMM99, JP05].

La transitivité des Réputation fondée sur les Interactions est définie comme la possibilité d'obtenir le degré de réputation qu'un agent associe à un autre en combinant, le long de « chaînes » plus ou moins longues, les réputations d'agents intermédiaires. Généralement, plus les chaînes sont longues, plus la réputation estimée par transitivité faiblit. La figure 3.7 illustre l'exemple 3.2.3 de transitivité des réputations.

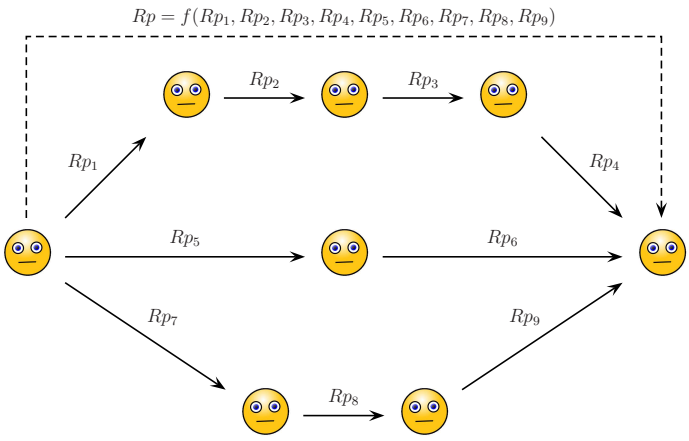


FIG. 3.7 – Transitivité de la réputation.

Exemple 3.2.3 *Soient, comme le montre la figure 3.7, trois chaînes de réputation reliant l'agent* `x` *et l'agent* `y`*, respectivement de longueur quatre, deux, trois. La réputation estimée par transitivité de l'agent* `x` *en l'agent* `y` *est évaluée en fusionnant, à l'aide d'une fonction* f*, les réputations* $rp1, \ldots, rp9$ *se trouvant le long de ces trois chaînes.*

De nombreux auteurs [CH96, GS00, JF01, AH97a, AH97b, Jøs96] estiment que la transitivité de la réputation n'existe pas dans la nature. En effet, les réputations sont des croyances subjectives, donc propres à chaque agent. De ce fait, il est impossible de combiner des réputations puisqu'elles ne sont ni présentes dans un même lieu, ni accessibles publiquement. Ainsi, le seul moyen qu'a un agent pour accéder aux croyances d'un autre agent sont les recommandations que veut bien fournir ce dernier. Or, d'une part, celles-ci ne sont pas nécessairement sincères et, d'autre part, un agent qui reçoit une recommandation n'est pas tenu de l'accepter (par exemple, parce qu'il ne fait pas confiance à l'émetteur de la recommandation). La transitivité des réputations n'est donc pas systématique et automatique. Ce caractère non systématique a engendré la notion de « transitivité conditionnelle » [AH97a]. Enfin, un argument avancé contre les modèles implémentant la transitivité des réputations est que ceux-ci agrègent les réputations le long d'une même chaîne en mélangeant différentes facettes [Abd04].

3.3 Synthèse

En suivant une approche descendante, nous sommes partis de la définition générale de la confiance et avons caractérisé les différents concepts au fur et à mesure de leur apparition. Nous avons ainsi circonscrit notre domaine de recherche et en avons obtenu une spécification précise.

Dans ce chapitre, nous avons étudié les différentes définitions de la confiance et de la réputation en nous appuyant sur des travaux principalement issus des sciences humaines, sociales et économiques. Grâce aux travaux de [MC01] nous avons centré nos efforts sur la seule classe de confiance interpersonnelle et avons défini la réputation comme les croyances sur lesquelles un agent s'appuie pour prendre la décision d'agir en confiance avec un autre agent. Nous avons ensuite étudié les réputations à l'aide d'une ontologie fonctionnelle de la réputation [CS05] et conclu qu'il existait différents types de réputation, que l'on peut distinguer en fonction du type et de la source de l'information

utilisée pour les construire [MHM02]. Finalement, nous avons abouti à une spécification des Réputations fondées sur les Interactions : *toute Réputation fondée sur les Interactions se définit à un instant donné, d'un agent envers un autre, pour une certaine facette et selon une certaine dimension.*

L'analyse en profondeur des propriétés de ces réputations peut alors être utilisée pour étudier les différents modèles computationnels de réputation. En particulier, il est possible d'étudier si et comment ces modèles implémentent les différentes propriétés. Pour ce faire, nous définissons la grille d'analyse des modèles computationnels de réputation suivante :

- Le cadre des travaux ;
- Les définitions utilisées ;
- Les propriétés prises en compte ;
- Les processus implémentés ;
- Certains paramètres spécifiques à la mise-en-œuvre computationnelle.

Le cadre des travaux doit être pris en compte car il permet souvent d'expliquer certains choix des modèles computationnels. Nous avons vu qu'il existe des définitions très différentes de la confiance et de la réputation. Il est possible d'utiliser les classes de confiance et les types de réputation pour analyser les définitions utilisées dans les modèles computationnels. Afin d'affiner l'étude des définitions, il est important de prendre aussi en compte les propriétés (subjectivité, facettes, dimensions, transitivité et graduation) que ces modèles attribuent aux réputations qu'ils utilisent. Nous avons vu aussi qu'il existe six processus liés à la réputation. Ceux-ci sont issus des propriétés de dynamisme et d'évolution des Réputations fondées sur les Interactions. Il est donc intéressant de comparer si et comment les différents modèles computationnels implémentent ces processus. Finalement, d'autres paramètres, plus spécifiques à la mise-en-œuvre computationnelle peuvent être pris en compte. Par exemple, quels types d'agents peuvent interagir dans le système.

Le chapitre suivant étudie les modèles computationnels de réputation actuels en fonction de la façon dont ils instancient ces différents éléments.

Chapitre 4

Modèles computationnels de réputation

Dans cette section, nous étudions les modèles computationnels de réputation. Il en existe une grande quantité et de nombreux modèles s'ajoutent chaque année [FBKS03, FBSS04, CBSS05]. De ce fait, il est pratiquement impossible d'être exhaustif dans leur description. Notre approche est donc la suivante : nous utilisons la grille d'analyse définie précédemment pour catégoriser les modèles et ne présentons ici que certains modèles, illustrant chaque catégorie.

Les modèles sont caractérisés selon l'ensemble des critères de la grille, mais le seul critère des processus a suffit pour construire les catégories. Ce critère, nous a permis d'établir des catégories reflétant la plus ou moins grande dépendance des modèles à l'intervention humaine. Les catégories sont les suivantes : modèles d'assistance à l'utilisateur, modèles avec évaluation, modèles avec punition, modèles avec raisonnement et enfin modèles avec décision.

Ce chapitre est structuré comme suit : tout d'abord, les différentes valeurs que peuvent prendre les critères de la grille sont détaillées. Ensuite, les différentes catégories de modèles sont présentées. La présentation des différentes catégories suit un ordre qui reflète la dépendance à l'intervention humaine. Nous présentons d'abord les catégories les plus dépendantes de l'intervention humaine : modèles d'assistance à l'utilisateur, modèles avec punition, modèles avec décision et finissons avec celles les moins dépendantes : modèles avec évaluation, modèles avec raisonnement.

4.1 Caractérisation des modèles

Dans cette section nous présentons brièvement les différentes valeurs que peuvent prendre les attributs de la grille d'analyse.

- Le **cadre** caractérise le domaine d'application du modèle et est décrit textuellement par quelques phrases.
- La **définition** correspond aux différentes classes de confiance (dispositionnelle, institutionnelle, interpersonnelle, de groupe, voir section 3.1.2) et différents types de réputation (directe, observée, propagée, stéréotypée, collective, voir section 3.2.3) que le modèle considéré implémente.
- Les **processus** sont généralement décrits à l'aide d'algorithmes. Nous nous intéressons principalement ici au fait qu'un algorithme est proposé ou non pour chaque processus. Nous ne donnons ici que le principe général des algorithmes proposés.
- Les **propriétés** :
  - *Subjectivité* : chaque type de réputation modélisé peut être subjectif ou non.
  - *Facettes* : chaque réputation peut s'adresser à l'agent dans son ensemble (on dit qu'elle est uni-facette), ou s'adresser à une facette particulière (on dit qu'elle est multi-facette). Parfois aussi, un nombre précis de facettes peut être indiqué.
  - *Dimensions* : chaque réputation peut être calculée selon une seule ou plusieurs dimensions. Parfois aussi, un nombre précis de dimensions peut être indiqué. Ce nombre appartient à l'ensemble $\{1, 2, 3, 4\}$, en référence aux quatre dimensions proposées par [MC01] : compétence, prévisibilité, bonne volonté et intégrité (voir section 3.2.5).
  - *Transitivité* : les modèles peuvent mettre en place un calcul par transitivité ou non.
  - *Graduation* : le domaine de valeur ou la structure de données employée pour représenter les réputations est précisé. Il est parfois nécessaire de différencier la représentation computationnelle des évaluations (issues du processus d'évaluation) et la représentation des réputations (issues du processus de punition).

4.2 Modèles d'assistance à l'utilisateur

Les modèles regroupés sous le nom de « modèles d'assistance à l'utilisateur » sont des modèles de réputation dans lesquels l'être humain a une place prépondérante. Le modèle lui-même ne gère que le stockage des valeurs de réputation et propose des interfaces permettant à l'utilisateur de les manipuler.

4.2.1 OpenPGP

Les différentes versions de PGP (« Pretty Good Privacy » [NAI99b]) sont des programmes de cryptographie qui peuvent servir à établir des certificats cryptographiques. Un certificat est une association entre une identification (ex. : numéro de carte d'identité) et une clef publique. De manière à garantir l'intégrité de cette association, elle doit être signée à l'aide d'une clef privée par un tiers. Dans la version originale de PGP, il s'agit d'un tiers de confiance : banque, gouvernement, grande entreprise, etc. L'originalité d'OpenPGP [OP05, Abd97] par rapport à PGP est de proposer que chaque individu puisse être le signataire de certificats. Il est alors nécessaire de gérer explicitement les réputations des signataires.

OpenPGP propose de manipuler deux réputations : l'une dans le certificat lui-même et prenant ses valeurs dans { `undefined`, `marginal`, `complete` } ; l'autre dans la capacité d'un certificat (en fait de l'individu référencé dans celui-ci) à signer d'autres certificats. Cette réputation est choisie dans : { `full`, `marginal`, `untrustworthy`, `don't know` }.

Lors de « signing parties », des individus se rencontrent physiquement et vérifient l'identité de chaque interlocuteur. Ils signent et enregistrent alors les certificats correspondants à l'aide de leur logiciel. Ils peuvent aussi affecter des valeurs aux réputations de ces certificats. Le logiciel peut ensuite calculer la réputation de certificats inconnus, par transitivité sur les différentes chaînes de réputation existant entre le bénéficiaire et la cible inconnue.

OpenPGP manipule donc deux classes de confiance, l'une dans les objets, l'autre de classe interpersonnelle, associée à une réputation de type directe. Cette dernière est subjective, uni-facette, potentiellement multi-dimensionnelle (selon comment l'humain l'estime), graduée et transitive. En dehors du calcul de transitivité, l'intégralité des processus est menée par des humains.

4.3 Modèles avec punition

Les modèles avec punition sont fortement dépendants de l'intervention humaine. N'automatisant que le calcul des réputations, ils n'implémentent que le processus de punition. D'autres processus, en particulier l'évaluation des comportements, nécessitent toujours une intervention humaine.

4.3.1 Modèles de réputation sur la toile

Il existe de nombreux sites Internet utilisant la réputation. eBay et OnSale proposent des modèles de réputation destinés à fiabiliser les enchères [eBa03, OnS03]. Ces deux sites fonctionnent selon des principes très similaires. Dans cette section, le fonctionnement du modèle de réputation d'eBay est détaillé ; seules les différences majeures entre les deux modèles sont soulignées.

Après chaque vente, les acheteurs sont invités à évaluer les vendeurs quant à leur comportement pendant la transaction. Pour ce faire, ils peuvent laisser sur le site une évaluation et un commentaire textuel. L'évaluation est une note dans $\{-1, 0, +1\}$. La réputation d'un vendeur est alors calculée automatiquement par addition de l'ensemble des notes laissées par les différents acheteurs. Il s'agit donc d'une valeur dans $[-\infty, +\infty[$. Dans eBay, un système simple de normalisation empêche un unique acheteur qui a laissé plusieurs évaluations d'un même vendeur d'avoir trop d'influence sur la valeur finale de réputation.

La décision d'agir en confiance ou non avec un vendeur reste sous l'entière responsabilité de l'être humain qui souhaite participer à une enchère. Le modèle de réputation ne fournit ici qu'un moyen pour un acheteur d'accéder à la réputation d'un vendeur. Lorsqu'il cherche à acquérir un objet, un acheteur peut recevoir plusieurs offres. Afin de décider de mener ou non la transaction et avec quel vendeur, l'acheteur peut consulter le profil de chaque vendeur. Ce profil se décompose en trois parties principales :

1. Différentes informations concernant le vendeur depuis son entrée dans le système : sa réputation, ainsi que quelques statistiques sur l'évolution des évaluations ;
2. L'évolution des évaluations sur différentes échelles de temps : le dernier mois, les six derniers mois, les douze derniers mois ;
3. L'historique complet des évaluations accompagnées des commentaires textuels.

Dans le cas où le vendeur n'a reçu aucune évaluation, il se voit doté d'une réputation de 0 sur eBay, mais n'a pas de réputation sur OnSale.

Les réputations manipulées par ces modèles sont calculées à partir d'évaluations laissées par des acheteurs. Pour un acheteur qui n'a jamais interagi avec un certain vendeur, il s'agit de réputation observée puisque les évaluations proviennent toutes d'individus ayant interagi directement avec la cible. Pour un acheteur qui a déjà interagi avec le vendeur, il s'agit d'un mélange entre de la réputation observée et de la réputation directe.

La réputation est non subjective, uni-facette, potentiellement multi-dimensionnelle (selon comment l'humain l'estime), graduée et non transitive. En dehors du processus de punition, l'intégralité des processus est menée par des humains.

[Del01b] émet l'hypothèse que ce système fonctionne relativement bien surtout du fait de la présence du commentaire textuel. De plus, afin de limiter les changements d'identité, OnSale requiert un numéro de carte bancaire à l'inscription.

4.3.2 Sporas et Histos

Toujours dans le domaine des enchères électroniques, [ZMM99] propose deux modèles de réputation complémentaires, Sopras et Histos. Nous avons séparé ces modèles des deux précédents car il s'agit de travaux de recherche qui ne sont pas déployés à aussi large échelle.

Dans Sporas, les acheteurs jugent les vendeurs à l'aide d'évaluations dans $[0.1, +1]$. La réputation finale est une valeur dans $[0, 3000]$. Ce modèle repose sur l'hypothèse (assez peu réaliste pour les systèmes ouverts) qu'un vendeur commence avec une réputation la plus basse possible (ici, 0) et ne doit jamais pouvoir descendre plus bas. La punition s'effectue par renforcement de l'ancienne valeur. Le renforcement est pondéré par le temps et par la réputation de l'évaluateur. Histos s'appuie sur l'existence d'un modèle de type Sporas et propose le calcul de réputations par transitivité.

Pour les mêmes raisons qu'eBay, Sporas manipule des réputations qui mélangent les types réputation directe et réputation observée. La réputation est non subjective, uni-facette, potentiellement multi-dimensionnelle (selon comment l'humain l'estime) et graduée. Dans Sporas elle est non transitive, dans Histos elle est transitive. En dehors du calcul des réputations, l'intégralité des processus est menée par des humains.

4.4 Modèles avec décision

Les modèles de réputation avec décision permettent non seulement de calculer automatiquement les réputations, mais aussi de prendre des décisions en s'appuyant sur celles-ci. Ces modèles sont plus indépendants de l'intervention humaine que ceux de la catégorie précédente, mais la révision n'est toujours pas entièrement automatisée.

4.4.1 Modèle de Abdul-Rahman et Hailes

Abdul-Rahman propose un modèle de réputation appliqué au partage de temps-processeur entre utilisateurs dans un réseau de type pair-à-pair [Abd04]. Les réputations des personnes qui veulent exécuter du code sont estimées par les personnes qui disposent de temps processeur afin de déterminer si ces dernières acceptent d'exécuter un code venant d'une autre personne. Pour cela, plusieurs réputations sont utilisées : la réputation dans un agent donné pour une facette donnée (type réputation directe), la réputation dans un agent en général (type réputation directe, mais uni-facette), une réputation collective et une réputation stéréotypée. Abdul-Rahman considère le fait de fournir des recommandations comme une facette d'un agent. Il distingue une deuxième réputation directe, celle dans un agent pour fournir des recommandations sur d'autres agents. L'utilisateur du logiciel pair-à-pair donne le retour sur le bon ou mauvais comportement de l'autre partie. Cette estimation est retournée à l'aide d'une valeur dans l'ensemble $\{-2, -1, 0, +1, +2\}$. Les réputations sont calculées à partir des historiques complets des interactions et sont elles aussi des valeurs dans $\{-2, -1, 0, +1, +2\}$.

L'apport principal de ce modèle par rapport à ceux de la catégorie précédente est un premier pas vers l'automatisation du processus de décision. L'utilisateur définit la politique à suivre à l'aide d'un langage fourni. L'agent peut alors appliquer le processus de décision automatiquement. Comme l'ensemble des réputations est discret, les différentes valeurs peuvent servir de seuil de décision dans ces politiques. Abdul-Rahman propose aussi d'utiliser la fiabilité des valeurs de réputation durant le processus de décision. Ces fiabilités sont estimées à partir de la réputation des fournisseurs de recommandations.

Abdul-Rahman propose un modèle manipulant de nombreux types de réputation liées aux différentes classes de confiance. Certaines des réputations primaires sont uni-facettes d'autres sont multi-facettes. Ces réputations sont

toutes subjectives et graduées dans des ensembles finis et discrets. Elles sont non transitives. La notion de dimension n'est ni présente ni déductible du modèle. Les processus de punition et de décision sont en grande partie indépendants de l'humain. Le processus de propagation semble avoir toujours lieu et les recommandations être toujours sincères. L'initialisation s'effectue en utilisant les réputations collective et stéréotypée. Concernant les autres processus, rien n'est précisé.

4.4.2 Modèle de Marsh

Marsh est le premier à avoir proposé d'informatiser le concept de confiance [Mar94]. Il propose un modèle général pour l'utilisation de la confiance dans le cadre de la coopération. Il ne s'est pas intéressé au processus de propagation ni aux réputations qui en sont issues. Il définit trois sortes de réputation : *Basique* qui est une réputation stéréotypée, *Générale* et *Situationnelle* qui sont des réputations directes, la première uni-facette, la seconde multi-facette. Les trois réputations sont représentées par des valeurs dans $[-1, +1[$. Marsh estime que si la confiance aveugle existait, alors le fait même de chercher à estimer la réputation d'un agent ne serait plus nécessaire. Il considère donc que la confiance aveugle n'existe pas et retire la valeur $+1$ du domaine de valeurs possibles pour les réputations. [Mar94] ne précise pas d'où provient l'information qui permet de calculer les réputations, ni comment les réputations sont initialisées. Il s'est intéressé principalement aux processus de punition et de décision. Les réputations sont calculées par rapport à l'ensemble des interactions.

L'apport principal de ce modèle est un processus de décision entièrement automatisé. Chaque décision est prise par seuillage : si la valeur de réputation *Situationnelle* est supérieure au seuil de coopération, alors l'agent coopère, sinon l'agent ne coopère pas. Marsh propose plusieurs définitions de ce seuil de coopération en fonction : des risques perçus par le bénéficiaire, de la compétence de la cible telle que perçue par le bénéficiaire et de l'importance de la situation pour le bénéficiaire.

Marsh propose un modèle manipulant des réputations fondées sur des interactions directes cible–bénéficiaire. Certaines sont uni-facettes d'autres sont multi-facettes. Ces réputations sont toutes subjectives, graduées et non transitives. La notion de dimension n'est ni présente ni déductible du modèle. Les processus de punition et de décision sont indépendants de l'humain, pour les autres processus, rien n'est précisé.

4.4.3 AFRAS

AFRAS est un modèle de réputation pour le commerce électronique où les réputations sont générales pour un agent [CMD03]. Elles sont représentées par des ensembles flous décrits par les quatre côtés d'un trapèze. L'évaluation issue du processus d'évaluation est fournie par un être humain. Elle prend la forme d'une valeur sélectionnée parmi un ensemble fini et discret et est traduite en un ensemble flou. La punition se déroule par renforcement de l'ancienne valeur en comparaison avec une telle évaluation. Cette punition s'effectue à l'aide de formules très similaires à celles utilisées dans Sporas [ZMM99], adaptées aux ensembles flous.

Le processus de décision s'appuie sur le modèle proposé par [CF98]. Comme dans le modèle précédent, la décision s'effectue par seuillage, mais de manière adaptée aux ensembles flous. Si la réputation d'un agent est supérieure à un certain seuil, alors l'acheteur achète au vendeur. Sinon, le bénéficiaire choisit les agents auxquels demander des recommandations par seuillage sur la réputation. Les agents qui reçoivent des demandes de recommandations décident s'ils répondent ou non de la même manière. Le processus de propagation utilise donc le processus de décision.

AFRAS manipule des réputations mélangeant le type réputation directe et le type réputation observée. Elles sont subjectives, uni-facette, graduées et non transitives. La notion de dimension n'est ni présente ni déductible du modèle. L'évaluation dépend de l'intervention humaine. Les processus de punition, de décision et de propagation sont entièrement automatisés. Pour les autres processus, rien n'est précisé.

4.5 Modèles avec évaluation

Les modèles avec évaluation marquent un seuil dans la hiérarchie des catégories que nous avons établie : ils constituent la première catégorie qui permet à des agents de calculer les réputations entièrement par eux-mêmes. De plus, lorsque les processus de révision et de décision sont intégralement automatisés, les agents peuvent réaliser les deux étapes primordiales sans intervention humaine et évoluer seuls dans un environnement incertain.

4.5.1 Modèle de Schillo et Funk

[SFR99, SF99] proposent un modèle de réputation fortement ancré dans la théorie des jeux. Le scénario que les auteurs utilisent est un dilemme du prisonnier itéré modifié, se déroulant comme une enchère. Les agents s'évaluent les uns les autres à l'aide de réputations représentées par des probabilités (donc dans $[0, 1]$). Les réputations déployées ici sont de types réputation directe et réputation observée.

Ce modèle se différencie de ceux des catégories précédentes par le fait qu'il propose une automatisation du processus d'évaluation. Cette évaluation est possible du fait qu'un agent annonce à l'avance le coup qu'il va jouer. Il est alors aisé pour un agent de comparer le coup effectivement joué au coup qui avait été annoncé. De ce fait, le processus d'évaluation peut être automatisé. Une valeur dans $\{0, 1\}$ est ainsi utilisée pour représenter le fait que la cible a été intègre et de bonne volonté [SF99].

La punition de la réputation directe est estimée par le nombre de bons comportements rapportés au nombre d'interactions. Les réputations observées sont calculées en fusionnant les recommandations à l'aide d'un algorithme probabiliste permettant de limiter l'influence des « correlated evidences » [Pea88].

Les réputations considérées dans ce modèle sont de type directe et observée. Elles sont subjectives, uni-facettes, bi-dimensionnelles, graduées et non transitives. L'intégralité du processus de révision est ici automatisé. Le processus de propagation a toujours lieu, sur demande des autres agents et les agents ne mentent jamais, ils cachent seulement de l'information de temps en temps. Les autres processus ne sont pas détaillés, en particulier le processus de décision est absent.

4.5.2 Modèle de Sen et Sajja

[SS02] propose un modèle de réputation pour aider des agents utilisateurs, qui ont des tâches à exécuter, à sélectionner des agents processeurs, qui peuvent exécuter ces tâches. Comme dans le modèle précédent, les réputations modélisées sont de type réputation directe et réputation observée. Elles estiment la compétence d'un agent processeur à exécuter une tâche. Elles sont représentées par des valeurs dans $[0, 1]$. La réputation d'un agent inconnu est initialisée à la valeur médiane 0.5.

Ce modèle se distingue aussi de la catégorie précédente par le fait qu'il

propose un processus d'évaluation. Ici, les agents utilisent une estimation de la qualité de l'exécution de la tâche, qu'ils rapportent ensuite à une valeur dans $\{0, 1\}$ estimant la qualité du service rendu. Bien que purement binaire, le processus d'évaluation est donc réalisé automatiquement. Les recommandations fournies par d'autres agents sont aussi des valeurs dans $\{0, 1\}$. Les réputations sont calculées par une formule de renforcement de l'ancienne valeur de réputation.

La décision est prise par seuillage : si la réputation directe est supérieure à 0.5, alors l'agent utilisateur fait confiance à l'agent processeur, sinon il ne lui fait pas confiance. Si la réputation directe ne suffit pas à l'agent pour décider, il peut demander des recommandations.

Les réputations considérées dans ce modèle sont de type directe et observée. Elles sont subjectives, uni-facettes, uni-dimensionnelles, graduées et non transitives. L'intégralité du processus de révision est ici automatisé, ainsi que le processus de décision. Ce modèle est donc implémentable dans des agents autonomes. Le processus de propagation a toujours lieu, sur demande des autres agents. Un agent qui ment le fait toujours et dit toujours l'inverse de ce qu'il croit.

4.6 Modèles avec raisonnement

Les modèles de réputation avec raisonnement automatique ajoutent encore à l'indépendance entre l'agent et l'intervention humaine. Ils permettent, en particulier, aux agents qui les implémentent d'utiliser des modèles plus riches, faisant intervenir de nombreuses facettes.

4.6.1 Modèle de Wang et Vassileva

[WV03] propose un modèle de réputation pour l'échange de fichiers dans les réseaux pair-à-pair. Les personnes qui fournissent des fichiers sont évalués par les personnes qui téléchargent à l'aide de réputations séparées par facettes. Les deux types de réputation utilisés ici sont la réputation directe et la réputation observée. Les auteurs représentent les réputations, différenciées par facettes, par des probabilités et les relient par des réseaux bayésiens. Contrairement aux modèles de la section précédente, l'évaluation est ici menée par l'utilisateur du client de partage de fichiers pair-à-pair. Il fournit un degré de satisfaction transformé en une évaluation dans $\{0, 1\}$. Le pro-

cessus de punition de la réputation directe se fait grâce à un renforcement paramétrable de l'ancienne valeur de réputation. Le processus de punition de la réputation observée s'opère par renforcement des probabilités du réseau bayésien du bénéficiaire. Ce renforcement s'effectue par comparaison entre le réseau bayésien fourni comme recommandation et celui du bénéficiaire.

La plus-value de ce modèle est le processus de raisonnement. Il s'opère par fusion des réputations selon différentes facettes à l'aide des réseaux bayésiens. Par exemple, la réputation d'un pair pour fournir des fichiers est liée à ses réputations pour fournir rapidement les fichiers, pour fournir des fichiers de bonne qualité et pour fournir des fichiers du type recherché. La réputation dans l'agent peut alors se calculer à l'aide de la règle de Bayes appliquée aux réputations selon les facettes de l'agent. Les réputations issues du processus de raisonnement permettent de décider par un seuillage simple.

Le modèle présenté ici manipule deux types de réputation : directe et observée. Elles sont subjectives, uni-dimensionnelles (compétence), graduées et non transitives. Une réputation générale dans un agent (uni-facette) peut être calculée à partir des réputations selon différentes facettes. Les processus de punition, de raisonnement et de décision sont automatisés. Le processus d'évaluation requiert l'intervention humaine. Le processus de propagation a toujours lieu, sur demande des autres agents et est sincère.

4.6.2 Modèle de Melaye et Demazeau

[MD05] s'intéressent principalement au processus de raisonnement. Dans ces travaux, un réseau bayésien est hiérarchisé en trois niveaux : les sources d'information (niveau bas) sont liées aux réputations multi-facettes (niveau intermédiaire) et les réputations multi-facettes sont liées à la réputation générale de l'agent (niveau haut). Ces travaux proposent une approche similaire à la précédente, à l'aide d'un filtre de Kalman pour calculer la réputation générale d'un agent en fonction des réputations multi-facettes.

Ce modèle manipule un seul type de réputation, la réputation directe. Elle est subjective, graduée et non transitive. La réputation de l'agent en général est calculée à l'aide des réputations selon les différentes facettes. Les auteurs évoquent différentes dimensions (compétence et bonne volonté). Ce modèle prend en compte les notions de fragilité et d'érosion des réputations avec le temps. Seuls les processus de raisonnement et de décision sont automatisés.

4.6.3 Modèle de Sabater i Mir et Sierra

[Sab02] propose un modèle de réputation pour le commerce électronique. Les commerçants s'évaluent à l'aide de réputations directe, propagée, collective et stéréotypée. Ces réputations prennent leurs valeurs dans $[-1, +1]$. Les commerçants négocient des contrats. Ceux-ci sont représentés de manière formelle dans le modèle à l'aide de conjonctions de paires attributs–valeurs. Une fois exécuté, un contrat peut être représenté de la même façon, les valeurs associées aux différents attributs pouvant différer du contrat prévu. Le processus d'évaluation consiste alors à comparer le contrat tel qu'il a été exécuté au contrat tel qu'il était prévu. La réputation directe est calculée par une moyenne pondérée des évaluations des contrats passés entre la cible et le bénéficiaire. La pondération dépend du temps (les anciennes évaluations ont moins d'importance) et de l'importance de chaque attribut du contrat pour le bénéficiaire. La réputation propagée est calculée par rapport à un ensemble de témoins attentivement sélectionnés afin de limiter le problème de « correlated evidence » [Pea88]. Les témoins sont sélectionnés s'ils ont beaucoup interagi avec la cible. La réputation collective est estimée en fonction des réputations des agents avec lesquels la cible a de forte relations sociales. La réputation stéréotypée dépend simplement du rôle de l'agent et n'évolue pas. Parallèlement à chaque valeur de réputation, est calculée sa fiabilité. Le processus de décision consiste à utiliser la réputation directe si sa fiabilité est suffisante. Si elle ne l'est pas, alors une combinaison linéaire des autres types de réputation est utilisée. Dans le cas d'agents inconnus, les réputations secondaires sont donc utilisées.

L'apport principal de ce modèle est de proposer un processus de raisonnement s'appuyant sur une ontologie du domaine. Par exemple, l'ontologie peut exprimer que la réputation d'une compagnie aérienne repose sur la réputation qu'elle a de posséder de bons avions, de ne jamais perdre de bagages et de fournir de bons repas. Cette ontologie relie les différentes facettes par des pondérations représentant l'importance avec laquelle elles interviennent. Ces pondérations sont alors utilisées pour fusionner, par combinaison linéaire, les réputations selon les différentes facettes en une réputation globale.

Ce modèle est le plus riche que nous ayons étudié puisqu'il implémente quasiment tous les types de réputations définis et tous les processus associés. Les réputations considérées sont subjectives, multi-facettes, potentiellement multi-dimensionnelles (selon les attributs du contrat considérés), graduées et non transitives. L'intégralité des processus est automatisée.

4.7 Synthèse

Dans ce chapitre, nous avons étudié le fonctionnement de différents modèles computationnels de réputation. Pour ce faire, nous nous sommes intéressés à la façon dont ils implémentent les différents aspects de la grille définie à la fin du chapitre précédent.

En particulier, partant de l'observation que peu de modèles implémentent l'intégralité des processus, nous nous sommes intéressés à caractériser les processus qu'ils implémentent. Nous avons alors pu définir des catégories de modèles. Ces catégories établissent une progression dans l'automatisation des processus liés aux réputations. Cette progression permet de passer de modèles d'assistance à l'utilisateur, fournissant une simple interface de manipulation des réputations à des modèles automatisant l'intégralité des processus.

	Processus						
	Init.	Révision		Rais.	Déci.	Prop.	
		Éval.	Puni.				
OpenPGP	O	**X**	H	H	H	H	-
eBay	O	H	**O**	H	H	H	
Zacharias	O	H	O	N	H	N	
AbdulRahman	O	H	O	N	**O**	I	
Marsh	N	N	O	N	O	N	
AFRAS	N	H	O	N	O	O	
Schillo	N	**O**	O	N	N	I	
Sen	O	O	O	N	O	I	
Wang	N	H	O	**O**	O	I	
Melaye	N	N	N	O	O	N	
Sabater	O	O	O	O	O	O	+

autonomie

Légende :							
O	oui	N	non	Init.	initialisation	Rais.	raisonnement
H	humain	I	inféré	Éval.	évaluation	Déci.	décision
X	non pertinent			Puni.	punition	Prop.	propagation

TAB. 4.1 – Synthèse des modèles computationnels de réputation étudiés.

La table 4.1 synthétise cette approche. O (resp. N) signifie qu'un algorithme est (resp. n'est pas) proposé pour le processus correspondant. H signifie que le modèle nécessite une intervention humaine. X signifie qu'un

champ n'est pas pertinent pour le modèle. Finalement, I signifie que le modèle n'explicite pas le processus entièrement, mais qu'il a été possible d'inférer en partie son fonctionnement. Les cases en gras pointent le critère discriminant entre deux catégories. Les traits doubles entre les lignes marquent les séparations entre les catégories de modèles.

Les modèles les plus « primaires », font intervenir l'humain à la fois pour calculer les valeurs de réputation et pour prendre une décision à partir ces valeurs (catégorie « assistance à l'utilisateur ») [OP05, Abd97]. Ensuite viennent les modèles « avec punition » [Abd04, Mar94], qui sont un peu moins attachés à l'intervention humaine puisqu'ils automatisent le processus de punition. Notons que tous les processus de punition présentés ici ont une évolution monotone différenciée. Seul [MD05] propose une évolution explicitement fragile. De plus, [ZMM99, Sab02, MD05] proposent des modèles où la réputation s'érode avec le temps. Quant aux modèles de la catégorie « avec décision » [Abd04, Mar94, CMD03], ils permettent non seulement de calculer les niveaux de réputation, mais aussi de prendre des décisions de faire confiance en s'appuyant sur ces niveaux. Il permettent ainsi de s'abstraire encore un peu plus de l'intervention humaine. Cependant, ils nécessitent toujours une telle intervention pour les évaluations.

Les modèles de la catégorie « avec évaluation » [SFR99, SF99, SS02] créent une césure dans la hiérarchisation des catégories. Le triple trait marque cette césure sur la table 4.1. En implémentant de manière totalement automatique le processus de révision, ils permettent d'envisager une implémentation dans des agents totalement indépendants de leur concepteur et / ou utilisateur, à condition d'implémenter à la fois les processus de révision et de décision. Enfin, les modèles « avec raisonnement » [WV03, Sab02] vont encore plus loin en implémentant le processus de raisonnement et, parfois, celui de propagation. Ils permettent ainsi l'utilisation par des agents de modèles multi-facettes avec un moyen automatique de lier les facettes entre elles.

En outre, notre étude des modèles computationnels de réputation nous a permis d'identifier trois types de comportements attendus :

- Un comportement *bienveillant* est un comportement toujours correct, par exemple l'envoi de recommandations sincères.
- Un comportement *trivial* consiste, par exemple, à mentir toujours, à mentir de temps en temps ou à mentir toujours de la même façon.
- Un comportement *malveillant* est un comportement difficile à caractériser et imprévisible, tel qu'un être humain pourrait avoir.

Ces comportements peuvent différer selon s'il s'agit des comportements at-

tendus quant au domaine d'application du modèle, ou par rapport à la capacité d'un agent à fournir des recommandations. Seuls les modèles [eBa03, OnS03, OP05] sont vraiment confrontés à des comportements malveillants, du fait qu'ils sont déployés à large échelle. Contrairement à ce que l'on pourrait attendre, les autres modèles ne sont pas tous préparés à affronter de tels comportements. Ainsi, [SF99, SS02] travaillent sur des comportements triviaux aussi bien pour le domaine d'application que pour les recommandations et [Abd04, WV03] s'appuient sur des comportements bienveillants pour les recommandations.

L'instanciation des autres aspects de la grille d'analyse (cadre et propriétés) est discutée en annexe A.1, page 185.

En conclusion, une hiérarchie des modèles a été établie, depuis les plus primaires, car de simple assistance à un utilisateur, vers les plus riches, implémentables dans des agents logiciels autonomes. En outre, le processus d'évaluation a été pointé comme étant le processus critique pour une automatisation complète du contrôle social.

Synthèse de l'état de l'art

Dans cette partie, nous avons étudié les différentes composantes du contrôle des interactions des agents. Nous avons tout d'abord vu comment modéliser les interactions à l'aide d'engagements sociaux. Nous avons ensuite vu comment définir l'acceptabilité des interactions à l'aide de normes. Enfin la présentation des processus d'évaluation, par lesquels il est possible de vérifier si une interaction respecte ou non les normes, nous a permis de conclure la première phase du contrôle qui consiste à caractériser les interactions. Nous avons alors étudié les notions de confiance et de réputation et leurs modèles computationnels dans le but de mener à bien la deuxième phase du contrôle, qui consiste en la sanction des interactions.

Nous analysons ci-dessous les limites, dans le cadre des systèmes multi-agents ouverts et décentralisés, des modèles que nous avons étudiés. Les modèles que nous proposons dans la partie suivante tentent de pallier ces limites.

Modèles d'engagement social

La plupart des modèles actuels d'engagement social sont fortement centralisés : les engagements sociaux sont modélisés par la plate-forme, de façon transparente pour les agents. Dans le cadre du contrôle social, il est nécessaire de revenir à la proposition originelle de [Sin91] qui suggérait que les agents puissent modéliser eux-mêmes explicitement les engagements qu'ils perçoivent. Dans un tel cadre, les modèles centralisés sont difficilement utilisables. En effet, les agents peuvent avoir des représentations locales des engagements sociaux partielles, imparfaites et qui diffèrent d'un agent à l'autre.

Par ailleurs, les modèles qui s'intéressent à l'argumentation introduisent des liens entre les engagements sociaux dont la gestion reste un problème ouvert. Ces liens ne sont pas indispensables à la définition de la simple sémantique des interactions, nous écartons donc les modèles d'engagement social

avec cycle de vie du contenu.

Enfin, les sanctions doivent être appliquées en fonction de l'acceptabilité des interactions. Or, ce sont les normes qui définissent cette acceptabilité. En conséquence, nous considérons que les sanctions doivent être attachées aux normes et non aux engagements sociaux. Les modèles d'engagement social avec sanctions sont donc écartés.

Modèles de norme

Afin de garantir l'ouverture d'un système multi-agent, les hypothèses posées sur l'implémentation interne des agents doivent être minimales. De ce fait, il n'est pas possible de garantir que les agents ont été implémentés de manière enrégimentée ou qu'un contrôle interne les régit. Il est donc nécessaire de prévoir (1) que les agents puissent violer les normes et (2) qu'un contrôle *externe*, en ligne et dynamique soit mis en place pour remédier aux éventuelles violations.

Les formalismes descriptifs semblent plus adaptés à la mise en place d'un tel contrôle du fait (i) de leur souplesse et de leur plus grande richesse d'expression (ii) du caractère indécidable des logiques déontiques.

Processus d'évaluation

Tous les processus d'évaluation que nous avons étudiés sont indécidables, centralisés et / ou nécessitent de se dérouler hors-ligne. Ils ne sont donc pas adaptés au contrôle social des Systèmes Multi-Agents Ouverts et Décentralisés.

Sanctions

Les sanctions de type matériel requièrent que l'agent qui est sanctionné donne ou fasse quelque-chose pour se faire pardonner. Pour être sûr qu'elles soient appliquées, elles nécessitent, en dernier recours, qu'une autorité qui en a la pouvoir puisse contraindre l'agent sanctionné à faire ce qu'il doit. Par exemple, ce sont généralement des institutions fiscales centralisées qui appliquent des sanctions pécuniaires, car elles disposent de relations particulières avec les banques leur permettant d'assurer que l'argent leur parviendra.

Les sanctions sociales, telles que les notions de confiance et de réputation reposent sur un principe différent. C'est l'agent qui sanctionne qui augmente ou diminue le niveau de la réputation ou de la confiance qu'il associe à l'agent sanctionné. Il peut ensuite décider en s'appuyant, sur ces niveaux, du comportement à avoir vis-à-vis de l'agent sanctionné. L'agent qui applique la sanction conserve donc un contrôle total sur celle-ci et l'agent qui la subit peut difficilement y échapper, même si l'agent qui punit ne dispose d'aucun pouvoir explicite sur lui.

Pour leur part, les sanctions de type psychologiques sont difficiles à appliquer dans le cadre d'agents logiciels, puisque les états mentaux émotifs de ces derniers sont très limités, voire absents. De plus, il est très difficile de vérifier ou de garantir leur application.

Les sanctions sociales sont donc les plus adaptées dans le cas de systèmes décentralisés composés d'agents logiciels. Nous nous sommes donc intéressés plus particulièrement aux concepts de confiance et de réputation. Étant donné la confusion qui règne dans les définitions de ces concepts, nous en avons mené une étude en profondeur. Nous avons pu restreindre nos considérations aux seules réputation fondée sur les interactions et en donner une spécification précise. Cette spécification a ensuite été traduite en une grille d'analyse des modèles computationnels de réputations, grâce à laquelle nous avons pu catégoriser les modèles. La conclusion de l'analyse des différentes catégories de modèles est la suivante : (i) peu de modèles proposent l'implémentation de l'intégralité des processus et (ii) le processus d'évaluation semble le plus difficile à automatiser : dans les modèles actuels, il repose sur une déclaration *a priori* et *explicite* du comportement que compte avoir la cible.

Deuxième partie

Modèle L.I.A.R.

Introduction au modèle

Dans cette partie, nous décrivons le modèle de contrôle social des interactions L.I.A.R. (pour « Liar Identification for Agent Reputation »), répondant aux besoins identifiés dans la partie précédente. Il s'agit d'un modèle adapté aux systèmes multi-agents ouverts et décentralisés visant à réguler les interactions communicatives. Son architecture générale est décrite dans la figure 4.1 (les ovales représentent des données et les rectangles des processus. Cette architecture précise la boucle de rétroaction (les flèches plus marquées dans la figure 4.1) caractérisant le contrôle social des interactions, en intégrant les engagements sociaux et les normes pour évaluer les réputations et sanctionner les menteurs.

Les interactions sont observées et représentées sous forme d'engagements sociaux. La conformité de ces derniers par rapport à des normes est mesurée et permet d'obtenir des évaluations des comportements observés, sous la forme de politiques sociales. Ces politiques sociales sont utilisées dans un processus de punition qui peut amener à baisser ou augmenter les niveaux des réputations des agents observés. Les niveaux des réputations sont ensuite utilisés pour raisonner et aboutir à une intention de faire (ou non) confiance à un autre agent, en fonction d'un certain contexte. Un processus de décision réalise cette intention de faire (ou non) confiance par un changement des états mentaux de l'agent. Les nouveaux états mentaux de l'agent peuvent finalement amener celui-ci à générer (ou à empêcher) de nouvelles interactions, afin de sanctionner positivement ou négativement les autres agents. D'autre part, des recommandations, qui ne sont pas nécessairement sincères, peuvent aussi être échangées entre agents. Ces recommandations sont des interactions particulières, prenant la forme de messages, et sont donc aussi transformées en engagements sociaux par le processus d'observation. Elles sont filtrées pour déterminer un ensemble de recommandations de confiance qui peut servir lors du processus de sanction.

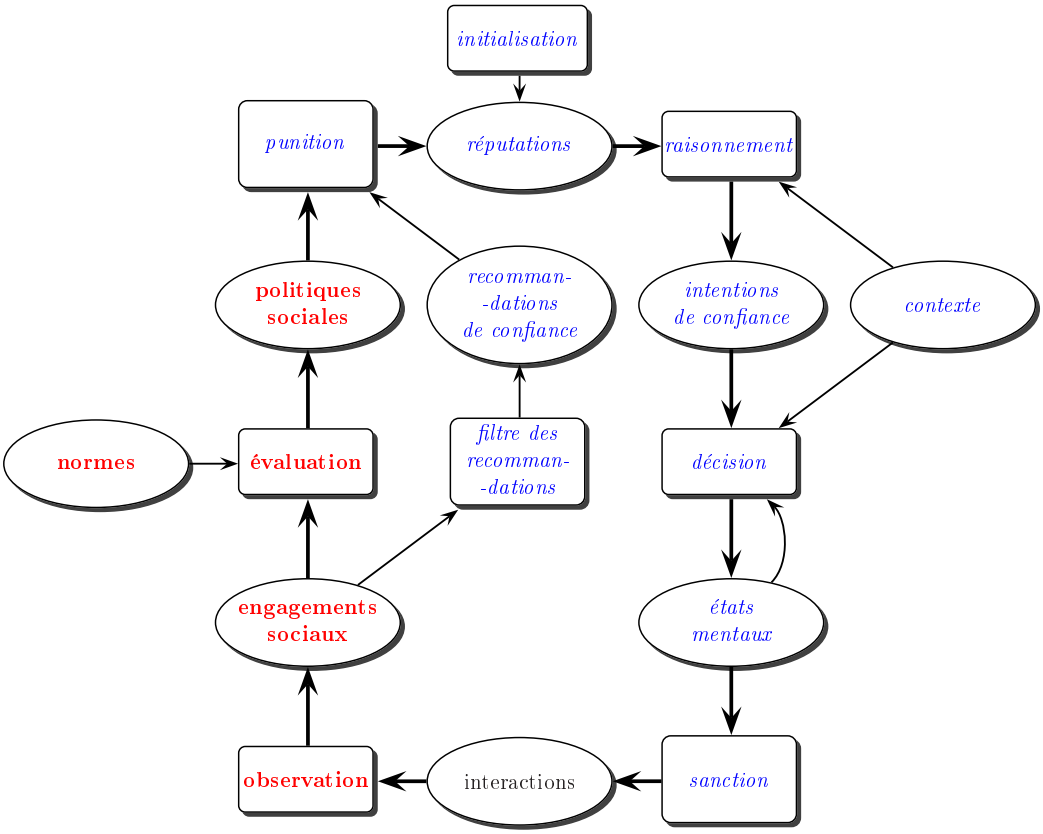


FIG. 4.1 – Architecture générale du modèle L.I.A.R.

Dans le chapitre 5, nous décrivons la partie gauche de cette architecture (en gras et rouge sur la figure 4.1). Nous y proposons un modèle d'engagement social qui permet aux agents de modéliser les interactions qu'ils observent, ainsi qu'un modèle de normes et un modèle de politiques sociales qui permettent de caractériser l'acceptabilité des interactions d'un point de vue social. Un processus d'évaluation est finalement proposé, sous la forme d'une détection de la violation des normes.

Dans le chapitre 6, nous nous intéressons à la partie droite de l'architecture (en italique et bleu sur la figure 4.1). Nous proposons un modèle de réputation s'appuyant sur les politiques sociales et permettant la prise de décisions.

En intégrant l'ensemble de ces modèles, nous disposons alors d'un contrôle social des interactions entièrement automatisé : les interactions observées sont sanctionnées par l'établissement d'un niveau de réputation reflétant le comportement d'un agent et ce niveau de réputation influe sur la décision d'agir ou non en confiance.

Chapitre 5

Modèles d'engagement social et de norme

Dans ce chapitre, nous posons les fondements du modèle L.I.A.R., pour le contrôle social des interactions. Tout d'abord, nous présentons un modèle d'engagement social, qui permet aux agents de représenter les interactions qu'ils observent dans le système, soit en interagissant eux-mêmes, soit par écoute flottante. Nous présentons ensuite un modèle de norme qui permet aux agents de définir les comportements acceptables ou non. Enfin, nous définissons un processus de détection de la violation de ces normes, qui détermine si les comportements observés sont respecteux des normes en vigueur et de caractériser leur conformité à ces normes à l'aide de politiques sociales. Dotés d'un tel dispositif, les agents disposent alors d'un historique d'évaluations des comportements qu'ils ont observés et sont en mesure de sanctionner leurs pairs. Le mécanisme de sanction est décrit dans le chapitre 6.

5.1 Modèle d'engagement social

Cette section présente un modèle d'engagement social qui permet la modélisation des interactions communicatives des agents. Ce modèle d'engagement social peut être utilisé par n'importe quel agent pour représenter les interactions qu'il observe, afin de détecter les éventuelles violations de normes. Pour cela, il est nécessaire de disposer d'un modèle posant le moins d'hypothèses possibles sur l'implémentation interne des agents, de façon à conserver au maximum l'ouverture du système.

5.1.1 Formulation

Dans cette section, nous présentons un modèle d'engagement social (de type « conditionnel » [FC02]) qui permet, dans le cadre de la détection décentralisée de mensonges, de modéliser des communications de manière externe aux agents.

Définition

Soit $\Omega(\mathtt{t})$ l'ensemble des agents présents dans le système à l'instant $\mathtt{t}$, $\mathtt{ob} \in \Omega(\mathtt{t})$ l'agent qui modélise l'engagement social, $\Omega_{\mathtt{ob}}(\mathtt{t}) \subseteq \Omega(\mathtt{t})$ l'ensemble des agents connus de l'agent $\mathtt{ob}$ à l'instant $\mathtt{t}$, $\mathcal{T}$ l'ensemble des instants, $\mathcal{P}$ l'ensemble des formules bien formées du calcul des prédicats et $\mathcal{E}_{sc} \stackrel{\text{def}}{=} \{$ `inactive`, `active`, `fulfilled`, `violated`, `cancelled` $\}$ l'ensemble des états dans lesquels peut se trouver un engagement social. Un engagement social peut alors être défini selon la définition 5.1.1 suivante :

Définition 5.1.1 *Un* ***engagement social*** *(*`SCom` *: « Social Commitment ») est modélisé comme suit :*

$$_{\mathtt{ob}}\mathtt{SCom}(\mathtt{db}, \mathtt{cr}, \mathtt{t_e}, \mathtt{st}, [\mathtt{cond},]\mathtt{cont})$$

Où :

- $\mathtt{ob} \in \Omega(\mathtt{t})$ est l'observateur de l'engagement social : l'agent qui construit l'objet représentant l'interaction observée.
- $\mathtt{db} \in \Omega_{\mathtt{ob}}(\mathtt{t})$ est le débiteur : l'agent qui est engagé.
- $\mathtt{cr} \in \Omega_{\mathtt{ob}}(\mathtt{t})$ est le créditeur : l'agent envers qui le débiteur est engagé.
- $\mathtt{t_e} \in \mathcal{T}$ est le moment d'émission : l'instant où l'engagement social a été créé.
- $\mathtt{st} \in \mathcal{E}_{sc}$ est l'état de l'engagement social.
- $\mathtt{cond} \in \mathcal{P}$ sont les conditions d'activation de l'engagement social. Dans le cadre de cette thèse, il s'agit d'une formule de la logique des prédicats. Ce champ est optionnel ; son omission correspond à une condition toujours vraie.
- $\mathtt{cont} \in \mathcal{P}$ représente le contenu de l'engagement social. Dans le cadre de cette thèse, il s'agit d'une formule de la logique des prédicats.

Dans la suite, nous nous autorisons à accéder aux différents attributs d'un engagement social à l'aide d'une syntaxe inspirée des langages à objets.

Par exemple, si **e** est un engagement social, alors **e**.db est le débiteur de l'engagement **e**.

Exemple

L'exemple 5.1.1, ci-dessous, illustre la modélisation d'un engagement social à l'aide du formalisme présenté précédemment. Cet engagement représente l'observation, par un agent ob du fait que l'agent a s'engage, à 13 :00, envers l'agent b sur le fait que le cinéma theater1 joue le film Shrek à 19 :00 en salle room1, avec une condition toujours vraie.

Exemple 5.1.1

$$_{\mathtt{ob}}\mathtt{SCom}(\mathtt{a},\mathtt{b},\mathtt{13:00},\mathtt{inactive},\mathtt{shows}(\mathtt{theater1},\mathtt{Shrek},\mathtt{19:00},\mathtt{room1}))$$

Cycle de vie

Le cycle de vie décrit l'ensemble des transitions possibles entre les états d'un engagement social. Le cycle de vie de l'engagement social est représenté par le diagramme d'états UML [OMG03] de la figure 5.1. Les états du diagramme représentent les états de l'engagement social et les transitions représentent les actions menées sur l'engagement social. Il est possible de différencier les transitions qui ont lieu suite à des actions perpétrées par des agents sur leurs engagements sociaux (has_been_created, has_been_cancelled) et les actions liées à des événements extérieurs (condition_met, content_fulfilled et content_violated).

Ce cycle de vie se déroule typiquement comme suit :

- Un engagement social est toujours créé dans l'état inactive (prédicat has_been_created vérifié).
- Dès que la condition d'activation cond est vérifiée, l'engagement social devient actif (état active, prédicat condition_met vérifié). Alors seulement, la satisfaction, la violation ou l'annulation de l'engagement peut avoir lieu.
- L'engagement social peut être annulé par le débiteur ou le créditeur (état cancelled, prédicat has_been_cancelled vérifié).
- L'engagement social peut passer en état fulfilled si le débiteur remplit son engagement (prédicat content_fulfilled vérifié).

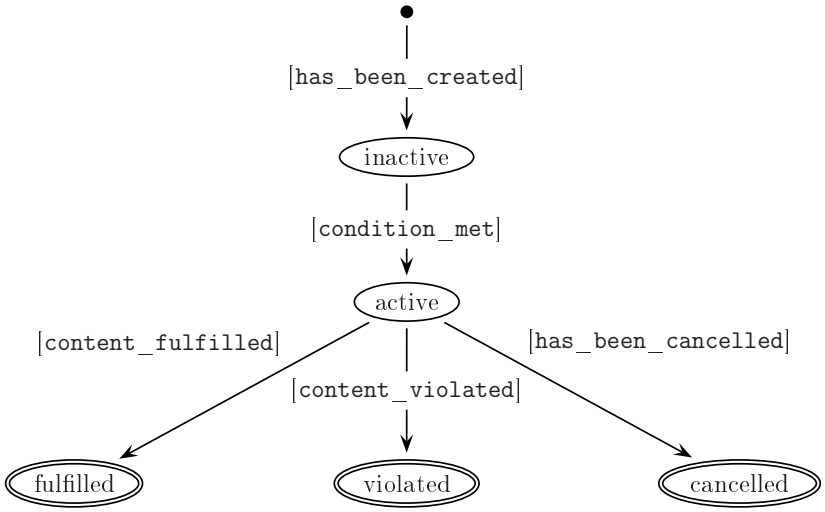


FIG. 5.1 – Cycle de vie d'un engagement social.

– L'engagement social peut passer en état `violated` si le débiteur ne remplit pas son engagement (prédicat `content_violated` vérifié).

Fonctions de manipulation

Un agent `ob` peut manipuler les engagements sociaux qu'il modélise à l'aide des fonctions ci-dessous :

– `ob.inconsistent_content` : $\mathcal{T} \times \mathfrak{P}(\mathcal{P}) \mapsto \{\texttt{true}, \texttt{false}\}$ renvoie `true` si l'ensemble des n contenus d'engagements sociaux passés en argument constitue un ensemble inconsistant à l'instant `t` $\in \mathcal{T}$, `false` sinon. $\mathfrak{P}(\mathcal{P})$ désigne l'ensemble des parties de $\mathcal{P}$.

– `ob.facets` : $\mathcal{P} \mapsto \mathfrak{P}(\mathcal{F})$ prend en argument un contenu d'engagement social ($\in \mathcal{P}$) et renvoie un ensemble de facettes qui lui sont associées. $\mathcal{F}$ est l'ensemble des facettes qu'il est possible de juger chez un agent. Les facettes qu'un contenu d'engagement social permet de juger peuvent être déduites par l'agent `ob` en fonction des thèmes sur lesquels porte le contenu de l'engagement social.

L'exemple 5.1.2 illustre un retour possible de la fonction `inconsistent_content` chez un agent `ob` donné : un cinéma ne pouvant pas jouer deux films différents dans une même salle à une même heure, les contenus `shows(theater1, Shrek, 19 :00, room1)` et `shows(theater1, Bambi,`

19 :00, room1) sont inconsistants.

Exemple 5.1.2

$$\begin{aligned}&\texttt{ob.inconsistent\_content}(\texttt{10:00},\\ &\texttt{shows}(\texttt{theater1}, \texttt{Shrek}, \texttt{19:00}, \texttt{room1}),\\ &\texttt{shows}(\texttt{theater1}, \texttt{Bambi}, \texttt{19:00}, \texttt{room1}))\\ &= \texttt{true}\end{aligned}$$

L'exemple 5.1.3 illustre un retour possible de la fonction facets chez le même agent ob : le contenu shows(theater1, Shrek, 19 :00, room1) est lié à la capacité d'un agent à fournir de l'information sur les horaires de cinéma.

Exemple 5.1.3

$$\begin{aligned}&\texttt{ob.facets}(\texttt{shows}(\texttt{theater1}, \texttt{Shrek}, \texttt{19:00}, \texttt{room1}))\\ &= \{\textit{"theater showtimes"}\}\end{aligned}$$

Historiques

Tout au long de son existence dans le système, un agent observe des interactions et les modélise sous forme d'engagements sociaux. Ces engagements sont stockés dans des historiques d'engagements sociaux. La définition 5.1.2 est celle d'historiques d'engagements sociaux regroupés en fonction de leur débiteur et de leur créditeur.

Définition 5.1.2 *L'historique d'engagements sociaux de l'agent* db *envers l'agent* cr *tel que se le représente l'agent* ob *à l'instant* t *est noté* ${}_{\texttt{ob}}\mathbf{CCS}^{\texttt{cr}}_{\texttt{db}}(\texttt{t})$ *(***CCS** *pour « Communicative Commitment Store ») :*

$${}_{\texttt{ob}}\mathbf{CCS}^{\texttt{cr}}_{\texttt{db}}(\texttt{t}) \stackrel{\text{def}}{=} \{\ {}_{\texttt{ob}}\texttt{SCom}(\texttt{db}, \texttt{cr}, \texttt{t}', \texttt{st}, \texttt{cond}, \texttt{cont})/\texttt{t}' \leq \texttt{t}\ \}$$

Où $\texttt{ob} \in \Omega(\texttt{t})$, $\texttt{db} \in \Omega_{\texttt{ob}}(\texttt{t})$, $\texttt{cr} \in \Omega_{\texttt{ob}}(\texttt{t})$, $\texttt{t} \in \mathcal{T}$, $\texttt{t}' \in \mathcal{T}$, $\texttt{st} \in \mathcal{E}_{sc}$, $\texttt{cond} \in \mathcal{P}$ *et* $\texttt{cont} \in \mathcal{P}$.

L'historique d'engagements sociaux de l'agent db envers l'agent cr tel que se le représente l'agent ob à l'instant t regroupe l'ensemble des engagements

sociaux qu'a pris l'agent db envers l'agent cr avant ou à l'instant t, tels que les perçoit l'agent ob.

Plus généralement, l'ensemble des engagements sociaux connus par un agent ob peut être regroupé dans un historique unique, comme le montre la définition 5.1.3.

Définition 5.1.3 *L'ensemble des engagements sociaux perçus par l'agent* ob *avant ou à l'instant* t *est regroupé dans son* ***historique général d'engagements sociaux*** *noté :*

$$ {}_{\mathtt{ob}}\mathbf{CCS}(\mathtt{t}) \stackrel{\text{def}}{=} \bigcup_{\substack{\mathtt{x} \in \Omega_{\mathtt{ob}}(\mathtt{t}),\\ \mathtt{y} \in \Omega_{\mathtt{ob}}(\mathtt{t})}} {}_{\mathtt{ob}}\mathbf{CCS}^{\mathtt{y}}_{\mathtt{x}}(\mathtt{t}) $$

5.1.2 Inconsistance d'engagements sociaux

La définition 5.1.4 suivante spécifie l'inconsistance d'un engagement social avec un ensemble d'autres engagements sociaux :

Définition 5.1.4 *Du point de vue d'un observateur* ob, ***un engagement social*** $\mathbf{c}$ ***crée une inconsistance avec un ensemble d'engagements sociaux*** $\mathcal{A}$ *s'il existe des engagements* $\{\mathbf{c_1}, \ldots, \mathbf{c_n}\}$ *dans* $\mathcal{A}$ *qui sont dans un état « positif »* (active *ou* fulfilled) *et dont les contenus sont inconsistants avec celui de* $\mathbf{c}$ *:*

$$ \mathtt{ob.inconsistent}(\mathtt{t}, \mathbf{c}, \mathcal{A}) \stackrel{\text{def}}{=} $$

$$ \mathbf{c} \in {}_{\mathtt{ob}}\mathbf{CCS}(\mathtt{t}) \wedge \mathbf{c}.\mathtt{st} \in \{\mathtt{active}, \mathtt{fulfilled}\} \wedge \forall \mathbf{c_i} \in \mathcal{A}, \mathbf{c_i} \in {}_{\mathtt{ob}}\mathbf{CCS}(\mathtt{t}) \wedge $$

$$ \exists \{\mathbf{c_1}, \ldots, \mathbf{c_n}\} \subseteq \mathcal{A} / $$

$$ \forall \mathbf{c_i} \in \{\mathbf{c_1}, \ldots, \mathbf{c_n}\}, \mathbf{c_i}.\mathtt{st} \in \{\mathtt{active}, \mathtt{fulfilled}\} \wedge $$

$$ \mathtt{ob.inconsistent\_content}(\mathtt{t}, \mathbf{c}.\mathtt{cont}, \mathbf{c_1}.\mathtt{cont}, \ldots, \mathbf{c_n}.\mathtt{cont}) $$

Il est à noter que l'état d'un engagement social prend en compte la condition d'activation. Il n'est donc pas nécessaire de faire apparaître cette condition dans la formule de la définition 5.1.4.

Exemple

L'exemple 5.1.4 suivant propose deux engagements sociaux. S'il est possible pour l'agent ob, qui modélise ces deux engagements, d'établir localement que les contenus shows(theater1, Shrek, 19 :00, salle1) et shows(theater1, Bambi, 19 :00, salle1) sont inconsistants alors le premier engagement crée une inconsistance avec tout ensemble comprenant le deuxième (et *vice-versa*). En effet, ces deux engagements sociaux sont état active et de contenus inconsistants. L'agent ob pourrait, par exemple, considérer que les contenus sont inconsistants du fait qu'ils portent sur une même facette (ici, l'aptitude à fournir des horaires de cinéma) et du fait qu'un cinéma ne peut pas jouer deux films différents à la même heure dans la même salle.

Exemple 5.1.4 *Soit un premier engagement social, de l'agent* a *vers l'agent* b *sur le fait que le cinéma* theater1 *joue le film* Shrek *à* 19 :00 *en salle* room1*, perçu à* 13 :00 *par l'agent* ob*, étant actuellement actif :*

$$_{\mathtt{ob}}\mathtt{SCom}(\mathtt{a}, \mathtt{b}, 13:00, \mathtt{active}, \mathtt{shows}(\mathtt{theater1}, \mathtt{Shrek}, 19:00, \mathtt{room1}))$$

Soit un deuxième engagement social de l'agent c *vers l'agent* d *sur le fait que le cinéma* theater1 *joue le film* Bambi *à* 19 :00 *en salle* room1*, perçu à* 18 :00 *par l'agent o, étant actuellement actif :*

$$_{\mathtt{ob}}\mathtt{SCom}(\mathtt{c}, \mathtt{d}, 18:00, \mathtt{active}, \mathtt{shows}(\mathtt{theater1}, \mathtt{Bambi}, 19:00, \mathtt{room1}))$$

L'agent ob *peut considérer que premier engagement crée une inconsistance avec tout ensemble comprenant le deuxième et vice-versa.*

5.1.3 Propriétés

Dans cette section, nous étudions quelques propriétés particulières du modèle d'engagement social que nous proposons. Ces propriétés découlent directement du caractère décentralisé des systèmes que nous considérons. Elles portent sur les historiques et les cycles de vie des engagements sociaux.

Historiques locaux

Du fait du caractère fortement décentralisé des systèmes que nous considérons, il n'existe pas de lieu commun où les agents pourraient partager une

représentation commune de l'ensemble des engagements sociaux pris dans le système. Chaque agent doit donc modéliser **localement** les engagements qu'il perçoit dans ses propres historiques d'engagements sociaux.

Chaque agent n'a alors accès qu'aux engagements sociaux contenus dans ses propres historiques d'engagements sociaux , ce qui ne constitue qu'un sous-ensemble des engagements sociaux qui ont été pris dans le système. En tant que concepteurs, notre point de vue sur le système est omniscient. La notion d'observateur, en notation préfixée et indicée, que nous avons ajoutée au modèle d'engagement social, permet de limiter nos considérations à l'ensemble des engagements considérés par un agent donné, tout en conservant notre point de vue omniscient sur le système.

Cycles de vie

Le modèle d'engagement social présenté dans cette section permet de représenter le fait que chaque agent possède ses propres historiques d'engagements sociaux. Or, les agents n'ont qu'une perception incomplète et souvent imparfaite du système dans lequel ils évoluent. De ce fait, les différentes représentations locales peuvent être globalement incohérentes. Par exemple, deux agents peuvent modéliser un même engagement social dans deux états différents. Dans cette section, nous montrons en quoi ceci implique que les cycles de vies des représentations locales des engagements sociaux ne sont pas nécessairement respectés.

Le scénario suivant illustre une situation qui peut s'avérer très courante dans les réseaux sans fils ou les réseaux pair-à-pairs auxquels nous nous intéressons dans le cadre de cette thèse.

Au cours d'un même dialogue, un débiteur `db` peut être amené à prendre un engagement social puis à l'annuler. Par exemple, le débiteur `db` informe le créditeur `cr` de la valeur de vérité d'un certain fait `p`. Si un observateur extérieur `ob` perçoit ce premier message, il peut reconstruire l'engagement social associé à cette communication (voir figure 5.2). Si, suite à un dysfonctionnement de ses capteurs ou du réseau de communication, l'observateur n'est plus en mesure de capter le second message, qui annule le premier, sa représentation locale de l'engagement social divergera non seulement de la réalité, mais aussi du point de vue des deux autres agents (figure 5.3).

L'agent `ob` pourrait alors être amené à croire que l'engagement a été violé si le débiteur ne remplit pas le contenu. Or, ce dernier n'est plus tenu de faire, puisqu'il a annulé son engagement.

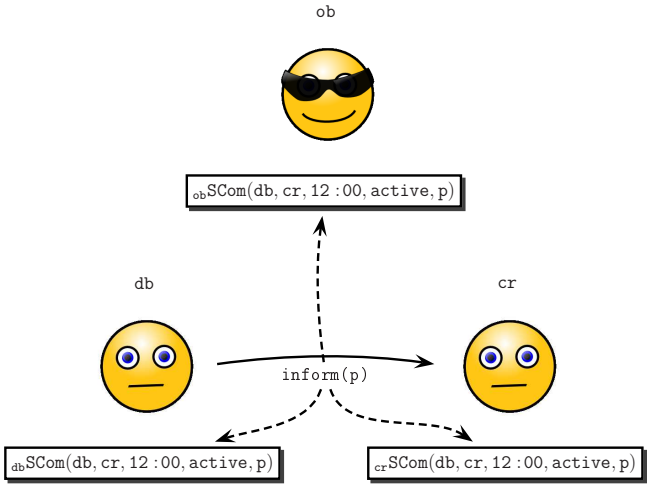


FIG. 5.2 – Prise d'un engagement social avec observation extérieure.

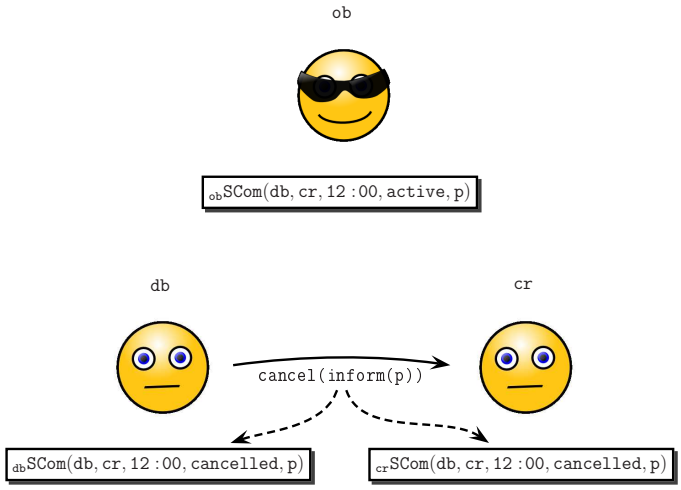


FIG. 5.3 – Annulation d'un engagement sans observation extérieure.

L'agent `ob` peut aussi s'aperçevoir de son erreur, par exemple, si un autre agent lui fournit une copie du message d'annulation. Dès qu'il se rendra compte que l'engagement social avait en fait été annulé, l'agent `ob` aura intérêt à changer l'état de sa représentation locale de cet engagement social pour refléter sa nouvelle perception du monde. Or, il n'existe pas de transition de l'état `violated` vers l'état `cancelled` dans le cycle de vie d'un engagement social décrit en section 5.1.1, page 81.

En conclusion, le cycle de vie décrit l'évolution d'un engagement social réel. Cependant,les transitions de ce cycle de vie ne sont pas nécessairement respectées dans les représentations locales des engagements sociaux, du fait de la perception incomplète et souvent imparfaite qu'ont les agents du système dans lequel ils évoluent.

Dimension temporelle

De la même manière que pour les engagements sociaux, nous considérons des historiques de politiques sociales. Nous avons donc, là encore, ajouté une dimension temporelle pour obtenir des *historiques* de politiques sociales. Ainsi, un agent est capable de retrouver l'état qu'avait une politique sociale donnée à tout instant du passé où cette politique existait. Cette distinction est importante dans le cadre des calculs des niveaux des réputations car elle permet d'avoir accès aux niveaux des réputations d'un agent à tout instant passé (voir chapitre 6, page 105).

5.2 Modèle de normes

Dans cette section, nous proposons un modèle de norme permettant de décrire les comportements d'un agent acceptables ou non. Afin de laisser la possibilité à n'importe quel agent d'entrer dans le système à n'importe quel instant et d'assurer ainsi l'ouverture du système, il est nécessaire de faire le moins possible d'hypothèses sur la constitution interne des agents. En conséquence, il n'est pas possible de garantir l'enrégimentation des agents. De plus, du fait du caractère décentralisé des systèmes que nous considérons, il n'est pas non plus possible d'assurer qu'une institution centralisée unanimement reconnue aura le pouvoir et les moyens de sanctionner tous les agents du système. Nous nous intéressons donc aux normes sociales, plus particulièrement aux s-normes. Nous proposons ici un modèle descriptif de normes permet-

tant à chaque agent de détecter quand les autres agents les violent (contrôle externe).

5.2.1 Normes

Dans cette section, nous présentons un modèle descriptif de norme qui permet de définir les comportements acceptables ou non.

Définition

Les normes (ici de type s-normes) sont créées par un groupe d'agents, les législateurs, pour réguler un certain groupe d'agents, les cibles. N'importe quel agent ayant connaissance de ces normes peut en détecter la violation, c'est-à-dire exercer le pouvoir judiciaire. Ces agents qui détectent les violations sont nommés évaluateurs (ce découpage est inspiré du principe de séparation des pouvoirs [Fra]). Les violations peuvent être punies par des agents exerçant le pouvoir exécutif, les punisseurs. La définition 5.2.1 suivante est celle d'une norme.

Définition 5.2.1 *Une **norme** est modélisée comme suit :*

$$\texttt{snorm}(\texttt{op}, \texttt{Tg}, \texttt{Ev}, \texttt{Pu}, \texttt{cond}, \texttt{cont}, \texttt{st})$$

Où :

- $\texttt{op} \in \{ \texttt{I}, \texttt{O}, \texttt{P} \}$ représente l'opérateur déontique caractérisant la norme : `I` représente l'interdiction, `O` représente l'obligation et `P` représente la permission.
- $\texttt{Tg} \subseteq \Omega(\texttt{t})$ représente la ou les entités soumises à la norme (cibles). `t` est l'instant où l'on considère la norme ;
- $\texttt{Ev} \subseteq \Omega(\texttt{t})$ représente la ou les entités qui exercent le pouvoir judiciaire, *i.e.* qui décident quand la norme est violée (évaluateurs).
- $\texttt{Pu} \subseteq \Omega(\texttt{t})$ représente la ou les entités qui exercent le pouvoir exécutif, *i.e.* qui appliquent les pénalités (punisseurs).
- $\texttt{cond} \in \mathcal{P}$ sont les conditions de validité de la norme. Il s'agit d'une formule de la logique des prédicats ;
- $\texttt{cont} \in \mathcal{P}$ est le contenu sur lequel porte la norme. Il s'agit d'une formule de la logique des prédicats ;

– $\mathtt{st} \in \mathcal{E}_{sn}$ représente l'état de la norme. Une norme peut être dans un état parmi l'ensemble : $\mathcal{E}_{sn} \stackrel{\text{def}}{=} \{ \mathtt{inactive}, \mathtt{active} \}$.

L'ensemble des normes connues par un agent $\mathtt{a}$ à un instant $\mathtt{t}$ est noté : ${}_{\mathtt{a}}\mathcal{N}(\mathtt{t})$.

Exemple

L'exemple 5.2.1 ci-dessous présente une norme interdisant aux agents de parler de contournement de mesures techniques de protection de contenu (DRM)[1].

Exemple 5.2.1

$$\mathtt{snorm}(\mathtt{I}, G(\mathtt{t}), \Omega(\mathtt{t}), \Omega(\mathtt{t}), \mathtt{true},$$
$$\forall \mathtt{x} \in G(\mathtt{t}), \mathtt{talk\_about\_DRM\_skirting}(\mathtt{x}), \mathtt{active})$$

Où $\mathtt{talk\_about\_DRM\_skirting}(\mathtt{x})$ est vrai si l'agent $\mathtt{x}$ parle de contournement de DRM.

Instanciation

De manière à pouvoir étudier la conformité du comportement d'un agent envers une norme à un instant donné, nous proposons, dans cette thèse, d'instancier les normes par des politiques sociales. Ces politiques sociales représentent alors l'engagement des différentes cibles à respecter les normes en vigueur dans le groupe auquel elles appartiennent.

Cette phase d'instanciation permet de :

– **spécifier le contenu d'une norme dans le référentiel de chacun des évaluateurs :** Comme le montrent le formalisme de la définition 5.2.1 et l'exemple 5.2.1, le contenu d'une norme est formulé comme une règle générique, avec un point de vue omniscient sur le système. Il est nécessaire que chaque évaluateur exprime le contenu dont il va devoir détecter la violation selon son propre point de vue sur le monde.

– **gérer la violabilité d'une même norme par différentes cibles :** la règle décrite par la norme peut, à un instant donné, être violée par plusieurs cibles. En instanciant chaque norme par plusieurs politiques

[1] Le sujet de cette norme est tiré d'une loi sur le droit d'auteur (DADVSI) en cours de discussion au parlement français.

sociales, chacune dirigée envers une seule cible, un évaluateur peut détecter des violations simultanées.
- **gérer la violabilité d'une même norme par une cible donnée, à de multiples reprises :** un même agent peut violer plusieurs fois la même règle. En re-instanciant une politique sociale à chaque fois qu'elle est violée (et tant que la norme correspondante est active), un évaluateur peut détecter de multiples violations d'une règle par un même agent. Cela permet, en outre, de garder des traces de chacune des violations qui ont eu lieu.
- **faire apparaître des pénalités propres à chaque punisseur :** le formalisme de norme ne spécifie pas de pénalité car c'est à chaque punisseur de décider des pénalités qu'il souhaite appliquer à une norme donnée. Deux punisseurs pouvant associer des pénalités différentes à une même norme. Il est donc nécessaire de différencier la description de la règle (la norme) qui ne contient pas de pénalité, de l'engagement à respecter la règle (la politique sociale) qui est, lui, associé à des pénalités.

5.2.2 Politiques sociales

Dans cette section, nous définissons les politiques sociales qui permettent aux évaluateurs d'instancier les normes.

Définition

Soit $\texttt{ev} \in \Omega(\texttt{t})$ l'agent qui instancie la politique sociale et $\mathcal{E}_{sp} \stackrel{\text{def}}{=} \{$ `inactive`, `active`, `justifying`, `violated`, `fulfilled`, `cancelled` $\}$ l'ensemble des états dans lesquels peut se trouver une politique sociale. Une politique sociale peut alors être définie selon la définition 5.2.2 suivante :

Définition 5.2.2 *Une **politique sociale** (*`SPol` *: « Social Policy ») est modélisée comme suit :*

$$_{\texttt{ev}}\texttt{SPol}(\texttt{db}, \texttt{cr}, \texttt{t}_\texttt{e}, \texttt{st}, [\texttt{cond},]\texttt{cont})$$

Où :
- $\texttt{ev} \in \Omega(\texttt{t})$ est l'évaluateur qui instancie la norme et crée l'objet représentant la politique sociale.

- $\texttt{db} \in \Omega_{\texttt{ev}}(\texttt{t})$ est le débiteur : l'agent qui est engagé pour cette politique sociale.
- $\texttt{cr} \in \Omega_{\texttt{ev}}(\texttt{t})$ est le créditeur : l'agent envers qui le débiteur est engagé pour cette politique sociale.
- $\texttt{t}_\texttt{e} \in \mathcal{T}$ est le moment de création : l'instant où la politique sociale a été créée.
- $\texttt{st} \in \mathcal{E}_{sp}$ est l'état de la politique sociale.
- $\texttt{cond} \in \mathcal{P}$ sont les conditions d'activation de la politique sociale. Dans le cadre de cette thèse, il s'agit d'une formule de la logique des prédicats. Ce champ est optionnel ; son omission correspond à une condition toujours vraie.
- $\texttt{cont} \in \mathcal{P}$ représente le contenu de la politique sociale. Dans le cadre de cette thèse, il s'agit d'une formule de la logique des prédicats.

Une politique sociale vise à réguler les interactions. Le contenu d'une politique sociale fait donc référence à des engagements sociaux. Pour ce faire, nous proposons que les engagements sociaux référencés soient représentés par des prédicats ayant la forme donnée dans la définition 5.1.1, page 80.

Une politique sociale est également associée à des pénalités. Ces dernières sont choisies et appliquées par chacun des punisseurs spécifiés dans la norme correspondante. Il est donc nécessaire que tout agent $\texttt{pu} \in \texttt{Pu}$ dispose de la fonction suivante (où $\mathcal{S}$ représente l'ensemble des politiques sociales) :

- $\texttt{pu.punishes} : \mathcal{S} \times \mathcal{T} \mapsto [0, +1]$ qui associe à une politique sociale et un instant donné une pénalité. Cette pénalité, choisie dans $[0, +1]$, représente l'importance relative accordée par l'agent pu à la politique sociale passée en argument, relativement aux autres politiques sociales. Pour déterminer la pénalité, l'agent pu peut, entre autres, s'appuyer sur l'état de la politique sociale.

Exemple

La politique sociale présentée dans l'exemple 5.2.2, ci-dessous, illustre une instance de la norme visant à interdire de parler de mesures techniques de protection de contenu, donnée dans l'exemple 5.2.1, page 90. Cette politique sociale a été instanciée par l'agent ev pour la cible x à 19 :00.

Exemple 5.2.2 *Soit un agent* `k` *connu de l'agent* `ev` *:* $\mathtt{k} \in \Omega_{\mathtt{ev}}(\mathtt{t})$ *et* $\mathbf{e}$ *un engagement social de l'agent* `x` *envers l'agent* `k` *:* $\mathbf{e} \in {}_{\mathtt{ev}}\mathbf{CCS}^{\mathtt{k}}_{\mathtt{x}}(\mathtt{t})$*. La politique sociale suivante interdit que l'engagement* $\mathbf{e}$ *porte sur la facette* *"DRM skirting"* *:*

$$
\begin{aligned}
&{}_{\mathtt{ev}}\mathtt{SPol}(\mathtt{x}, \mathtt{ev}, 19:00, \mathtt{active}, \mathtt{x} \in G(\mathtt{t}) \wedge \mathtt{ev} \in \mathtt{Ev},\\
&\quad \neg(\mathtt{ev.facets}(\mathbf{e}.\mathtt{cont}) \supseteq \{\textit{"DRM skirting"}\}))
\end{aligned}
$$

Cycle de vie

Le cycle de vie décrit l'ensemble des transitions possibles entre les états d'une politique sociale. Le cycle de vie d'une politique sociale est représenté par le diagramme d'états UML [OMG03] de la figure 5.4. Les états du diagramme représentent les états de la politique sociale et les transitions représentent les évolutions de la politique sociale.

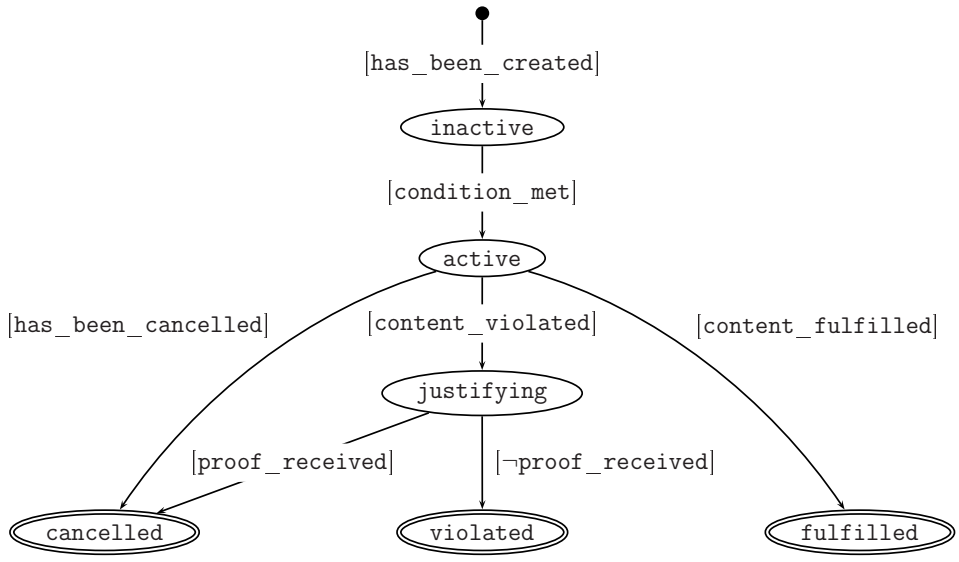


FIG. 5.4 – Cycle de vie d'une politique sociale.

Ce cycle de vie se déroule typiquement comme suit :

- Une politique sociale est toujours créée dans l'état `inactive` (prédicat `has_been_created` vérifié).

- Dès que la condition d'activation `cond` est vérifiée, la politique sociale devient active (état `active`, prédicat `condition_met` vérifié).
- La politique sociale peut être annulée (état `cancelled`, prédicat `has_been_cancelled` vérifié), par exemple si la norme qu'elle instancie a été inactivée.
- L'évaluateur peut considérer que le contenu de la politique sociale a été rempli (prédicat `content_fulfilled` vérifié) et faire passer la politique sociale en état `fulfilled`.
- L'évaluateur peut considérer que le contenu de la politique sociale n'a pas été respecté (prédicat `content_violated` vérifié) et faire passer la politique sociale en état `justifying`. Il lance alors un protocole de demande de justification afin de vérifier s'il y a effectivement eu violation, ou bien si ses perceptions locales sont erronées auquel cas, la violation n'a pas lieu d'être. Ce protocole est décrit section 5.3, page 98.
- Suivant l'aboutissement du protocole de demande de justification, l'évaluateur peut considérer que la politique sociale a effectivement été violée, si aucune preuve contredisant la violation n'est apportée (prédicat `proof_received` non vérifié) et faire passer la politique sociale en état `violated`. Dans le cas contraire, *i.e.* si une preuve est apportée que la violation n'a pas lieu d'être (prédicat `proof_received` vérifié), alors l'agent fait passer la politique sociale en état `cancelled`.

Fonctions de manipulation

Un agent évaluateur `ev` peut manipuler les politiques sociales qu'il modélise à l'aide des fonctions ci-dessous :

- `ev.facets` : $\mathcal{P} \mapsto \mathfrak{P}(\mathcal{F})$ prend en argument un contenu d'une politique sociale ($\in \mathcal{P}$) et renvoie l'ensemble de facettes que l'agent `ev` associe à ce contenu. $\mathcal{F}$ est l'ensemble des facettes qu'il est possible de juger chez un agent. Les facettes qu'un contenu de politique sociale permet de juger sont déduites par l'agent `ev` à partir des facettes des engagements sociaux référencés dans le contenu de la politique sociale. $\mathfrak{P}(\mathcal{P})$ désigne l'ensemble des parties de $\mathcal{P}$.
- `ev.dimensions` : $\mathcal{S} \mapsto \mathfrak{P}(\mathcal{D})$ prend en argument une politique sociale ($\in \mathcal{S}$) et renvoie un ensemble de dimensions que celle-ci permet de juger. $\mathcal{D}$ est l'ensemble des dimensions qu'il est possible de juger chez un agent. Dans le cadre de cette thèse, nous considérons l'ensemble de dimensions le plus complet, celui proposé par [MC01] : $\mathcal{D} \stackrel{\text{def}}{=} \{$

$\texttt{integrity}, \texttt{competence}, \texttt{benevolence}, \texttt{previsibility}\ \}$.

Les exemples 5.2.3 et 5.2.4 illustrent des retours possibles des fonctions `facets` et `dimensions` chez un agent `ev`.

Exemple 5.2.3 *Dans cet exemple, nous considérons la même politique sociale que dans l'exemple 5.2.2, page 93. Celle-ci est notée* **sp**. *La facette que l'agent* `ev` *associe à ce contenu est le contournement de mesure techniques de protections :*

$$\texttt{ev.facets}(\mathbf{sp}.\texttt{cont}) = \{\texttt{"DRM skirting"}\}$$

Exemple 5.2.4 *Dans cet exemple, nous considérons la politique sociale (notée* **sp***) de l'exemple 5.2.2, page 93. Un agent* `ev` *donné peut associer à cette politique sociale la dimension d'intégrité :*

$$\texttt{ev.dimensions}(\mathbf{sp}) = \{\texttt{integrity}\}$$

Historiques

Les politiques sociales sont stockées par les évaluateurs dans des historiques locaux, dont nous précisons les définitions dans cette section. La définition 5.2.3 est celle d'historiques de politiques sociales groupées en fonction de leur débiteur et de leur créditeur.

Définition 5.2.3 ***L'historique de politiques sociales de l'agent*** `db` ***envers l'agent*** `cr` ***tel que se le représente l'agent*** `ev` ***à l'instant*** `t` *est noté* ${}_{\texttt{ev}}\mathbf{NCS}^{\texttt{cr}}_{\texttt{db}}(\texttt{t})$ *(***NCS** *pour « Normative Commitment Store ») :*

$${}_{\texttt{ev}}\mathbf{NCS}^{\texttt{cr}}_{\texttt{db}}(\texttt{t}) \stackrel{\text{def}}{=} \{\ {}_{\texttt{ev}}\texttt{SPol}(\texttt{db}, \texttt{cr}, \texttt{t}', \texttt{st}, \texttt{cond}, \texttt{cont})/\texttt{t}' \leq \texttt{t}\ \}$$

Où, $\Omega(\texttt{t})$ *est l'ensemble des agents du système,* $\texttt{ev} \in \Omega(\texttt{t})$ *est un évaluateur,* $\Omega_{\texttt{ev}}(\texttt{t})$ *est l'ensemble des agents connus de l'agent* `ev` *à l'instant* `t`, $\texttt{db} \in \Omega_{\texttt{ev}}(\texttt{t})$, $\texttt{cr} \in \Omega_{\texttt{ev}}(\texttt{t})$, $\texttt{t} \in \mathcal{T}$, $\texttt{t}' \in \mathcal{T}$, $\texttt{st} \in \mathcal{E}_{sp}$, $\texttt{cond} \in \mathcal{P}$ *et* $\texttt{cont} \in \mathcal{P}$.

L'historique de politiques sociales de l'agent `db` envers l'agent `cr` tel que se le représente l'agent `ev` à l'instant `t` regroupe l'ensemble des politiques

sociales dont l'agent db est le débiteur et l'agent cr le créditeur et qui ont été instanciées avant ou à l'instant t, par l'agent ev.

Plus généralement, l'ensemble des politiques sociales connues de l'agent ev peut être regroupé dans un historique unique, comme le montre la définition 5.2.4.

Définition 5.2.4 *L'ensemble des politiques sociales instanciées par l'agent* ev *avant ou à l'instant* t *est regroupé dans l'**historique général de politiques sociales** noté :*

$$_{\mathtt{ev}}\mathbf{NCS}(\mathtt{t}) \stackrel{\text{def}}{=} \bigcup_{\substack{\mathtt{x} \in \Omega_{\mathtt{ev}}(\mathtt{t}),\\ \mathtt{y} \in \Omega_{\mathtt{ev}}(\mathtt{t})}} {}_{\mathtt{ev}}\mathbf{NCS}^{\mathtt{y}}_{\mathtt{x}}(\mathtt{t})$$

5.2.3 Processus d'instanciation des normes

À chaque fois qu'une nouvelle interaction est observée par un agent ev, c'est-à-dire dès qu'un engagement social est ajouté aux historiques d'engagements sociaux de l'agent ev, ce dernier instancie les normes dont il a connaissance pour cet engagement social. Ainsi, il génère des politiques sociales qui lui permettront de vérifier si l'interaction observée est conforme ou non aux normes.

Le processus d'instanciation se déroule comme suit. Pour chaque norme dont le débiteur de l'engagement social est une cible et dont l'agent ev est un évaluateur, le processus d'instanciation génère des politiques sociales. Le débiteur de ces politiques sociales est le débiteur de l'engagement social et le créditeur est l'évaluateur ev. Les contenus des politiques sociales correspondent aux contenus des normes qu'elles instancient. Cependant, le contenu d'une politique sociale spécialise le contenu de la norme, à la fois en l'instanciant pour l'interaction considérée et en l'adaptant au point de vue local de l'évaluateur sur le système. Plusieurs politiques sociales peuvent être générées pour une même norme, si plusieurs instanciations de son contenu sont possibles.

Par ailleurs, de manière à gérer correctement les liens entre les cycles de vie de la politique sociale et la norme qu'elle instancie, la condition de la politique sociale intègre le fait que cette dernière ne doit être valable que

tant que le débiteur de l'engagement social reste une cible de la norme et tant que l'agent `ev` reste un évaluateur de la norme. Ainsi, dès que l'une de ces deux conditions n'est plus vérifiées, la politique sociale sera annulée.

La politique sociale donnée dans l'exemple 5.2.2, page 93 requiert de la part d'un engagement social particulier **e** qu'il ne porte pas sur la facette `"DRM skirting"`. Il s'agit d'une instance particulière de la norme de l'exemple 5.2.1, page 90, qui interdit aux agents de parler de contournement de mesures techniques de protection de contenu (DRM). En effet, le débiteur de la politique sociale est le débiteur de l'engagement social **e**. Le créditeur est l'évaluateur qui instancie la norme. Le contenu est spécifique à un engagement social donné et fait référence aux historiques locaux d'engagement sociaux de l'agent `ev`, qui ne correspondent qu'à la vue partielle qu'à l'agent `ev` sur les engagements sociaux pris dans le système. Le contenu de la politique sociale traduit donc le contenu de la norme dans des termes propres à l'interaction considérée et au point de vue de l'évaluateur sur le système. Enfin, les conditions permettent d'annuler la politique sociale dès que le débiteur n'est plus concerné par la norme ou dès que l'agent qui a procédé à l'instanciation n'est plus évaluateur de la norme.

Des exemples de tels processus d'instanciation sont détaillés dans le chapitre 7, page 137.

5.2.4 Propriétés

Dans cette section, nous étudions quelques propriétés particulières du modèle de politique sociale que nous proposons. Ces propriétés découlent directement du caractère décentralisé des systèmes que nous considérons. Elles portent principalement sur les historiques, l'instanciation et le cycle de vie.

Dimension temporelle

De la même manière que pour les engagements sociaux, nous considérons des historiques de politiques sociales. Nous avons donc, là encore, ajouté une dimension temporelle pour obtenir des *historiques* de politiques sociales. Ainsi, un agent est capable de retrouver l'état qu'avait une politique sociale donnée à tout instant du passé où cette politique existait. Cette distinction est importante dans le cadre des calculs des niveaux des réputations car elle

permet d'avoir accès aux niveaux des réputations d'un agent à tout instant passé (voir chapitre 6, page 105).

Instanciation et cycle de vie

Dans cette thèse nous permettons aux agents de se représenter les normes qu'ils doivent respecter à l'aide d'un modèle descriptif de normes. Nous allons cependant plus loin que la simple définition des normes, puisque nous cherchons aussi à mettre en place un système permettant aux agents de détecter le respect ou non de ces normes. C'est dans cette optique que nous avons introduit les politiques sociales.

La détection du respect ou non des normes repose sur la notion d'instanciation de celles-ci en des politiques sociales. L'instanciation permet de prendre en compte certaines caractéristiques particulières des normes et de la décentralisation : multi-violabilité des normes, adaptation des pénalités au point de vue de chaque punisseur et adaptation des contenus des politiques sociales au point de vue de chaque évaluateur.

Le caractère fortement décentralisé des systèmes que nous considérons laisse la possibilité que les points de vues locaux des agents soient globalement incohérents. Il est ainsi possible qu'un agent ait des représentations locales des engagements sociaux obsolètes. Ceci peut l'amener à détecter une violation qui n'a pas lieu d'être. C'est pour limiter l'impact de telles situations que nous avons défini un cycle de vie spécifique aux politiques sociales, en introduisant un état particulier `justifying`.

5.3 Détection de violation des normes

Dans cette section nous proposons un processus décentralisé de détection de la violation de politiques sociales. Ce processus général de détection de violation utilise un processus de justification que nous détaillons en second lieu.

5.3.1 Processus de détection de violation

La figure 5.5 décrit le processus de détection de violation par un diagramme de séquence AUML [HOH+03]. Celui-ci se déroule comme suit :

1. Des propagateurs transmettent une observation à un évaluateur.

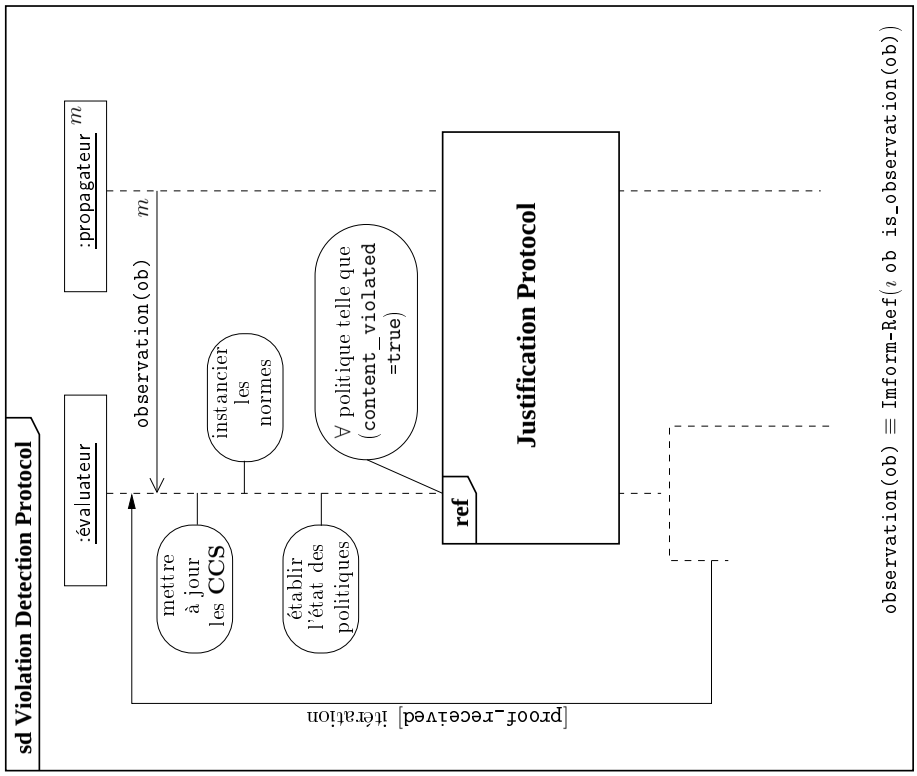


FIG. 5.5 – Processus de détection de violation.

2. L'évaluateur intègre l'observation dans ses historiques locaux d'engagements sociaux (**CCS**), puis instancie les normes dont il a connaissance et enfin tente d'établir l'état de ces politiques sociales.
3. Toutes les politiques sociales qui aboutissent à un état terminal (`fulfilled` ou `cancelled`) sont conservées, mais leur traitement s'arrête là. Toutes les politiques sociales dont le contenu est violé (*i.e.* pour lesquelles le prédicat `content_violated` est vrai, *cf.* section 5.2.2) sont passées en état `justifying` et un processus de justification est enclenché pour chacun de leurs contenus.
4. Lors de la terminaison de ces processus de justification, si des preuves ont été reçues (prédicat `proof_received` vérifié), cela signifie que les historiques d'engagements sociaux de cet évaluateur n'étaient pas à jour. Les preuves reçues jouent alors le même rôle que les observations reçues à l'étape 1 et l'évaluateur recommence le processus en étape 2, en commençant par mettre à jour ses historiques d'engagements sociaux avec les nouvelles observations qu'il a reçues en guise de preuve.
5. Dans le cas où aucune preuve n'est fournie (prédicat `proof_received` non vérifié), les politiques sociales sont simplement emmagasinées dans les historiques de politiques sociales. L'évaluateur peut alors transmettre à différents propagateurs certaines des évaluations qu'il vient de mener.

Les agents ne sont pas tenus de prendre en compte systématiquement les messages qui leur sont transmis. Par exemple, lorsqu'un évaluateur reçoit une observation (étape 1), il peut décider de ne pas la retenir s'il ne fait pas confiance à l'émetteur.

5.3.2 Processus de justification

Un évaluateur `ev` qui détecte que le contenu d'une politique sociale en état `active` n'a pas été respecté (*i.e.* le prédicat `content_violated` est vrai) fait passer celle-ci en état `justifying` et déclenche un processus de justification afin de vérifier si cette détection n'est pas liée à l'obsolescence de ses représentations locales d'engagements sociaux.

Ce processus de justification est décrit par le diagramme de séquence AUML [HOH+03] de la figure 5.6. Il se déroule comme suit :

1. L'évaluateur `ev` envoie le contenu qui crée une violation au débiteur de la politique sociale, en guise de demande de justification. Le message

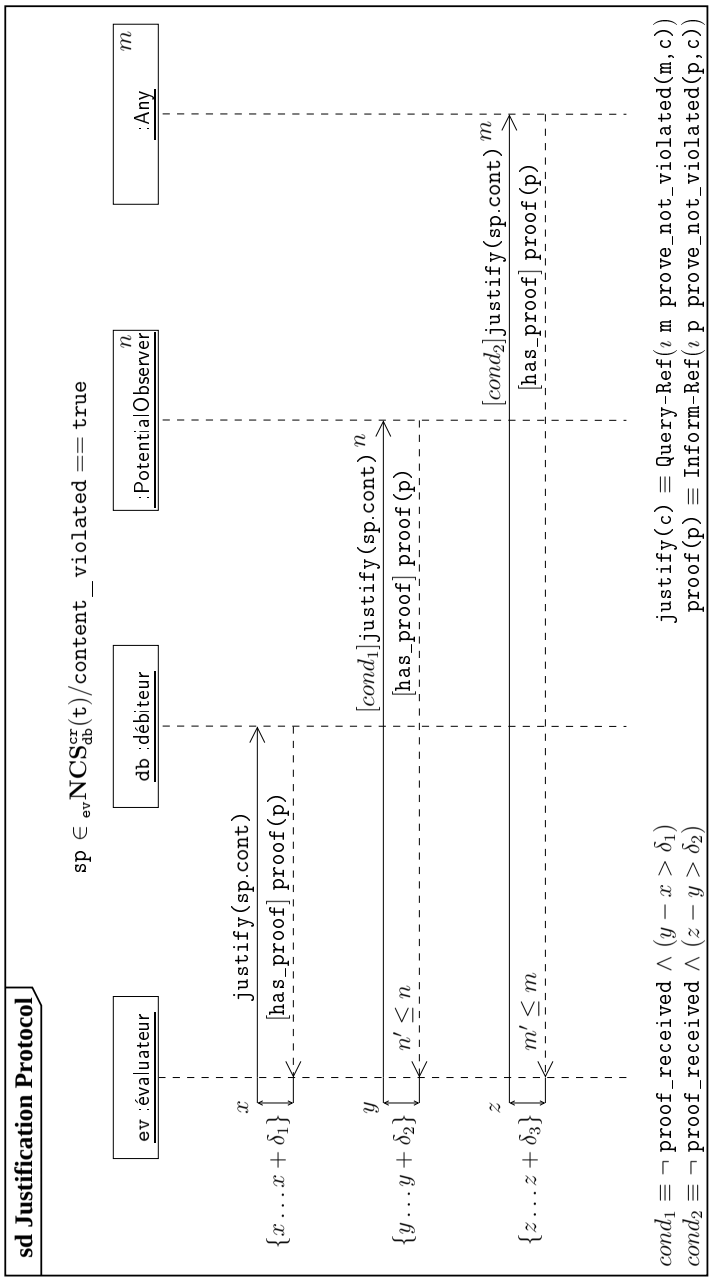


FIG. 5.6 – Processus de justification.

envoyé est de type `justify` et correspond à une requête de référence (`Query-Ref` en FIPA-ACL [FIP02]) pour un message `m` qui prouverait que le contenu `c` de la politique sociale **sp** n'a pas été violé.

2. L'agent `ev` attend durant un certain temps δ_1 des réponses. De façon à garantir une certaine équité, l'évaluateur peut, à l'étape précédente, fournir le δ_1 au débiteur, c'est-à-dire le prévenir du délai qu'il lui donne pour répondre.
3. Si, au bout du temps δ_1, le débiteur n'a pas répondu de manière satisfaisante, *i.e.* `proof_received` non validé (*cf.* $cond_1$), alors l'agent `ev` peut élargir le champ de ses investigations : il peut envoyer la demande de justification à d'autres agents, les PotentialObserver, par exemple ceux dont il pense qu'ils ont pu observer le comportement du débiteur de la politique sociale (comme les créditeurs des engagements référencés dans la politique sociale). Il procède alors à des étapes similaires aux étapes 1 et 2 avec un délai δ_2. S'il n'a toujours pas de réponse, alors il peut encore élargir son champ d'investigation à toutes ses accointances, avec un délai δ_3.
4. Si, à la fin de toutes ses investigations, l'évaluateur n'a pas de réponse satisfaisante, alors il considère que la politique sociale a bien été violée et la passe en état `violated`. Dans le cas contraire, il met à jour ses représentations locales avec les preuves reçues et annule la politique sociale.

Les agents qui reçoivent une telle demande de justification peuvent agir comme suit :

- Ils peuvent tout simplement ignorer la demande de justification.
- S'ils détectent eux aussi une violation, ils peuvent déclencher à leur tour un processus de demande de justification. Ils jouent alors le même rôle d'évaluateur que l'agent `ev`, mais dans le cadre d'un autre processus de justification se déroulant en parallèle.
- S'ils ne détectent pas de violation, c'est qu'ils ont des « preuves » (prédicat `has_proof` vérifié) que l'agent mis en cause n'a pas violé la politique sociale, par exemple, parce qu'il a annulé certains des engagements sociaux référés dans le contenu de la politique sociale. Dans ce cas de figure, ils peuvent fournir ces « preuves » à l'agent `ev`. Le message envoyé est de type `proof` (décrit par un performatif `Inform-Ref` en FIPA-ACL), donnant en référence un message `p` prouvant que le contenu `c` de la politique sociale **sp** n'a pas été violé.

Dans le cadre de cette thèse, le type de réponse qu'attend l'agent `ev`, les « preuves », sont des messages digitalement signés. Ces messages doivent prouver que les représentations locales de l'agent `ev` étaient obsolètes et que la violation qu'il a détectée n'a pas lieu d'être. La signature digitale des messages garantit la propriété de non-répudiation (à l'émission) [NAI99a] : un agent qui a signé digitalement un message ne peut pas prétendre ultérieurement ne pas l'avoir envoyé. Cette propriété de non-répudiation assure qu'un message signé digitalement constitue bien une « preuve ».

Dans le cadre du processus décrit ci-dessus, une preuve n'est valable que si elle lève *a posteriori* le soupçon de violation. C'est la raison pour laquelle nous avons ajouté une dimension temporelle aux modèles d'engagement social et de politique sociale.

Il est intéressant de noter que, les agents étant autonomes, ils ne sont pas tenus de répondre aux requêtes qu'ils reçoivent ni de prendre en compte systématiquement les réponses reçues. Le processus est défini ici dans son déroulement idéal. Dans ce cas, ils s'exposent à des sanctions de la part de l'agent qui a envoyé la requête, si celui-ci était amené à découvrir qu'ils pouvaient répondre.

5.4 Conclusion

Dans ce chapitre, nous avons tout d'abord présenté un modèle d'engagements sociaux permettant de représenter les interactions entre agents. Ce modèle est adapté aux systèmes décentralisés car il permet à chaque agent de se représenter localement les engagements qu'il perçoit et parce qu'il prend en compte le fait que deux agents peuvent avoir des perceptions différentes du monde qui les entoure.

Nous avons ensuite proposé des modèles de norme sociale et de politique sociale permettant de définir les comportements acceptables. Ces modèles sont, eux aussi, adaptés aux systèmes décentralisés et permettent non seulement de représenter les comportements acceptables, mais aussi d'en détecter la violation.

Enfin, à l'aide de ces modèles nous avons pu proposer un protocole que chaque agent peut déployer pour détecter des violations des normes qu'il connaît. Ce protocole prend en compte la possibilité que l'agent qui croit détecter une violation ait des représentations locales obsolètes. Ainsi, la détection de violation fait appel à un protocole de justification qui aboutit soit

à l'actualisation des représentations locales des historiques d'engagements sociaux de l'évaluateur, soit à la confirmation de l'occurrence d'une violation et à la mise à jour des historiques de politiques sociales. Ces protocoles de justification et de détection de violation sont, eux aussi, adaptés aux systèmes décentralisés. En effet, les différents rôles (observateur, évaluateur, punisseur et propagateur) peuvent être joués par des agents quelconques.

Chapitre 6

Modèle de réputation pour l'interaction

Dans ce chapitre, nous nous intéressons à la phase de sanction du contrôle social des interactions (*cf.* figure 4.1, page 76, parties supérieure et droite, en italique et en bleu).

Nous avons vu qu'il existait trois types de sanction : les sanctions matérielles, les sanctions sociales et les sanctions psychologiques. Les sanctions sociales telles que la confiance et la réputation sont les plus adaptées aux SMAOD puisque les agents qui les appliquent gardent un contrôle total dessus et que l'agent qui les subissent peuvent difficilement y échapper. C'est pourquoi nous proposons ici un modèle de réputation pour mener à bien la deuxième phase du contrôle social.

Le modèle de réputation est décrit selon la même grille d'analyse que celle utilisée pour le chapitre 4 : nous commençons par caractériser les types de réputation utilisés dans le modèle ainsi que leur propriétés. Ensuite, nous définissons les processus caractérisant un modèle de réputation.

6.1 Définitions

Les différents types de réputation utilisés dans le modèle L.I.A.R. peuvent être caractérisés en fonction de la source de l'information et du type de l'information qui circule entre les agents. Dans cette section, nous proposons tout d'abord un ensemble de rôles qui permet de distinguer des sources potentielles d'information, puis nous caractérisons les différents types d'information que

les agents jouant ces rôles peuvent échanger.

6.1.1 Rôles

Dans le domaine des systèmes multi-agents, la notion de rôle a été définie de différentes manières [FG99, HSB02, BOvV05]. Il nous paraît donc nécessaire de préciser, avant toute chose, que ce terme est employé ici dans un sens très général, par analogie avec le travail de [CP02]. Il s'agit plutôt de labels que des agents affectent à d'autres agents. Ce ne sont pas les agents qui décident explicitement par eux-mêmes de s'enrôler.

Définition des rôles

Nous avons identifié sept rôles que les agents peuvent jouer (les processus cités font référence à la figure 4.1, page 76) :

- Une *cible* est un agent qui est évalué du fait qu'il interagit directement avec le participant. Ce rôle est donc lié aux interactions.
- Un *participant* est un agent qui interagit directement avec la cible. Ce rôle est donc lié aux interactions.
- Un *observateur* est un agent qui modélise des messages par des engagements sociaux. Ce rôle est attaché au processus d'*observation*.
- Un *évaluateur* est un agent qui évalue des observations, c'est-à-dire qui confronte un ensemble d'engagements sociaux et un ensemble de normes et en déduit l'état d'une politique sociale. Ce rôle est associé au processus d'*évaluation*. Un tel évaluateur, capable de détecter la violation de normes a été spécifié dans la section 5.3, page 98.
- Un *punisseur* est un agent capable de punir un autre agent par une hausse ou une baisse de réputation en fonction du comportement qu'il a eu, c'est-à-dire d'établir un niveau de réputation à partir d'un ensemble de politiques sociales. Ce rôle est lié au processus de *punition*.
- Un *bénéficiaire* est un agent en mesure de raisonner et de prendre des décisions de faire confiance. Pour ce faire, il s'appuie sur les niveaux des réputations pour en déduire des intentions de confiance qu'il utilise alors pour modifier ses états mentaux. Ce rôle correspond aux processus de *raisonnement* et de *décision*.
- Un *propagateur* est un agent qui transmet des recommandations. Ce rôle est associé au processus de *propagation*.

6.1.2 Types d'information

Dans les systèmes multi-agents ouverts et décentralisés, les agents sont susceptibles de percevoir différents éléments : interactions directes, interactions indirectes, et recommandations.

Définition 6.1.1 *Une interaction est* ***directe*** *pour un agent* `a` *si celui-ci est le participant ou la cible.*

Définition 6.1.2 *Une interaction est* ***indirecte*** *pour un agent* `a`*, si celui-ci n'est ni le participant, ni la cible.*

Une interaction, au sens général, désigne soit une interaction directe, soit une interaction indirecte.

Définition 6.1.3 *Une* ***recommandation*** *est une interaction dont l'émetteur est un propagateur et dont le contenu porte sur une observation (cf. définition 6.1.4), sur une évaluation (cf. définition 6.1.5) ou sur un niveau de réputation. Une recommandation n'est pas nécessairement sincère.*

Définition 6.1.4 *Une* ***observation*** *est le résultat du processus d'observation : un engagement social dans notre cas.*

Définition 6.1.5 *Une* ***évaluation*** *est le résultat du processus d'évaluation des engagements sociaux : une politique sociale dans notre cas.*

Exemple 6.1.1 *Un message, tel que perçu par son émetteur ou son récepteur, constitue un exemple d'interaction directe. Ce même message, s'il est perçu par écoute flottante par un observateur, constitue une interaction indirecte. Si le contenu du message porte sur un engagement social, sur une politique sociale ou sur un niveau de réputation, alors il s'agit d'une recommandation.*

6.1.3 Types de réputation

Dans cette section, nous définissons différents types de réputation à partir du type d'information que reçoit un bénéficiaire donné. Dans les figures de cette section, les rôles sont donnés selon le point de vue du bénéficiaire.

Information reçue par le bénéficiaire et types de réputation

En s'appuyant sur les types d'information déterminés précédemment, les agents peuvent différencier les types de réputation suivants :

- **Réputation fondée sur les Interactions Directes** (`DIbRp`) dont le niveau est estimé à partir d'interactions directes.
- **Réputation fondée sur les Interactions Indirectes** (`IIbRp`) dont le niveau est estimé à partir d'interactions indirectes.
- **Réputation fondée sur les Recommandations** (`RcbRp`) dont le niveau est estimé en agrégeant des recommandations transmises par des propagateurs. En fonction de ce que contiennent les recommandations, on peut distinguer trois sous-types de `RcbRp` :
  - **Réputation fondée sur les Recommandations d'Observations** (`ObsRcbRp`) dont le niveau est estimé en agrégeant des observations transmises par des propagateurs.
  - **Réputation fondée sur les Recommandations d'Évaluations** (`EvRcbRp`) dont le niveau est estimé en agrégeant des évaluations transmises par des propagateurs.
  - **Réputation fondée sur les Recommandations de Réputation** (`RpRcbRp`) dont le niveau est estimé en agrégeant des niveaux de réputation transmises par des propagateurs.

Nous détaillons dans les sections suivantes ces trois types de réputation.

Réputation fondée sur les Interactions Directes

La figure 6.1 présente le cas où un même agent (Alice) joue le rôle de participant, d'observateur, d'évaluateur, de punisseur et de bénéficiaire. Il n'y a pas de propagateur et la cible est un agent différent, Bertrand. Le type d'information qu'utilise Alice pour construire sa réputation est une interaction directe. La réputation ainsi déterminée est donc appelée Réputation fondée sur les Interactions Directes.

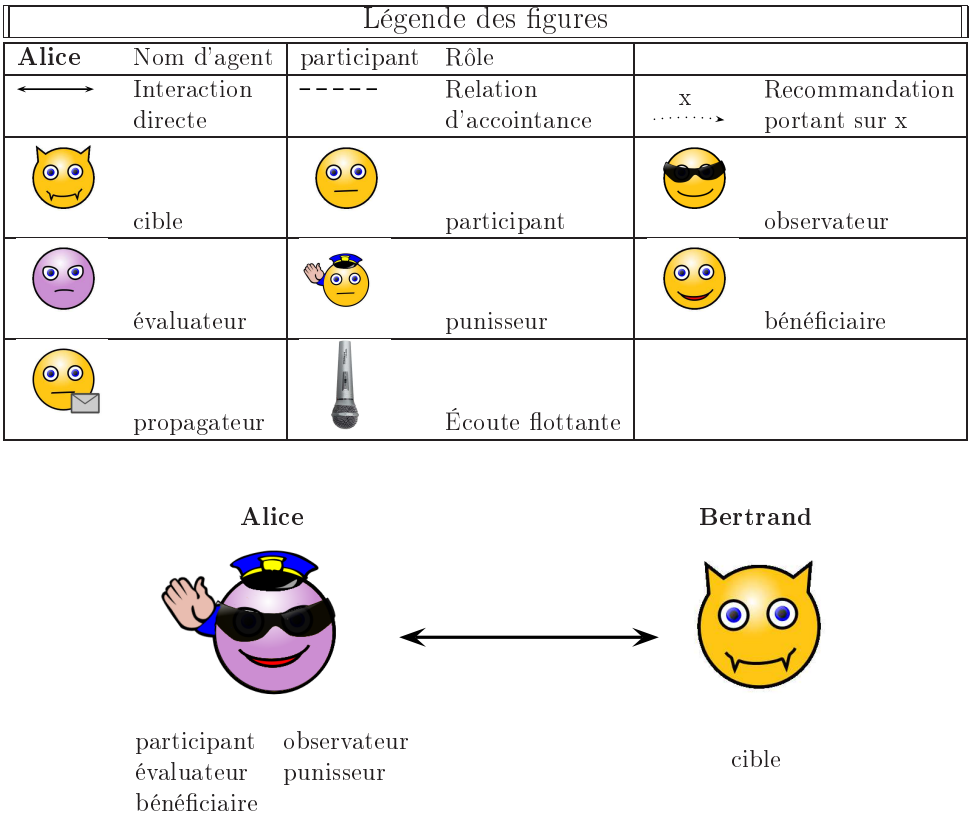


FIG. 6.1 – Réputation fondée sur les Interactions Directes.

Réputation fondée sur les Interactions Indirectes

La figure 6.2 illustre le cas de la Réputation fondée sur les Interactions Indirectes, où deux agents différents jouent les rôles d'observateur (Charles) et de participant (Alice).

Réputation fondée sur les Recommandations

La figure 6.3 illustre le cas de la Réputation fondée sur les Recommandations d'Observations. Des agents (ici Alice, Bertrand et Charles) sont à la fois observateur et propagateur. Ces agents ont observé des interactions (directes ou indirectes) et transmettent leurs observations à travers des chaînes

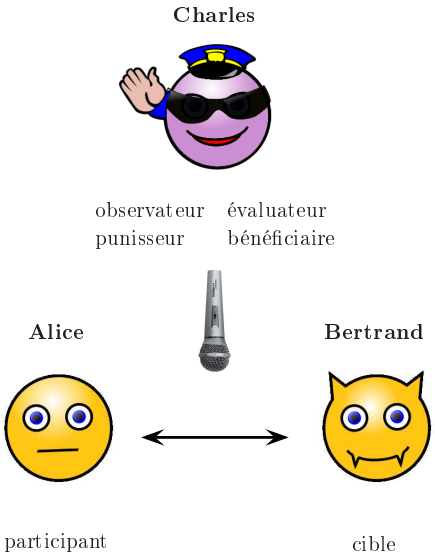


FIG. 6.2 – Réputation fondée sur les Interactions Indirectes.

de propagation plus ou moins longues, à un autre agent (Daniel), qui joue les rôles d'évaluateur, de punisseur et de bénéficiaire. La principale différence avec les types de réputation précédents réside dans le fait que des agents différents jouent les rôles d'observateur (Alice, Bertrand et Charles) et d'évaluateur (Daniel). Le type de l'information passée, par les propagateurs, depuis l'observateur jusqu'au bénéficiaire est une observation. Ainsi, le bénéficiaire reçoit une recommandation portant sur une observation. Nous appelons la réputation construite dans ce cas une Réputation fondée sur les Recommandations d'Observations. Le bénéficiaire est ici nécessairement un évaluateur et un punisseur, puisqu'il doit intégrer l'information qu'il reçoit à son propre modèle de réputation afin d'être en mesure de raisonner et de prendre des décisions de faire confiance.

L'observateur peut avoir obtenu son observation de deux manières : par interaction directe ou indirecte. Les rôles de cible et participant ne sont pas représentés dans la figure 6.3 de façon à laisser ouvertes les deux possibilités. D'autre part, dans le cas particulier de l'interaction directe (où l'observateur est aussi le participant), l'information que cet observateur transmet est l'observation d'une interaction directe. La réputation est alors celle définie

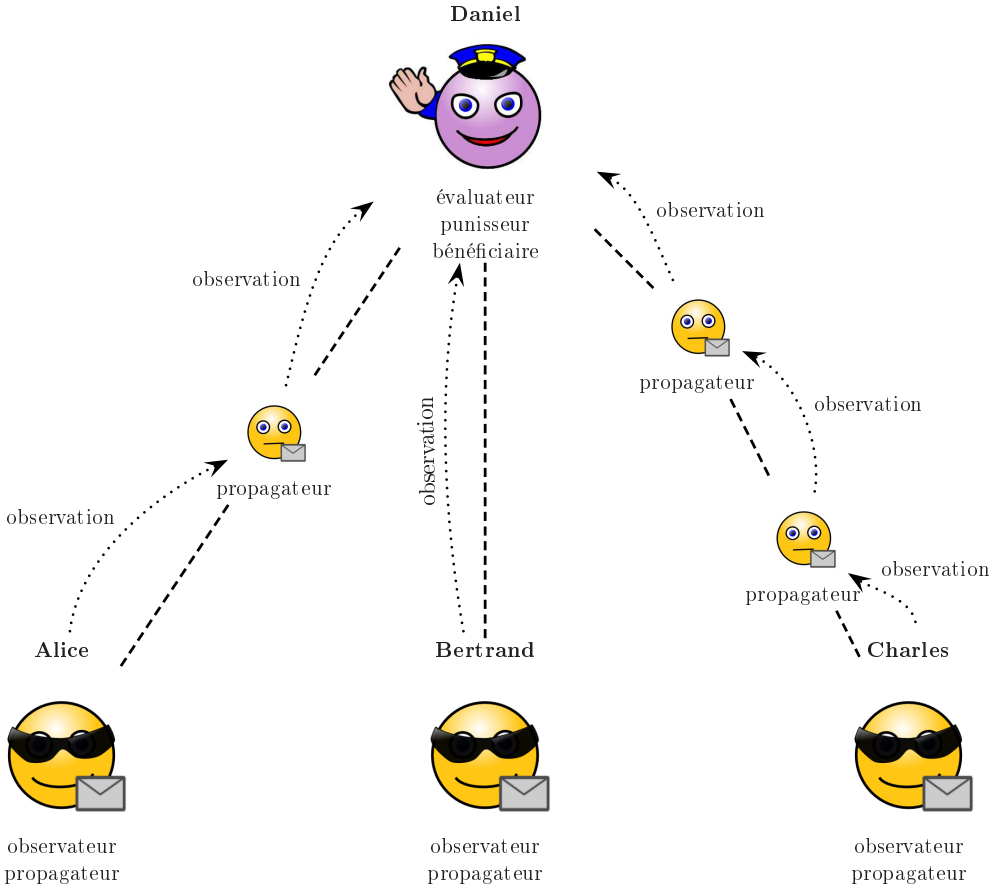


FIG. 6.3 – Réputation fondée sur les Recommandations d'Observations.

comme la « réputation observée » par [MHM02] (voir figure 3.5, page 45).

La figure 6.4 illustre le cas de la Réputation fondée sur les Recommandations d'Évaluations. Ce cas est similaire au précédent : les agents Alice, Bertrand et Charles transmettent des recommandations à Daniel. Cette fois-ci, les recommandations portent sur des évaluations : les agents jouant les rôles d'évaluateur (Alice, Bertrand et Charles) et de punisseur (Daniel) sont différents. Les propagateurs ont pu obtenir leur évaluation à partir de différents éléments : interaction directe ou indirecte ou encore recommandation d'observation qu'ils ont évaluée eux-mêmes, ou recommandation d'évaluation qu'ils transmettent simplement. L'information est passée entre les mains d'un certain nombre d'agents avant de parvenir au bénéficiaire et a été diversement

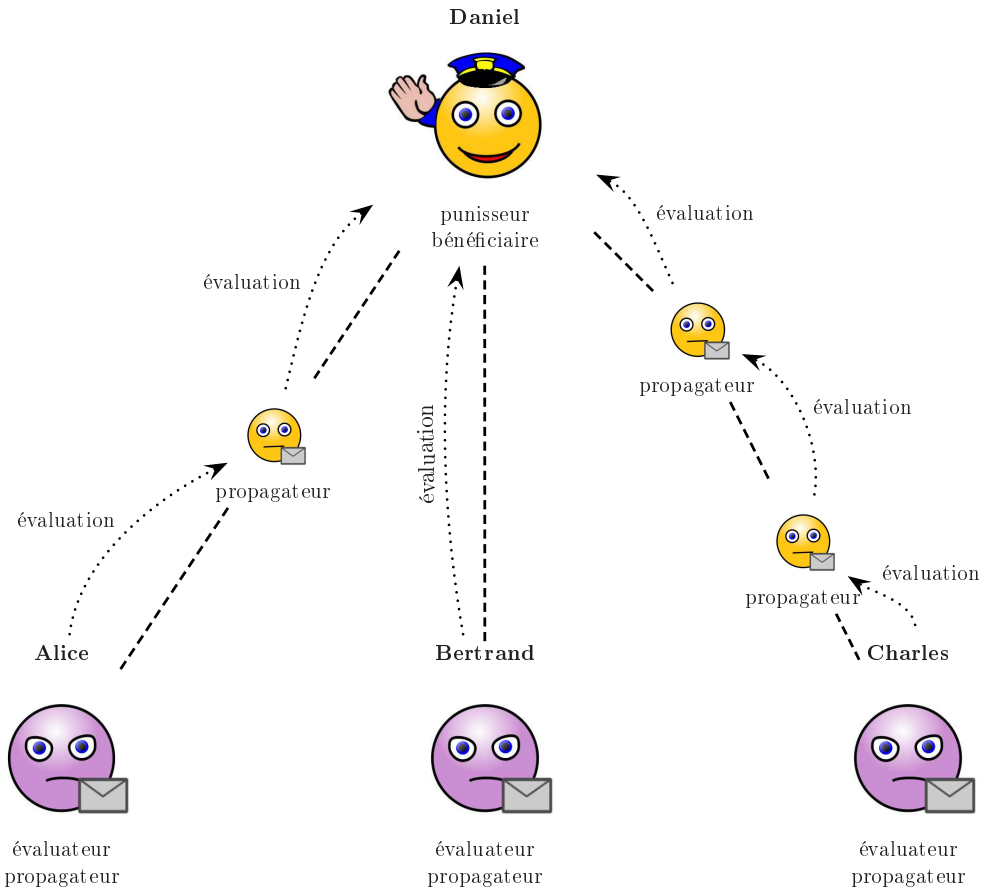


FIG. 6.4 – Réputation fondée sur les Recommandations d'Évaluations.

interprétée. La réputation que le bénéficiaire peut construire est alors appelée Réputation fondée sur les Recommandations d'Évaluations. Le bénéficiaire est ici nécessairement un punisseur, puisqu'il doit intégrer l'information qu'il reçoit à son propre modèle de réputation afin d'être en mesure de raisonner et prendre des décisions de faire confiance.

Dans la figure 6.4, les rôles sont présentés selon le point de vue du bénéficiaire. Les rôles de cible, de participant et d'observateur ne sont pas représentés, d'une part pour laisser ouvertes les différentes possibilités pour les évaluateurs d'avoir obtenu leurs évaluations et, d'autre part, car il n'est pas nécessairement possible pour le bénéficiaire de déterminer quels sont les agents qui ont effectivement joué ces rôles.

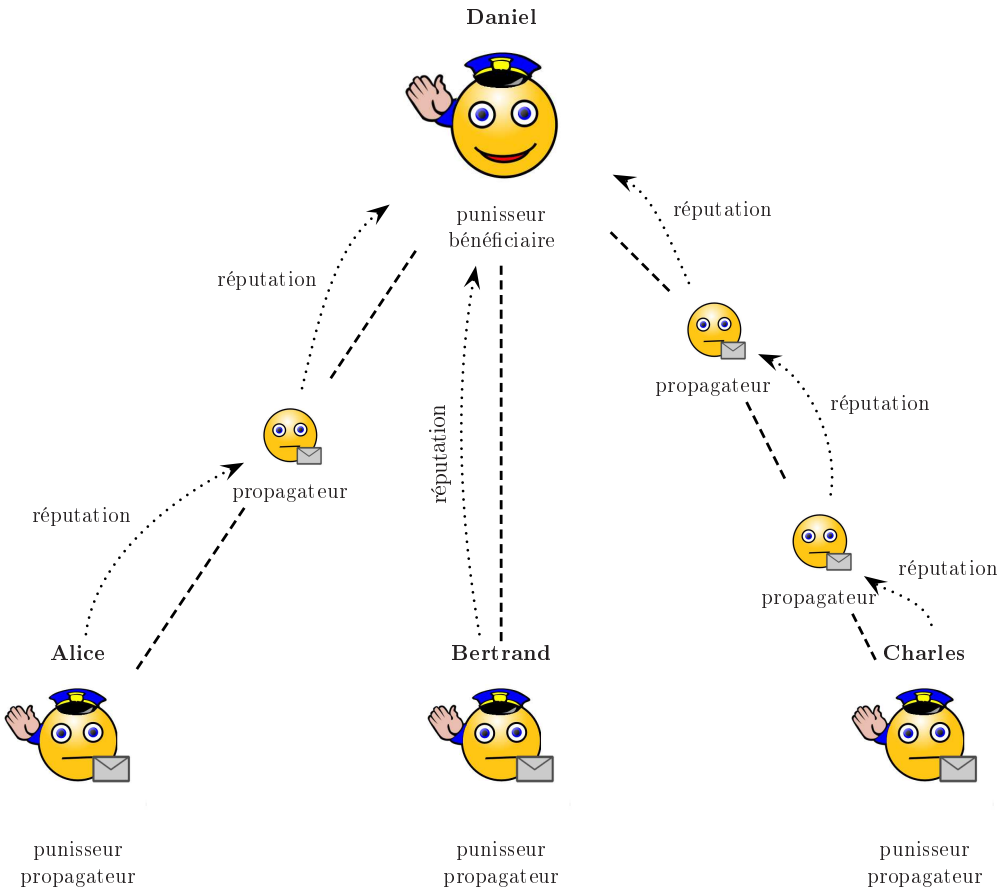


FIG. 6.5 – Réputation fondée sur les Recommandations de Réputation.

La figure 6.5 présente un cas similaire aux précédents, mais où le bénéficiaire n'est pas le seul punisseur. L'information est passée entre les mains d'un certain nombre d'agents avant de parvenir au bénéficiaire et a été interprétée de différentes manières. En effet, les propagateurs peuvent avoir obtenu le niveau de réputation qu'ils transmettent de diverses manières : interaction directes ou indirectes qu'il ont évaluées et punies eux-mêmes, recommandations d'observation ou d'évaluation à partir desquelles ils ont établi un niveau de réputation, recommandation de réputation qu'il transmettent simplement, etc. La réputation que le bénéficiaire peut construire est alors appelée Réputation fondée sur les Recommandations de Réputation. Le bénéficiaire est ici nécessairement un punisseur, puisqu'il doit intégrer l'information qu'il reçoit

à son propre modèle de réputation.

6.2 Propriétés

Dans cette section, nous étudions les propriétés des réputations : subjectivité, multi-facette, multi-dimension, dynamisme, graduation et transitivité.

6.2.1 Formulation

Les réputations que le modèle L.I.A.R. manipule, quelque soit leur type, sont des croyances sociales, dans le sens où elles caractérisent une relation entre deux agents : le bénéficiaire et la cible. Ces différentes réputations sont **subjectives** puisque deux bénéficiaires peuvent avoir des estimations différentes de la réputation d'une même cible. Elles sont **multi-facettes** puisqu'une cible peut être appréciée différemment selon chaque facette qu'elle présente. Elles sont **multi-dimensionnelles** puisque chacune de ces facettes peut être jugée selon plusieurs dimensions. Elles sont aussi **dynamiques** puisque leur évolution dépend du comportement qu'a la cible au fil du temps. Enfin, en conséquence de la propriété de subjectivité, nous considérons que ni les niveaux de réputation, ni les décisions **ne sont transitives**. En effet, du fait que les agents n'ont pas directement accès aux niveaux de réputations que les autres maintiennent, nous considérons que les réputations ne sont pas transitives. De plus, du fait que les niveaux des réputations locales à un agent peuvent l'amener a décider de ne pas agir en confiance avec une cible alors qu'il existe une chaîne de confiance le liant à cette cible (et *vice-versa*), nous considérons que les décisions ne sont pas transitives non plus.

Du fait de ces propriétés, les réputations peuvent être différenciées en fonction de : l'agent qui l'estime, l'agent qu'elle cible, la facette de l'agent qu'elle caractérise, la dimension selon laquelle cette facette est estimée et l'instant auquel elle est calculée. Nous modélisons donc les différents types de réputation définis précédemment comme suit :

Définition 6.2.1 *La **Réputation fondée sur les Interactions Directes** est définie, à un instant* `t`*, pour un bénéficiaire* `x` *envers une cible* `y` *pour une facette* `f` *et selon la dimension* `d`*. Elle est notée formellement :* $\mathtt{DIbRp}_{\mathtt{x}}^{\mathtt{y}}(\mathtt{f},\mathtt{d},\mathtt{t})$.

Définition 6.2.2 *La* ***Réputation fondée sur les Interactions Indirectes*** *est définie, à un instant* t*, pour un bénéficiaire* x *envers une cible* y *pour une facette* f *et selon la dimension* d*. Elle est notée formellement :* $\mathtt{IIbRp}^{\mathtt{y}}_{\mathtt{x}}(\mathtt{f},\mathtt{d},\mathtt{t})$.

Définition 6.2.3 *La* ***Réputation fondée sur les Recommandations d'Observations*** *est définie, à un instant* t*, pour un bénéficiaire* x *envers une cible* y *pour une facette* f *et selon la dimension* d*. Elle est notée formellement :* $\mathtt{ObsRcbRp}^{\mathtt{y}}_{\mathtt{x}}(\mathtt{f},\mathtt{d},\mathtt{t})$.

Définition 6.2.4 *La* ***Réputation fondée sur les Recommandations d'Évaluations*** *est définie, à un instant* t*, pour un bénéficiaire* x *envers une cible* y *pour une facette* f *et selon la dimension* d*. Elle est notée formellement :* $\mathtt{EvRcbRp}^{\mathtt{y}}_{\mathtt{x}}(\mathtt{f},\mathtt{d},\mathtt{t})$.

Définition 6.2.5 *La* ***Réputation fondée sur les Recommandations de Réputation*** *est définie, à un instant* t*, pour un bénéficiaire* x *envers une cible* y *pour une facette* f *et selon la dimension* d*. Elle est notée formellement :* $\mathtt{RpRcbRp}^{\mathtt{y}}_{\mathtt{x}}(\mathtt{f},\mathtt{d},\mathtt{t})$.

6.2.2 Graduation et représentation computationnelle

Pour permettre aux agents logiciels de manipuler des réputations, il est tout d'abord nécessaire de fixer une représentation computationnelle de ces croyances. Cette section propose une telle représentation.

Nous considérons dans cette thèse que les réputations sont un moyen, pour un agent, d'ordonner partiellement ses accointances. En effet, dans certains cas, il est possible de comparer deux agents à l'aide de leurs niveaux de réputation, l'un étant, par exemple, supérieur à l'autre. Dans d'autres cas, cette comparaison n'est pas possible, par exemple si aucune information n'est accessible sur l'un des deux agents.

De nombreuses représentations computationnelles sont disponibles afin représenter un tel ordre partiel (graphes, ensembles réels, etc.). Nous avons choisi de représenter un niveau de réputation par une valeur dans $[-1, +1] \cup$ `unknown`. La sémantique associée à ces valeurs est la suivante :

- Un agent a une réputation de -1 si les observations montrent qu'il s'est toujours mal comporté.
- Un agent a une réputation de $+1$ si les observations montrent qu'il s'est toujours bien comporté.
- Un agent dont la réputation est 0 est un agent suffisamment connu (le bénéficiaire a accumulé suffisamment d'information) pour qu'un avis ait été formé à son propos, mais cet avis n'est ni positif, ni négatif : il est neutre.
- La réputation peut prendre la valeur particulière `unknown`, qui indique qu'il n'y a pas assez d'information à propos de la cible pour construire une réputation. Cette valeur permet d'exprimer l'ignorance et de la distinguer du cas précédent de neutralité, pour lequel de l'information est présente.

L'ensemble ci-dessus a, par ailleurs, l'avantage d'être **infini**, donc de ne pas limiter *a priori* le nombre d'accointances auxquelles un agent pourra associer un niveau de réputation. D'autre part, cet ensemble est **continu** : il est toujours possible de placer un niveau de réputation entre deux autres.

6.3 Processus

Grâce à la représentation computationnelle proposée précédemment, des agents logiciels sont en mesure de manipuler des valeurs représentant les niveaux des réputations qu'ils associent aux autres agents. Dans cette section, nous définissons des processus qui permettent aux agents de manipuler ces valeurs. Nous définissons six processus : initialisation, évaluation, punition, raisonnement, décision et propagation.

6.3.1 Initialisation

Le processus d'initialisation consiste à définir le niveau de la réputation d'un agent en l'absence de suffisamment d'information sur celui-ci. L'agent qui lance ce processus joue le rôle d'évaluateur. Grâce à la représentation computationnelle que nous avons choisie, et en particulier à l'introduction

de la valeur spéciale `unknown`, le processus d'initialisation de notre modèle est simple. En effet, dans le modèle L.I.A.R., un agent attribue un niveau de réputation `unknown` à tout agent pour lequel il n'a pas d'information sur laquelle construire une réputation.

6.3.2 Évaluation

Le processus d'évaluation consiste à caractériser plus ou moins positivement ou négativement un comportement observé à l'aide d'un ensemble de normes. C'est un agent jouant le rôle d'évaluateur qui lance ce processus. Le processus d'évaluation employé dans le cadre de cette thèse est le processus de détection de la violation de normes défini dans la section 5.3, page 98.

Ce processus se déroule en deux temps : une phase d'instanciation des normes en politiques sociales, puis une phase de détermination de l'état de ces politiques sociales. Lorsqu'un agent dispose d'une nouvelle observation d'une interaction, il l'ajoute à ses historiques d'engagements sociaux. Il instancie alors les normes qu'il connaît. Cette étape génère un ensemble de politiques sociales. L'agent cherche ensuite à établir l'état de ces politiques sociales. En particulier, il déclenche des processus de justification pour toutes les politiques en état `justifying`. À l'aboutissement de ces processus de justification, l'agent dispose d'un ensemble de politiques sociales en état terminal, qui viennent enrichir ses historiques de politiques sociales. L'agent est alors en mesure d'établir des niveaux de réputation à partir de ces ensembles.

6.3.3 Punition

Dans cette section, nous proposons des algorithmes pour le processus de punition pour chacun des types de réputation que nous avons définis. Un processus de punition est capable de traduire un ensemble d'évaluations (politiques sociales en état terminal) ou de recommandations en une hausse ou baisse de réputation. Ce sont les punisseurs qui sont en charge de mener à bien ces processus. Les algorithmes que nous proposons s'appuient sur le contenu des historiques de politiques sociales et les ensembles de recommandations à un instant donné.

Nous commençons par définir certains sous-ensembles des historiques de politiques sociales, afin de différencier les interactions en fonction de leur caractère direct ou indirect, ainsi qu'en fonction des facettes et des dimensions qu'elles concernent. Nous proposons ensuite un moyen de quantifier

ces ensembles. Enfin, nous présentons des fonctions calculant des niveaux de Réputation fondée sur les Interactions Directes et de Réputation fondée sur les Interactions Indirectes à partir des quantités et des états des politiques sociales se trouvant dans ces sous-ensembles des historiques et des fonctions calculant les niveaux des Réputation fondée sur les Recommandations à partir des ensembles de recommandations.

Interactions directes et indirectes

Afin de calculer la valeur de réputation d'un agent tg (cible) à un instant t, un agent pu (punisseur) considère les politiques sociales dont il a connaissance et qui concernent l'agent tg, c'est-à-dire les politiques sociales dont l'agent tg est le débiteur. Pour différencier les politiques sociales qui font référence à des interactions directes ou indirectes, il est nécessaire de considérer le contenu des politiques sociales. Si celui-ci fait référence à un engagement social dont l'agent pu est créditeur, alors il s'agit d'une interaction directe, sinon d'une interaction indirecte. Afin de différencier les politiques sociales en fonction du caractère direct ou non de l'interaction qu'elles évaluent, nous définissons des ensembles de politiques sociales suivants (où $\mathcal{A} \subseteq \Omega_{\mathtt{pu}}(\mathtt{t})$ est un ensemble d'agents) :

$$
{}_{\mathtt{pu}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in {}_{\mathtt{pu}}\mathbf{NCS}(\mathtt{t})/\mathbf{sp}.\mathtt{db} = \mathtt{tg} \wedge
$$
$$
\mathtt{pu.has\_creditor}(\mathbf{sp}.\mathtt{cont}, \mathtt{x})/\mathtt{x} \in \mathcal{A}\}
$$

Où $\mathtt{pu.has\_creditor}(\mathbf{sp}.\mathtt{cont}, \mathtt{x})$ est vrai si x est créditeur dans le contenu $\mathbf{sp}.\mathtt{cont}$. En d'autres termes, le prédicat est vrai si x est une victime potentielle de la violation d'un engagement de l'agent tg. Les ensembles de ce type où $\mathcal{A} = \{\mathtt{pu}\}$ référencent seulement des interactions directes et les ensembles tels que $\mathtt{pu} \notin \mathcal{A}$ référencent uniquement des interactions indirectes.

Dans les définitions de ces historiques de politiques sociales, nous exprimons la facette par un paramètre α et la dimension par un autre paramètre δ, afin de regrouper les politiques sociales liées à une même facette et une même dimension dans les ensembles ${}_{\mathtt{pu}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\alpha, \delta, \mathtt{t}) \subseteq {}_{\mathtt{pu}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\mathtt{t})$ tels que :

$$
{}_{\mathtt{pu}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\alpha, \delta, \mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in {}_{\mathtt{pu}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\mathtt{t})/
$$
$$
\delta \in \mathtt{pu.dimensions}(\mathbf{sp}) \wedge \alpha \in \mathtt{pu.facets}(\mathbf{sp}.\mathtt{cont})\}
$$

À partir d'un tel ensemble de politiques sociales ${}_{\mathtt{pu}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t})$, nous définissons les trois sous-ensembles suivants :

- **Fulfilled NCS**
  ${}_{\mathtt{pu}}\mathbf{FNCS}^{\mathcal{A}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in {}_{\mathtt{pu}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t})/\ \mathbf{sp}.\mathtt{st} = \mathtt{fulfilled}\}$;
- **Violated NCS**
  ${}_{\mathtt{pu}}\mathbf{VNCS}^{\mathcal{A}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in {}_{\mathtt{pu}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t})/\ \mathbf{sp}.\mathtt{st} = \mathtt{violated}\}$;
- **Cancelled NCS**
  ${}_{\mathtt{pu}}\mathbf{CNCS}^{\mathcal{A}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in {}_{\mathtt{pu}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t})/\ \mathbf{sp}.\mathtt{st} = \mathtt{cancelled}\}$;

Pour simplifier la notation, nous référençons ces ensembles avec ${}_{\mathtt{pu}}\mathcal{X}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t})$ où $\mathcal{X} \in \{F,V,C\}$.

Importance des historiques de politiques sociales

Afin d'associer un poids aux différentes politiques sociales, nous utilisons les pénalités qui leur sont attachées. Pour associer un poids aux différents ensembles de politiques sociales identifiés précédemment, nous additionnons simplement les pénalités de toutes les politiques sociales qu'ils contiennent. Ainsi, si ${}_{\mathtt{pu}}E(\mathtt{t})$ est un ensemble de politiques sociales connues de l'agent pu à l'instant t, alors la fonction Imp calcule la somme des pénalités associées aux politiques sociales contenues dans l'ensemble ${}_{\mathtt{pu}}E(\mathtt{t})$:

$$\mathtt{Imp}({}_{\mathtt{pu}}E(\mathtt{t})) \stackrel{\text{def}}{=} \sum_{\mathbf{sp} \in {}_{\mathtt{pu}}E(\mathtt{t})} \mathtt{pu}.\mathtt{punishes}(\mathbf{sp},\mathtt{t})$$

Les pénalités étant définies en section 5.2.2, page 91, par une importance dans $[0,+1]$, nous avons, pour tout ensemble E : $\mathtt{Imp}(E) \in [0,|E|]$.

Réputation fondée sur les Interactions Directes

La *Réputation fondée sur les Interactions Directes* (abrégée DIbRp) attachée à un agent tg (cible) par un agent pu (punisseur) pour la facette α selon une dimension δ à l'instant t, est calculée à partir des interactions directes entre l'agent qui punit, pu, et l'agent évalué, tg. La DIbRp agrège les différentes politiques sociales évaluant des interactions directes comme suit :

$$\mathtt{DIbRp}^{\mathtt{tg}}_{\mathtt{pu}}(\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \frac{\sum_{\mathcal{X}\in\{F,V,C\}} \tau_{\mathcal{X}} \times \mathtt{Imp}({}_{\mathtt{pu}}\mathcal{X}\mathbf{NCS}^{\{\mathtt{pu}\}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t}))}{\sum_{\mathcal{X}\in\{F,V,C\}} |\tau_{\mathcal{X}}| \times \mathtt{Imp}({}_{\mathtt{pu}}\mathcal{X}\mathbf{NCS}^{\{\mathtt{pu}\}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t}))}$$

La fonction fait apparaître deux niveaux de pondération : d'une part, chaque politique sociale est pondérée d'une importance (à travers la fonction `Imp`), et d'autre part, les différents états ont des importances relatives : τ_F, τ_V et τ_C, qui sont les poids associés aux politiques sociales remplies, violées et annulées, respectivement. Ces pondérations sont des valeurs réelles que chaque agent est libre de fixer, avec pour seule contrainte que $\tau_F > 0$, $\tau_V < 0$.

Réputation fondée sur les Interactions Indirectes

La *Réputation fondée sur les Interactions Indirectes* (abrégée `IIbRp`) attachée à un agent `tg` par un agent `pu` pour la facette α selon une dimension δ à l'instant `t`, est calculée de la même manière que la `DIbRp`, mais avec d'autres historiques de politiques sociales :

$$\mathtt{IIbRp}^{\mathtt{tg}}_{\mathtt{pu}}(\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \frac{\sum_{\mathcal{X}\in\{F,V,C\}} \tau'_{\mathcal{X}} \times \mathtt{Imp}({}_{\mathtt{pu}}\mathcal{X}\mathbf{NCS}^{\Omega_{\mathtt{pu}}(\mathtt{t})\setminus\{\mathtt{pu}\}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t}))}{\sum_{\mathcal{X}\in\{F,V,C\}} |\tau'_{\mathcal{X}}| \times \mathtt{Imp}({}_{\mathtt{pu}}\mathcal{X}\mathbf{NCS}^{\Omega_{\mathtt{pu}}(\mathtt{t})\setminus\{\mathtt{pu}\}}_{\mathtt{tg}}(\alpha,\delta,\mathtt{t}))}$$

La fonction ci-dessus est semblable à la précédente. Cependant, contrairement au cas de la `DIbRp`, les politiques sociales considérées ici ne sont pas celles où `tg` est le créditeur, mais les politiques sociales concernant des interactions indirectes, c'est-à-dire où les créditeurs sont tous les autres agents : $\Omega_{\mathtt{pu}}(\mathtt{t}) \setminus \{\mathtt{pu}\}$. Les paramètres $\tau'_{\mathcal{F}}$ et $\tau'_{\mathcal{V}}$ sont soumis aux mêmes contraintes que les paramètres $\tau_{\mathcal{F}}$ et $\tau_{\mathcal{V}}$.

Réputation fondée sur les Recommandations

Les différents types de *Réputation fondée sur les Recommandations* nécessitent que certains agents (propagateurs) aient communiqué des recommandations sur la cible. La manière dont un propagateur a obtenu l'information qu'il transmet n'importe pas ici. Il est, par ailleurs, tout à fait possible que sa recommandation soit un mensonge. Dans tous les cas, les recommandations sont des communications ; elles se trouvent donc modélisées dans les historiques d'engagements sociaux des agents.

Le punisseur utilise son processus de raisonnement afin de filtrer les mauvaises recommandations et de déterminer l'ensemble des « recommandations de confiance » et celui des « propagateurs de confiance ». Le processus de

raisonnement utilisé dans le but de déterminer ces deux ensembles est exactement le même que pour n'importe quel autre raisonnement sur les réputations. Ce processus, `reasons`, prend en entrée une cible, une facette, une dimension, un ensemble de seuils de raisonnement et un instant. La facette utilisée pour le filtrage des recommandations est la facette de recommandation : `"recommend"`, la dimension sur laquelle sont jugés les propagateurs est l'intégrité : `integrity` et l'ensemble de seuils lié au raisonnement sur les recommandations est noté `RcLev`. La définition précise du processus de raisonnement, de ses entrées et de ses sorties est donnée dans la section 6.3.4. Le processus `reasons` renvoie une intention de confiance, composée d'un couple de valeur : d'une part, si l'intention est de faire confiance ou non (`trust_int`, qui prend sa valeur dans $\{\texttt{trust}, \texttt{distrust}\}$ et, d'autre part, la force de l'intention de confiance (`trust_val`, valeur réelle). Ici, la sortie `trust_int` est utilisée pour choisir les recommandations et les propagateurs de confiance.

En utilisant ce processus, l'agent `pu` (punisseur) obtient, à un instant `t` un ensemble de recommandations de confiance, noté ${}_{\texttt{pu}}\texttt{TRc}(\texttt{t})$:

$$
\begin{aligned}
{}_{\texttt{pu}}\texttt{TRc}(\texttt{t}) \stackrel{\text{def}}{=} \{\mathbf{sc} \in {}_{\texttt{pu}}\mathbf{CCS}(\texttt{t}) / \mathbf{sc}.\texttt{cr} = \texttt{pu} \wedge \\
\texttt{"recommend"} \in \texttt{pu.facets}(\mathbf{sc}.\texttt{cont}) \\
\wedge \texttt{pu.reasons}(\mathbf{sc}.\texttt{db}, \texttt{"recommend"}, \texttt{integrity}, \texttt{RcLev}, \texttt{t}).\texttt{trust\_int} = \texttt{trust}\}
\end{aligned}
$$

L'ensemble de recommandations de confiance est constitué de l'ensemble des engagements sociaux reçus, avant ou à l'instant `t`, par l'agent `pu`, dont le contenu porte (entre autres) sur la facette de recommandation et pour lesquels `pu` à l'intention de faire confiance au débiteur (un propagateur) pour son intégrité dans ses recommandations.

Réputation fondée sur les Recommandations d'Observations. La Réputation fondée sur les Recommandations d'Observations (abrégée `ObsRcbRp`) se calcule à partir des recommandations portant sur des observations. L'ensemble de ces recommandations noté `TObsRc` est défini comme suit :

$$
\begin{aligned}
{}_{\texttt{pu}}\texttt{TObsRc}(\texttt{tg}, \alpha, \texttt{t}) \stackrel{\text{def}}{=} \{\mathbf{rc} \in {}_{\texttt{pu}}\texttt{TRc}(\texttt{t}) / \\
\mathbf{sc} = \mathbf{rc}.\texttt{cont} \wedge \mathbf{sc} \in {}_{\bullet}\mathbf{CCS}(\texttt{t}) \wedge \mathbf{sc}.\texttt{db} = \texttt{tg} \wedge \mathbf{sc}.\texttt{t}_{\texttt{e}} \leq \texttt{t} \wedge \\
\alpha \in \texttt{pu.facets}(\mathbf{sc}.\texttt{cont})\}
\end{aligned}
$$

Où ${}_{\bullet}\mathbf{CCS}(\texttt{t}) = \bigcup_{\texttt{ag} \in \Omega_{\texttt{pu}}(\texttt{t})} {}_{\texttt{ag}}\mathbf{CCS}(\texttt{t})$.

Les recommandations d'observation concernant une cible `tg` pour une facette α sont des recommandations de confiance dont le contenu est un engagement social pris par `tg` et dont le contenu porte, au moins, sur la facette α.

À partir des engagements sociaux contenus dans cet ensemble de recommandations d'observation, l'agent `pu` peut déclencher un processus de détection de violation des normes. Pour ce faire, il commence par instancier les normes dont il a connaissance avec l'ensemble des engagements sociaux qu'il connaît : ceux acquis par interaction directe ou indirecte et ceux reçus par recommandation. Il obtient ainsi un ensemble de politiques sociales. L'agent peut alors établir l'état final de ces politiques sociales, en utilisant des processus de justification pour les politiques qui seraient encore en état `justifying`.

Le calcul du niveau de Réputation fondée sur les Recommandations d'Observations se déroule ensuite de manière similaire au calcul du niveau des Réputation fondée sur les Interactions Directes et Réputation fondée sur les Interactions Indirectes : l'ensemble de politiques sociales ainsi obtenu est noté $\mathcal{SP}(t)$ et peut alors être séparé en trois sous-ensembles en fonction de l'état des politiques sociales :

- **Recommandation d'Observation FNCS**, noté **ObsRcFNCS**, qui contient les politiques sociales en état `fulfilled` et qui impliquent des observations reçues par recommandation, portant sur une facette et une dimension donnée :

$$
\begin{aligned}
{}_{\mathtt{pu}}\mathbf{ObsRcFNCS}(\mathtt{tg}, \alpha, \delta, \mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in \mathcal{SP}(t)/\mathbf{sp}.\mathtt{st} = \mathtt{fulfilled} \wedge \\
\alpha \in \mathtt{pu.facets}(\mathbf{sp}.\mathtt{cont}) \wedge \delta \in \mathtt{pu.dimensions}(\mathbf{sp}) \wedge \\
\mathtt{refers}(\mathbf{sp}.\mathtt{cont}, {}_{\mathtt{pu}}\mathtt{TObsRc}(\mathtt{tg}, \alpha, \mathtt{t}))\}
\end{aligned}
$$

  où `refers` est vrai (resp. faux) si le contenu de la politique sociale référence fait (resp. ne fait pas) référence à un engagement social reçu par recommandation.
- **Recommandation d'Observation VNCS**, noté **ObsRcVNCS**, qui contient les politiques sociales en état `violated` :

$$
\begin{aligned}
{}_{\mathtt{pu}}\mathbf{ObsRcVNCS}(\mathtt{tg}, \alpha, \delta, \mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in \mathcal{SP}(t)/\mathbf{sp}.\mathtt{st} = \mathtt{violated} \wedge \\
\alpha \in \mathtt{pu.facets}(\mathbf{sp}.\mathtt{cont}) \wedge \delta \in \mathtt{pu.dimensions}(\mathbf{sp}) \wedge \\
\mathtt{refers}(\mathbf{sp}.\mathtt{cont}, {}_{\mathtt{pu}}\mathtt{TObsRc}(\mathtt{tg}, \alpha, \mathtt{t}))\}
\end{aligned}
$$

– **Recommandation d'Observation CNCS**, noté **ObsRcCNCS**, qui contient les politiques sociales en état `cancelled` :

$$_{\mathtt{pu}}\mathbf{ObsRcCNCS}(\mathtt{tg},\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in \mathcal{SP}(t)/\mathbf{sp}.\mathtt{st} = \mathtt{cancelled} \wedge \\ \alpha \in \mathtt{pu.facets}(\mathbf{sp}.\mathtt{cont}) \wedge \delta \in \mathtt{pu.dimensions}(\mathbf{sp}) \wedge \\ \mathtt{refers}(\mathbf{sp}.\mathtt{cont}, {}_{\mathtt{pu}}\mathtt{TObsRc}(\mathtt{tg},\alpha,\mathtt{t}))\}$$

Le niveau de la Réputation fondée sur les Recommandations d'Observations de l'agent `pu` en l'agent `tg` pour la facette α et la dimension δ à l'instant `t` peut alors être calculé comme suit :

$$\mathtt{ObsRcbRp}^{\mathtt{tg}}_{\mathtt{pu}}(\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \frac{\sum_{\mathcal{X}\in\{F,V,C\}} \tau''_{\mathcal{X}} \times \mathtt{Imp}({}_{\mathtt{pu}}\mathbf{ObsRc}\mathcal{X}\mathbf{NCS}(\mathtt{tg},\alpha,\delta,\mathtt{t}))}{\sum_{\mathcal{X}\in\{F,V,C\}} |\tau''_{\mathcal{X}}| \times \mathtt{Imp}({}_{\mathtt{pu}}\mathbf{ObsRc}\mathcal{X}\mathbf{NCS}(\mathtt{tg},\alpha,\delta,\mathtt{t}))}$$

où ${}_{\mathtt{pu}}\mathbf{ObsRc}\mathcal{X}\mathbf{NCS}(\mathtt{tg},\alpha,\delta,\mathtt{t})$ avec $\mathcal{X} \in \{F,V,C\}$ abrège respectivement ${}_{\mathtt{pu}}\mathbf{ObsRcFNCS}(\mathtt{tg},\alpha,\delta,\mathtt{t})$, ${}_{\mathtt{pu}}\mathbf{ObsRcVNCS}(\mathtt{tg},\alpha,\delta,\mathtt{t})$ et ${}_{\mathtt{pu}}\mathbf{ObsRcCNCS}(\mathtt{tg},\alpha,\delta,\mathtt{t})$. Les paramètres $\tau''_{\mathcal{F}}$ et $\tau''_{\mathcal{V}}$ sont soumis aux mêmes contraintes que les paramètres $\tau_{\mathcal{F}}$ et $\tau_{\mathcal{V}}$, c'est-à-dire $\tau''_{\mathcal{F}} > 0$ et $\tau''_{\mathcal{V}} < 0$.

Réputation fondée sur les Recommandations d'Évaluations. La Réputation fondée sur les Recommandations d'Évaluations (abrégée `EvRcbRp`) se calcule à partir des recommandations portant sur des évaluations. L'ensemble de ces recommandations, noté `TEvRc` est défini comme suit :

$$_{\mathtt{pu}}\mathtt{TEvRc}(\mathtt{tg},\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{rc} \in {}_{\mathtt{pu}}\mathtt{TRc}(\mathtt{t})/ \\ \mathbf{sp} = \mathbf{rc}.\mathtt{cont} \wedge \mathbf{sp} \in {}_{\bullet}\mathbf{NCS}(\mathtt{t}) \wedge \mathbf{sp}.\mathtt{db} = \mathtt{tg} \wedge \mathbf{sp}.\mathtt{t_e} \leq \mathtt{t} \\ \wedge \alpha \in \mathtt{pu.facets}(\mathbf{sp}.\mathtt{cont}) \wedge \delta \in \mathtt{pu.dimensions}(\mathbf{sp})\}$$

Où ${}_{\bullet}\mathbf{NCS}(\mathtt{t}) = \bigcup_{\mathtt{ag}\in\Omega_{\mathtt{pu}}(\mathtt{t})} {}_{\mathtt{ag}}\mathbf{NCS}(\mathtt{t})$.

Les recommandations d'évaluation concernant une cible `tg` pour une facette α et une dimension δ sont des recommandations de confiance dont le contenu est une politique sociale dont `tg` est débiteur, qui permettent de juger la dimension δ et dont le contenu porte sur la facette α.

Le calcul du niveau de Réputation fondée sur les Recommandations d'Évaluations se déroule ensuite de manière similaire au calcul du niveau des

Réputation fondée sur les Interactions Directes, Réputation fondée sur les Interactions Indirectes et Réputation fondée sur les Recommandations d'Observations : l'ensemble de politiques sociales obtenu par recommandation est noté $\mathcal{SP}'(t)$ et peut alors être séparé en trois sous-ensembles en fonction de l'état des politiques sociales :

- **Recommandation d'Évaluation FNCS**, noté **EvRcFNCS**, qui contient les politiques sociales reçues par recommandation, en état `fulfilled` et concernant la facette α et la dimension δ :

$$_{\mathtt{pu}}\mathbf{EvRcFNCS}(\mathtt{tg},\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in \mathcal{SP}'(t)/\mathbf{sp}.\mathtt{st} = \mathtt{fulfilled} \wedge \alpha \in \mathtt{pu.facets}(\mathbf{sp}.\mathtt{cont}) \wedge \delta \in \mathtt{pu.dimensions}(\mathbf{sp})\}$$

- **Recommandation d'Évaluation VNCS**, noté **EvRcVNCS**, qui contient les politiques sociales en état `violated` :

$$_{\mathtt{pu}}\mathbf{EvRcVNCS}(\mathtt{tg},\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in \mathcal{SP}'(t)/\mathbf{sp}.\mathtt{st} = \mathtt{violated} \wedge \alpha \in \mathtt{pu.facets}(\mathbf{sp}.\mathtt{cont}) \wedge \delta \in \mathtt{pu.dimensions}(\mathbf{sp})\}$$

- **Recommandation d'Évaluation CNCS**, noté **EvRcCNCS**, qui contient les politiques sociales en état `cancelled` :

$$_{\mathtt{pu}}\mathbf{EvRcCNCS}(\mathtt{tg},\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{sp} \in \mathcal{SP}'(t)/\mathbf{sp}.\mathtt{st} = \mathtt{cancelled} \wedge \alpha \in \mathtt{pu.facets}(\mathbf{sp}.\mathtt{cont}) \wedge \delta \in \mathtt{pu.dimensions}(\mathbf{sp})\}$$

Le niveau de la Réputation fondée sur les Recommandations d'Évaluations de l'agent `pu` pour la cible `tg`, la facette α et la dimension δ à l'instant `t` peut alors être calculé comme suit :

$$\mathtt{EvRcbRp}^{\mathtt{tg}}_{\mathtt{pu}}(\alpha,\delta,\mathtt{t}) \stackrel{\text{def}}{=} \frac{\sum_{\mathcal{X}\in\{F,V,C\}} \tau'''_{\mathcal{X}} \times \mathtt{Imp}(_{\mathtt{pu}}\mathbf{EvRc}\mathcal{X}\mathbf{NCS}(\mathtt{tg},\alpha,\delta,\mathtt{t}))}{\sum_{\mathcal{X}\in\{F,V,C\}} |\tau'''_{\mathcal{X}}| \times \mathtt{Imp}(_{\mathtt{pu}}\mathbf{EvRc}\mathcal{X}\mathbf{NCS}(\mathtt{tg},\alpha,\delta,\mathtt{t}))}$$

Les paramètres $\tau'''_{\mathcal{F}}$ et $\tau'''_{\mathcal{V}}$ sont soumis aux mêmes contraintes que les paramètres $\tau_{\mathcal{F}}$ et $\tau_{\mathcal{V}}$, c'est-à-dire $\tau'''_{\mathcal{F}} > 0$ et $\tau'''_{\mathcal{V}} < 0$.

Réputation fondée sur les Recommandations de Réputation. La Réputation fondée sur les Recommandations de Réputation (abrégée RpRcbRp) se calcule à partir des recommandations portant sur des niveaux de réputation. L'ensemble de ces recommandations, noté TRpRc est défini comme suit :

$$
\begin{aligned}
{}_{\mathtt{pu}}\mathtt{TRpRc}(\mathtt{tg}, \alpha, \delta, \mathtt{t}) \stackrel{\text{def}}{=} \{\mathbf{rc} \in {}_{\mathtt{pu}}\mathtt{TRc}(\mathtt{t}) / \\
\mathbf{rc}.\mathtt{cont} = \mathtt{PrRp}(\mathtt{x}, \mathtt{tg}, \alpha, \delta, \mathtt{t}', \mathtt{level}) \wedge \mathtt{t}' \leq \mathtt{t}\}
\end{aligned}
$$

où $\mathtt{PrRp}(\mathtt{x}, \mathtt{tg}, \alpha, \delta, \mathtt{t}', \mathtt{level})$ est la réputation calculée par l'agent x (punisseur) à propos de l'agent tg (cible) à l'instant t′ (précédant t) pour la facette α selon la dimension δ et dont le niveau communiqué est $\mathtt{level} \in [-1, +1] \cup \mathtt{unknown}$. Afin d'alléger les écritures, nous considérons dans la suite que le calculateur du niveau de réputation ainsi que le niveau lui-même peuvent être accédés comme suit : si **rc** est une recommandation de réputation, alors $\mathfrak{pu} = \mathbf{rc}.\mathtt{cont.pu}$ est le punisseur qui a calculé le niveau de réputation et $\mathfrak{lev} = \mathbf{rc}.\mathtt{cont.level}$ est le niveau de réputation communiqué.

Le niveau de la Réputation fondée sur les Recommandations de Réputation de l'agent pu en l'agent tg pour la facette α et la dimension δ à l'instant t peut alors être calculé comme suit (où $\alpha' = \texttt{"recommend"}$ et $\delta' = \mathtt{competence}$) :

$$
\mathtt{RpRcbRp}^{\mathtt{tg}}_{\mathtt{pu}}(\alpha, \delta, \mathtt{t}) \stackrel{\text{def}}{=}
$$

$$
\frac{\displaystyle\sum_{\mathbf{rc} \in {}_{\mathtt{pu}}\mathtt{TRpRc}(\mathtt{tg}, \alpha, \delta, \mathtt{t})} \mathfrak{lev} \times \mathtt{pu.reasons}(\mathfrak{pu}, \alpha', \delta', \mathtt{RcLev}, \mathtt{t}).\mathtt{trust\_val}}{\displaystyle\sum_{\mathbf{rc} \in {}_{\mathtt{pu}}\mathtt{TRpRc}(\mathtt{tg}, \alpha, \delta, \mathtt{t})} \mathtt{pu.reasons}(\mathfrak{pu}, \alpha', \delta', \mathtt{RcLev}, \mathtt{t}).\mathtt{trust\_val}}
$$

La Réputation fondée sur les Recommandations de Réputation est donc calculée par la moyenne pondérée des valeurs de réputation fournies dans les recommandations de confiance. Les pondérations sont les intentions de confiance accordées aux calculateurs des niveaux de réputation (punisseurs) fournis pour la facette α et la dimension δ. Ces valeurs sont considérées positives.

Ainsi, le calcul de la RpRcbRp fait intervenir un double filtrage. Tout d'abord, la réputation du propagateur est utilisée au cours d'un processus de raisonnement pour décider si la recommandation est de confiance ou non. La dimension utilisée lors de ce filtrage est la dimension d'intégrité (integrity).

Ensuite et pour les recommandations de confiance seulement, la réputation de l'agent qui a calculé le niveau communiqué est utilisée pour pondérer le poids de la recommandation. Cette fois-ci, les réputations selon la dimension de compétence (`competence`) sont utilisées. Ce double filtrage permet de prendre en compte le fait que l'agent qui a calculé le niveau de réputation (qui est donc un punisseur) et l'agent qui a communiqué ce niveau à l'agent `pu` (qui est donc un propagateur) peuvent être différents et avoir des comportements différemment malfaisant.

6.3.4 Raisonnement

Le processus de raisonnement consiste, pour un bénéficiaire `bn`, à dériver les conséquences des niveaux des réputations qu'il associe à une cible `tg`. Le processus `reasons` fonctionne par seuillage, comme suggéré dans [Luh79, Gam00a, FC99]. Ce processus prend en entrée une cible, une facette, une dimension et un ensemble de seuils de raisonnement. Les seuils sont des valeurs réelles positives notées $\theta^{\text{trust}}_{\mathcal{X}\texttt{bRp}}$, $\theta^{\text{distrust}}_{\mathcal{X}\texttt{bRp}}$, $\theta^{\text{relevance}}_{\mathcal{X}\texttt{bRp}}$, où $\mathcal{X} \in \{$ `DI`, `II`, `ObsRc`, `EvRc`, `RpRc` $\}$. Ils sont regroupés dans la structure notée `Lev`. Un exemple de tels seuils a été donné dans le cadre de la recomandation, page 121. Le processus de raisonnement renvoie une intention de confiance décomposée en deux parties : d'une part, une partie booléenne établissant si l'intention est de faire confiance ou non et, d'autre part, une partie réelle représentant la force de l'intention de confiance. Ces composantes sont notées respectivement `trust_int` et `trust_val`.

Les agents maintiennent différents types de réputation en parallèle : `DIbRp`, `IIbRp`, `ObsRcbRp`, `EvRcbRp` et `RpRcbRp`. L'intégration de ces différents types de réputation dans un processus de raisonnement n'est pas trivial puisque l'agent doit choisir quel type de réputation utiliser dans la situation de raisonnement considérée. De plus, le type de réputation qui serait le plus approprié à un instant donné peut être associé à une valeur qui n'est actuellement pas pertinente, par exemple si le bénéficiaire n'a pas assez d'information sur la cible (qu'il s'agisse d'interactions directes, indirectes ou de recommandations).

Afin de mener à bien le processus de raisonnement, les différents types de réputation sont ordonnés. L'ordre n'est pas fixé : chaque agent peut classer les différents types de réputation suivant ses propres préférences. Par exemple, un agent peut classer les Réputation fondée sur les Recommandations après la Réputation fondée sur les Interactions Directes et la Réputation fondée

sur les Interactions Indirectes s'il estime que de l'information extérieure est moins fiables que ses propres observations. Au contraire, il peut les placer avant s'il considère que les autres agents sont plus compétents que lui dans la situation de raisonnement considérée.

À titre d'illustration, nous considérons dans cette section que le bénéficiaire utilise une notion de fiabilité s'appuyant sur le niveau d'interprétation et de contrôle qu'il a sur l'information pour déterminer l'ordre des différents types de réputation. Ainsi, en s'appuyant sur niveau d'interprétation, il classe les différents types de Réputation fondée sur les Recommandations comme suit (par ordre croissant de fiabilité) : `ObsRcbRp`, `EvRcbRp` puis `RpRcbRp`. L'agent considère une observation comme une information moins interprétée qu'une évaluation, qui est elle-même considérée comme moins interprétée qu'un niveau de réputation. En s'appuyant sur le contrôle qu'il a sur l'information, le bénéficiaire classe les types de réputations comme suit (par ordre croissant de fiabilité) : `DIbRp`, `IIbRp`, puis les différentes `RcbRp`. Le bénéficiaire considère plus fiable une information tirée d'une interaction à laquelle il a participé qu'une information tirée d'une écoute flottante, puisque ses capteurs peuvent être défaillants ou qu'il peut perdre le contact avec une partie de son réseau d'accointances. D'autre part, il considère une interaction directe ou indirecte plus fiable qu'une recommandation puisque les propagateurs peuvent mentir et / ou juger les autres sur des critères très différents de ceux qu'il utilise. Finalement, l'ordre que nous considérons ici, à titre d'illustration, est le suivant (par ordre croissant de fiabilité) : `DIbRp`, `IIbRp`, `ObsRcbRp`, `EvRcbRp`, puis `RpRcbRp`.

La figure 6.6 schématise le processus de raisonnement noté $\texttt{bn.reasons}(\texttt{tg}, \alpha, \delta, \texttt{Lev}, \texttt{t})$. Les différents types de réputation affichés dans la figure sont fixés à l'instant du raisonnement, pour une facette α et une dimension δ données et ont pour cible l'agent `tg`. Ainsi `DIbRp` dans la figure représente $\texttt{DIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t})$.

Le processus se déroule comme suit : l'agent `pu` essaie d'abord d'utiliser la valeur de réputation qu'il considère la plus fiable (`DIbRp` sur la figure 6.6). Ce type de réputation peut être suffisant pour décider d'avoir l'intention de faire (resp. ne pas faire) confiance à la cible. Si la valeur associée à la `DIbRp` est plus grande que le seuil $\texttt{Lev}.\theta^{\text{trust}}_{\texttt{DIbRp}}$, alors l'agent décide d'avoir l'intention de faire confiance. À l'opposé, si la valeur est plus petite que le seuil $\texttt{Lev}.\theta^{\text{distrust}}_{\texttt{DIbRp}}$, alors l'agent décide d'avoir l'intention de ne pas faire confiance à la cible. Si la `DIbRp` est en état `unknown` ou si elle n'est pas discriminante (entre les

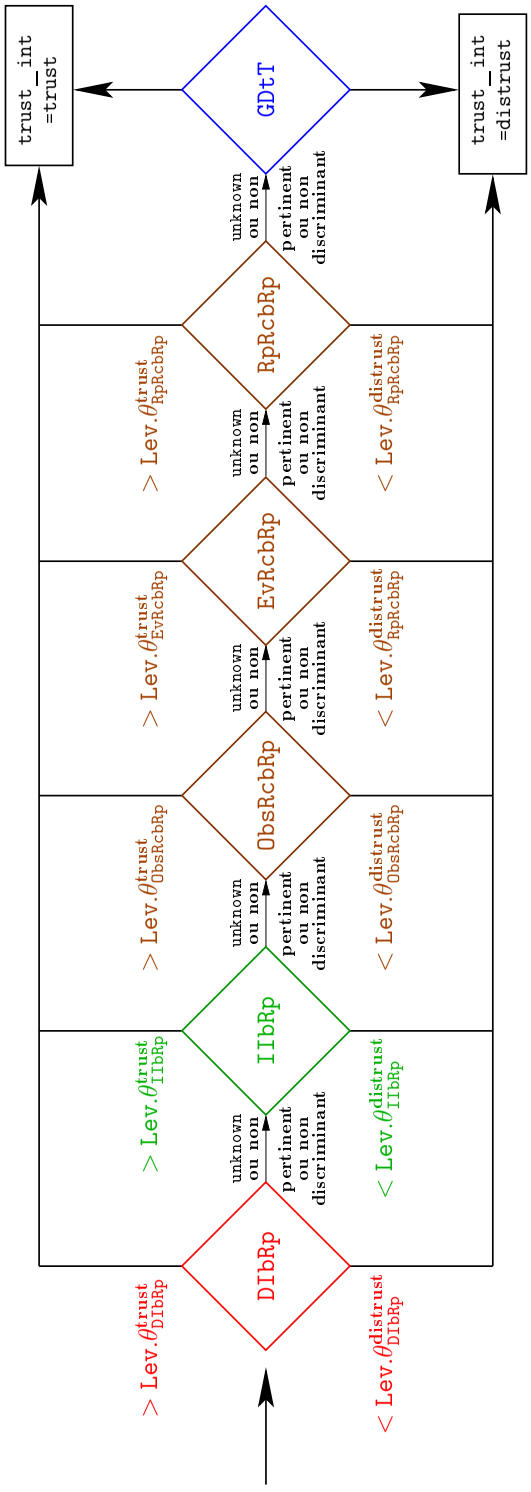


FIG. 6.6 – Processus de raisonnement.

deux seuils $\texttt{Lev}.\theta^{\text{trust}}_{\texttt{DIbRp}}$ et $\texttt{Lev}.\theta^{\text{distrust}}_{\texttt{DIbRp}}$) ou si elle n'est pas pertinente (pas assez d'interactions directes), la DIbRp n'est pas suffisante pour fixer si l'intention est de faire confiance ou non.

Un processus similaire est alors utilisé sur le prochain type de réputation (IIbRp sur la figure). La valeur est comparée à deux seuils $\texttt{Lev}.\theta^{\text{trust}}_{\texttt{IIbRp}}$ et $\texttt{Lev}.\theta^{\text{distrust}}_{\texttt{IIbRp}}$. Si la valeur ne permet pas d'aboutir à une intention de confiance, l'agent utilise les types de réputation suivants (ObsRcbRp, EvRcbRp puis RpRcbRp sur la figure) avec les seuils correspondants ($\texttt{Lev}.\theta^{\text{trust}}_{\texttt{ObsRcbRp}}$ et $\texttt{Lev}.\theta^{\text{distrust}}_{\texttt{ObsRcbRp}}$, $\texttt{Lev}.\theta^{\text{trust}}_{\texttt{EvRcbRp}}$ et $\texttt{Lev}.\theta^{\text{distrust}}_{\texttt{EvRcbRp}}$, $\texttt{Lev}.\theta^{\text{trust}}_{\texttt{RpRcbRp}}$ et $\texttt{Lev}.\theta^{\text{distrust}}_{\texttt{RpRcbRp}}$). Finalement, si aucune des valeurs précédentes n'a permis d'aboutir à une intention de confiance, la Prédisposition Générale à faire Confiance est utilisée.

La Prédisposition Générale à faire Confiance (abrégée GDtT) est une sorte de réputation par défaut. Elle n'est pas attachée à une cible particulière et représente l'inclination du bénéficiaire à accorder sa confiance à un autre agent quand il n'a pas d'information à son propos. Il ne s'agit pas d'une réputation interpersonnelle car elle n'est pas ciblée sur un agent particulier. Elle n'est pas étudiée en détails ici, car nous la considérons associée à une valeur fixe.

La pertinence de la valeur de DIbRp (resp. IIbRp, ObsRcbRp, EvRcbRp et RpRcbRp) est établie à l'aide d'un seuil $\texttt{Lev}.\theta^{\text{relevance}}_{\texttt{DIbRp}} \in [0, +\infty[$ (resp. $\texttt{Lev}.\theta^{\text{relevance}}_{\texttt{IIbRp}}$, $\texttt{Lev}.\theta^{\text{relevance}}_{\texttt{ObsRcbRp}}$, $\texttt{Lev}.\theta^{\text{relevance}}_{\texttt{EvRcbRp}}$ et $\texttt{Lev}.\theta^{\text{relevance}}_{\texttt{RpRcbRp}}$) qui représente le nombre d'interactions directes (resp. interactions indirectes et différents types de recommandations) à partir duquel l'agent considère que sa valeur de réputation est pertinente. À une pertinence nulle correspond une valeur de réputation unknown.

La force de l'intention de confiance correspond à la valeur de réputation qui a permis de fixer l'intention de confiance. Par exemple, si la DIbRp est suffisante pour fixer l'intention de confiance, alors trust_val est égale à la valeur de DIbRp. Les seuils de raisonnement étant des valeurs positives, la force de l'intention est elle-même une valeur positive.

À la fin de ce mécanisme, l'agent a établi une intention de faire confiance pour une facette et une dimension données. Le mécanisme présenté ci-dessus peut être itéré pour toutes les combinaisons possibles de facettes et de dimensions que l'agent pu jugera pertinentes par rapport au contexte de raisonnement. Le bénéficiaire bn génère ainsi un ensemble d'intentions de confiance.

Le processus $\texttt{bn.reasons}(\texttt{tg}, \alpha, \delta, \texttt{Lev}, \texttt{t})$ est écrit sous la forme d'un algorithme en annexe B.1).

6.3.5 Décision

Le processus de décision consiste, pour un bénéficiaire `bn`, à prendre des décisions d'agir en confiance ou non dans un contexte donné. Ce processus prend en entrée une cible, un contexte et un instant. En sortie, les états mentaux courants du bénéficiaire sont modifiés en fonction des décisions qu'il prend d'agir en confiance ou non avec la ou les cibles.

Afin de prendre une décision, un agent peut s'appuyer sur différentes informations : intentions de confiance issues du processus de raisonnement, connaissance sur la disponibilité des partenaires potentiels, émotions... Dans le cadre de cette thèse, nous considérons qu'il ne s'appuie que sur des intentions de confiance issues du processus de raisonnement. Le bénéficiaire `bn` peut alors prendre deux types de décisions :

sélection *décider si un agent donné est digne de confiance.* Par exemple, être en mesure de décider si un agent donné `tg` (cible) est digne de se voir révéler une information sensible ou de se faire déléguer une tâche importante.

tri *ordonner un ensemble d'agents en fonction de la confiance qui leur est accordée.* Par exemple, choisir, parmi un ensemble de m agents, un sous-ensemble des $n \leq m$ agents les plus dignes de confiance, à qui il serait possible d'envoyer de l'information sensible ou bien avec qui il serait possible de coopérer.

Dans le cas d'une décision de type sélection, l'agent `bn` peut simplement tester si la partie booléenne de l'intention de confiance qu'il a envers la cible penche dans le sens de lui faire confiance ou non. Dans le cas positif, l'agent `bn` change ses états mentaux de façon à envoyer le message ou à déléguer la tâche à `tg`. Dans le cas négatif, `bn` change ses états mentaux de façon à ne pas envoyer le message ou déléguer la tâche à `tg`.

Dans le cas d'une décision de type tri, l'agent utilisera plutôt la composante réelle des intentions de confiance afin de trier les agents. Il modifiera ensuite ses états mentaux de façon à réaliser l'action souhaitée (envoyer un message ou déléguer un tâche) avec les agents de confiance, c'est-à-dire les agents pour lesquels l'intention est de faire confiance et qui ont les pondérations les plus fortes.

L'annexe B.2, page 192 propose des implémentations des processus de décision de type sélection et tri.

6.3.6 Propagation

Le processus de propagation est exécuté par un propagateur et consiste à choisir pourquoi, quand, comment et à qui diffuser des recommandations.

Nous considérons dans cette thèse que, quand une information est correcte, le fait de la partager profite à toute la société d'agents, puisqu'elle renforce l'effet du contrôle social. De ce fait, les agents ont intérêt à partager l'information le plus souvent possible. Les différents types d'information que les agents peuvent partager sont : des observations (engagements sociaux), des évaluations (politiques sociales) et des niveaux de réputation.

Les agents peuvent échanger ces recommandations de deux façons différentes : par une approche de type « push », où un propagateur envoie spontanément à une partie de son réseau d'accointances des recommandations et une approche « pull », où un propagateur reçoit des requêtes pour envoyer des recommandations. Dans le cadre de ces deux processus, le propagateur doit décider si et à qui envoyer des recommandations. Pour ce faire, il s'appuie sur son modèle de réputation et utilise son processus de décision. Dans le cas du processus de type « push », une décision de type tri est utilisée afin de sélectionner un ensemble d'agents. Dans le cas du processus de type « pull », il s'agit plutôt d'une décision de type sélection, où le propagateur choisit de répondre ou non à la requête.

Au cours du processus de décision visant à déterminer si et à qui envoyer des recommandations, le propagateur s'appuie sur un principe de réciprocité qui consiste à diffuser des recommandations aux agents qui sont eux-mêmes de « bons » recommandeurs. Pour sélectionner les « bons » recommandeurs, le propagateur a recours à une décision ciblant la facette `"recommend"` et les dimensions `competence` et `integrity`. Il sélectionne donc les agents compétents et intègres dans leurs propres recommandations.

6.4 Conclusion

Dans ce chapitre, nous avons proposé un modèle de réputation qui permet à des agents de maintenir, en parallèle, différents types de réputation, mais aussi de prendre des décisions en s'appuyant sur le niveau de ces réputations.

Dans un premier temps, nous avons étudié les rôles que jouent les différents agents au cours du processus de punition ainsi que les types d'information échangés par les agents jouant ces rôles. Ceci nous a permis de définir

les différents types de réputation que le modèle utilise.

Nous avons ensuite défini formellement ces types de réputation fondée sur les interactions, en fonction de leurs propriétés de subjectivité, multi-facette, multi-dimension, dynamisme et transitivité, grâce aux rôles et aux types de données identifiés précédemment. Nous avons alors choisi une représentation computationnelle adaptée.

Enfin, nous avons proposé des implémentations pour les processus d'initialisation, d'évaluation, de punition, de raisonnement, de décision et de propagation. Le processus d'évaluation que nous utilisons ici est en fait le processus de détection de violation de normes que nous avons défini dans le chapitre précédent. Le processus de punition des Réputation fondée sur les Recommandations s'appuie sur le processus de raisonnement pour sélectionner les recommandations de confiance et les pondérer. Le processus de propagation s'appuie sur le processus de raisonnement pour décider à quel agents envoyer des recommandations.

Ainsi, n'importe quel agent implémentant ce modèle de réputation est en mesure d'estimer, de manière totalement autonome, la réputation d'une cible en fonction des engagements sociaux qu'elle a pris et des normes qui les régissent.

Conclusions au modèle

Dans cette partie, nous avons présenté le modèle L.I.A.R. Celui-ci permet à tout agent qui l'implémente de modéliser les interactions qu'il perçoit, de les confronter à des normes dont il a connaissance et de sanctionner les interactions perçues en fonction de leur respect ou non des normes.

Dans le chapitre 5, nous avons proposé un modèle d'engagement social qui permet aux agents de modéliser les interactions qu'ils perçoivent en faisant un minimum d'hypothèses sur l'implémentation interne des autres agents. Ce modèle est donc particulièrement adapté aux systèmes ouverts et décentralisés, dans lesquels des agents hétérogènes peuvent interagir. Nous avons ensuite proposé un modèle de norme qui permet de définir les règles de comportement. Ces règles sont décrites en des termes généraux et omniscients sur le système. De manière à ce que n'importe quel agent puisse participer au contrôle social, nous avons proposé un moyen pour les agents d'instancier ces normes dans leur point de vue local. Pour ce faire, nous avons défini un modèle de politiques sociales. Enfin, nous avons proposé un processus de détection de la violation des normes que n'importe quel agent peut lancer et qui prend en compte le fait que les agents n'ont que des perceptions partielles du système dans lequel ils évoluent.

Nous avons ensuite, dans le chapitre 6, défini un modèle de réputation qui permet aux agents de sanctionner le respect ou la violation des normes. Ce modèle définit différents types de réputation fondée sur les interactions, définis formellement en fonction des rôles que les agents peuvent jouer au cours du processus de punition, du type d'information qu'ils peuvent échanger et de leurs propriétés. En s'appuyant sur la détection de violation des normes présentée dans le chapitre précédent, les agents qui implémentent ce modèle de réputation sont en mesure de punir les autres agents par des hausses ou des baisses de niveaux de réputation. Les niveaux des réputations sont ensuite utilisés par les agents pour prendre des décisions concernant le fait d'interagir

en confiance ou non avec les autres.

En permettant aux agents eux-mêmes de modéliser et de sanctionner les interactions qu'ils perçoivent, et en ne nécessitant pas de pouvoirs particuliers de certains agents sur les autres, le modèle L.I.A.R. contribue au contrôle social des systèmes multi-agents ouverts et décentralisés. De plus, il préserve l'autonomie des agents vis-à-vis de l'intervention humaine, du fait qu'il est totalement automatisé.

Troisième partie

Application

Chapitre 7

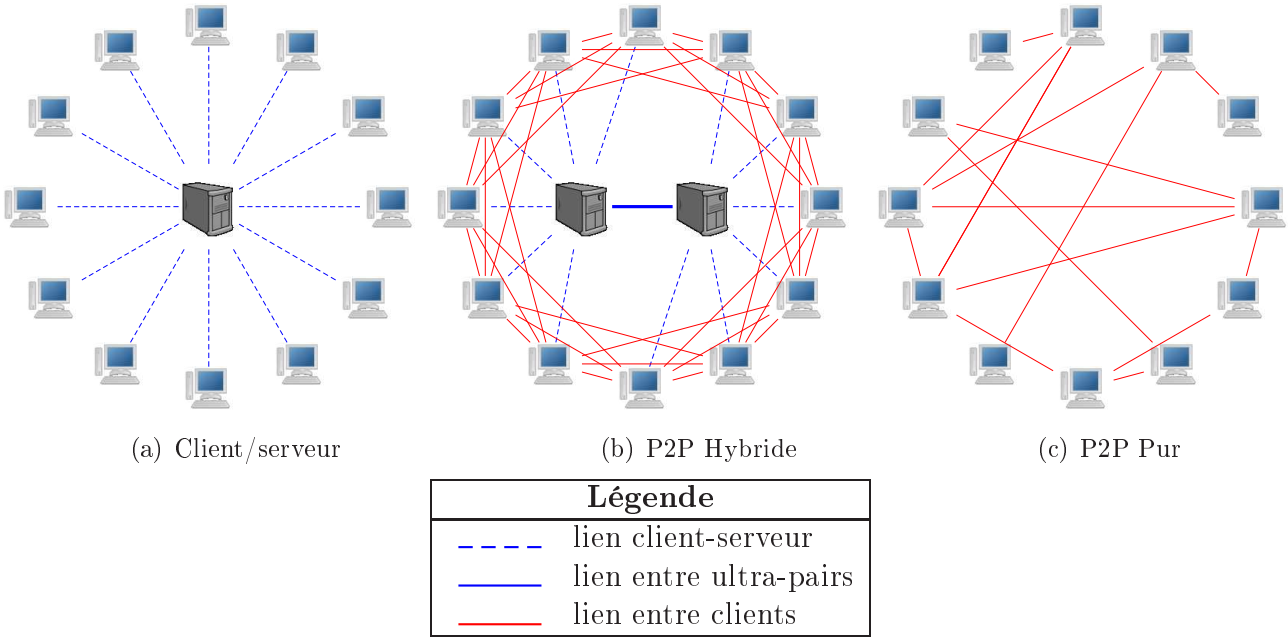
Régulation des communications dans un réseau pair-à-pair

Dans ce chapitre, nous considérons un scénario d'échange d'informations dans un réseau pair-à-pair, où les pairs sont doublés d'agents implémentant le modèle L.I.A.R., afin de contrôler les interactions des autres agents. Après différentes expérimentations, nous discutons brièvement les résultats.

Nous motivons tout d'abord l'utilisation du modèle L.I.A.R. dans le cadre de réseaux pair-à-pair « purs », puis nous présentons le scénario qui sert de trame aux expérimentations. Nous décrivons ensuite les normes qui permettent de réguler les engagements sociaux que prennent les agents dans ce scénario. Les différents comportements que peuvent avoir les agents sont ensuite précisés. Enfin, nous exposons une grille d'expérimentation et présentons les résultats de l'évaluation du modèle L.I.A.R. selon cette grille.

7.1 Motivations

Il existe différents types de réseaux de communication entre ordinateurs. L'architecture « client / serveur », présentée dans la figure 7.1(a) est très asymétrique. D'un côté, des serveurs proposent des ressources et, de l'autre côté, les clients émettent des requêtes pour ces ressources. Les clients ne peuvent pas proposer de ressources et les serveurs ne peuvent pas en requérir. L'architecture « client / serveur » passe mal à l'échelle du fait de sa trop forte centralisation. Pour pallier ces problèmes, des architectures dites « pair-à-pair » (P2P [AT03, Cla04]) ont été proposées.



(a) Client/serveur (b) P2P Hybride (c) P2P Pur

FIG. 7.1 – Différentes architectures de réseau.

Dans l'architecture « P2P hybride », présentée dans la figure 7.1(b), les rôles de client et de serveur sont moins différenciés. Les « pairs simples » peuvent communiquer entre eux et s'échanger des ressources. Les « ultra-pairs » assurent de plus les tâches collectives du système : mise en relation des pairs, gestion de l'ouverture, transmission des messages... Cette architecture est plus robuste que l'architecture client / serveur, puisque l'indisponibilité d'un ultra-pair a un impact plus limité. Cependant, les ultra-pairs constituent toujours les maillons faibles du réseau.

L'architecture « P2P pur », présentée dans la figure 7.1(c), est parfaitement symétrique : chaque pair joue à la fois le rôle de client et de serveur. Il est prévu dès la conception que tout pair puisse entrer ou sortir du réseau à tout instant. En contrepartie, tous les pairs doivent participer aux tâches collectives du système. Cette architecture supporte mal l'introduction de pairs mal configurés ou malveillants, particulièrement s'ils ne remplissent pas correctement la partie des tâches collectives qui leur incombe. Nous proposons, dans ce chapitre, d'utiliser le modèle L.I.A.R. dans de tels réseaux afin de contrôler les communications des pairs.

7.2 Scénario

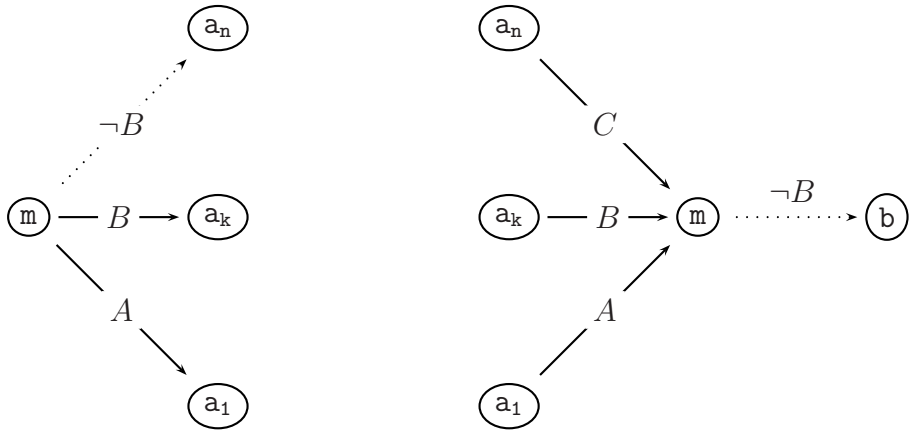
Pour illustrer les capacités de contrôle des interactions du modèle L.I.A.R., nous avons retenu un scénario d'échange d'informations, c'est-à-dire un scénario dont l'essence même est un type particulier d'interaction : la communication. Ce scénario est le suivant : dans un réseau P2P pur, des pairs échangent de l'information sur les horaires de passage des films dans différentes salles de cinéma. Un agent est associé à chaque pair et dispose d'horaires complets ou partiels pour certaines salles de cinéma. Chaque agent peut tout aussi bien requérir que fournir des horaires aux autres agents.

Le réseau P2P pur qui sert de substrat à ce scénario est de type GNUtella 1 [Gnu00]. Dans un tel réseau, les requêtes sont diffusées par inondation. Les messages peuvent donc être observés non seulement directement par leurs émetteurs et leurs récepteurs, mais aussi indirectement par tous les agents qui se trouvent sur le chemin de l'un à l'autre. De ce fait, de nombreux agents perçoivent les communications par écoute flottante.

Tous les agents qui implémentent le modèle L.I.A.R. modélisent les interactions qu'ils perçoivent par des engagements sociaux. Par exemple, un agent qui implémente le modèle L.I.A.R. considérera qu'un agent qui répond à une requête s'engage implicitement et publiquement vis-à-vis de son interlocuteur. Il estimera alors qu'il existe un engagement social portant sur le contenu de la réponse, dont l'émetteur de la réponse est le débiteur et dont l'émetteur de la requête le créditeur. Cet engagement restera en état `active` tant qu'il n'aura pas explicitement été annulé et passera en état `cancelled` dès qu'il aura été annulé.

7.3 Normes de régulation

Dans le cadre du scénario d'échange d'horaires de salles de cinéma défini précédemment, le type de comportements que nous cherchons à réprimer est la contradiction. En effet, les contradictions sont souvent révélatrices de comportements malveillants, comme les mensonges. Dans cette section, nous définissons deux types de contradictions et les normes qui permettent de les interdire.



(a) Contradiction en émission. (b) Contradiction en transmission.

FIG. 7.2 – Contradictions en émission et en transmission.

7.3.1 Contradictions

La figure 7.2 décrit les deux types de contradictions que nous cherchons à interdire : les contradictions en émission et en transmission. Dans la situation 7.2(a), l'agent $\mathtt{m}$ est engagé envers un agent $\mathtt{a_1}$ sur A et envers un agent $\mathtt{a_k}$ sur B. Il s'engage alors sur $\neg B$ envers l'agent $\mathtt{a_n}$ (flèche pointillée). Ce dernier engagement social crée alors une inconsistance avec l'ensemble des engagements qu'avait déjà l'agent $\mathtt{m}$. L'agent $\mathtt{m}$ se contredit. Une telle contradiction apparaît, par exemple, si un agent s'engage sur le fait qu'un cinéma joue deux films différents à la même heure dans la même salle. Nous appelons ce type de contradiction « la contradiction en émission » :

Définition 7.3.1 *Une **contradiction en émission** a lieu quand un agent est débiteur d'engagements sociaux inconsistants.*

Dans la situation 7.2(b), l'agent $\mathtt{m}$ est créditeur d'engagements sociaux en état positif (`active`, `fulfilled`) portant sur les faits A, B et C, venant des agents $\mathtt{a_1}, \mathtt{a_k}, \mathtt{a_n}$. Il prend alors un engagement social portant sur un contenu $\neg B$ envers l'agent $\mathtt{b}$ (flèche pointillée dans la figure). Ce dernier engagement social crée une inconsistance. L'agent contredit des engagements qu'il n'a pas explicitement rejetés. Une telle contradiction apparaît, par exemple, si

un agent s'engage sur le fait qu'un cinéma joue un certain film dans une certaine salle à une certaine heure, alors qu'un autre agent est déjà engagé auprès de lui sur le fait qu'un autre film est joué dans cette même salle à cette même heure. Nous appelons ce type de contradiction « la contradiction en transmission » :

Définition 7.3.2 *Une* ***contradiction en transmission*** *a lieu quand un agent est débiteur d'un engagement social qui engendre une inconsistance avec des engagements sociaux dont il est créditeur et qui sont en état positif.*

7.3.2 Normes

Afin d'expliciter les règles interdisant les contradictions en émission et en transmission, nous définissons les normes suivantes. Ces normes interdisent des situations où des engagements sociaux créent des contradictions, tout en laissant les agents libres de sanctionner les violations qu'ils détectent comme ils l'entendent.

Norme interdisant la contradiction en émission

Les agents doivent respecter la première norme suivante, qui interdit les contradictions en émission :

$$\forall \mathtt{t} \in \mathcal{T}, \mathtt{snorm}(\mathtt{I}, \Omega(\mathtt{t}), \Omega(\mathtt{t}), \Omega(\mathtt{t}), \exists \mathtt{x} \in \Omega(\mathtt{t}) / \mathtt{cont\_emission}(\mathtt{t}, \mathtt{x}), \mathtt{active})$$

Le prédicat $\mathtt{cont\_emission}(\mathtt{t}, \mathtt{x})$ exprime le fait que l'agent $\mathtt{x}$ provoque une contradiction en émission à l'instant $\mathtt{t}$. Ce prédicat s'écrit comme suit (où $\mathbf{CCS}^{\star}_{\mathtt{x}}(\mathtt{t}) = \bigcup_{\mathtt{z} \in \Omega(\mathtt{t})} CS^{\mathtt{z}}_{\mathtt{x}}(\mathtt{t}))$) :

$$\mathtt{cont\_emission}(\mathtt{t}, \mathtt{x})$$
$$\equiv$$
$$\exists \mathtt{y} \in \Omega(\mathtt{t}), \exists \mathbf{c} \in \mathbf{CCS}^{\mathtt{y}}_{\mathtt{x}}(\mathtt{t}) / \mathtt{inconsistent}(\mathtt{t}, \mathbf{c}, \mathbf{CCS}^{\star}_{\mathtt{x}}(\mathtt{t}) \setminus \{\mathbf{c}\})$$

Cette formule exprime le fait que l'agent $\mathtt{x}$ est débiteur d'un engagement social qui crée une inconsistance avec l'ensemble des autres engagements

sociaux dont il est débiteur. Cette formule traduit donc la contradiction en émission.

Notons, par ailleurs, que cette norme n'empêche pas les agents de changer d'avis. Elle les contraint seulement à annuler des engagements sociaux dont les contenus pourraient s'avérer inconsistants avec le contenu d'un nouvel engagement social qu'ils souhaitent prendre. Pour prouver qu'il ne s'est pas contredit, un agent devra, au cours d'un éventuel processus de justification (*cf.* section 5.3), fournir un message digitalement signé attestant qu'il a annulé l'un quelconque des engagements sociaux impliqués dans l'inconsistance.

Norme interdisant la contradiction en transmission

La norme suivante interdit les contradictions en transmission :

$$\forall \mathtt{t} \in \mathcal{T}, \mathtt{snorm}(\mathtt{I}, \Omega(\mathtt{t}), \Omega(\mathtt{t}), \Omega(\mathtt{t}),$$
$$\exists \mathtt{x} \in \Omega(\mathtt{t}) / \mathtt{cont\_transmission}(\mathtt{t}, \mathtt{x}), \mathtt{active})$$

Le prédicat $\mathtt{cont\_transmission}(\mathtt{t}, \mathtt{x})$ exprime le fait que l'agent $\mathtt{x}$ provoque une contradiction en transmission à l'instant $\mathtt{t}$. Ce prédicat s'écrit (où $\mathbf{CCS}^{\mathtt{x}}_{\star}(\mathtt{t}) = \bigcup_{\mathtt{z} \in \Omega(\mathtt{t})} CS^{\mathtt{x}}_{\mathtt{z}}(\mathtt{t})$) :

$$\mathtt{cont\_transmission}(\mathtt{t}, \mathtt{x})$$
$$\equiv$$
$$\exists \mathtt{y} \in \Omega(\mathtt{t}), \exists \mathbf{c} \in \mathbf{CCS}^{\mathtt{y}}_{\mathtt{x}}(\mathtt{t}) / \mathtt{inconsistent}(\mathtt{t}, \mathbf{c}, \mathbf{CCS}^{\mathtt{x}}_{\star}(\mathtt{t}))$$

Cette formule exprime le fait que l'agent $\mathtt{x}$ prend un engagement social $\mathbf{c}$ qui provoque une inconsistance avec des engagements sociaux qui ont été pris précédemment envers lui.

Pour ne pas être pris en flagrant délit de contradiction en transmission, l'agent doit, avant de s'engager à nouveau, annuler au moins l'un des engagements sociaux dont il est créditeur.

7.4 Comportement des agents

Dans le cadre du scénario défini précédemment, les agents remplissent deux fonctions : ils partagent des informations et jugent les autres agents

en fonction des informations qu'ils partagent. Les normes interviennent dans les deux cas : d'une part, chaque agent peut être lui-même la source de violations en fonction des engagements qu'il prend ; d'autre part, les agents doivent instancier les normes pour détecter les violations des autres. Dans cette section, nous présentons les différents comportements que peuvent avoir les agents en tant que générateurs de violations et en tant que détecteurs de violation.

7.4.1 Comportement en tant que générateur de violations

Les agents prennent deux types d'engagements sociaux : des engagements sur les horaires de salles de cinéma et des recommandations. Les premiers sont liés à la la facette `"theater showtimes"` et les seconds à la facette `"recommend"`. Dans les deux cas, les agents peuvent générer des contradictions. Quand un agent décide de prendre un nouvel engagement social, il est possible que ce nouvel engagement social crée une inconsistance avec de précédents engagements sociaux dont il est débiteur ou créditeur. Afin de ne pas se faire sanctionner par les autres agents, il devrait annuler tous ces engagements sociaux inconsistants. Afin d'évaluer le modèle L.I.A.R., nous considérons que tous les agents n'agissent pas ainsi.

Un agent est caractérisé par deux paramètres : `violationRate` et `lieRate`. Le premier paramètre correspond aux contradictions qu'il va générer pour la facette `"theater showtimes"`, le second pour la facette `"recommend"`. Ainsi, le paramètre `violationRate` définit le taux d'engagements sociaux inconsistants, portant sur la facette `"theater showtimes"`, que l'agent ne va pas annuler. Le paramètre `lieRate`, correspond au taux d'engagements sociaux inconsistants, portant sur la facette `"recommend"`, qu'un agent ne va pas annuler. Dans ce dernier cas, les inconsistances sont générées par l'envoi de recommandations mensongères : l'agent envoie une valeur aléatoire dans $[-1, +1]$. Les paramètres `violationRate` et `lieRate` sont assignés aux agents comme suit : chaque agent $i \in 0, \ldots, N$ est associé à un `violationRate` et un `lieRate` de i/N.

7.4.2 Comportement en tant que détecteur de violation

Une des particularités du modèle L.I.A.R. est de fonctionner par l'instanciation des normes en politiques sociales. Différents agents peuvent employer

différentes stratégies lors de l'instanciation des normes.

Nous proposons ici deux stratégies d'instanciations : l'une rancunière et l'autre indulgente :

- Dans le cas des agents *rancuniers* (« rancorous » dans les figures de cette section), les normes sont instanciées une fois pour toutes. Cette stratégie s'implémente simplement en regardant, à chaque fois qu'un nouvel engagement social est observé, s'il crée une inconsistance avec de précédents engagements sociaux. Si oui, la politique sociale est générée en état `violated`, sinon en état `fulfilled`.
- Les agents *indulgents* (« forgiving » dans les figures de cette section) instancient aussi les normes en politiques sociales à chaque fois qu'un nouvel engagement social est observé. Cependant, dans le cas où un agent annule *a posteriori* certains de ses engagements, les agents indulgents révisent leurs historiques de politiques sociales, pour annuler celles qui ne reflètent plus une inconsistance.

Par ailleurs, les agents doivent aussi être en mesure de prendre des décisions de faire ou non confiance en fonction des violations qu'ils détectent. Pour ce faire, nous avons implémenté différents types de réputation dans les agents : Réputation fondée sur les Interactions Directes, Réputation fondée sur les Interactions Indirectes et Réputation fondée sur les Recommandations de Réputation. Les autres types de Réputation fondée sur les Recommandations ont été écartés car, dans le cadre de nos simulations, ils ne diffèrent de la Réputation fondée sur les Interactions Directes et de la Réputation fondée sur les Interactions Indirectes que par le système de filtrage. Le système de filtrage est illustré à l'aide de la Réputation fondée sur les Recommandations de Réputation.

De manière à ce que les agents puissent prendre des décisions en s'appuyant sur ces réputations, les valeurs des pondérations du processus de punition ($\tau_{\mathcal{X}}$, $\mathcal{X} \in \{$ `fulfilled`, `violated`, `cancelled` $\}$) et les seuils de raisonnement ($\theta^{\text{trust}}_{\mathcal{X}\texttt{bRp}}$, $\theta^{\text{distrust}}_{\mathcal{X}\texttt{bRp}}$, $\theta^{\text{relevance}}_{\mathcal{X}\texttt{bRp}}$, $\mathcal{X} \in \{$ `DI`, `II`, `ObsRc`, `EvRc`, `RpRc` $\}$) doivent aussi être déterminées. Dans les expérimentations présentées ici, nous les avons fixées comme suit (voir Annexe C.4 pour les raisons de ces choix) :

- $\theta^{\text{trust}}_{\mathcal{X}\texttt{bRp}} = 0.8, \forall \mathcal{X} \in \{$ `DI`, `II`, `ObsRc`, `EvRc`, `RpRc` $\}$.
- $\theta^{\text{distrust}}_{\mathcal{X}\texttt{bRp}} = 0.5, \forall \mathcal{X} \in \{$ `DI`, `II`, `ObsRc`, `EvRc`, `RpRc` $\}$.
- $\theta^{\text{relevance}}_{\texttt{DIbRp}} = 10$, $\theta^{\text{relevance}}_{\texttt{IIbRp}} = 7$, $\theta^{\text{relevance}}_{\texttt{RpRcbRp}} = 5$.

Sauf indication contraire, les pénalités associées aux politiques sociales sont fixées à 1.0.

7.5 Expérimentations

Dans cette section, nous commençons par présenter une grille d'expérimentation, puis présentons les résultats obtenus avec le modèle L.I.A.R., selon les différents critères de cette grille, dans le cadre du scénario d'échange d'information défini précédemment. Les résultats présentés sont des moyennes sur dix simulations.

7.5.1 Grille d'expérimentation

Les expérimentations que nous menons dans cette section sont organisées de la manière décrite ci-après. Nous commençons par comparer les différentes stratégies d'instanciation de normes en termes de nombre de violations détectées et de niveau de Réputation fondée sur les Interactions Directes calculé. Nous utilisons ensuite un travail que nous avons mené dans le cadre du ART-testbed pour définir les expérimentations présentées dans cette partie [FKM+05b]. Dans ce travail, nous avons établi les critères suivants :

Multi-$\star$ Un modèle de réputation doit être en mesure d'estimer la réputation d'un agent selon différents aspects. Nous ne proposons pas d'expérimentations allant dans ce sens car le modèle L.I.A.R. est, par construction, multi-dimensionnel et multi-facette.

Rapidement convergent Un modèle de réputation doit être capable d'estimer rapidement la réputation d'un nouveau venu. Cette propriété est particulièrement nécessaire dans les environnements virtuels, car les agents peuvent changer d'identité en entrant et sortant rapidement du système. La convergence peut être estimée en mesurant le temps que met le modèle à tendre vers des valeurs suffisamment précises [DKG+04].

Précis Un modèle de réputation doit être un indicateur correct du comportement futur d'un agent. La précision d'un modèle peut être mesurée en termes de similarité entre la valeur de réputation calculée et le comportement réel d'un agent [Ful03, KP04, WsI04].

Adaptatif Un modèle de réputation doit être capable de s'adapter aux changements dynamiques du comportement des agents. Ceux-ci peuvent, en effet, soudainement devenir moins compétents ou user de stratégies malignes en trichant de manière non régulière [FB04].

Efficace Les modèles de réputation doivent estimer les niveaux efficacement, en termes de coûts et de temps [GH04, YIO04]. L'efficacité computationnelle peut être mesurée par le temps que met un algorithme à mettre à jour un niveau de réputation.

Par ailleurs, le modèle de réputation doit aider les agents à prendre des décisions permettant de :

Identifier et isoler les agents malveillants Les agents doivent être capables d'identifier et d'isoler les agents malveillants ou incompétents en refusant d'interagir avec eux. [BK01, BSD00].

Décider si et avec qui interagir Étant donné un groupe d'agents avec qui une interaction pourrait être profitable, un agent doit être capable de choisir un partenaire qui a de fortes chances de remplir ses engagements. Il est, par exemple, possible d'estimer le taux de succès de décisions de faire confiance en rapportant le nombre d'interactions terminées avec succès sur le nombre d'interactions total [FPCC04, SFR00].

Évaluer l'utilité d'une interaction Un agent doit estimer l'utilité d'une interaction, ou le degré avec lequel un contrat va d'être tenu, afin de mieux en négocier les termes [NP04]. Nous ne proposons pas d'expérimentation pour cette propriété, car le modèle L.I.A.R. est conçu et testé ici dans le cadre d'engagements non explicites et non négociés.

7.5.2 Paramètres de simulation

Pour tester le modèle L.I.A.R., nous simulons une plate-forme multi-agent dans laquelle des agents s'engagent socialement les uns envers les autres.

Les faits concernant les horaires de salles de cinéma sont symbolisés comme suit : les faits du type `shows`(`theater1`, `Shrek`, `19 :00`, `room1`) sont représentés par une lettre comme A. Les faits avec lesquels ces derniers peuvent créer des inconsistances sont regroupés sous la notation $\neg A$.

Les paramètres de la simulation sont les suivants :

- `NB_ITERATIONS` : le nombre d'itérations pendant lequel la simulation est répétée.
- `NB_AGENTS` : le nombre d'agents qui participent à la simulation.
- `NB_FACTS` : le nombre de faits différents sur lesquels un agent peut s'engager socialement. Pour chaque fait, un agent peut s'engager sur ce fait ou sa négation. Le nombre total de faits sur lesquels un agent peut s'engager est donc 2∗`NB_FACTS`.

– NB_DI_BY_ITERATION : le nombre d'interactions directes dont chaque agent est la source à chaque pas de temps, *i.e.* le nombre d'engagements sociaux dont chaque agent est débiteur.

Le déroulement de la simulation est le suivant ($\mathcal{A}$ est un ensemble d'agents tel que $|\mathcal{A}|$ =NB_AGENTS) :

```
Pour n = 1 à NB_ITERATIONS faire :
  Pour chaque agent db de A :
    CR = choisir(NB_DI_BY_ITERATION, A\{db})
    Pour chaque agent cr de CR :
      Générer un fait f_cr parmi les 2*NB_FACTS possibles
      En fonction du violationRate, désengager db de
      certains de ses engagements inconsistants avec f_cr
      Engager db sur f_cr envers le cr correspondant
    Fin
  Fin
Fin
```

Où *choisir*(NB_DI_BY_ITERATION, $\mathcal{A}\backslash\{db\}$) retourne un ensemble de NB_DI_BY_ITERATION agents différents choisis aléatoirement dans $\mathcal{A}\backslash\{db\}$.

Suite à diverses expérimentations, dont les résultats sont synthétisés en Annexe C.1, nous avons fixés les paramètres de la simulation comme suit :

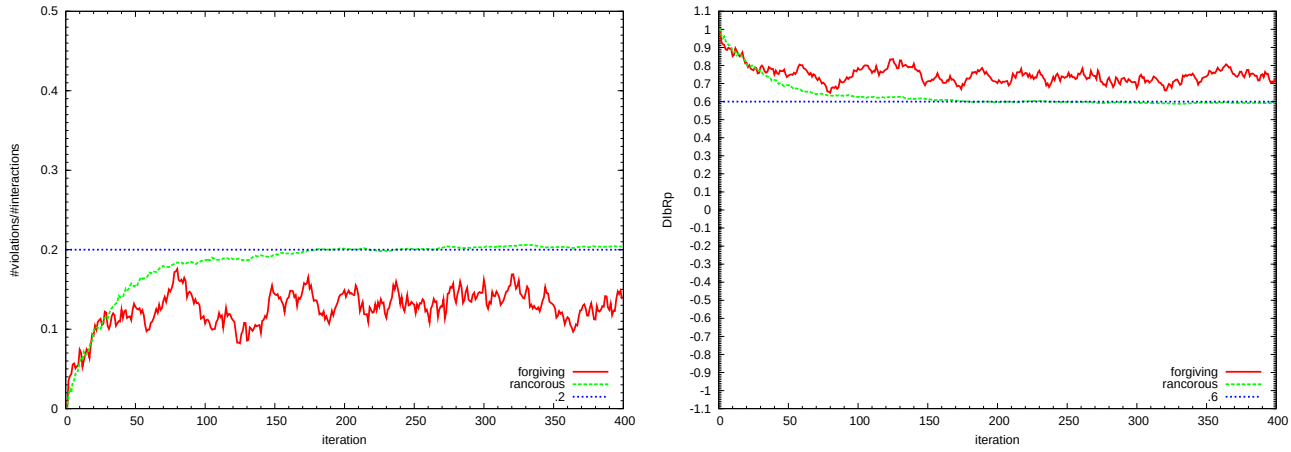
– NB_ITERATIONS : 400.
– NB_AGENTS : 11.
– NB_FACTS : 30.
– NB_DI_BY_ITERATION : 2.

Sauf indication contraire, ces paramètres sont utilisés dans toutes les simulations qui suivent.

7.5.3 Comparaison des stratégies

Dans cette section, nous présentons une comparaison du taux de violations détectées et du niveau calculé pour la Réputation fondée sur les Interactions Directes en fonction de la stratégie d'instanciation déployée par l'agent. Dans ces expériences, tous les agents ont un violationRate de 0.2.

Les figures 7.3(a) et 7.3(b) présentent, respectivement, les taux de violations détectées et le niveau de Réputation fondée sur les Interactions Directes (DIbRp) calculé, selon que la stratégie employée est indulgente ou rancunière.



(a) taux violations (b) `DIbRp`

FIG. 7.3 – Taux de violations détectées et `DIbRp` selon la stratégie.

La ligne à 0.2 dans la figure 7.3(a) indique le `violationRate`e l'agent considéré ; la ligne à 0.6 dans la figure 7.3(b) trace la réputation qu'aurait un agent qui génèrerait 20 % de violations, selon la formule donnée en section 6.3.3, page 117.

Les agents indulgents annulent certaines des politiques sociales, quand elles sont associées à des engagements sociaux que la cible a annulés. De ce fait, le nombre de politiques sociales en état `violated` et le nombre de politiques sociales total que considère l'agent indulgent ne sont pas stables dans le temps. Les courbes représentant le nombre de violations détectées et la réputation estimée par les agents indulgents sont donc plus variables. De plus, les agents indulgents sous-estiment légèrement le taux de violation (et, en conséquence, sur-estiment légèrement la réputation) du fait de l'influence du nombre de faits (*cf.* Annexe C.1, page 195).

Cette influence du nombre de faits est cependant contrebalancée chez les agents rancuniers par une estimation du taux de violations supérieur à ce que la cible génère. Ceci est lié au fait que les agents rancuniers considèrent qu'une violation tient toujours, même si la cible a annulé *a posteriori* ses engagements inconsistants, alors que l'agent qui génère ces violations ne prend plus en compte les engagements qu'il a annulés.

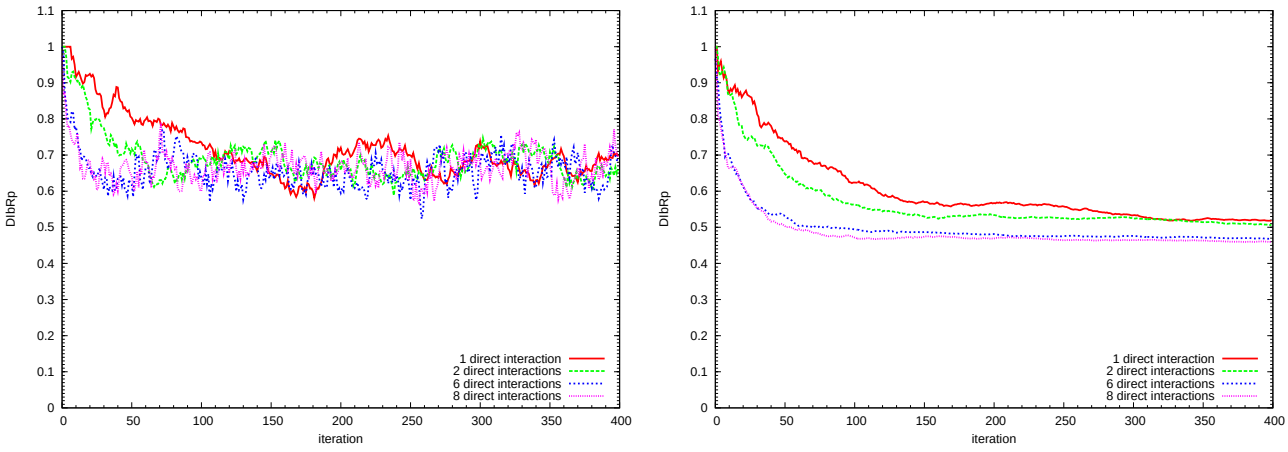
7.5.4 Convergence et précision

Dans cette section, nous étudions la convergence[1] et la précision des différents types de réputation en fonction des différentes stratégies.

Il est important de noter que la décentralisation est une caractéristique fondamentale du modèle L.I.A.R. et qu'elle a pour conséquence que le niveau de réputation qu'estime un agent est fortement dépendant de ce qu'il a pu observer. Ainsi, la précision des réputations calculées par des bénéficiaires autres que la cible elle-même est toute relative. Dans cette section, nous sommes parfois amenés à tracer les réputations que les agents estiment pour eux-mêmes (dont ils sont à la fois cible et bénéficiaire), afin d'obtenir des courbes qui ne sont pas perturbées par l'incomplétude des historiques locaux d'engagements sociaux et de politiques sociales.

Dans les expérimentations suivantes, les agents 0 à 5 emploient la stratégie indulgente, les agents 6 à 10, la stratégie rancunière.

Réputation fondée sur les Interactions Directes



(a) indulgents (b) rancuniers

FIG. 7.4 – `DIbRp` en fonction du nombre d'interactions directes.

[1] Il est important de noter que les réputations ne « convergent » que sous la contrainte que les agents conservent un comportement constant, ce qui est relativement rare dans la réalité. Cependant, pour l'intérêt des expérimentations présentées ici, les agents ont des `violationRate` et `lieRate` fixés dans le temps.

La figure 7.4(a) présente l'évolution de la Réputation fondée sur les Interactions Directes, en fonction du nombre d'interactions directes par pas de temps, telle que perçue par un agent indulgent, pour lui-même. La réputation tracée est celle d'un agent ayant un `violationRate` de 0.2 ($\texttt{DIbRp}_2^2$, dans une configuration où l'agent 2 a une stratégie indulgente).

L'augmentation du nombre d'interactions directes influe sur le temps de convergence du niveau de Réputation fondée sur les Interactions Directes calculé. Cette influence est cependant limitée, du fait de la forte variabilité des ensembles de politiques sociales violées et total sur lesquels les agents indulgents s'appuient pour calculer la Réputation fondée sur les Interactions Directes (*cf.* section 7.5.3).

La figure 7.4(b) présente l'évolution de la Réputation fondée sur les Interactions Directes, en fonction du nombre d'interactions directes par pas de temps, telle que perçue par un agent rancunier, pour lui-même. La réputation tracée est celle d'un agent ayant un `violationRate` de 0.2 ($\texttt{DIbRp}_2^2$, dans une configuration telle que l'agent 2 déploie une stratégie rancunière).

Le nombre d'interactions directes influe sur le temps de convergence du niveau de Réputation fondée sur les Interactions Directes calculé par les agents rancuniers. En effet, les agents rancuniers considèrent toujours l'ensemble des politiques sociales qu'ils ont générées depuis leur entrée dans le système. Plus le nombre d'interactions augmente, plus l'agent dispose d'un large ensemble de politiques sociales sur lequel mener ses calculs. En conséquence, plus il y a d'interactions directes, plus le niveau de réputation est estimé rapidement.

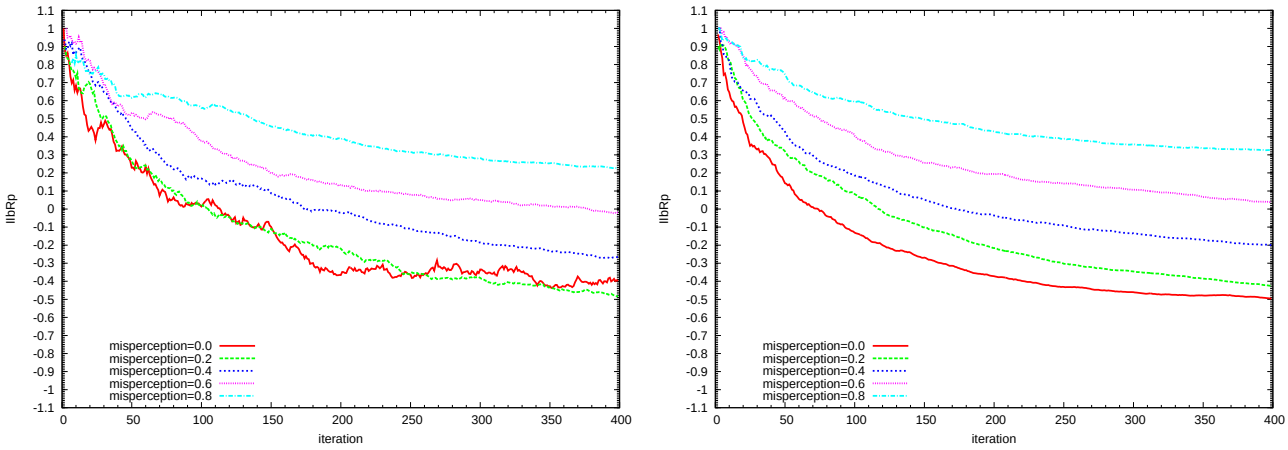
Du fait que la méthode qu'emploient les agents pour générer les engagements sociaux ne garantit pas la production d'une proportion donnée de violations, la précision est difficile à définir dans le cadre des simulations présentées ici. En revanche, les figures présentées ici montrent qu'à `violationRate` constant, les niveaux de réputations calculés se « stabilisent » autour d'une certaine valeur. Ces dernières sont, par ailleurs, différentes selon la stratégie de l'agent.

Réputation fondée sur les Interactions Indirectes

Afin de tester le comportement de la Réputation fondée sur les Interactions Indirectes, nous avons introduit de nouveaux paramètres à la simulation. Chaque engagement social qui est pris dans le système est désormais observé par un certain nombre d'agents, noté `NB_II_BY_ITERATION`. Étant donné que les calculs de la Réputation fondée sur les Interactions Directes

et de la Réputation fondée sur les Interactions Indirectes sont similaires, ce nouveau paramètre intervient sur la convergence de cette dernière de la même façon que le paramètre `NB_DI_BY_ITERATION` intervient sur la convergence de la Réputation fondée sur les Interactions Directes. Nous avons fixé ce paramètre à 2 dans les expérimentations proposées ci-après.

Nous avons aussi ajouté le paramètre `MISPERCEPTION_RATE` qui caractérise le taux de mauvaises perceptions des violations. Ce paramètre intervient lorsqu'un agent observe une interaction qui n'est pas directe pour lui. Le paramètre définit le nombre de fois où l'état de l'engagement social perçu ne sera pas son état réel, mais un état généré aléatoirement, parmi l'ensemble des autres états possibles. Une telle configuration implique que des violations peuvent ne pas être perçues, mais pas l'inverse.



(a) indulgents (b) rancuniers

FIG. 7.5 – `IIbRp` en fonction du taux de mauvaises perceptions.

La figure 7.5(a) (resp. 7.5(b)) présente l'évolution de la Réputation fondée sur les Interactions Indirectes (`IIbRp`), en fonction du taux de mauvaises perceptions, associé par un agent indulgent (resp. rancunier) à un agent ayant un `violationRate` de 0.8 ($\texttt{IIbRp}^8_0$, resp. $\texttt{IIbRp}^8_{10}$).

Du fait que la Réputation fondée sur les Interactions Indirectes est calculée à partir d'historiques incomplets et imparfaits, le nombre de violations détectées peut être relativement différent du nombre de violations effectivement générées. C'est la raison pour laquelle la précision est relativement difficile à estimer. Cependant, dans le cas où il n'y a pas de mauvaises per-

ceptions (« misperception=0.0 » dans la figure), les réputations tendent bien vers -0.6.

Quelque soit la stratégie d'instanciation, les figures montrent une différence marquée dans les valeurs obtenues en fonction de l'augmentation du taux de mauvaises perceptions. Du fait que les violations ne sont pas toutes perçues correctement, les réputations sont sur-estimées.

Réputation fondée sur les Recommandations de Réputation

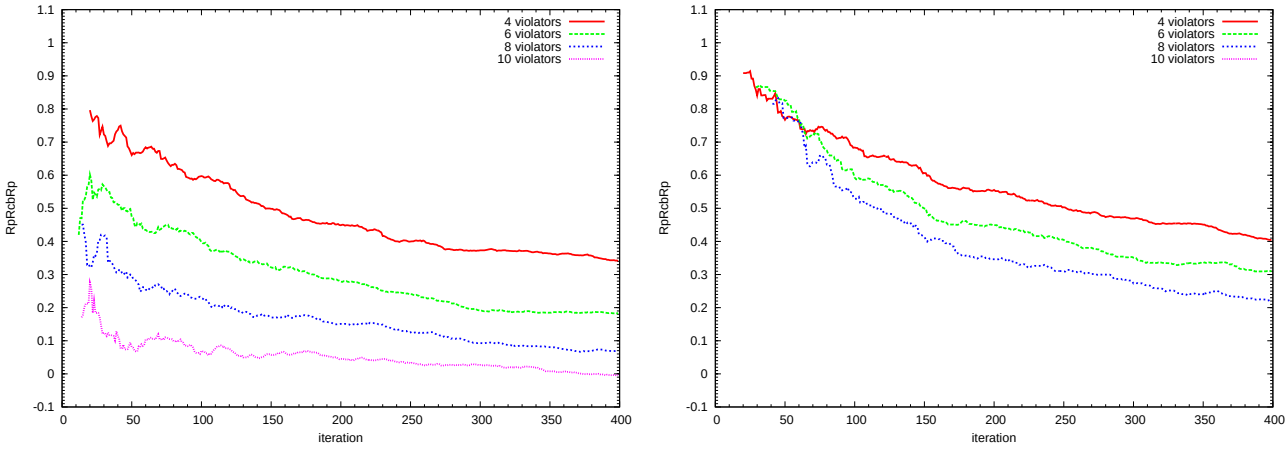
Afin de tester le comportement de la Réputation fondée sur les Recommandations de Réputation (`RpRcbRp`), nous avons introduit de nouveaux paramètres à la simulation. Les agents envoient désormais aussi des recommandations, c'est-à-dire prennent des engagements dont le contenu est un niveau de réputation. Il est alors possible de calculer les réputations en fonction de la facette qu'elles adressent (horaires de cinéma : `"theater showtimes"` ou recommandations : `"recommend"`). Puisqu'elles sont calculées sur le même principe, les réputations selon la facette de recommandation ont les mêmes propriétés, en termes d'influence de la stratégie de l'agents, de précision et de convergence, que les réputations selon l'autre facette (décrites ci-dessus).

Un nouveau paramètre, `NB_RC_BY_ITERATION`, définit le nombre de recommandations que chaque agent envoie à chaque itération. Ce paramètre intervient de façon similaire sur la convergence de la Réputation fondée sur les Recommandations de Réputation que les paramètres `NB_DI_BY_ITERATION` et `NB_II_BY_ITERATION` influent sur les autres réputations, dans le cas des agents rancuniers : plus il est grand, plus la convergence est rapide. Ceci est dû aux calculs sous forme de moyenne sur un ensemble s'agrandissant avec le nombre de recommandations. Nous avons fixé ce paramètre à 2 dans les expérimentations suivantes.

Nous avons aussi ajouté un paramètre qui permet d'activer ou non le double filtrage de la Réputation fondée sur les Recommandations de Réputation et un paramètre `NB_VIOLATORS` permettant de modifier le nombre de violateurs dans la population. Dans les simulations présentées ici, les violateurs ont tous un `violationRate` de 0.8.

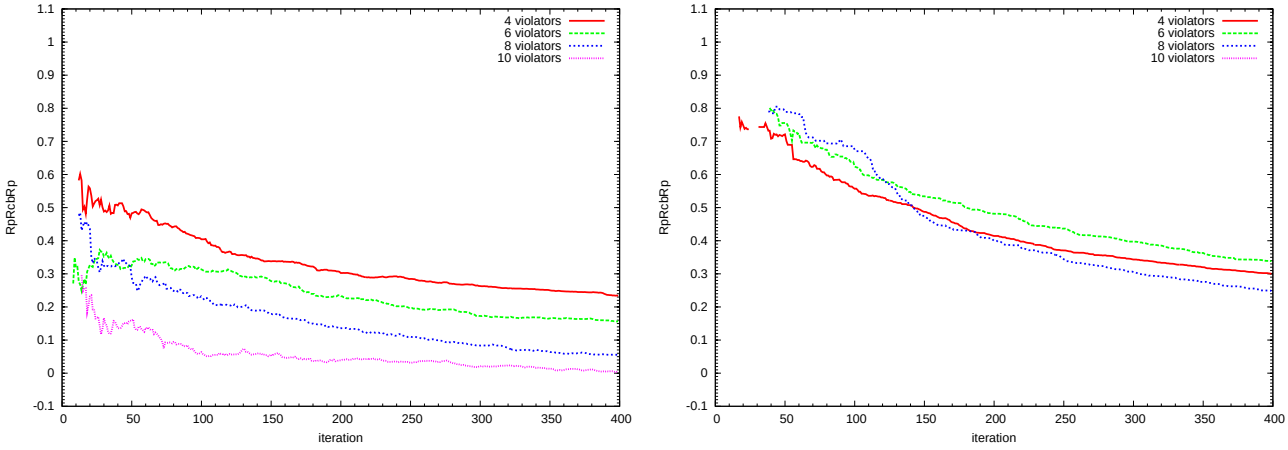
Les figures 7.6 et 7.7 tracent l'évolution de la Réputation fondée sur les Recommandations de Réputation que l'agent 10 associe à l'agent 1, en fonction de la mise en place ou non du filtrage et du nombre de violateurs présents dans la population.

Du fait qu'elle est calculée sur un ensemble de valeurs de réputation, qui



(a) sans filtrage (b) avec filtrage

FIG. 7.6 – $\texttt{RpRcbRp}^1_{10}$, filtrage et nombre de violateurs (indulgents).



(a) sans filtrage (b) avec filtrage

FIG. 7.7 – $\texttt{RpRcbRp}^1_{10}$, filtrage et nombre de violateurs (rancuniers).

ont elles-mêmes été calculées à partir de nombreuses politiques sociales, la Réputation fondée sur les Recommandations de Réputation est peu variable et converge rapidement. En revanche, il faut un certain temps avant qu'elle ne quitte le statut `unknown`. Ceci est lié à deux paramètres : d'une part, le nombre de recommandations reçues (qui dépend de `NB_RC_BY_ITERATION` et $\theta^{\text{relevance}}_{\texttt{RpRcbRp}}$) et, d'autre part, de la sortie du processus de raisonnement employé dans le filtrage (celui-ci réduisant encore le nombre de recommandations aux seules recommandations de confiance).

Les violations dans les recommandations consistant en des valeurs aléatoires dans $[-1, +1]$, leur moyenne tend vers 0. En l'absence de filtrage, les agents accumulent de telles recommandations en grand nombre et les réputations qu'ils estiment tendent donc elles aussi fortement vers 0. L'augmentation de la proportion d'agents malveillants augmente le nombre de telles recommandations et accélère donc la convergence vers 0 (figures 7.6(a) et 7.7(a)).

Les figures 7.6(b) et 7.7(b) montrent que le double filtrage permet de limiter fortement l'influence de ces mauvaises recommandations. Dans le cas où le nombre de violateurs est trop grand (10), le double filtrage maintient les réputations en état `unknown`, d'où l'absence de courbe associée à cette proportion.

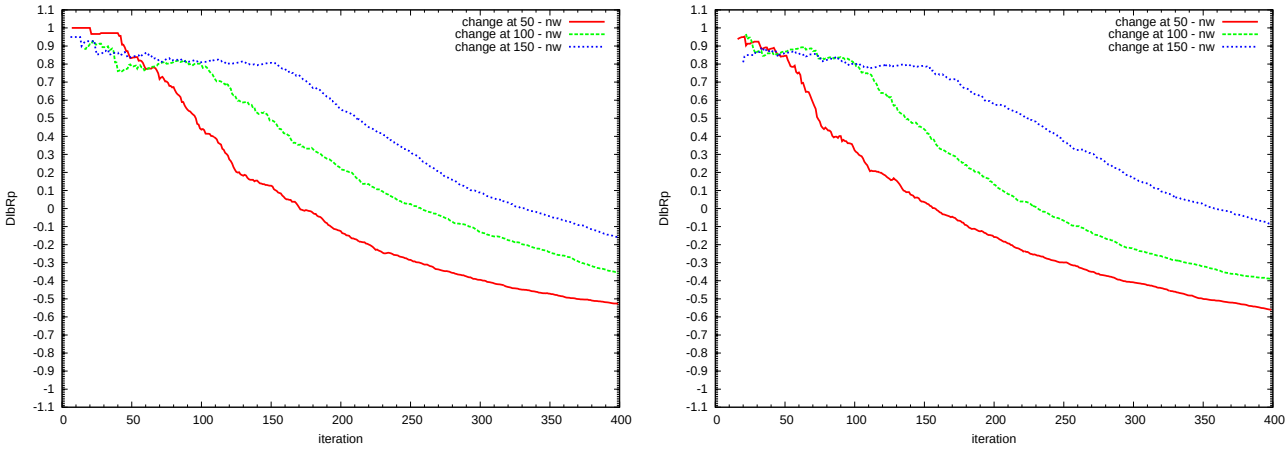
7.5.5 Adaptabilité

Dans cette section, nous présentons les résultats concernant l'adaptabilité des différents types de réputation. Nous étudions cette adaptabilité selon deux critères : l'inertie et la fragilité.

Inertie

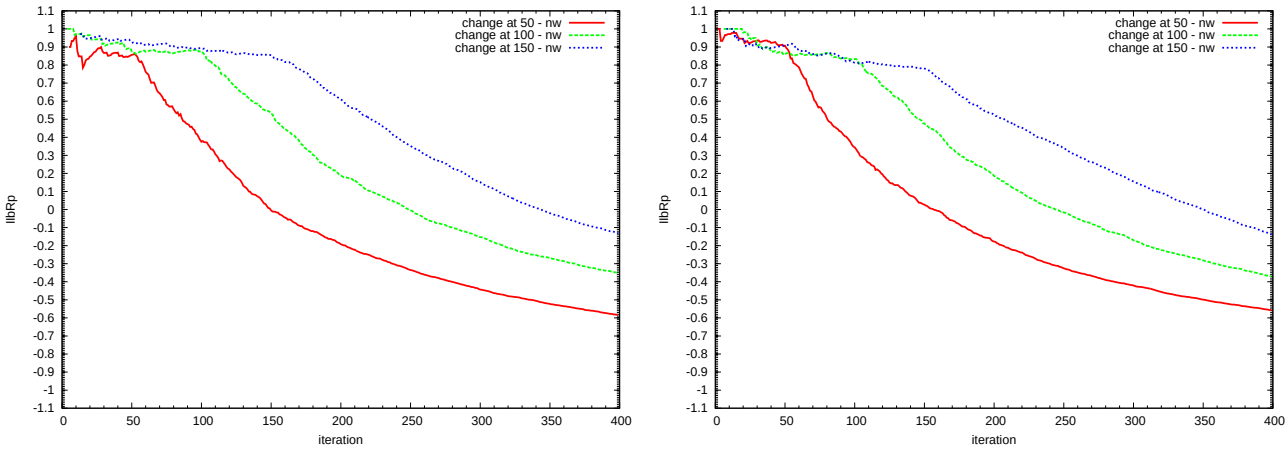
Afin de tester l'inertie des niveaux de réputation calculés par les bénéficiaires, nous avons introduit un nouveau paramètre : `CHANGE_BEHAVIOUR_TIME`, qui détermine l'itération à partir de laquelle les agents changent brutalement de comportements (leurs `violationRate` passent à -1.0).

Les figures 7.8 et 7.9 présentent le temps de réponse de la Réputation fondée sur les Interactions Directes et de la Réputation fondée sur les Interactions Indirectes en fonction des stratégies des agents, quand un changement brutal de comportement intervient à l'itération 50, ou 100, ou 150.



(a) indulgents (b) rancuniers

FIG. 7.8 – Inertie de la `DIbRp`.

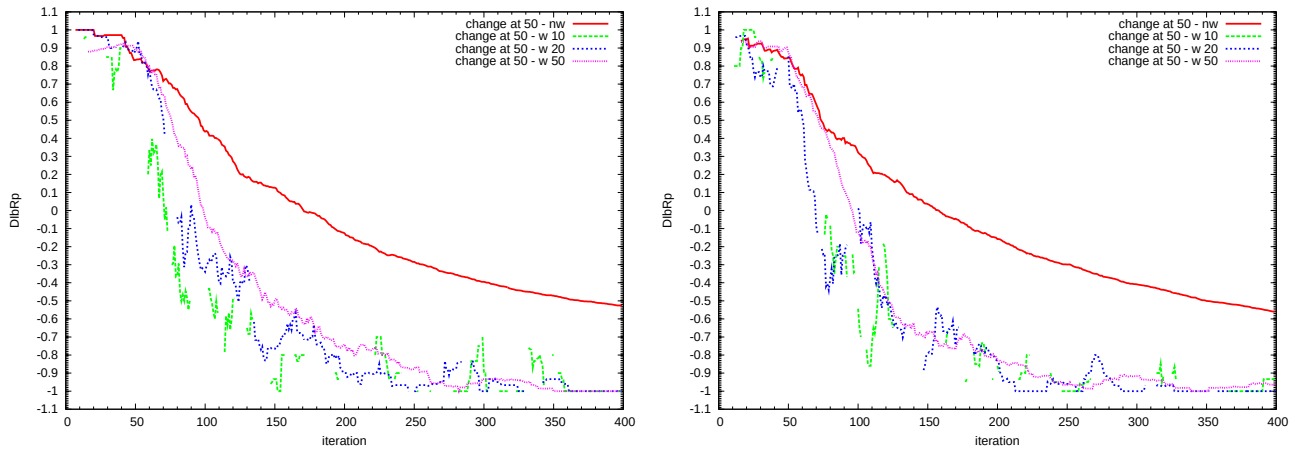


(a) indulgents (b) rancuniers

FIG. 7.9 – Inertie de la `IIbRp`.

Chez les agents rancuniers comme chez les agents indulgents, les calculs des niveaux de ces réputations font intervenir des moyennes sur des ensembles de politiques sociales qui s'agrandissent avec le temps. En conséquence, les valeurs font preuve d'une grande inertie : plus le changement intervient tard, plus la baisse de réputation est lente. C'est ce que montre la pente plus faible des courbes juste après un changement plus tardif dans les figures 7.8 et 7.9.

Il est possible de limiter cette inertie en utilisant des fenêtres temporelles. Un nouveau paramètre a été ajouté pour cela : `WINDOW_SIZE`. Ce dernier détermine la taille de la fenêtre temporelle en dessous de laquelle les politiques sociales ne sont pas considérées (et dont l'extrémité maximale est l'itération courante).



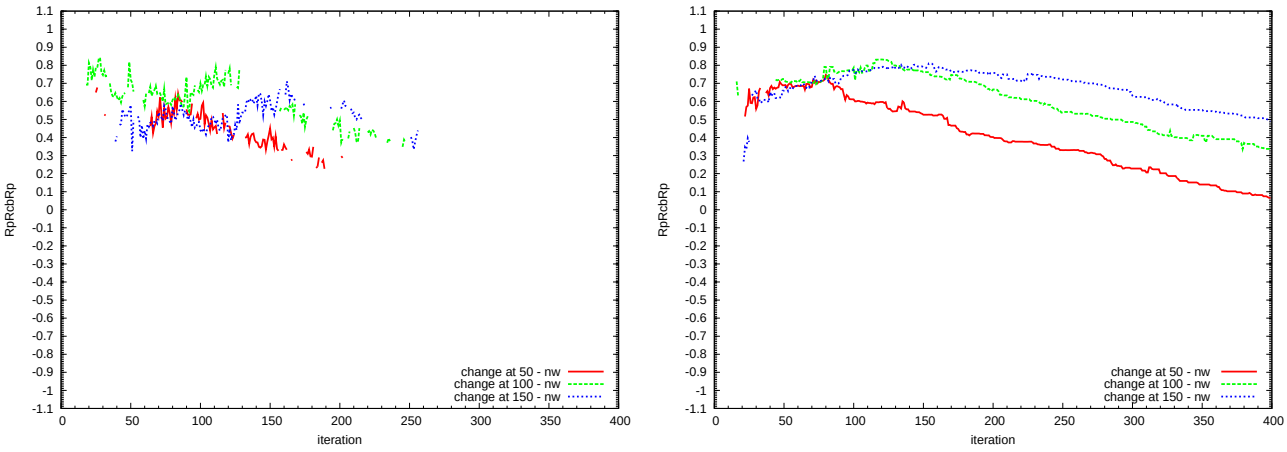
(a) indulgents (b) rancuniers

FIG. 7.10 – Inertie de la `DIbRp` avec fenêtres temporelles.

La figure 7.10 présente l'inertie de la Réputation fondée sur les Interactions Directes quand un changement brutal a lieu à l'itération 100, sans fenêtre temporelle (nw dans la figure) et en présence de fenêtres temporelles de tailles 10 (w 10 dans la figure), 20 (w 20) et 50 (w 50). Il apparaît que la réputation change plus vite (*i.e.* a moins d'inertie) quand la fenêtre est plus petite. Cependant, plus la fenêtre est petite, plus l'ensemble de politiques sociales considéré est petit. Il arrive même que, dans la durée définie par la fenêtre, il n'existe pas de politiques sociales. De ce fait, la réputation redevient `unknown`, d'où les interruptions dans les courbes.

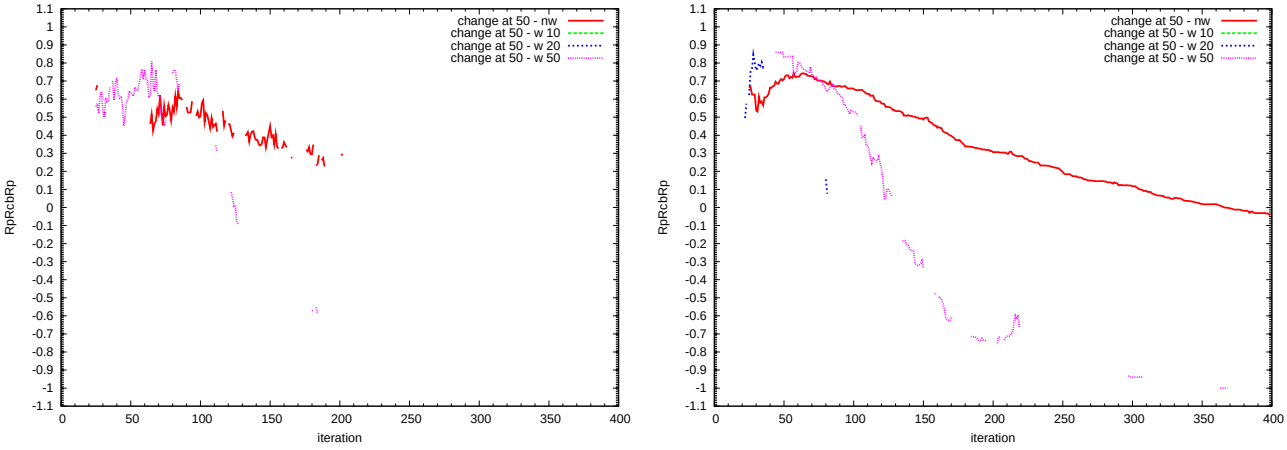
Du fait de la similitude des formules employées pour calculer la Répu-

tation fondée sur les Interactions Directes et la Réputation fondée sur les Interactions Indirectes, les résultats sont semblables pour ce deuxième type de réputation.



(a) indulgents (b) rancuniers

FIG. 7.11 – Inertie de la `RpRcbRp`.



(a) indulgents (b) rancuniers

FIG. 7.12 – Inertie de la `RpRcbRp` avec fenêtre temporelle.

Les figures 7.11 et 7.12 présentent le temps de réponse de la Réputation fondée sur les Recommandations de Réputation, respectivement pour

les agents indulgents et pour les agents rancuniers. Cette dernière étant calculée à partir de valeurs de Réputation fondée sur les Interactions Directes communiquées par d'autres agents, son inertie est liée à celle de ces valeurs. Par ailleurs, elle est calculée par moyenne sur l'ensemble des recommandations reçues depuis le début de la simulation. Elle fait donc aussi preuve d'une inertie propre. Globalement, la Réputation fondée sur les Recommandations de Réputation a donc une très forte inertie.

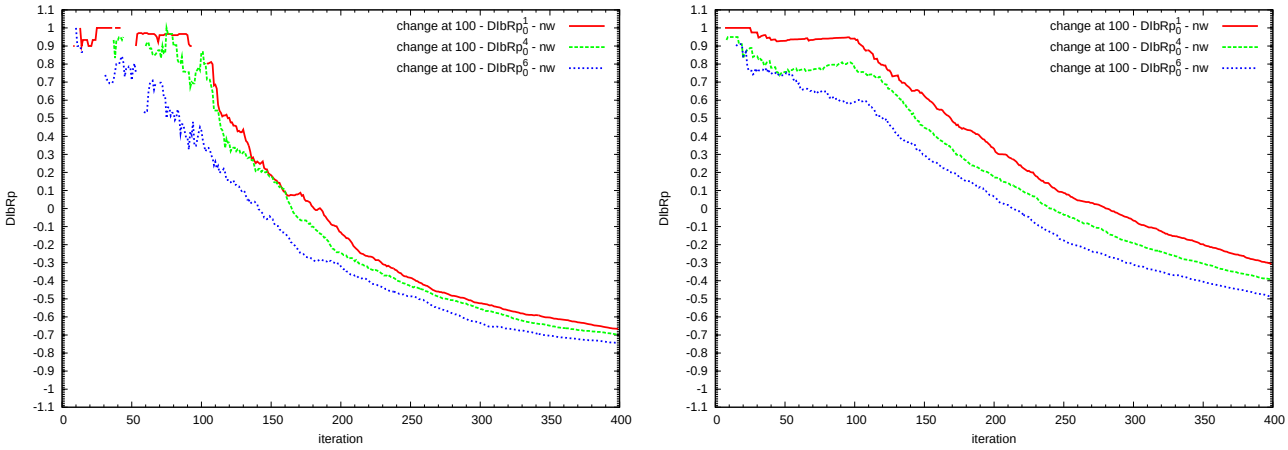
Les figures 7.12(a) et 7.12(b) comparent l'inertie de la Réputation fondée sur les Recommandations de Réputation sans fenêtre temporelle (nw) ou avec des fenêtres temporelles de tailles 10 (w 10 dans la figure), 20 (w 20) et 50 (w 50) sur les recommandations, respectivement pour la stratégie indulgente et pour la stratégie rancunière. Les mêmes remarques que pour les autres types de réputations peuvent être faites : l'utilisation d'une fenêtre permet de limiter fortement l'inertie, mais réduit le nombre de recommandations à partir desquelles le calcul de réputation est fait, ce qui rend la réputation plus souvent `unknown`. Cet effet est d'ailleurs exacerbé dans le cas de la Réputation fondée sur les Recommandations de Réputation, puisque le double filtrage réduit déjà beaucoup le nombre de recommandations utilisé dans le calcul. Les figures 7.12(a) et 7.12(b) montrent ainsi que seule une fenêtre de 50 permet d'obtenir des résultats.

Fragilité

Dans cette section, nous étudions la fragilité des niveaux de réputation calculés.

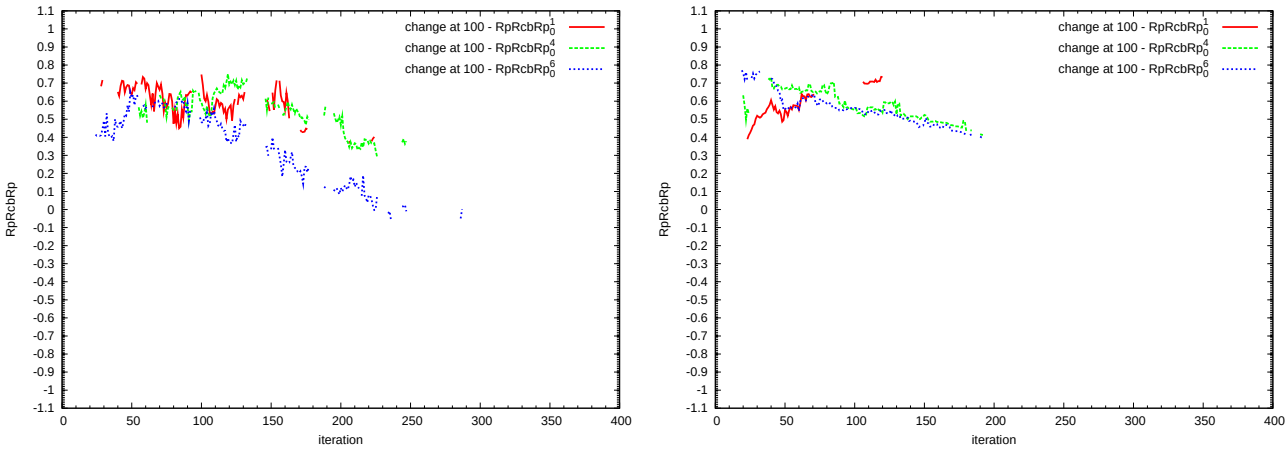
Les figures précédentes 7.8, 7.9 et 7.11 montrent que les différents types de réputations ne sont pas fragiles au sens de la fragilité temporelle (*cf.* section 3.2.5, page 48). En effet, du fait de l'inertie, il est plus long pour un agent de perdre sa réputation que de l'établir. Avec une fenêtre, il est possible de diminuer le temps que met la réputation à décroître, donc éventuellement d'introduire une certaine fragilité temporelle.

Nous avons tout d'abord regardé si les réputations font preuve de fragilité du point de vue de l'interaction (*cf.* section 3.2.5, page 48). Les figures 7.13(a), 7.13(b), tracent la diminution de la Réputation fondée sur les Interactions Directes pour des agents ayant différents `violationRate` (agent 1 : 0.1, agent 4 : 0.4 et agent 6 : 0.6), en fonction de la stratégie rancunière ou indulgente. Les figures 7.14(a) et 7.14(b) tracent des courbes similaires pour la Réputation fondée sur les Recommandations de Réputation.



(a) indulgents (b) rancuniers

FIG. 7.13 – Fragilité de la `DIbRp`.



(a) indulgents (b) rancuniers

FIG. 7.14 – Fragilité de la `RpRcbRp`.

Quelle que soit la stratégie, la diminution n'est pas plus forte chez un agent de forte réputation (les courbes restent parallèles). Les réputations ne sont donc pas fragiles au sens de la fragilité du point de vue de l'itération.

Les annexes C.2, page 198 et C.3, page 200 présentent d'autres résultats permettant d'introduire une fragilité (temporelle ou du point de vue de l'interaction) dans le calcul des réputations.

7.5.6 Décision

Dans cette section, nous étudions les décisions que permet de prendre le modèle L.I.A.R. Nous montrons ici la pertinence de ce modèle pour le contrôle social des interactions, en présentant les graphes globaux des relations de confiance qui se forment au fil du temps. Ceux-ci reflètent la façon dont les agents qui implémentent ce modèle éliminent les agents qui ont des mauvais comportements et renforcent leurs interactions avec ceux de forte réputation.

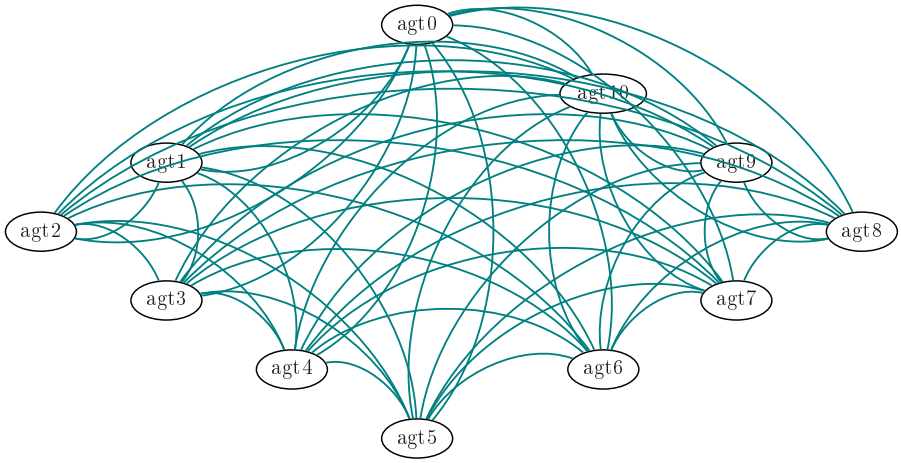


FIG. 7.15 – Graphe global des relations de confiance à l'étape 0 (rancuniers)

Les figures 7.15, 7.16, 7.17, 7.18 et 7.19, montrent l'évolution du graphe global des relations de confiance d'agents rancuniers au fil du temps, dans une configuration où les agents 0 à 7 ont un `violationRate` et un `lieRate` de 0.8. Pour leur part, les agents 8, 9 et 10 ne se contredisent jamais.

Les liens reflètent des intentions de confiance *réciproques* entre les paires d'agents. Ils sont affichés avec une épaisseur plus grande et une couleur plus

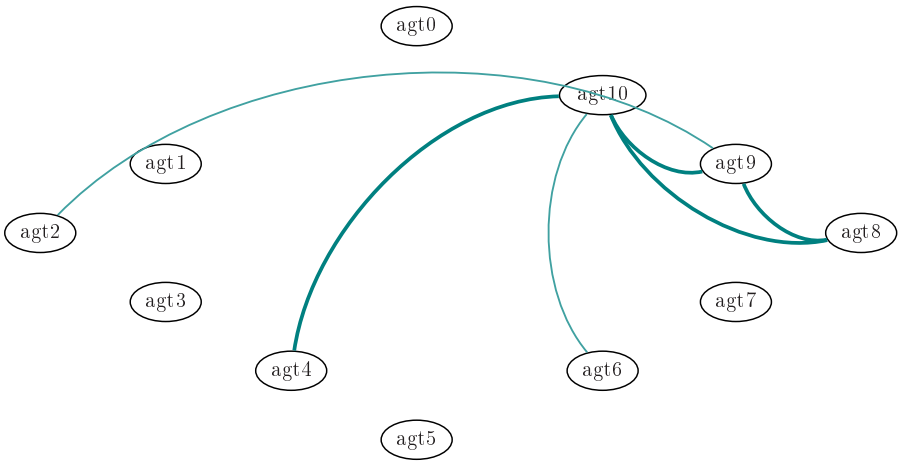


FIG. 7.16 – Graphe global des relations de confiance à l'étape 50 (rancuniers)

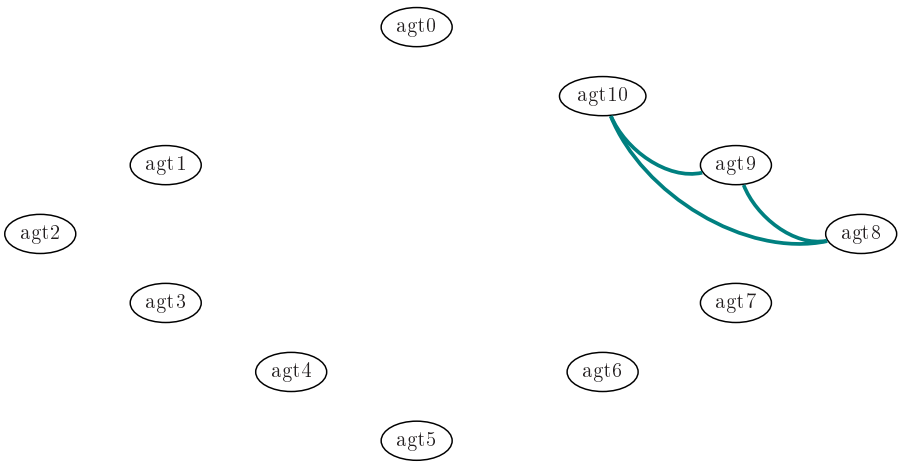


FIG. 7.17 – Graphe global des relations de confiance à l'étape 100 (rancuniers)

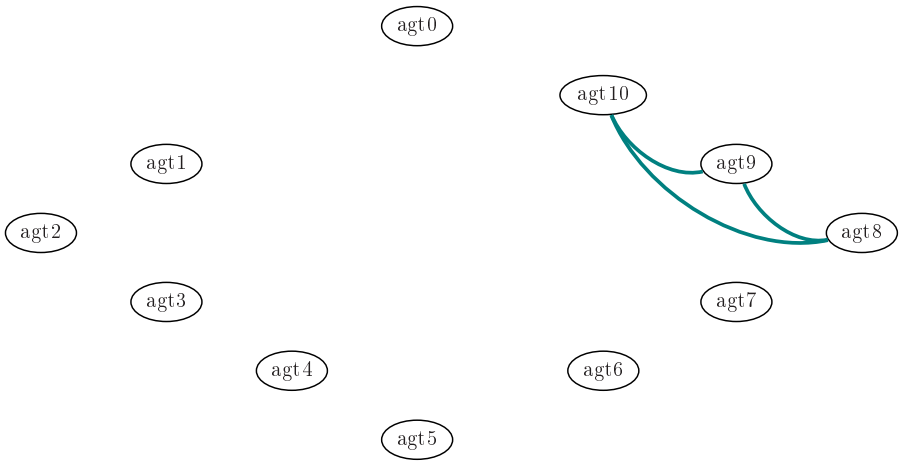


FIG. 7.18 – Graphe global des relations de confiance à l'étape 150 (rancuniers)

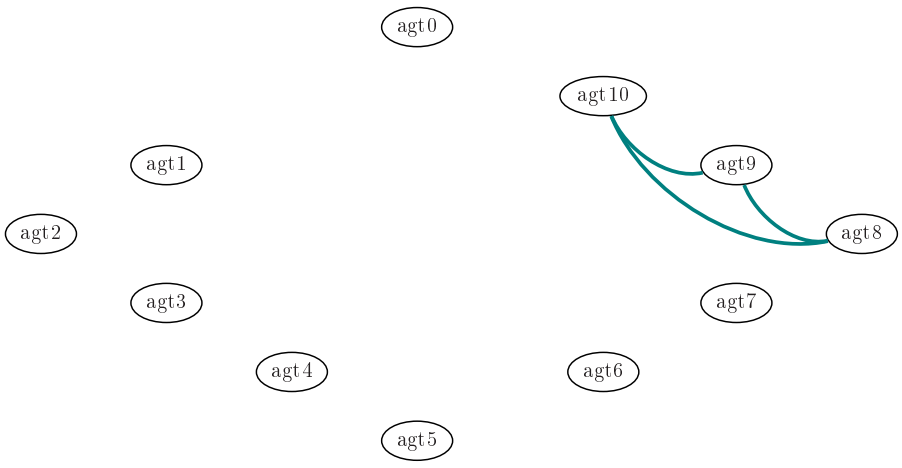


FIG. 7.19 – Graphe global des relations de confiance à l'étape 200 (rancuniers)

foncée quand les intentions de confiance sont plus fortes.

Au départ (itération 0), les agents ne se sont jamais rencontrés ; les intentions de confiance reposent sur la Prédisposition Générale à faire Confiance qui est à `true` par défaut. Tous les agents sont donc reliés par de faibles intentions de confiance. Le graphe tend ensuite rapidement, entre l'itération 50 et l'itération 100 à rejetter les agents 0 à 7 et à reconnaître que les agents 8, 9 et 10 sont de confiance.

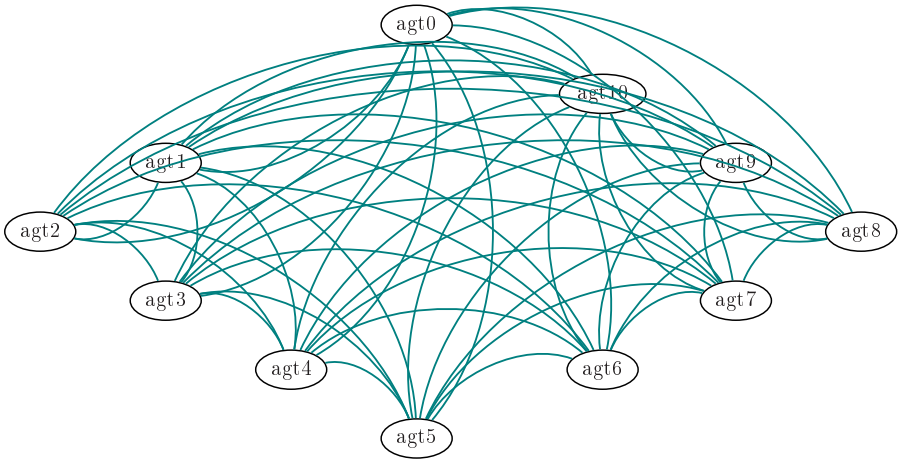


FIG. 7.20 – Graphe global des relations de confiance à l'étape 0 (indulgents)

Les figures 7.20, 7.21, 7.22 et 7.23, montrent l'évolution du graphe global des relations de confiance d'agents indulgents. Dans ce cas, l'identification définitive des agents de confiance et le rejet des agents qui se comportent mal prend plus de temps. Ceux-ci se déroulent entre les itérations 150 et 200.

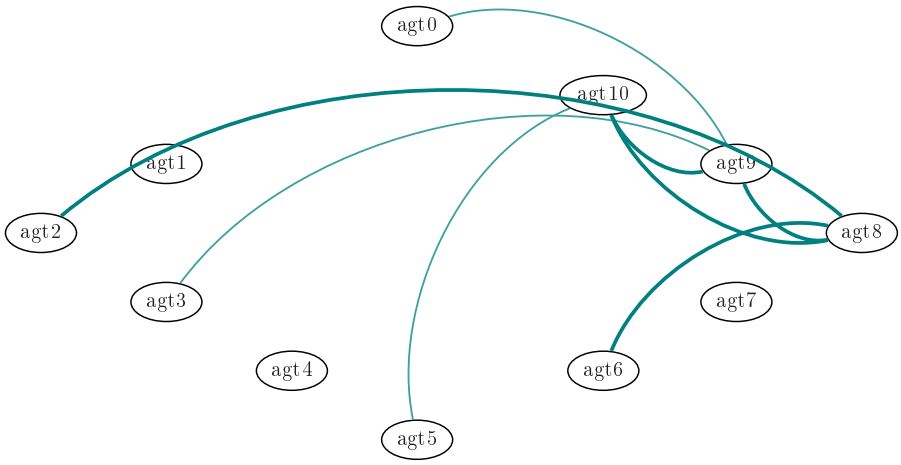


FIG. 7.21 – Graphe global des relations de confiance à l'étape 50 (indulgents)

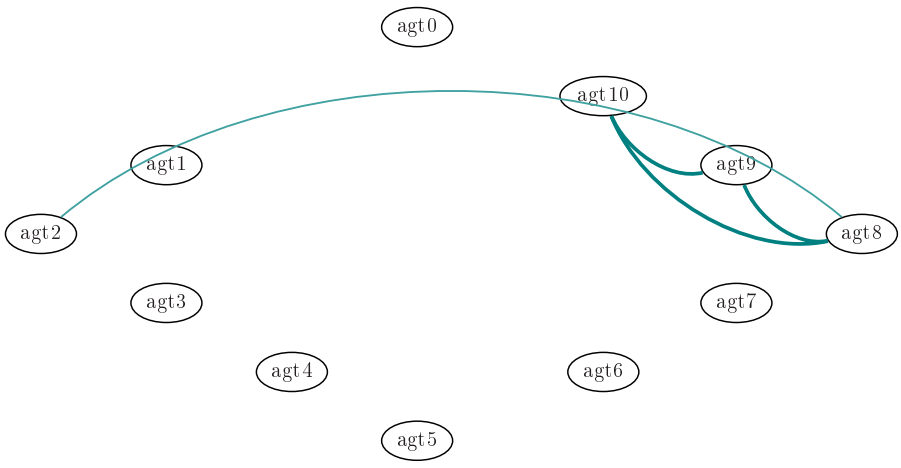


FIG. 7.22 – Graphe global des relations de confiance à l'étape 100 (indulgents)

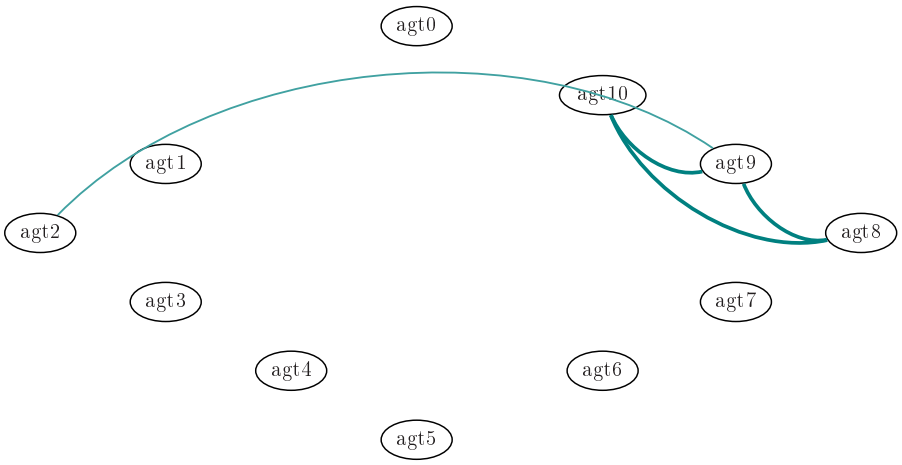


FIG. 7.23 – Graphe global des relations de confiance à l'étape 150 (indulgents)

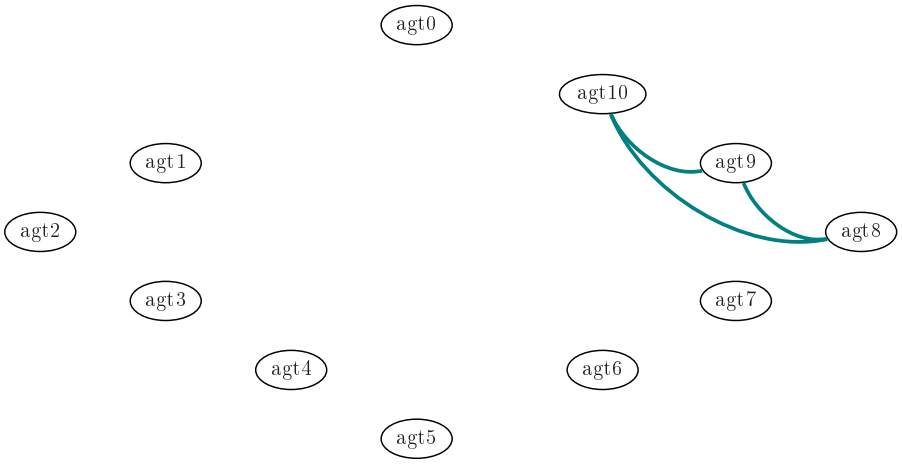


FIG. 7.24 – Graphe global des relations de confiance à l'étape 200 (indulgents)

7.6 Conclusions

Dans ce chapitre, nous avons mené différentes expérimentations pour étudier le comportement du modèle L.I.A.R. selon une grille définie dans le cadre d'une collaboration avec d'autres chercheurs du domaine au sein du projet ART-testbed.

Nous avons tout d'abord comparé les différentes stratégies d'instanciation des normes en termes de nombre de violations détectées et de niveau calculé pour la Réputation fondée sur les Interactions Directes. Les agents indulgents font preuve d'une certaine instabilité dans les niveaux qu'ils calculent. Les agents rancuniers considèrent plus de violations que n'en sont générées à un instant donné du fait qu'ils ne reviennent pas sur leur avis.

Nous avons ensuite étudié la précision et la vitesse de convergence des différents types de réputations. La convergence dépend de nombreux facteurs, comme le nombre d'interactions directes, indirectes ou de recommandations. Plus d'information permet généralement une convergence plus rapide. La précision est, quant à elle, dépendante du fait que les agents n'ont que des points de vue partiels sur le système. Elle est meilleure si l'agent dispose de plus d'information. La Réputation fondée sur les Interactions Indirectes est aussi sous l'influence du taux de mauvaise perception et la Réputation fondée sur les Recommandations de Réputation sous l'influence de la proportion de menteurs dans la population d'agents déployés. Cependant, dans le cas de la Réputation fondée sur les Recommandations de Réputation, le double filtrage limite l'effet des mauvaises recommandations.

Par la suite, nous avons étudié l'adaptabilité des réputations selon deux critères : l'inertie et la fragilité. Nous avons montré que les formules que nous proposons dans la partie II impliquent une forte inertie et ne créent pas de fragilité. Nous avons aussi montré que l'utilisation de fenêtres temporelles permettait de réduire l'inertie et, éventuellement, d'introduire une certaine fragilité temporelle. D'autres pistes pour introduire la fragilité sont présentées dans les annexes C.2 et C.3.

Afin de ne pas surcharger ce chapitre, nous présentons en annexe C.5 quelques résultats concernant l'efficacité du modèle L.I.A.R.. Les temps de calculs des niveaux de réputation sont faibles et ont une progression linéaire avec le temps.

Enfin, nous avons montré que le modèle autorisait les agents qui l'implémentent à prendre des décisions qui, d'une part, leur permettent d'éliminer les agents malveillants et, d'autre part, de renforcer leurs relations avec les

agents de confiance.

Quatrième partie

Conclusion et Perspectives

Chapitre 8

Conclusions et perspectives

Dans ce chapitre, nous présentons les conclusions et les perspectives de ce travail de recherche. Nous commençons par rappeler la problématique générale et les objectifs, puis exposons notre démarche, nos contributions et leurs limites. Enfin, nous dressons les perspectives de ce travail.

8.1 Problématique et objectifs

Dans cette thèse, nous avons abordé la problématique du contrôle des interactions des agents dans les Systèmes Multi-Agents Ouverts et Décentralisés.

Dans un tel cadre, un contrôle adaptatif, auto-organisé et appliqué par les agents eux-mêmes, c'est-à-dire un contrôle *social*, est plus adapté. Nous avons donc décidé de définir un modèle de réputation qu'un agent peut implémenter, afin de participer au contrôle social des interactions des autres agents. Le modèle de réputation doit permettre aux agents qui l'implémentent de **caractériser** les interactions qu'ils perçoivent et de les **sanctionner** en conséquence. La sanction peut être positive ou négative, suivant la nature des interactions. Durant la phase de caractérisation, les agents doivent confronter les interactions qu'ils perçoivent à des définitions de l'acceptabilité des interactions. Pour ce faire, il est nécessaire que les agents soient capables : de **modéliser les interactions** qu'ils perçoivent, de **définir l'acceptabilité** des interactions et de déployer un processus permettant d'**évaluer** les premières par rapport aux secondes.

8.2 Démarche suivie

Pour résoudre le problème défini précédemment, nous avons suivi la démarche suivante. Dans la partie I, nous avons commencé par étudier les modèles existants dans la littérature concernant les différentes composantes du contrôle social.

Dans le chapitre 2, nous avons étudié comment un agent peut modéliser les interactions qu'il perçoit à l'aide d'engagements sociaux. Ensuite, nous avons vu comment il peut caractériser l'acceptabilité des interactions à l'aide de normes. Nous avons aussi étudié par quels processus un agent peut comparer les interactions qu'il a observées pour déterminer leur degré d'acceptabilité. Enfin, nous avons catalogué les différentes sanctions qu'un agent peut appliquer à l'un de ses pairs. Nous avons ainsi observé que les notions de confiance et de réputation constituent un moyen pour un agent de sanctionner ses pairs de manière adaptée aux systèmes décentralisés.

Les chapitres 3 et 4 se sont ensuite focalisés sur ces notions de confiance et de réputation. Le chapitre 3 s'est concentré sur leurs définitions dans les sciences économiques, humaines et sociales. Pour sa part, le chapitre 4 s'est intéressé aux modèles computationnels. Nous avons alors établi que les modèles de réputation existants souffrent de lacunes au niveau de la phase de caractérisation des interactions. Or, les modèles d'engagement social et de norme ainsi que les processus d'évaluation existants (qui permettent aux agents d'estimer l'acceptabilité d'une interaction observée) sont peu adaptés au cadre décentralisé. Nous avons alors proposé, dans la partie II, le modèle L.I.A.R., qui contribue au contrôle social des interactions dans les SMAOD. Enfin, dans la partie III, nous avons proposé quelques expérimentations sur le modèle L.I.A.R., dans le cadre d'un scénario d'échange d'informations dans un réseau pair-à-pair.

8.3 Contributions

La première contribution majeure du présent travail de recherche concerne l'état de l'art sur les modèles de réputation. Par une étude approfondie des concepts de confiance et de réputation dans les sciences économiques, humaines et sociales, nous avons pu caractériser précisément ce que sont la réputation fondée sur les interactions et la décision de faire confiance. Au cours de cette étude, nous avons aussi extrait une grille d'analyse, qui nous

a permis de définir des catégories pour les modèles computationnels de réputation, en fonction des propriétés qu'ils implémentent ou non. Nous avons alors remarqué que ces modèles pêchent principalement par leur implantation du processus d'évaluation. Quand celui-ci n'est pas absent, il requiert l'intervention humaine ou bien s'applique à des situations très restreintes.

La deuxième contribution majeure du travail présenté ici réside non pas dans le fait même que nous proposions des modèles d'engagement social, de norme et de réputation, mais surtout dans leur intégration au sein du modèle L.I.A.R. Cela permet d'implémenter de manière entièrement automatique la boucle de rétroaction caractérisant l'apprentissage des réputations et de poser les premiers jalons d'un modèle de contrôle social dans un système multi-agent ouvert et décentralisé.

Parmi les contributions plus précises, nous avons défini un modèle de réputation utilisant un domaine de représentation original et permettant aux agents de maintenir séparément différents types de réputation. Les différents types de réputation utilisés dans ce modèle ont été définis formellement, grâce aux rôles que les agents peuvent jouer, aux types d'information qu'ils peuvent échanger au cours du processus de punition et aux propriétés des réputations. Ainsi, les réputations sont distinguées par facette et par dimension. Afin de permettre aux agents de déterminer les niveaux des réputations qu'ils associent aux autres, nous avons aussi proposé un modèle d'engagement social, un modèle de normes et un processus de détection de la violation (ou du respect) de ces normes. Ces derniers permettent aux agents de déterminer de manière totalement automatique des évaluations des interactions qu'ils observent. Ces modèles sont adaptés aux systèmes décentralisés, puisqu'ils prennent en compte le fait que les agents ont leurs propres représentations locales des engagements sociaux et des normes, qui peuvent être incomplètes et imparfaites.

8.4 Limites

Parmi les différents modèles et processus que nous avons proposé dans cette thèse, nous avons déterminé les limites suivantes.

Les limites du modèle d'engagement social concernent principalement son expressivité. Dans le cadre de cette thèse, les agents utilisent des engagements sociaux pour modéliser une simple interaction. La gestion de l'évolution du contenu d'un engagement social ne s'est pas avérée nécessaire. En consé-

quence, le modèle d'engagement social n'est pas adapté aux systèmes où les agents sont autorisés à argumenter pour défendre leurs points de vues différents sur le monde.

Les limites concernant le modèle de normes se situent majoritairement au niveau des types de normes considérées. Le modèle de normes de L.I.A.R. permet de modéliser des s-normes, au sens de la typologie de [Tuo95, TBT95]. Dans le cadre d'institutions centralisées, ce sont plutôt des r-normes qui sont utilisées. Le modèle L.I.A.R. ne peut donc actuellement pas tirer parti de la présence d'institutions centralisées pour permettre aux agents de mettre à jour leurs réputations.

Le processus d'évaluation repose sur une signature digitale des messages qui s'avère difficile à mettre en place dans les systèmes décentralisés, comme expliqué dans [AJ02].

Enfin, certains processus du modèle de réputation n'ont pas été étudiés dans toute leur étendue. Ainsi en est-il de l'ordre des réputations qui est utilisé dans le processus de raisonnement. Pour sa part, le processus de propagation des réputations se fonde sur l'hypothèse que les agents envoient le plus possible de recommandations. En effet, si celles-ci sont correctes, cela bénéficie à l'ensemble du système de contrôle. Cependant, du point de vue d'un agent, la stratégie consistant à diffuser largement l'information qu'il possède n'est pas nécessairement la plus pertinente. Ce processus repose aussi sur une règle de réciprocité assez élémentaire, consistant à n'envoyer des recommandations qu'aux agents qui ont eux-mêmes une bonne réputation comme recommandeurs. Enfin, les recommandations que les agents échangent ne peuvent être comprises par le récepteur que s'il en connaît la sémantique.

8.5 Perspectives

Afin de répondre aux limites identifiées précédemment, il est possible d'identifier des perspectives à plus ou moins long terme. Nous présentons d'abord les perspectives à court terme, puis celles à plus long terme.

Parmi les perspectives à court terme, une extension possible de notre modèle d'engagement social serait de lui ajouter la possibilité de gérer le cycle de vie du contenu. Le modèle L.I.A.R. permettrait alors aux agents d'argumenter, lorsque leurs points de vue sur le monde diffèrent.

Le modèle de normes pourrait être enrichi par la prise en compte d'autres types de normes, comme les r-normes. Les agents implémentant le modèle

L.I.A.R. pourraient alors tirer parti de la présence d'institutions centralisées. Dans [GVSM06], nous établissons quelques pistes en ce sens.

Un processus de détection de violation des normes plus souple, par exemple s'appuyant sur la réputation plutôt que sur des signatures électroniques, éviterait la mise en place difficile de systèmes cryptographiques.

Les travaux à plus long terme concernent une étude plus approfondie des processus de raisonnement et de propagation des réputations. Tout d'abord, il serait intéressant d'étudier les éléments extérieurs qui peuvent influencer le choix d'un agent sur l'ordre des réputations lors du processus de raisonnement. Ensuite, l'étude des relations entre les notions de confiance et de *« privacy »* permettrait de déterminer plus précisément quand, à qui et pourquoi les agents doivent ou ne doivent pas partager l'information qu'il possèdent sur leur propre réputation ou celle des autres. Enfin, l'utilisation d'une ontologie [CS05, VCSB07] de la réputation autoriserait les agents ayant des modèles de réputation différents à communiquer et à déterminer plus précisément comment diffuser les recommandations.

Enfin, dans les travaux en cours, il semble important de poursuivre les expérimentations, par exemple en déployant dans la population des agents ayant des paramètres différents ($\tau_{\mathcal{X}}$, $\theta^{\text{trust}}_{\mathcal{X}\texttt{bRp}}$, etc.) ou bien en autorisant les paramètres caractérisant le taux de contradiction à évoluer au cours du temps, pour plus de réalisme. Enfin, il pourrait être intéressant de confronter le modèle L.I.A.R. à d'autres modèles de la littérature. Cependant, les cadres d'applications et les métriques employées divergent tellement entre les modèles actuels, qu'il est difficile d'opérer une telle comparaison. C'est pourquoi nous nous sommes impliqués dans le projet ART-testbed [Tes04], visant à définir un banc d'essais pour les modèles de réputation. Ce projet étant très récent, des expérimentations avec le modèle L.I.A.R. sont en cours, mais leurs résultats ne peuvent être présentés dans le cadre de cette thèse.

Publications directes

[GVSM06] A. Grizard, L. Vercouter, T. Stratulat, and G. Muller. A peer-to-peer normative system to achieve social order. In V. Dignum, N. Fornara, and P. Noriega, editors, Proceedings of the Workshop on "Coordination, Organization, Institutions and Norms" (COIN) at Autonomous Agents and Multi-Agent Systems (AAMAS'06), Lecture Notes in Computer Science, Hakodate, Japan, May 2006. Springer-Verlag, Berlin, Germany. (in press). 175

[MV04a] G. Muller and L. Vercouter. Détection décentralisée d'agents menteurs. In O. Boissier and Z. Guessoum, editors, Actes des Journées Francophones sur les Systèmes Multi-Agents (JFSMA'04), pages 243–248. Hermès Sciences, novembre 2004.

[MV04b] G. Muller and L. Vercouter. Liar detection within agent communication. In W. van der Hoek, A. Lomuscio, E. de Vink, and M. Wooldridge, editors, Proceedings of the Workshop on "Logic and Communication in Multi-Agent Systems" (LCMAS'04), pages 4–16, Nancy, France, August 2004.

[MV05a] G. Muller and L. Vercouter. Decentralized monitoring of agent communications with a reputation model. In R. Falcone, S. Barber, J. Sabater, and M.P. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'04), volume 3577 of Lecture Notes in Artificial Intelligence. Trusting Agents for trusting Electronic Societies, pages 144–161. Springer-Verlag, Berlin, Germany, 2005.

[MV05b] G. Muller and L. Vercouter. Using social commitments to control the agents' freedom of speech. In F. Dignum *et al.*, editor, Proceedings of the Workshop on "Agent Communications,

Languages and Conversation Policies" at Autonomous Agents and Multi-Agent Systems (AAMAS'05), volume 3859 of Lecture Notes in Artificial Intelligence, pages 109–123, Utrecht, The Netherlands, July 2005. Springer-Verlag, Berlin, Germany.

[MVB03] G. Muller, L. Vercouter, and O. Boissier. Towards a general definition of trust and its application to openness in MAS. In R. Falcone, K.S. Barber, L. Korba, and M. Singh, editors, Proceedings of the Workshop on "Deception, Fraud and Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'03), pages 49–56, Melbourne, Australia, July 2003.

[MVB05] G. Muller, L. Vercouter, and O. Boissier. A trust model for the reliability of agent communications. In C. Castelfranchi, S. K. Barber, J. Sabater, and M. P. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'05), Utrecht, The Netherlands, July 2005.

Durant ces travaux de thèse, nous avons également participé activement au groupe de travail international « ART testbed » (Agent Reputation and Trust testbed) dont l'objectif est de fournir un banc d'essai procurant un environnement d'expérimentation et de comparaison pour les modèles de réputation et de confiance. Plusieurs communications ont été réalisées sur la plateforme développée afin de la faire connaître largement auprès de différentes communautés scientifiques (pour cette raison les publications [FKM+05c] et [FKM+05e], ont sensiblement le même contenu, mais ont été présentées à différentes communautés scientifiques ; la même remarque étant valable pour les démonstrations [FKM+05d, FKM+05a, FKM+06a, FKM+06b]). La liste des publications produites par le groupe ART testbed suit.

Publications indirectes

[FKM+05a] K. Fullam, T. Klos, G. Muller, J. Sabater, A. Schlosser, Z. Topol, K. S. Barber, J. S. Rosenschein, L. Vercouter, and M. Voss. A demonstration of the agent reputation and trust (ART) testbed : Experimentation and competition for Trust in Agent Societies. In Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'05), Utrecht, The Netherlands, July 2005. ACM Press, New York, NY, United States of America.

[FKM+05b] K. Fullam, T. Klos, G. Muller, J. Sabater, A. Schlosser, Z. Topol, K. S. Barber, J. S. Rosenschein, L. Vercouter, and M. Voss. A specification of the agent reputation and trust (ART) testbed : Experimentation and competition for Trust in Agent Societies. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M. P. Singh, and M. Wooldridge, editors, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'05), pages 512–518, Utrecht, The Netherlands, July 2005. ACM Press, New York, NY, United States of America. 145

[FKM+05c] K. Fullam, T. Klos, G. Muller, J. Sabater, Z. Topol, K. S. Barber, J. S. Rosenschein, and L. Vercouter. The agent reputation and trust (ART) testbed architecture. In C. Castelfranchi, S. Barber, J. Sabater, and M. P. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'05), volume 2, pages 50–62, July 2005.

[FKM+05d] K. K. Fullam, T. B. Klos, G. Muller, J. Sabater, Z. Topol, K. S. Barber, J. S. Rosenschein, and L. Vercouter. Le banc d'essais ART (agent reputation and trust) pour les modèles de confiance. In A. Drogoul and É. Ramat, editors, Actes des Journées

Francophones sur les Systèmes Multi-Agents (JFSMA'05), pages 175–179, Calais, France, novembre 2005. Hermès Sciences.

[FKM+05e] K. K. Fullam, T. B. Klos, G. Muller, J. Sabater i Mir, Z. Topol, K. S. Barber, J. S. Rosenschein, and L. Vercouter. The agent reputation and trust (art) testbed architecture. In B. López, J. Meléndez, P. Radeva, and J. Vitrià, editors, Proceedings of the Congrès Català d'Intel.ligència Artificial (CCIA'05), volume 131 of Frontiers in Artificial Intelligence and Applications, pages 389–396, Alguer, Spain, October 2005. Associació Catalana d'Intelligència Artificial, IOS Press, Amsterdam, The Netherlands.

[FKM+06a] K. K. Fullam, T. B. Klos, G. Muller, J. Sabater-Mir, K. S. Barber, and L. Vercouter. The agent reputation and trust (`art`) testbed. In K. Stølen, W. H. Winsborough, F. Martinelli, and F. Massacci, editors, Proceedings of the International Conference on Trust Management (iTrust'06), volume 3986 of Lecture Notes in Computer Science, pages 439–442, Pisa, Italy, May 2006. Springer-Verlag, Berlin, Germany.

[FKM+06b] Karen K. Fullam, Tomas B. Klos, Guillaume Muller, Jordi Sabater-Mir, K. Suzanne Barber, and Laurent Vercouter. The agent reputation and trust (ART) testbed. In Belgium-Netherlands Conference on Artificial Intelligence (BNAIC'06), pages 449–450, 2006.

Cinquième partie

Annexes

Annexe A

Compléments sur l'état de l'art

Dans cette annexe, nous donnons quelques complément concernant la grille qui nous a servi à catégoriser les modèles computationnels de réputation.

A.1 Propriétés des modèles computationnels

Pour une discussion détaillée sur l'implémentation des processus, voir section 4.7, page 65. En substance, en partant de l'observation que chaque modèle computationnel de réputation n'implémente généralement qu'une faible partie des processus de manipulation des réputations, nous avons pu construire une catégorisation des modèles : depuis ceux qui nécessitent une constante intervention humaine, jusqu'à ceux qui peuvent être implémentés directement dans des agents logiciels autonomes. Le processus qui freine le plus l'implémentation des modèles de réputation dans des agents est le processus de révision instantanée.

La table A.1 et les discussions menées ici se réfèrent principalement aux aspects : cadre, propriétés et mise-en-œuvre de la grille d'analyse 4.1, page 54. Dans cette table, O (resp. N) signifie que la propriété est (resp. n'est pas) implémentée dans le modèle. H signifie que le modèle nécessite une intervention humaine. X signifie qu'un champ n'est pas pertinent pour le modèle. Finalement, B, T et M correspondent, respectivement, aux comportements attendus : bienveillants, triviaux ou malveillants. Les chiffres correspondent aux nombres de facettes ou dimensions considérées dans le modèle. Un + signifie « plusieurs ». La notation 1/+ signifie que certaines réputations sont,

par exemple, uni-facettes alors que d'autres sont multi-facettes.

Il ressort de l'analyse de cette table que la plupart des modèles, en dehors des modèles de réputations effectivement déployés sur la toile [eBa03, OnS03, ZMM99], manipulent des réputations subjectives.

D'autre part, durant une première phase exploratoire [Mar94], les modèles étaient relativement riches (en particulier multi-facettes), mais difficilement implémentables dans des agents autonomes. Durant une deuxième phase [SFR99, SF99, SS02], les modèles se sont simplifiés (uni-facettes), mais sont devenus implémentables dans des agents autonomes. Plus récemment, une troisème phase a vu le jour, où des modèles riches commencent à être proposés pour des agents logiciels [WV03, Sab02], en particulier grâce à la prise en compte du caractère social de l'agent.

Si l'introduction de nombreuses facettes est assez courante dans les modèles actuels [WV03, MD05, Sab02, Abd04], en revanche, l'introduction de multiples dimensions est encore peu commune [MD05, Sab02].

De très nombreux domaines de valeurs sont utilisés pour représenter les réputations, mais de plus en plus, ils s'enrichissent : contrairement aux ensemble discrets et finis de certains modèles [Abd04], de plus en plus de chercheurs utilisent des ensembles continus et infinis tel le domaine $[-1, +1]$ [Sab02, Mar94]. D'autres vont encore plus loin et proposent des représentations plus riches telles que les réseaux bayésiens [WV03, MD05] ou les ensembles flous [CMD03, RPBF05]. De la même manière, on observe une tendance à l'enrichissement des modèles quant aux nombre de types de réputations différents qu'ils prennent en compte (voir table A.2, où Directe+Observée signifie « mélange de réputation Directe et Observée » et Observé|Propagée signifie « réputation Observée ou Propagée »).

De plus en plus de chercheurs considèrent que les réputations ne sont pas transitives [Abd04, Sab02], par conséquence de leur subjectivité : un agent n'ayant pas accès aux croyances d'un autre, il ne peut donc avoir une valeur fiable de la réputation que celui-ci accorde à un troisième agent (*cf.* discussion section 3.2.5, page 50).

Enfin, les comportements attendus indiqués dans la table A.1 sont le plus possible ceux qui sont précisés par les auteurs. Généralement, il s'agit de comportements malveillants, tant au niveau des différentes facettes considérées, que pour les recommandations. On remarque que, paradoxalement, certains auteurs attendent des comportements bienveillants ou triviaux pour les recommandations [SFR99, SF99, SS02, WV03]. Quand cet aspect n'est pas précisé dans le modèle [OP05, Abd97, eBa03], nous l'avons estimé par

	Cadre	Propriétés						Comportements	
					graduation				
		subj.	fac.	dim.	évaluation	réputation	trans.	Recom.	Réput.
OpenPGP	partage de clefs	O	1	H	X	{u,m,c}/{f,m,u,d}	O	X	M
eBay	enchères	N	1	H	{-1,0,+1}	$[0,+\infty[$	N	M	M
Zacharias	enchères	N	1	H	[0.1,1]	[0,3000]	O/N	M	M
AbdulRahman	pair-à-pair	O	1/+	X	{-2,...,+2}	{-2,...,+2}	N	B	M
Marsh	coopération	O	1/+	X	N	$[0,+1[$	N	X	M
AFRAS	e-commerce	O	1	X	ensemble flou	ensemble flou	N	M	M
Schillo	théorie des jeux	O	1	2	{0,1}	[0,1]	N	T	T
Sen	exécution de tâches	O	1	1	{0,1}	[0,1]	N	T	T
Wang	pair-à-pair	O	1/+	1	{0,1}	[0,1]	N	B	M
Melaye	X	O	1/+	+	{0,1}	[0,1]	N	X	M
Sabater	e-commerce	O	+	+	[-1,1]	[-1,+1]	N	M	M

TAB. A.1 – Caractéristiques des modèles computationnels.

Légende :							
O	oui	N	non	subj.	subjectivité	Recom.	Recommandation
H	humain	X	non pertinent	fac.	facettes	Réput.	Réputation
B	bienveillant	T	trivial	dim.	dimensions		
M	malveillant	+	plusieurs	trans.	transitivité		
a/b	parfois a, parfois b						

	Cadre	Types
OpenPGP	partage de clefs	Directe
eBay	enchères	Directe+Observée
Zacharias	enchères	Directe+Observée (+fiabilités)
AbdulRahman	pair-à-pair	Directe Collective Stéréotypée
Marsh	coopération	Directe Stéréotypée
AFRAS	e-commerce	Directe+Observée
Schillo	théorie des jeux	Directe Observée
Sen	exécution de tâches	Directe Observée
Wang	pair-à-pair	Directe Observée
Melaye	X	Directe
Sabater	e-commerce	Directe Propagée Collective Stéréotypée (+fiabilités)

TAB. A.2 – Types de réputation dans les modèles computationnels.

rapport à la mise en application du modèle.

Annexe B

Compléments sur le modèle L.I.A.R.

Dans cette annexe, nous proposons quelques compléments à la description du modèle L.I.A.R. donnée en partie II, page 73. En particulier, nous précisons les algorithmes de raisonnement et de décision.

B.1 Pseudo-code du processus de raisonnement

Dans cette section est proposé le pseudo-code du processus de raisonnement. Dans le pseudo-code qui suit, nous avons :

$$
\begin{aligned}
{}_{\mathtt{bn}}\mathbf{NCS}'^{\mathcal{A}}_{\mathtt{tg}}(\alpha, \delta, \mathtt{t}) = \{\ \mathbf{sp} \in {}_{\mathtt{bn}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\mathtt{t}) / \\
\alpha \in \mathtt{bn.facets}(\mathbf{sp}.\mathtt{cont}) \wedge \delta \in \mathtt{bn.dimensions}(\mathbf{sp}) \\
\wedge(\mathbf{sp}.\mathtt{st} \in \{\mathtt{fulfilled}, \mathtt{violated}, \mathtt{cancelled}\})\ \}
\end{aligned}
$$

où ${}_{\mathtt{bn}}\mathbf{NCS}^{\mathcal{A}}_{\mathtt{tg}}(\mathtt{t})$ est défini page 118. Les $\mathbf{NCS}'$ correspondent aux historiques de politiques sociales en état terminal, groupées par facettes et par dimensions.

Nous définissons aussi *intention_t* comme une structure contenant un booléen et une valeur réelle :

```
STRUCT intention_t :
 trust_int : BOOL
 trust_val : FLOAT
ENDSTRUCT
```

$\underline{\texttt{bn.reasons}(\texttt{tg}, \alpha, \delta, \texttt{Lev}, \texttt{t}) \ : \text{set of } \textit{intention\_t}}$
BEGIN
 res : set of $\textit{intention\_t} \leftarrow \emptyset$
 inten : $\textit{intention\_t}$

 // si (assez d'interactions directes et
 // valeur non unknown et valeur discriminante)
 IF $(|_{\texttt{bn}}\mathbf{NCS}'^{\{\texttt{bn}\}}_{\texttt{tg}}(\alpha, \texttt{ctxt}.\delta, \texttt{t})| > \texttt{ctxt}.\theta^{\text{relevance}}_{\texttt{DIbRp}}$
  AND $\texttt{DIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t}) \neq \texttt{unknown}$
  AND $(\texttt{DIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t}) > \texttt{ctxt}.\theta^{\text{trust}}_{\texttt{DIbRp}}$
  OR $\texttt{DIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t}) < \texttt{ctxt}.\theta^{\text{distrust}}_{\texttt{DIbRp}}))$
 // si suffisant pour confiance
 THEN IF $(\texttt{DIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t}) > \texttt{ctxt}.\theta^{\text{trust}}_{\texttt{DIbRp}})$
  THEN inten.trust_int $\leftarrow$ trust
   inten.trust_val $\leftarrow \texttt{DIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t})$
  // si suffisant pour défiance
  ELSE inten.trust_int $\leftarrow$ distrust
   inten.trust_val $\leftarrow \texttt{DIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t})$
  ENDIF
 // si (assez d'interactions indirectes et
 // valeur non unknown et valeur discriminante)
 ELSE IF $(|_{\texttt{bn}}\mathbf{NCS}'^{\Omega_{\texttt{bn}}(\texttt{t})\setminus\{\texttt{bn}\}}_{\texttt{tg}}(\alpha, \delta, \texttt{t})| > \texttt{ctxt}.\theta^{\text{relevance}}_{\texttt{IIbRp}}$
    AND $\texttt{IIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t}) \neq \texttt{unknown}$
    AND $(\texttt{IIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t}) > \texttt{ctxt}.\theta^{\text{trust}}_{\texttt{IIbRp}}$
    OR $\texttt{IIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t}) < \texttt{ctxt}.\theta^{\text{distrust}}_{\texttt{IIbRp}}))$
  // si suffisant pour confiance
  THEN IF $(\texttt{IIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t}) > \texttt{ctxt}.\theta^{\text{trust}}_{\texttt{IIbRp}})$
   THEN inten.trust_int $\leftarrow$ trust
    inten.trust_val $\leftarrow \texttt{IIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t})$
   // si suffisant pour défiance
   ELSE inten.trust_int $\leftarrow$ distrust
    inten.trust_val $\leftarrow \texttt{IIbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t})$
   ENDIF
  // si (assez de recommandations d'observation et
  // valeur non unknown et valeur discriminante)
  ELSE IF $(|\bigcup_{\mathcal{X}\in\{F,V,C\}} {}_{\texttt{bn}}\mathbf{ObsRc}\mathcal{X}\mathbf{NCS}(\texttt{tg}, \alpha, \delta, \texttt{t})| > \texttt{ctxt}.\theta^{\text{relevance}}_{\texttt{ObsRcbRp}}$
     AND $\texttt{ObsRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha, \delta, \texttt{t}) \neq \texttt{unknown}$

AND ($\texttt{ObsRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) > \texttt{ctxt}.\theta^{\text{trust}}_{\texttt{ObsRcbRp}}$
OR $\texttt{ObsRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) < \texttt{ctxt}.\theta^{\text{distrust}}_{\texttt{ObsRcbRp}}$))
// si suffisant pour confiance
THEN IF ($\texttt{ObsRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) > \texttt{ctxt}.\theta^{\text{trust}}_{\texttt{ObsRcbRp}}$)
THEN inten.trust_int ← trust
inten.trust_val ← $\texttt{ObsRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t})$
// si suffisant pour défiance
ELSE inten.trust_int ← distrust
inten.trust_val ← $\texttt{ObsRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t})$
ENDIF
// si (assez de recommandations d'évaluation et
// valeur non unknown et valeur discriminante)
ELSE IF ($|\bigcup_{\mathcal{X}\in\{F,V,C\}} {}_{\texttt{bn}}\mathbf{EvRc}\mathcal{X}\mathbf{NCS}(\texttt{tg},\alpha,\delta,\texttt{t})| > \texttt{ctxt}.\theta^{\text{relevance}}_{\texttt{EvRcbRp}}$
AND $\texttt{EvRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) \neq \texttt{unknown}$
AND ($\texttt{EvRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) > \texttt{ctxt}.\theta^{\text{trust}}_{\texttt{EvRcbRp}}$
OR $\texttt{EvRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) < \texttt{ctxt}.\theta^{\text{distrust}}_{\texttt{EvRcbRp}}$))
// si suffisant pour confiance
THEN IF ($\texttt{EvRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) > \texttt{ctxt}.\theta^{\text{trust}}_{\texttt{EvRcbRp}}$)
THEN inten.trust_int ← trust
inten.trust_val ← $\texttt{EvRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t})$
// si suffisant pour défiance
ELSE inten.trust_int ← distrust
inten.trust_val ← $\texttt{EvRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t})$
ENDIF
// si (assez de recommandations de réputation et
// valeur non unknown et valeur discriminante)
ELSE IF ($|{}_{\texttt{bn}}\texttt{TRpRc}(\texttt{tg},\alpha,\delta,\texttt{t})| > \texttt{ctxt}.\theta^{\text{relevance}}_{\texttt{RpRcbRp}}$
AND $\texttt{RpRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) \neq \texttt{unknown}$
AND ($\texttt{RpRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) > \texttt{ctxt}.\theta^{\text{trust}}_{\texttt{RpRcbRp}}$
OR $\texttt{RpRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) < \texttt{ctxt}.\theta^{\text{distrust}}_{\texttt{RpRcbRp}}$))
// si suffisant pour confiance
THEN IF ($\texttt{RpRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t}) > \texttt{ctxt}.\theta^{\text{trust}}_{\texttt{RpRcbRp}}$)
THEN inten.trust_int ← trust
inten.trust_val ← $\texttt{RpRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t})$
// si suffisant pour défiance
ELSE inten.trust_int ← distrust
inten.trust_val ← $\texttt{RpRcbRp}^{\texttt{tg}}_{\texttt{bn}}(\alpha,\delta,\texttt{t})$

```
        ENDIF
       // cas par défaut
       ELSE IF (bnGDtT=true)
        THEN inten.trust_int ← trust
         inten.trust_val ← 0.0
        ELSE inten.trust_int ← distrust
         inten.trust_val ← 0.0
        ENDIF
       ENDIF
      ENDIF
     ENDIF
    ENDIF
   ENDIF
  ENDIF
  res.add(inten)
RETURN(res)
END
```

B.2 Pseudo-code des processus de décision

Dans cette section sont proposés des pseudo-codes pour les deux types de décisions que peut prendre un agent.

Une décision de type sélection, correspond à décider si un agent donné est digne de confiance. Nous proposons l'algorithme suivant pour que le bénéficiaire bn décide s'il est d'envoyer ou non un message de contenu sensible content à la cible tg :

```
bn.décision(tg, ctxt, t) :_
BEGIN
  // choisir la facette et la dimension les plus adaptées au contexte
  bn.selectFacDim(ctxt, α, δ)
  IF (bn.reasons(tg, α, δ, ctxt.Lev, t).trust_int=trust)
  // modifier les états mentaux pour décider d'envoyer le message
  THEN bn.add_intention(bn.send_message(tg, content))
  // modifier les états mentaux pour décider de ne pas envoyer le message
  ELSE
  THEN bn.add_intention(¬bn.send_message(tg, content))
END
```

Où `selectFacDim` retourne la facette α et la dimension δ que `bn` considèrent les plus adaptées au contexte `ctxt`. `add_intention` correspond à l'ajout d'une intention (de type BDI [RG91], à ne pas confondre avec les intentions de confiance) aux états mentaux de l'agent. L'algorithme de décision proposé ici consiste simplement à tester si la partie booléenne de l'intention de confiance penche dans le sens de faire confiance à l'agent `tg` ou, au contraire, à ne pas lui faire confiance. Dans le cas positif, une intention d'envoyer le message est ajoutée aux états mentaux de `bn`, dans le cas négatif, une intention de ne pas envoyer le message est ajoutée aux états mentaux de `bn`.

Une décision de type tri, consiste à ordonner un ensemble d'agents en fonction de la confiance qui leur est accordée. Nous proposons l'algorithme suivant pour sélectionner un sous-ensemble de $n \leq m$ agents les plus dignes de confiance, à qui il serait possible d'envoyer de l'information sensible :

`bn.décision(Tg, ctxt, t, n)` :_

```
BEGIN
 STRUCT trust_intention_type :
  trust_int : BOOL
  trust_val : FLOAT
 ENDSTRUCT

 STRUCT cell :
  t_int : trust_intention_type
  position : INTEGER
 ENDSTRUCT

 t_int_and_pos : cell[|Tg|]
 cpt : INTEGER ← 0
 nb : INTEGER ← 0
 i : INTEGER ← 0
 ti : trust_intention_type

 bn.selectFacDim(ctxt, α, δ)
 FORALL tg ∈ Tg DO
  ti ← bn.reasons(tg, α, δ, ctxt.Lev, t)
  t_int_and_pos[i].t_int ← ti
  t_int_and_pos[i].position ← i
  i = i + 1
```

```
 DONE
 quick_sort_decreasing(t_int_and_pos)
 FORALL cpt ∈ [0..|Tg| − 1] DO
  IF ((nb < n) AND (t_int_and_pos[i].t_int=trust))
  THEN bn.add_intention(
   bn.send_message(Tg.get(t_int_and_pos[i].position), content))
  nb = nb + 1
  ELSE bn.add_intention(
   ¬bn.send_message(Tg.get(t_int_and_pos[i].position), content))
  ENDIF
 DONE
END
```

Où `quick_sort_decreasing` effectue un tri rapide par ordre décroissant du tableau, en s'appuyant sur la composante réelle des intentions de confiance contenues dans le tableau.

Cet algorithme consiste simplement à trier les agents en fonction de la composante réelle des intentions de confiance, puis à ajouter une intention de réaliser l'envoi de message pour les (au plus) n agents ayant les pondérations les plus fortes et pour lesquels l'intention est de faire confiance. Pour les autres agents, une intention de ne pas envoyer le message est créée.

Annexe C

Compléments aux expérimentations

Dans cette annexe, nous fournissons quelques expérimentations complémentaires à celles fournies dans la partie III, page 135. Ces dernières permettent de préciser l'influence de certains des paramètres du modèle L.I.A.R.

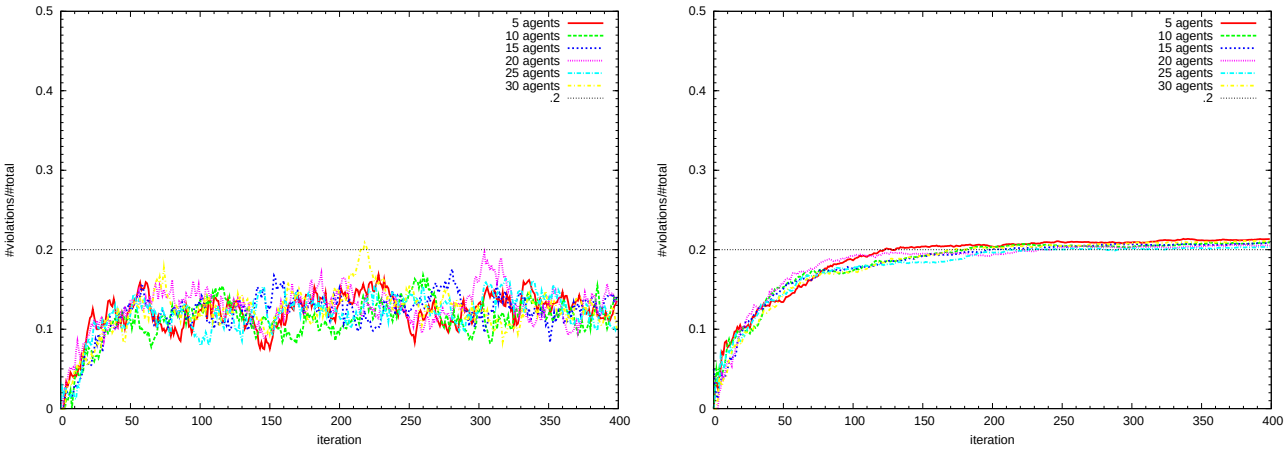
C.1 Choix des paramètres de simulation

Dans cette annexe, nous synthétisons les résultats des expérimentations qui nous ont permis de fixer les paramètres des simulations, en termes de nombre d'agents, de faits, d'interactions directes et d'itérations.

C.1.1 Nombre d'agents

Pour déterminer le nombre d'agents, nous avons lancé des simulations avec des agents ayant tous un `violationRate` de 0.2 et avons estimé le nombre de politiques sociales détectées comme étant violées par rapport au nombre total de politiques sociales, en fonction des deux stratégies d'instanciation. Les résultats sont présentés dans la figure C.1

Les résultats sont différents selon la stratégie déployée, mais le nombre d'agents n'influe pas sur la détection des violations de manière significative. Nous pouvons donc fixer le nombre d'agents comme il nous semble le plus judicieux. Nous fixons ce nombre d'agents à 11, d'une part pour réduire les ressources employées par la simulation et, d'autre part, de manière à



(a) indulgents (b) rancuniers

FIG. C.1 – Détermination du nombre d'agents.

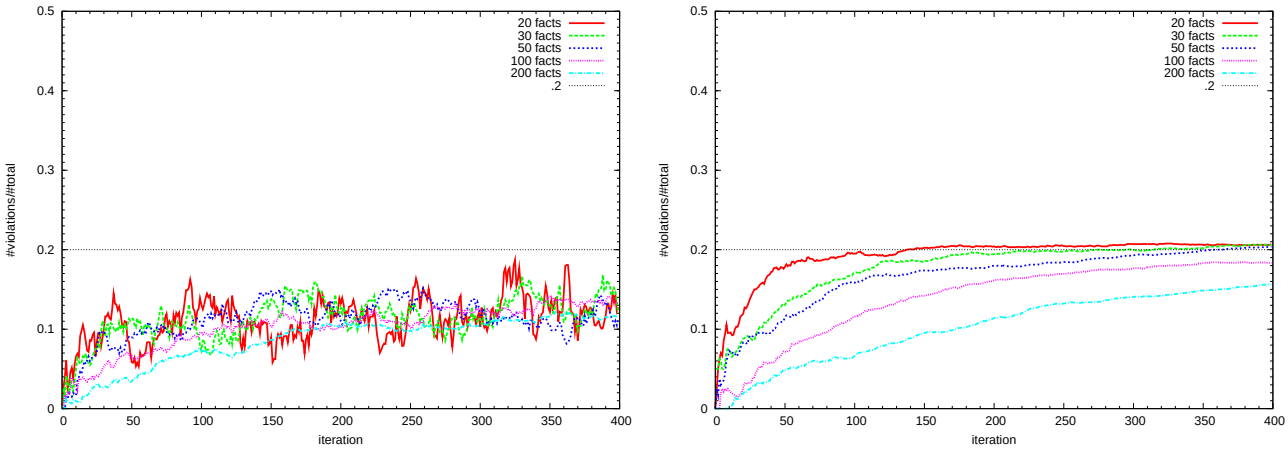
pouvoir répartir les `violationRate` et `lieRate` de façon pertinente : chaque agent $i \in 0, \ldots, 10$ est associé à un `violationRate` et un `lieRate` de $i/10$.

C.1.2 Nombre de faits

Pour déterminer le nombre de faits, nous avons lancé des simulations avec des agents ayant tous un `violationRate` de 0.2 et avons estimé le nombre de politiques sociales détectées comme étant violées par rapport au nombre total de politiques sociales, en fonction des deux stratégies d'instanciation. Les résultats sont présentés dans la figure C.2.

Quelle que soit la stratégie, le nombre de faits influe sur la vitesse de convergence du rapport entre le nombre de politiques sociales violées et total (voir la pente des courbes au démarrage). En effet, plus il y a de faits, plus un agent a de possibilités pour s'engager sans créer d'inconsistance. En conséquence, plus il a de faits, plus il faut de temps pour qu'un agent ait généré suffisamment d'engagements pour s'être contredit avec une moyenne de 20 %. Le temps de convergence du taux de politiques sociales détectées comme étant violées par rapport au nombre total de politiques sociales est donc plus grand si le nombre de faits différents sur lesquels un agent peut s'engager est plus grand. Par ailleurs, un nombre de faits plus important lisse les courbes dans le cas des agents indulgents.

Les agents rancuniers sont plus sensibles à cet effet, car ils considèrent tou-



(a) indulgents (b) rancuniers

FIG. C.2 – Détermination du nombre de faits.

jours l'ensemble des politiques sociales qu'ils instancient, alors que les agents indulgents finissent par ignorer certaines politiques sociales. Le poids d'une violation diminue donc avec le temps chez les agents rancuniers (inertie).

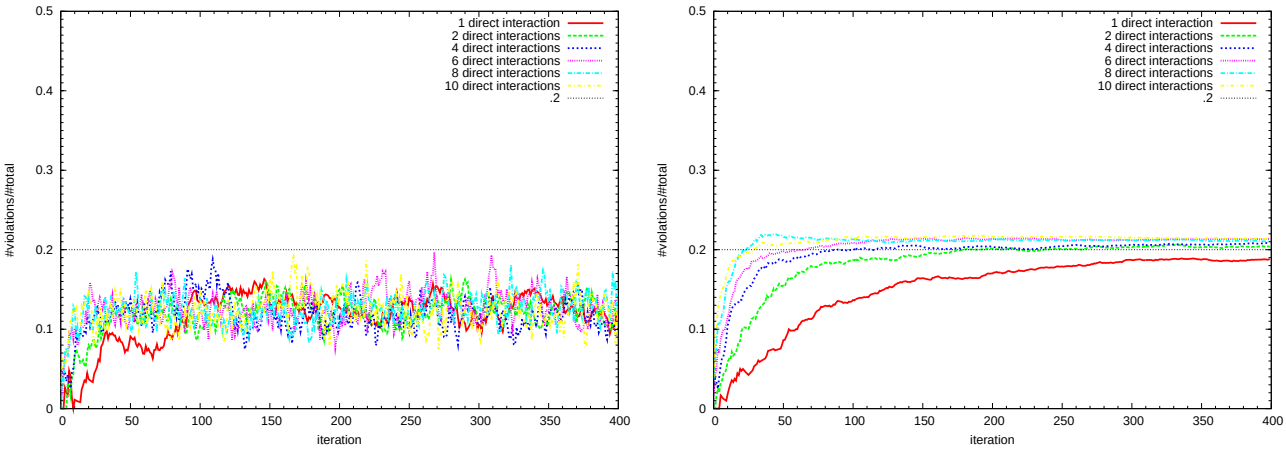
Pour conclure, pour un nombre d'agents fixé à 11 et un nombre de faits inférieur à 50, il faut moins de 100 itérations pour obtenir la convergence. Nous fixons le nombre de faits à 30.

C.1.3 Nombre d'interactions directes

Pour déterminer le nombre d'interactions directes, nous avons lancé des simulations avec des agents ayant tous un `violationRate` de 0.2 et avons estimé le nombre de politiques sociales détectées comme étant violées par rapport au nombre total de politiques sociales, en fonction des deux stratégies d'instanciation. Les résultats sont présentés dans la figure C.3.

Quelle que soit la stratégie, le nombre d'interactions directes par itération influe sur le temps que mettent les agents à détecter une moyenne de 20 % de contradictions. En effet, plus il y a d'interactions, plus un agent a de matière pour calculer le rapport du nombre de politiques sociales violées sur le nombre total de politiques. En conséquence, plus il a d'interactions, moins il faut de temps pour que la convergence apparaisse.

Par ailleurs, pour la même raison que précédemment (inertie), les agents rancuniers sont plus sensibles à cet effet.



(a) indulgents (b) rancuniers

FIG. C.3 – Détermination du nombre d'interactions directes.

Pour conclure, pour un nombre d'agents fixé à 11, 2 interactions directes suffisent pour obtenir une convergence en moins de 100 itérations. Nous fixons le nombre d'interactions directes à 2 par défaut.

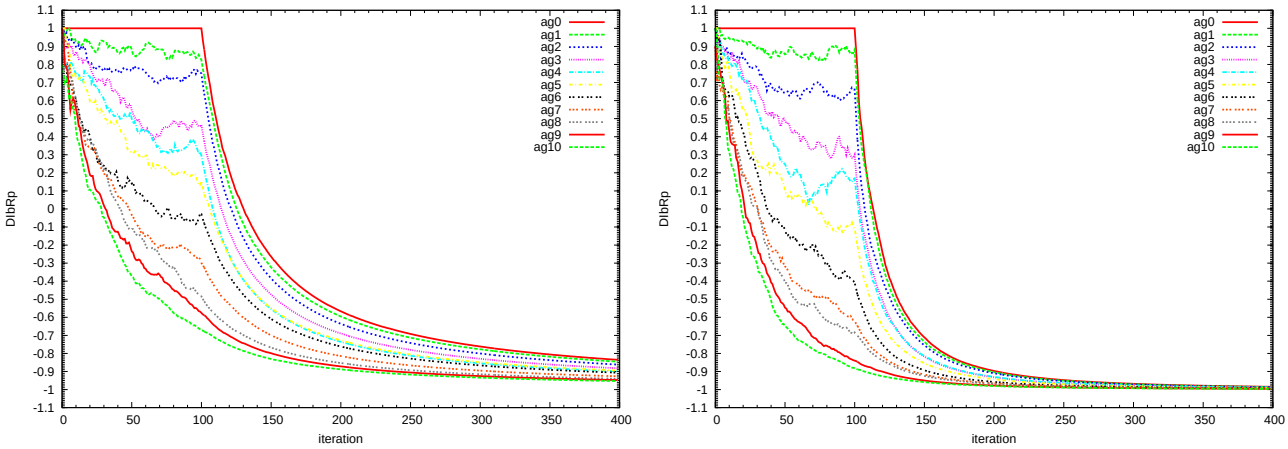
C.1.4 Nombre d'iterations

Étant donnés les paramètres fixés ci-dessus, il est possible d'observer une convergence avant 100 itérations. Cependant, dans d'autres expériences, les convergences peuvent se produire plus tard (c'est le cas, en particulier, quand nous abordons l'inertie, puisque nous faisons changer le comportements des agents à des instants assez avancés). Nous fixons donc de manière globale le nombre d'itérations à 400 pour les expériences. Ceci permet d'assurer que les phénomènes intéressants pourront être observés et de vérifier que des phénomènes parasites ne se produisent pas ensuite.

C.2 Pénalité des normes

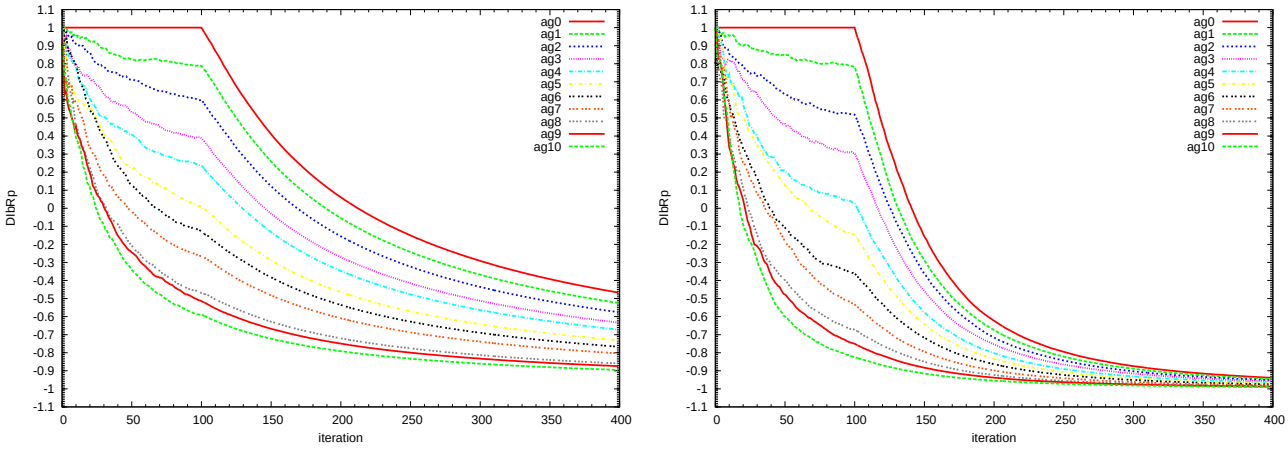
Dans cette annexe, nous présentons quelques résultats sur l'utilisation de la pénalité associée aux normes.

Les figures C.4(a), C.4(b), C.5(a) et C.5(b) présentent l'évolution du comportement de la Réputation fondée sur les Interactions Directes d'agents



(a) pénalité=1.0 (b) pénalité variable

FIG. C.4 – Influence de la pénalité des normes (indulgents).



(a) pénalité=1.0 (b) pénalité variable

FIG. C.5 – Influence de la pénalité des normes (rancuniers).

ayant différents `violationRate` quand un changement brusque de comportement intervient à l'itération 100, en fonction des stratégies d'instanciation des normes. Dans les figures C.4(a) et C.5(a) la pénalité associée aux normes vaut toujours 1.0. Dans les figures C.4(b) et C.5(b), elle vaut 1.0 quand la norme est remplie et i quand elle est violée, où i est le nombre d'engagements sociaux impliqués dans la contradiction.

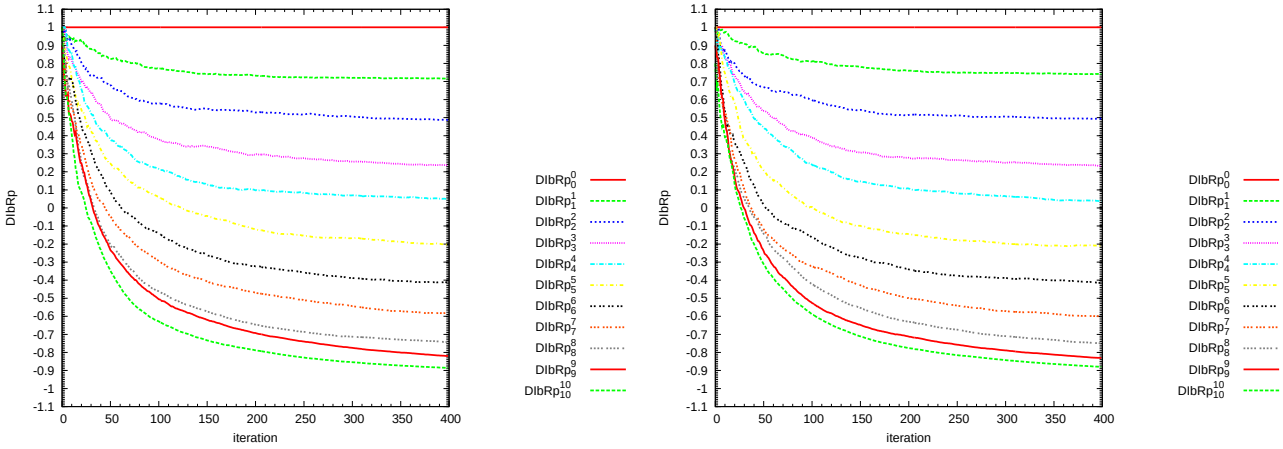
L'introduction de ce type de variabilité dans la pénalité induit une fragilité du point de vue de l'interaction (*cf.* section 3.2.5, page 48), puisque la chute de la réputation est plus forte pour les agents qui avaient une réputation déjà forte. Elle permet aussi d'introduire une fragilité temporelle, puisqu'il faudra, par exemple, à un agent cinq politiques sociales en état `fulfilled` pour contrer l'effet d'une violation impliquant cinq engagements sociaux. L'effet de cette configuration est plus visible chez les agents rancuniers du fait de l'inertie.

C.3 Influence des paramètres $\tau_{\mathcal{X}}$

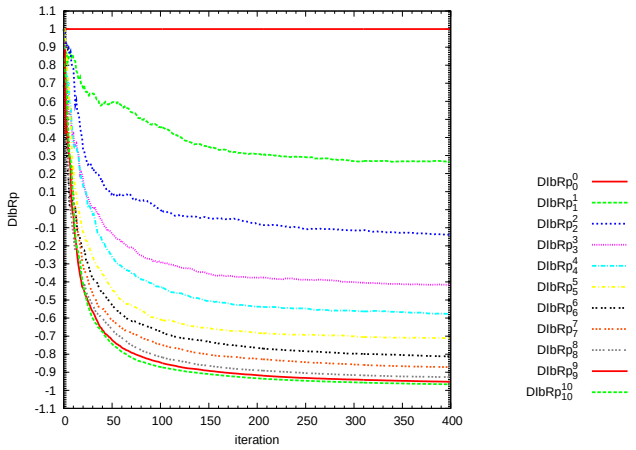
Dans cette section, nous étudions l'influence des paramètres $\tau_{\mathcal{X}}$ sur le calcul des Réputation fondée sur les Interactions Directes. Du fait que les caculs sont similaires pour la Réputation fondée sur les Interactions Indirectes, les résultats présentés ici sont aussi valables pour ce type de réputation.

Les figures C.6(a), C.6(b) et C.6(c) présentent l'influence du rapport entre les paramètres τ_V et τ_F sur la Réputation fondée sur les Interactions Directes, pour la stratégie rancunière (les courbes de la stratégie indulgentes sont trop instables pour être pertinentes ici). Quand les deux paramètres sont du même ordre de grandeur, l'influence sur le niveau de la Réputation fondée sur les Interactions Directes est faible. En revanche, quand le paramètre τ_V a plus de poids que le paramètre τ_F, les niveaux de réputation calculés sont plus rapprochés dans les valeurs basses (réputation de -1.0) et plus écartés dans les valeurs hautes (réputation de $+1.0$). La différence entre deux agents de faible réputation est moins importante qu'entre deux agents de forte réputation.

Ce paramètre pourrait permettre d'introduire une fragilité du point de vue de l'interaction (*cf.* section 3.2.5, page 48), puisqu'une mauvaise expérience fait plus chuter la réputation pour les agents qui ont déjà une forte réputation que pour les agents de faible réputation.



(a) $\tau_V = -1.0$, $\tau_F = 1.0$ (b) $\tau_V = -4.0$, $\tau_F = 4.0$

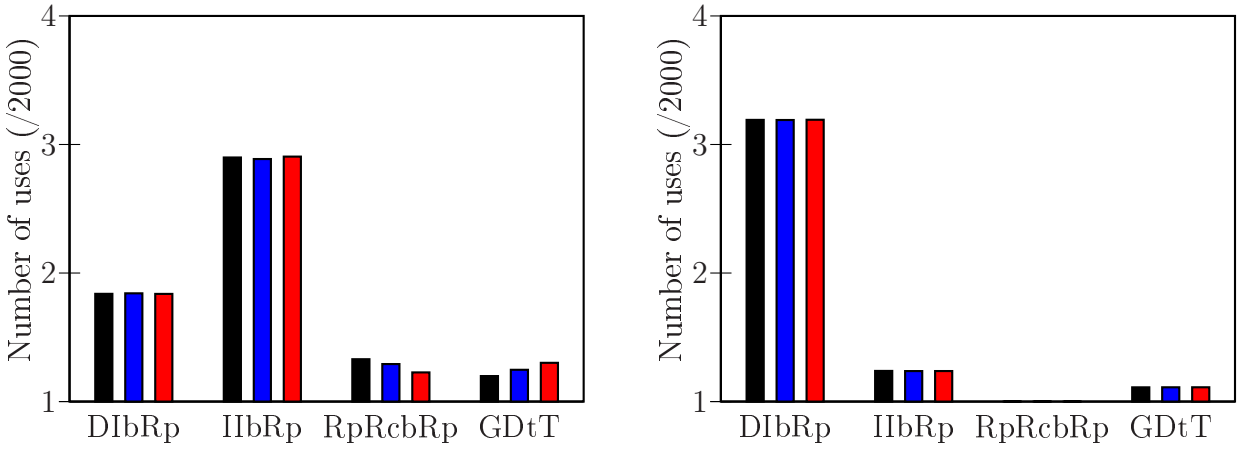


(c) $\tau_F = -4.0$, $\tau_F = 1.0$

FIG. C.6 – Influence des paramètres $\tau_{\mathcal{X}}$ (rancuniers).

C.4 Influence des paramètres $\theta^{\mathbf{trust}}_{\mathcal{X}\texttt{bRp}}$, ...

Les seuils $\theta^{\text{trust}}_{\mathcal{X}\texttt{bRp}}$, $\theta^{\text{distrust}}_{\mathcal{X}\texttt{bRp}}$, $\theta^{\text{relevance}}_{\mathcal{X}\texttt{bRp}}$,... sont utilisés dans le processus de raisonnement, les deux premiers pour déterminer si l'intention est de faire confiance ou si elle tend vers la défiance et le dernier pour déterminer la quantité d'information nécessaire pour considérer pertinente la valeur de réputation considérée. Dans cette section, nous proposons quelques résultats concernant leur influence sur le comportement général du modèle L.I.A.R.

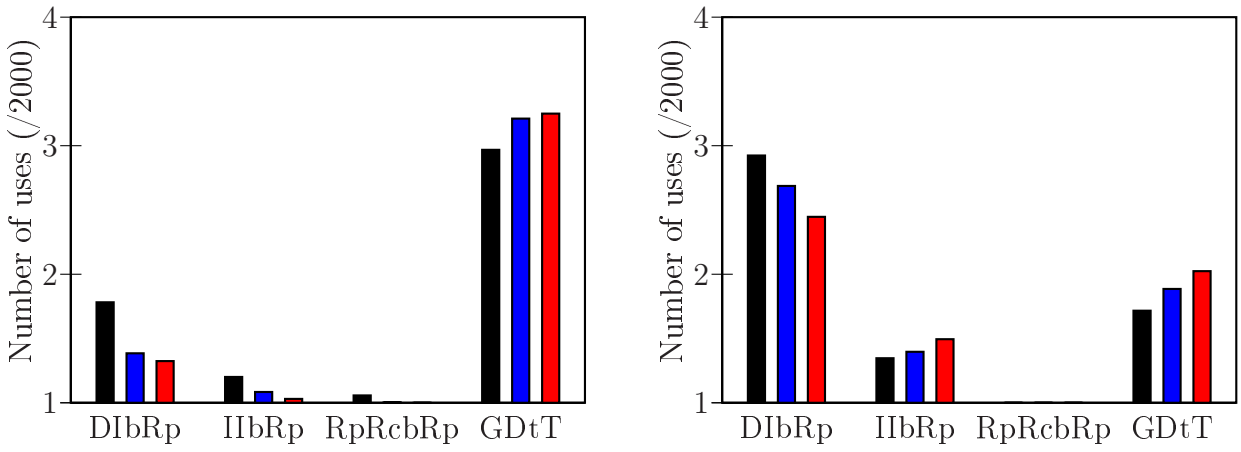


(a) indulgents (b) rancuniers

FIG. C.7 – Influence des paramètres $\theta^{\text{trust}}_{\mathcal{X}\texttt{bRp}}$, $\theta^{\text{distrust}}_{\mathcal{X}\texttt{bRp}}$.

La figure C.7 présente l'évolution du nombre de fois où chaque type de réputation est utilisé lors du processus de raisonnement quand les seuils $\theta^{\text{trust}}_{\mathcal{X}\texttt{bRp}}$, $\theta^{\text{distrust}}_{\mathcal{X}\texttt{bRp}}$ sont tous définis à 0.0, puis 0.5, puis 0.8. L'influence de ces changements est légèrement plus forte chez les agents indulgents, mais les tendances générales restent inchangées.

La figure C.8 présente l'évolution du nombre de fois où chaque type de réputation est utilisé lors du processus de raisonnement quand les seuils $\theta^{\text{relevance}}_{\mathcal{X}\texttt{bRp}}$ sont définis comme suit : $\theta^{\text{relevance}}_{\texttt{DIbRp}}$=10, $\theta^{\text{relevance}}_{\texttt{IIbRp}}$=20 et $\theta^{\text{relevance}}_{\texttt{RpRcbRp}}$=30, puis 0.5, puis $\theta^{\text{relevance}}_{\texttt{DIbRp}}$=20, $\theta^{\text{relevance}}_{\texttt{IIbRp}}$=30 et $\theta^{\text{relevance}}_{\texttt{RpRcbRp}}$=40, puis $\theta^{\text{relevance}}_{\texttt{DIbRp}}$=30, $\theta^{\text{relevance}}_{\texttt{IIbRp}}$=40 et $\theta^{\text{relevance}}_{\texttt{RpRcbRp}}$=50, puis 0.5. L'influence de ces changements est légèrement plus forte chez les agents indulgents, mais les tendances générales restent inchangées.



(a) indulgents

(b) rancuniers

FIG. C.8 – Influence des paramètres $\theta^{\text{relevance}}_{\mathcal{X}_{\texttt{bRp}}}$.

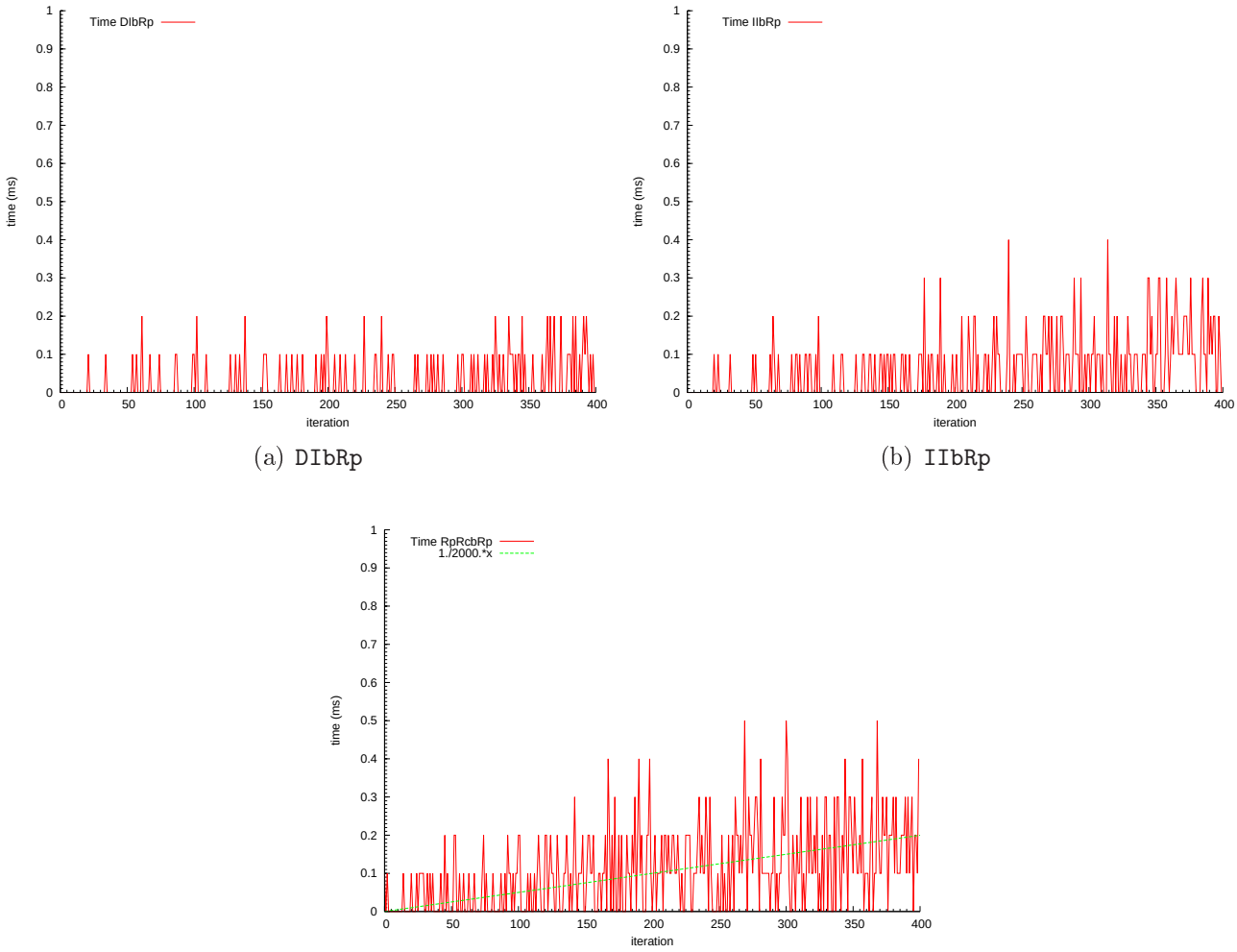
La quantité d'information dont dispose un agent pour calculer les niveaux de réputation est assez faible [Sab02]. De ce fait, il paraîtrait peu réaliste qu'un agent requière, par exemple, plus de 50 expériences directes avant d'estimer pouvoir calculer un niveau de Réputation fondée sur les Interactions Directes pertinent. Les seuils ont donc volontairement été fixés à de petites valeurs, pour plus de réalisme. La figure C.8 montre d'ailleurs que plus les seuils de pertinence augmentent, plus la Prédisposition Générale à faire Confiance est utilisée, au détriment des autres types de réputation. Si les seuils étaient fixés à de trop hautes valeurs, les réputations fondées sur l'interaction deviendraient inutiles.

C.5 Efficacité du modèle L.I.A.R.

Dans cette annexe, nous nous intéressons à l'efficacité du modèle L.I.A.R., c'est-à-dire au temps qu'il met pour calculer les différents types de réputations.

Dans les expérimentations présentées ici, nous ne cherchons pas à avoir des temps exacts d'exécution, mais une simple estimation de la forme que prend l'augmentation des temps de calculs avec l'augmentation du nombre d'interactions directes, indirectes ou de recommandations. De ce fait, nous avons employé la technique consistant à prendre la valeur de l'horloge au

début et à la fin de chaque fonction et à calculer la différence (d'où les pics fréquents dans les figures ci-dessous, liés à l'utilisation du processeur par d'autres tâches).



(a) `DIbRp` (b) `IIbRp`

(c) `RpRcbRp`

FIG. C.9 – Temps de calcul des différents types de réputation.

Le calcul des niveaux des différents types de réputations s'appuie sur des ensembles de politiques sociales qui grandissent avec le temps. Comme le montre la figure C.9, l'efficacité du module de punition diminue donc avec le temps. Cette diminution se fait de manière approximativement linéaire.

Ces temps de calculs sont raisonnables puisque, dans le pire des cas (celui de la Réputation fondée sur les Recommandations de Réputation, qui fait intervenir un double filtrage) et sur un Pentium IV à 1.7 Ghz, nous obtenons les temps les plus longs aux environs de 2 millisecondes dans une situation où il y a 800 recommandations à filtrer et à traiter.

Bibliographie

[Abd97] A. Abdul-Rahman. The PGP trust model. EDI-Forum : The Journal of Electronic Commerce, April 1997. 50, 55, 66

[Abd04] A. Abdul-Rahman. A Framework for Decentralized Trust Reasoning. PhD thesis, University of London, London, United Kingdom, December 2004. 45, 51, 58, 66, 67

[AC98] L. Amgoud and C. Cayrol. On the acceptability of arguments in preference-based argumentation. In Proceedings of Uncertainty in Artificial Intelligence (UAI'98), pages 1–7, Madison, Wisconsin, United States of America, July 1998. Morgan Kaufmann, San Francisco, CA, United States of America. 11

[AD01] K. Aberer and Z. Despotovic. Managing trust in a peer-2-peer information system. In H. Paques, L. Liu, and D. Grossman, editors, Proceedings of the Conference on "Information and Knowledge Management" (CIKM'2001), pages 310–317, New York, NY, United States of America, 2001. ACM Press. 45

[AGVSD05] H. Aldewereld, D. Grossi, J. Vàzquez-Salceda, and Frank Dignum. Designing normative behaviour by the use of landmarks. In O. Boissier, J. Padget, V. Dignum, G. Lindemann, E. Matson, S. Ossowski, J. S. Sichman, and J. Vázquez-Salceda, editors, Proceedings of the Workshop on "Agents, Norms and Institutions for REgulated Multi-agent systems" (ANIREM) at Autonomous Agents and Multi-Agent Systems (AAMAS'05), volume 3913 of Lecture Notes in Computer Science, pages 157–169, Utrecht, The Netherlands, July 2005. Springer-Verlag, Berlin, Germany. 23

[AH97a] A. Abdul-Rahman and S. Hailes. A distributed trust model. In Proceedings of the Workshop on "New Security Paradigms"

(NSPW'97), pages 48–60, Langdale, Cumbria, United Kingdom, 1997. ACM Press, New York, NY, United States of America. 51

[AH97b] A. Abdul-Rahman and S. Hailes. Using recommendations for managing trust in distributed systems. In Proceedings of the Malaysia International Conference on Communication (MICC'97), Kuala Lumpur, Malaysia, November 1997. IEEE Computer Society. 51

[AH00] A. Abdul-Rahman and S. Hailes. Supporting trust in virtual communities. In Proceedings of the Hawaii International Conference on System Sciences (HICSS-33), volume 6, page 6007, Maui, Hawaii, United States of America, January 2000. IEEE Computer Society, Washington, DC, United States of America. 39, 40, 48, 49

[AHTW97] S. Abu-Hakima, M. Toloo, and T. White. A multi-agent systems approach for fraud detection in personal communication systems. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'97), number NRC 41553, pages 1–8, Nagoya, Japan, August 1997. 26

[AJ02] G. Avoine and P. Junod. PGP : comment éviter les mauvaises surprises ? Multi-System and Internet Security Cookbook (MISC), 3, juillet–août 2002. 174

[AT03] S. Androutsellis-Theotokis. A survey of peer-to-peer file sharing technologies. Technical Report WHP-2002-003, ELTRUN, 2003. http ://www.eltrun.aueb.gr/whitepapers/p2p_2002.pdf. 137

[Aus62] J. L. Austin. How to do things with words. Oxford University Press, 1962. 10, 11

[Bar83] B. Barber. The Logic and Limits of Trust, chapter The meanings of trust : Technical competence and fiduciary responsibility, pages 7–25. Rutgers University Press, Rutgers, NJ, United States of America, 1983. 33

[Bat00] P. Bateson. Trust. Making and Breaking Cooperative Relations, chapter The Biological Evolution of Cooperation and Trust, pages 14–30. Basil Blackwell, New York, NY, United States of

America, 2000. (electronic edition, Department of Sociology, University of Oxford, Oxford, United Kingdom). 32

[BBK94] T. Beth, M. Borcherding, and B. Klein. Valuation of trust in open networks. In Proceedings of the European Symposium on "Research in Computer Security", pages 3–18, London, United Kingdom, 1994. Springer-Verlag. 45

[BD96] C. Baeijs and Y. Demazeau. Les organisations dans les systèmes multi-agents. In Actes de la Journée Nationale du Groupe de Recherche GdR-PRC IA, pages 35–46, Toulouse, France, février 1996. 2

[BDS00] A. Biswas, S. Debnath, and S. Sen. Believing others : Pros and cons. In Proceedings of the International Conference on "Multi-Agent Systems" (ICMAS'00), pages 279–285, Boston, MA, United States of America, July 2000. 45

[BF03] K. S. Barber and K. Fullam. Applying reputation models to continuous belief revision. In R. Falcone, S. K. Barber, L. Korba, and M. P. Singh, editors, Proceedings of the Workshop on "Trust, Privacy, Deception, and Fraud in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'03), pages 6–15, Melbourne, Australia, July 2003. 45

[BFK99] M. Blaze, J. Feigenbaum, and A.D. Keromytis. Keynote : Trust management for public-key infrastructures. In Proceedings of the Workshop on "Security Protocols" (SP'98), volume 1550 of Lecture Notes in Computer Science, pages 59–63, Cambridge, United Kingdom, April 1999. Springer-Verlag, Berlin, Germany. 2

[BFL96] M. Blaze, J. Feigenbaum, and J. Lacy. Decentralized trust management. In Proceedings of the IEEE Symposium on "Security and Privacy", pages 164–173, Oakland, CA, United States of America, May 1996. IEEE Computer Society, Washington, DC, United States of America. 2

[BGG04] O. Boissier, S. S. Gitton, and P. Glize. Systèmes Multi-Agents, volume ARAGO 29, chapter Caractéristiques des systèmes et des applications, pages 25–54. Éditions TEC et DOC, Observatoire Français des Techniques Avancées (OFTA) edition, 2004. 1

[BGM98] M. Barbuceanu, T. Gray, and S. Mankovski. Coordinating with obligations. In K. P. Sycara and M. Wooldridge, editors, Proceedings of Autonomous Agents (Agents'98), pages 62–69, Minneapolis, MN, United States of America, May 1998. ACM Press, New York, NY, United States of America. 21

[BK01] K.S. Barber and J. Kim. Belief revision process based on trust : Agents evaluating reputation of information sources. In R. Falcone, M. Singh, and Y.-H. Tan, editors, Proceedings of the Workshop on "Deception, Fraud, and Trust in Agent Societies" at Autonomous Agents (AA'00), volume 2246 of Lecture Notes in Artificial Intelligence. Trust in Cybersocieties : integrating the human and artificial perspectives, pages 73–82, Barcelona, Spain, June 2001. Springer-Verlag, Berlin, Germany. 146

[BL00] G. Boella and L. Lesmo. Deliberate normative agents. In C. Dellarocas and R. Conte, editors, Proceedings of the Workshop on "Norms and Institutions in Multi-Agent Systems" at Autonomous Agents (AA'00), pages 15–25, Barcelona, Spain, 2000. 21, 24

[BMCD03] J. Bentahar, B. Moulin, and B. Chaib-Draa. Towards a formal framework for conversational agents. In M.-P. Huget and F. Dignum, editors, Proceedings of the Workshop on "Agent Communication Languages and Conversation Policies" at Autonomous Agents and Multi-Agent Systems (AAMAS'03), Melbourne, Australia, July 2003. 12, 14, 15

[Bom99] M. Boman. Norms in artificial decision making. Artificial Intelligence and Law, 7(1) :17–35, 1999. 21

[BOvV05] G. Boella, J. Odell, L. van der Torre, and H. Verhagen, editors. American Association for Artificial Intelligence (AAAI) Fall Symposium : Roles, an interdisciplinary perspective, Arlington, VA, United States of America, November 2005. http ://normas.di.unito.it/zope/roles05. 106

[Bri06] Britanica. Encyclopedia britanica, March 2006. http ://www.britannica.com/. 18

[Bro00] É. Brousseau. Confiance et Rationalité, chapter Confiance et Contrat, Confiance ou Contrat, pages 1–15. INRA Édition, 2000. 35, 36, 48

[BSD00] A. Biswas, S. Sen, and S. Debnath. Limiting deception in a group of social agents. Applied Artificial Intelligence, 14 :785–797, 2000. 146

[BvdT05] G. Boella and L. van der Torre. Enforceable social laws. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M.P. Singh, and M. Wooldridge, editors, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'05), volume 2, pages 682–689, Utrecht, The Netherlands, July 2005. ACM Press, New York, NY, United States of America. 21

[BZL03] B. Bhargava, Y. Zhong, and Y. Lu. Fraud formalization and detection. In Proceedings of Data Warehousing and Knowledge Discovery (DaWak'03), volume 2737 of Lecture Notes in Computer Science, pages 330–339, Prague, Czech Republic, September 2003. Springer-Verlag, Berlin, Germany. 49

[CB05] C. Carabelea and O. Boissier. Coordinating agents in organizations using social commitments. In G. Boella and L. van der Torre, editors, Proceedings of the Workshop on "Coordination and Organisation" (CoOrg'05), volume 150 of Electronic Notes in Theoretical Computer Science, pages 73–91, Namur, Belgium, April 2005. Elsevier Science B.V., Amsterdam, The Netherlands. 24

[CBSS05] C. Castelfranchi, S. K. Barber, J. Sabater, and M. P. Singh, editors. Proceedings of the Workshop on "Trust in Agent Societies", Autonomous Agent and Multi-Agent Systems (AAMAS'05), Utrecht, The Netherlands, July 2005. 33, 53

[Cd90] J. A. Campbell and M. P. d'Inverno. Knowledge interchange protocols. In Y. Demazeau and J.-P. Müller, editors, Proceedings of the Workshop on "Modelling Autonomous Agents in a Multi-Agent World", Decentralized AI, pages 63–80, Cambridge, United Kingdom, August 1990. Elsevier Science B.V., Amsterdam, The Netherlands. 10

[CDJT00] C. Castelfranchi, F. Dignum, C. Jonker, and J. Treur. Deliberative normative agents : Principles and architecture. In N. Jennings and Y. Lespérance, editors, Proceedings of the Workshop on "Agent Theories, Architectures, and Languages" (ATAL'99), volume 1757 of Lecture Notes in Artificial Intelligence.

Intelligent Agents VI, pages 364–378, Orlando, FL, United States America, 2000. Springer-Verlag, Berlin, Germany. 21, 24

[CF98] C. Castelfranchi and R. Falcone. Principles of trust for mas : Cognitive anatomy, social importance, and quantification. In Y. Demazeau, editor, Proceedings of the International Conference on Multi-Agent Systems (ICMAS'98), pages 72–79, Paris, France, 1998. IEEE Computer Society, Washington, DC, United States of America. 2, 32, 33, 34, 35, 36, 45, 47, 48, 60

[CFP03] C. Castelfranchi, R. Falcone, and G. Pezzulo. Trust in information sources as a source for trust : a fuzzy approach. In Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'03), pages 89–96, Melbourne, Australia, July 2003. ACM Press, New York, NY, United States of America. 35

[CH96] B. Christianson and W. S. Harbison. Why isn't trust transitive ? In T. Mark and A. Lomas, editors, Proceedings of the Workshop on "Security Protocols", volume 1189 of Lecture Notes in Computer Science, pages 171–176, University of Cambridge, April 1996. Springer-Verlag, London, United Kingdom. 51

[Che80] B. F. Chellas. Modal Logic an Introduction. Cambridge University Press, Cambridge, United Kingdom, 1980. 25

[CL95] P. Cohen and H. Levesque. Communicative actions for artificial agents. In Proceedings of the International Conference on Multi Agent Systems (ICMAS'95), pages 65–72, Cambridge, MA, United States of America, 1995. MIT Press. 10

[Cla04] R. Clarke. Peer-to-peer (p2p) – an overview, 2004. http ://www.anu.edu.au/people/Roger.Clarke/EC/P2POview.html. 137

[CLPS02] M. H. Cahill, D. Lambert, J. C. Pinheiro, and D. X. Sun. Detecting fraud in the real world. Kluwer Academic Publishers, Norwell, MA, United States of America, 2002. 26

[CMD03] J. Carbo, J. M. Molina, and J. Dávila Muro. Trust management through fuzzy reputation. International Journal of Cooperative Information Systems, 12 :135–155, 2003. 46, 60, 66

[Col01] Columbia. Columbia encyclopedia, May 2001. http ://www.bartleby.com/65/. 18

[Cor83] D.D. Corkill. A framework for Organizational Self-Design in Distributed Problem-Solving Networks. PhD thesis, Department of Computer Science, University of Massachusetts, Amherst, MA, United States of America, February 1983. (also published as Technical Report COINS-TR-82-33, December 1982). 2

[COTZ00] C. Castelfranchi, A. Omicini, R. Tolksdorf, and F. Zambonelli. Engineering social order. In A. Ominici, R. Tolksdorf, and F. Zambonelli, editors, Proceedings of Engineering Societies in the Agents World (ESAW'00), volume 1972 of Lecture Notes in Computer Science, pages 1–18, Berlin, Germany, December 2000. Springer-Verlag, Berlin, Germany. 2

[CP99] C. Castelfranchi and R. Pedone. A review on trust in information technology, 1999. http ://alfebiite.ee.ic.ac.uk/docs/papers/D1/ab-d1-cas+ped-trust.pdf. 48

[CP02] R. Conte and M. Paolucci. Reputation in Artificial Societies. Social Beliefs for Social Order. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002. 2, 32, 35, 36, 38, 40, 41, 106

[CS05] S. Casare and J. Sichman. Towards a functional ontology of reputation. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M.P. Singh, and M. Wooldridge, editors, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'05), volume 2, pages 505–511, Utrecht, The Netherlands, July 2005. ACM Press, New York, NY, United States of America. 33, 35, 38, 39, 40, 41, 43, 51, 175

[Das90] P. Dasgupta. Trust. Making and Breaking Cooperative Relations, chapter Trust as a commodity. Basil Blackwell, New York, NY, United States of America, 1990. (electronic edition, Department of Sociology, University of Oxford, Oxford, United Kingdom). 32, 33, 35, 36, 48

[Del00] Chrysanthos Dellarocas. Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior. In Proceedings of the ACM Conference on "Electronic

Commerce" (EC'00), pages 150–157, New York, NY, United States of America, October 2000. ACM Press. 32

[Del01a] C. Dellarocas. Analyzing the economic efficiency of eBay-like online reputation reporting mechanisms. In Proceedings of the ACM Conference on "Electronic Commerce" (EC'01), pages 171–179, Tempa, FL, United States of America, October 2001. ACM Press, New York, NY, United States of America. 32

[Del01b] C. Dellarocas. Building trust on-line : The design of reliable reputation reporting : Mechanisms for online trading communities. Working Paper 4180-01, MIT Sloan, Cambridge, MA, United States of America, 2001. 45, 57

[Del03] C. Dellarocas. The digitalization of word of mouth : Promise and challenges of online reputation mechanisms. Technical report, Center for eBusiness at MIT, Cambridge, MA, United States of America, March 2003. http ://ebusiness.mit.edu/research/papers/173_Dellarocas_Word_of_Mouth.pdf. 48

[Dem04] R. Demolombe. Reasoning about trust : A formal logical framework. In C. D. Jensen, S. Poslad, and T. Dimitrakos, editors, Proceedings of the International Conference on Trust Management (iTrust'04), volume 2995 of Lecture Notes in Computer Science, pages 291–303, Oxford, United Kingdom, January 2004. Springer-Verlag, Berlin, Germany. 37, 39, 40

[Deu62] M. Deutsch. Cooperation and trust : Some theoretical notes. In M.R. Jones, editor, Proceedings of the Nebraska Symposium on "Motivation", pages 275–320, Lincoln, NE, United States of America, 1962. Nebraska University Press. 32

[Dig99] F. Dignum. Autonomous agents with norms. In Artificial Intelligence and Law, volume 7, pages 69–79, 1999. 21, 24

[Dig02] Frank Dignum. Abstract norms and electronic institutions. In G. Lindemann, D. Moldt, M. Paolucci, and B. Yu, editors, Proceedings of the Workshop on "Regulated Agent-Based Social Systems : Theories and Applications" (RASTA) at Autonomous Agents and Multi-Agent Systems (AAMAS'02), pages 93–103, Bologna, Italy, July 2002. 18

[DKG+04] L. Ding, P. Kolari, S. Ganjugunte, T. Finin, and A. Joshi. On modeling and evaluating trust networks inference. In

R. Falcone, K.S. Barber, J. Sabater, and M. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'04), pages 21–32, New York, NY, United States of America, July 2004. 145

[DM90] E. H. Durfee and T. A. Montgomery. A hierarchical protocol for coordinating mulitagent behaviors. In Proceedings of the National Conference on "Artificial Intelligence", pages 86–93, Boston, MA, United States of America, July–August 1990. AAAI Press / MIT Press. 10

[DMSC00] F. Dignum, D. Morley, L. Sonenberg, and L. Cavedon. Towards socially sophisticated BDI agents. In Proceedings of the International Conference on Multi-Agent Systems (ICMAS'00), pages 111–118, Boston, MA, United States of America, 2000. IEEE Computer Society, Washington, DC, United States of America. 19, 20, 21

[dSdL05] V. Torres da Silva and C. J.P. de Lucena. Classifying and describing agent contracts and norms. In Proceedings of the Workshop on "Agents, Norms and Institutions for REgulated Multi-agent systems" (ANIREM) at Autonomous Agents and Multi-Agent Systems (AAMAS'05), pages 33–45, Utrecht, The Netherlands, July 2005. 23

[Dun94] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning logic programming and n-person games. Artificial Intelligence, 77(2) :321–357, April 1994. 11

[DvL97] F. Dignum and B. van Linder. Modelling social agents : Communication as action. In J.P. Müller, M.J. Wooldridge, and N.R. Jennings, editors, Proceedings of the Workshop on "Intelligent Agents III, Agent Theories, Architectures, and Languages" at the European Conference on Artificial Intelligence (ECAI'96), volume 1193 of Lecture Notes in Artificial Intelligence, pages 205–218, Brighton, United Kingdom, 1997. Springer-Verlag, London, United Kingdom. 10

[eBa03] eBay. eBay auction website, 2003. http ://www.ebay.com. 56, 67

[Elg93] D. Elgesem. Action Theory And Modal Logic. PhD thesis, Institttut for filosofi, Universitetet Oslo, Oslo, Norway, 1993. 25

[ENT+02] C. English, P. Nixon, S. Terzis, A. McGettrick, and H. Lowe. Security models for trusting network appliances. In Proceedings of the International Workshop on "Networked Appliances", pages 39–44, Liverpool, United Kingdom, October 2002. IEEE Computer Society. 45

[Fab96] P. Fabiani. Dynamics of beliefs and strategy of perception. In W. Wahlster, editor, Proceedings of European Conference on "Artificial Intelligence" (ECAI'96), pages 8–12, Budapest, Hungary, 1996. John Wiley & Sons, Ltd. 49

[FB04] K. Fullam and K. S. Barber. A temporal policy for trusting information. In R. Falcone, K.S. Barber, J. Sabater, and M. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'04), pages 47–57, New York, NY, United States of America, July 2004. 145

[FBKS03] R. Falcone, S. K. Barber, L. Korba, and M. P. Singh, editors. Proceedings of the Workshop on "Trust, Privacy, Deception, and Fraud in Agent Societies" at Autonomous Agent and Multi-Agent Systems (AAMAS'03), Melbourne, Australia, July 2003. 33, 53

[FBSS04] R. Falcone, S. K. Barber, J. Sabater, and M. P. Singh, editors. Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agent and Multi-Agent Systems (AAMAS'04), New York, NY, United States of America, July 2004. 33, 53

[FC99] R. Falcone and C. Castelfranchi. The dynamics of trust : from beliefs to action. In Proceeding of the Workshop on "Deception, Fraud and Trust in Agent Societies" at Autonomous Agents (AA'99), pages 41–54, Seattle, WA, United States of America, May 1999. 126

[FC02] N. Fornara and M. Colombetti. Operational specification of a commitment-based agent communication language. In Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'02), pages 536–542, Bologna, Italy, 2002. ACM Press, New York, NY, United States of America. 11, 80

[FC03] N. Fornara and M. Colombetti. Defining interaction protocols using a commitment-based agent communication language. In Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'03), pages 520–527, Melbourne, Australia, July 2003. ACM Press, New York, NY, United States of America. 11, 12, 13

[FG99] J. Ferber and O. Gutknecht. Operational semantics of a role-based agent architecture. In N.R. Jennings and Y. Lespérance, editors, Proceedings of the Workshop on "Agent Theories, Architectures and Languages" (ATAL'99), volume 1757 of Lecture Notes in Artificial Intelligence. Intelligent Agents VI, pages 205–217, Orlando, FL, United States of America, July 1999. Springer-Verlag, Berlin, Germany. 106

[FIP02] FIPA. FIPA communicative act library specification. Technical Report SC00037J, Fundation For Intelligent Physical Agents (FIPA), December 2002. Standard Status. 10, 102

[FKM+05a] K. Fullam, T. Klos, G. Muller, J. Sabater, A. Schlosser, Z. Topol, K. S. Barber, J. S. Rosenschein, L. Vercouter, and M. Voss. A demonstration of the agent reputation and trust (ART) testbed : Experimentation and competition for Trust in Agent Societies. In Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'05), Utrecht, The Netherlands, July 2005. ACM Press, New York, NY, United States of America.

[FKM+05b] K. Fullam, T. Klos, G. Muller, J. Sabater, A. Schlosser, Z. Topol, K. S. Barber, J. S. Rosenschein, L. Vercouter, and M. Voss. A specification of the agent reputation and trust (ART) testbed : Experimentation and competition for Trust in Agent Societies. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M. P. Singh, and M. Wooldridge, editors, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'05), pages 512–518, Utrecht, The Netherlands, July 2005. ACM Press, New York, NY, United States of America. 145

[FKM+05c] K. Fullam, T. Klos, G. Muller, J. Sabater, Z. Topol, K. S. Barber, J. S. Rosenschein, and L. Vercouter. The agent reputation and trust (ART) testbed architecture. In C. Castelfranchi, S. Barber, J. Sabater, and M. P. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at

Autonomous Agents and Multi-Agent Systems (AAMAS'05), volume 2, pages 50–62, July 2005.

[FKM+05d] K. K. Fullam, T. B. Klos, G. Muller, J. Sabater, Z. Topol, K. S. Barber, J. S. Rosenschein, and L. Vercouter. Le banc d'essais ART (agent reputation and trust) pour les modèles de confiance. In A. Drogoul and É. Ramat, editors, Actes des Journées Francophones sur les Systèmes Multi-Agents (JFSMA'05), pages 175–179, Calais, France, novembre 2005. Hermès Sciences.

[FKM+05e] K. K. Fullam, T. B. Klos, G. Muller, J. Sabater i Mir, Z. Topol, K. S. Barber, J. S. Rosenschein, and L. Vercouter. The agent reputation and trust (art) testbed architecture. In B. López, J. Meléndez, P. Radeva, and J. Vitrià, editors, Proceedings of the Congrès Català d'Intel.ligència Artificial (CCIA'05), volume 131 of Frontiers in Artificial Intelligence and Applications, pages 389–396, Alguer, Spain, October 2005. Associació Catalana d'Intelligència Artificial, IOS Press, Amsterdam, The Netherlands.

[FKM+06a] K. K. Fullam, T. B. Klos, G. Muller, J. Sabater-Mir, K. S. Barber, and L. Vercouter. The agent reputation and trust (`art`) testbed. In K. Stølen, W. H. Winsborough, F. Martinelli, and F. Massacci, editors, Proceedings of the International Conference on Trust Management (iTrust'06), volume 3986 of Lecture Notes in Computer Science, pages 439–442, Pisa, Italy, May 2006. Springer-Verlag, Berlin, Germany.

[FKM+06b] Karen K. Fullam, Tomas B. Klos, Guillaume Muller, Jordi Sabater-Mir, K. Suzanne Barber, and Laurent Vercouter. The agent reputation and trust (ART) testbed. In Belgium-Netherlands Conference on Artificial Intelligence (BNAIC'06), pages 449–450, 2006.

[Fox81] M.S. Fox. An organizational view on distributed systems. IEEE Transactions on Systems, Man and Cybernetics, (SMC-11), 1 :70–80, 1981. 2

[FPCC04] R. Falcone, G. Pezzulo, C. Castelfranchi, and G. Calvi. Trusting the agents and the environment leads to successful delegation : A contract net simulation. In R. Falcone, K.S. Barber,

J. Sabater, and M. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'04), pages 33–39, New York, NY, United States of America, July 2004. 146

[Fra] La Documentation Française. La séparation des pouvoirs. http ://www.vie-publique.fr/decouverte-institutions/institutions/approfondissements/separation-pouvoirs.html. 18, 89

[FTL98] B. S. Firozabadi, Y.-H. Tan, and R. M. Lee. Formal definitions of fraud. In Proceedings of the Workshop on "Deontic Logic" (DEON'98), Bologna, Italy, January 1998. 25, 27

[Fuk95] F. Fukuyama. Trust : the social virtues and the creation of prosperity. The Free Press, New York, NY, United States of America, 1995. 32

[Ful03] K. Fullam. An expressive belief revision framework based on information valuation. Master's thesis, Department of Electrical and Computer Engineering, University of Texas, Austin, TX, United States of America, 2003. 145

[Gam00a] D. Gambetta. Trust. Making and Breaking Cooperative Relations, chapter Can We Trust Trust ?, pages 213–237. Basil Blackwell, New York, NY, United States of America, 2000. (electronic edition, Department of Sociology, University of Oxford, Oxford, United Kingdom). 32, 45, 49, 126

[Gam00b] Diego Gambetta, editor. Trust. Making and Breaking Cooperative Relations. Basil Blackwell, New York, NY, United States of America, 2000. (electronic edition, Department of Sociology, University of Oxford, Oxford, United Kingdom). 34

[GBKD05] B. Gâteau, O. Boissier, D. Khadraoui, and E. Dubois. MOISE-Inst : an organizational model for specifying rights and duties of autonomous agents. In M. P. Gleizes, G. A. Kaminka, A. Nowé, S. Ossowski, K. Tuyls, and K. Verbeeck, editors, Proceedings of the Workshop on "Coordination and Organisation" (CoOrg'05), pages 484–485, Namur, Belgium, December 2005. Koninklijke Vlaamse Academie van België voor Wetenschappen en Kunsten, Brussel, Belgium. 23, 24, 25

[GCNRA05] A. Garcia-Camino, P. Noriega, and J. Rodriguez-Aguilar. Implementing norms in electronic institutions. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M.P. Singh, and M. Wooldridge, editors, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'05), volume 2, pages 667–673, Utrecht, The Netherlands, July 2005. ACM Press, New York, NY, United States of America. 23, 25

[GDT04] GDT. Le grand dictionaire terminologique, 2004. http ://w3.granddictionnaire.com. 18, 32

[GH04] R. Ghanea-Hercock. The cost of trust. In R. Falcone, K.S. Barber, J. Sabater, and M. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'04), pages 58–64, New York, NY, United States of America, July 2004. 146

[Gnu00] Gnutella. Gnutella 0.48 specifications, 2000. http ://rfc-gnutella.sourceforge.net/developer/stable/. 139

[Gra03] T. Grandison. Trust Management for Internet Applications. PhD thesis, Imperial College London, London, United Kingdom, July 2003. 48

[GS00] T. Grandison and M. Sloman. A survey of trust in internet applications. IEEE Communications Surveys and Tutorials, 3(4), October–December 2000. 45, 48, 51

[GVSM06] A. Grizard, L. Vercouter, T. Stratulat, and G. Muller. A peer-to-peer normative system to achieve social order. In V. Dignum, N. Fornara, and P. Noriega, editors, Proceedings of the Workshop on "Coordination, Organization, Institutions and Norms" (COIN) at Autonomous Agents and Multi-Agent Systems (AAMAS'06), Lecture Notes in Computer Science, Hakodate, Japan, May 2006. Springer-Verlag, Berlin, Germany. (in press). 175

[Hab84] J. Habermas. The Theory of Communicative Action, Volume 1 & 2. Polity Press, Cambridge, Uinted Kingdom, 1984. 10, 11, 12

[Ham70] C.L. Hamblin. Fallacies. Methuen, London, United Kingdom, 1970. 11

[Han03] M. Hannoun. MOISE : un modèle organisationnel pour les systèmes multi-agents. PhD thesis, Université Jean Monnet et École des Mines de Saint-Étienne, Saint-Étienne, France, Décembre 2003. 2

[HDM03] T. Hughes, J. Denny, and P.A. Muckelbauer. Dynamic trust applied to ad hoc network resources. In R. Falcone, S. K. Barber, L. Korba, and M. P. Singh, editors, Proceedings of the Workshop on "Trust, Privacy, Deception, and Fraud in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'03), pages 33–40, Melbourne, Australia, July 2003. 35

[Her88] L. Hertzberg. On the attitude of trust. Inquiry, 31(3) :307–322, September 1988. 32

[HOH+03] M.-Ph. Huget, J. Odell, Ø. Haugen, M. Nodine, S. Cranefield, R. Levy, and L. Padgham. FIPA modeling : Interaction diagrams. Technical report, Foundation for Intelligent Physical Agents, 2003. http ://www.auml.org/auml/documents/ID-03-07-02.pdf. 98, 100

[HSB02] J. F. Hübner, J. S. Sichman, and O. Boissier. A model for the structural, functional, and deontic specification of organizations in multi-agent systems. In Proceedings of the Brazilian Artificial Intelligence Symposium (SBIA'02), pages 118 – 128, Berlin, Germany, 2002. Springer-Verlag. 106

[JAS] JASSS. Journal of Artificial Societies and Social Simulation. (electronic edition) http ://jasss.soc.surrey.ac.uk/JASSS.html. 31

[Jen97] D. Jensen. Prospective assessment of AI technologies for fraud detection : a case study. In T. Fawcett, I. Haimowitz, F. Provost, and S. Stolfo, editors, Proceedings of the Workshop on "Artificial Intelligence Approaches to Fraud Detection and Risk Management", pages 34–38. American Association for Artificial Intelligence Press, July 1997. 26

[JF01] A. J. I. Jones and B. S. Firozabadi. On the characterisation of a trusting agent – aspects of a formal approach. In C. Castelfranchi and Y.-H. Tan, editors, Proceedings of the Workshop on "Deception, Trust, Fraud in Agent Societies" at Autonomous

Agents (AA'00), Trust and Deception in Virtual Societies, pages 157–168, Barcelona, Spain, June 2001. Kluwer Academic Publishers, Norwell, MA, United States of America. 51

[JoM] JoM. Journal of Marketing. 31

[Jon93] A.J.I. Jones. Towards a formal theory of defeasible deontic conditionals. Annals of Mathematics and Artificial Intelligence, 9(1-2) :151–166, 1993. 22

[Jøs96] A. Jøsang. The right type of trust for distributed systems. In C. Meadows, editor, Proceedings of the Workshop on "New Security Paradigms" (NSPW'96), pages 119–131, Lake Arrowhead, CA, United States of America, 1996. ACM Press, New York, NY, United States of America. 51

[JP05] A. Jøsang and S. Pope. Semantic constraints for trust transitivity. In S. Hartmann and M. Stumptner, editors, Proceedings of Asia-Pacific Conference on Conceptual Modelling (APCCM'05), volume 43 of Conferences in Research and Practice in IT, pages 59–68, Newcastle, Australia, January–February 2005. Australian Computer Society, Darlinghurst, Australia. 50

[JS93] A.J.I. Jones and M. Sergot. Deontic Logic in Computer Science : Normative System Specification, chapter On the characterisation of law and computer systems : The normative systems perspective. John Wiley & Sons, 1993. 21

[JT02] C.M. Jonker and J. Treur. A dynamic perspective on an agent's mental states and interaction with its environment. In C. Castelfranchi and W.L. Johnson, editors, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'02), pages 865–872, Bologna, Italy, July 2002. ACM Press, New York, NY, United States of America. 49

[KN05] M. J. Kollingbaum and T. J. Norman. Informed deliberation during norm-governed practical reasoning. In O. Boissier, J. Padget, V. Dignum, G. Lindemann, E. Matson, S. Ossowski, J. S. Sichman, and J. Vázquez-Salceda, editors, Proceedings of the Workshop on "Agents, Norms and Institutions for REgulated Multi-agent systems" (ANIREM) at Autonomous Agents and Multi-Agent Systems (AAMAS'05), volume 3913 of Lecture

Notes in Artificial Intelligence, pages 19–31, Utrecht, The Netherlands, July 2005. Springer-Verlag, Berlin, Germany. 23, 24

[KP04] T. Klos and H. La Poutré. Using reputation-based trust for assessing agent reliability. In R. Falcone, K.S. Barber, J. Sabater, and M. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'04), pages 75–82, New York, NY, United States of America, July 2004. 145

[Lab96] Y. Labrou. Semantics for an Agent Communication Language. PhD thesis, University of Maryland, Baltimore, MD, United States of America, September 1996. 10

[Lag92] O. Lagenspetz. Legitimacy and trust. Philosophical Investigation, 15(1) :1–21, 1992. 32

[Lam01] P. Lamsal. Understanding trust and security. Online-Reference, October 2001. http ://www.cs.helsinki.fi/u/lamsal/asgn/trust/UnderstandingTrustAndSecurity.pdf. Accessed December 2003. 45

[LB95] R. J. Lewicki and B. B. Bunker. Conflict, Cooperation and Justice : Essays Inspired by the Work of Morton Deutsch, chapter Trust in Relationships : A Model of Development and Decline. Jossey-Bass Publishers, San Francisco, CA, United States of America, 1995. 33

[LL04] F. López y López and M. Luck. A model of normative multi-agent systems and dynamic relationships. In G. Lindemann, D. Moldt, and M. Paolucci, editors, Proceedings of the Workshop on "Regulated Agent-Based Social Systems : Theories and Applications" (RASTA) at Autonomous Agents and Multi-Agent Systems (AAMAS'02), volume 2934 of Lecture Notes in Computer Science, pages 259–280, Bologna, Italy, January 2004. Springer-Verlag, Berlin, Germany. 23

[LLd02] F. López y López, M. Luck, and M. d'Inverno. Constraining autonomy through norms. In C. Castelfranchi and W.L. Johnson, editors, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'02), pages 674–681, Bologna, Italy, 2002. ACM Press, New York, NY, United States of America. 18, 19, 21, 24

[LLd04] F. López y López, M. Luck, and M. d'Inverno. Normative agent reasoning in dynamic societies. In Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'04), volume 2, pages 732–739, New York, NY, United States of America, 2004. IEEE Computer Society, Washington, DC, United States of America. 21, 23

[LS04] A. Lomuscio and M. Sergot. A formalisation of violation, error recovery, and enforcement in the bit transmission problem. Journal of Applied Logic (selected articles from DEON'02), 1(1) :93–116, March 2004. 26

[Luh79] N. Luhmann. Trust and Power. John Wiley & Sons, 1979. 32, 126

[Luh00] N. Luhmann. Trust. Making and breaking cooperative relations, chapter Familiarity, Confidence, Trust, pages 94–107. Basil Blackwell, New York, NY, United States of America, 2000. (electronic edition, Department of Sociology, University of Oxford, Oxford, United Kingdom). 32

[Mar94] S. Marsh. Formalizing Trust as a Computational Concept. PhD thesis, Department of Computer Science and Mathematics, University of Stirling, Scotland, United Kingdom, April 1994. 48, 59, 66

[MC01] D.H. McKnight and N.L. Chervany. Trust and distrust definitions : One bite at a time. In Proceedings of the Workshop on "Deception, Fraud and Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'01), volume 2246 of Lecture Notes In Computer Science, pages 27–54, Montreal, Canada, May 2001. Springler-Verlag, London, United Kingdom. 2, 31, 33, 34, 35, 36, 37, 40, 47, 48, 51, 54, 94

[MC02] D.H. McKnight and N.L. Chervany. What trust means in e-commerce customer relationships : An interdisciplinary conceptual typology. International Journal of Electronic Commerce (IJEC'02), 6(2) :33–57, 2002. 33

[MD05] D. Melaye and Y. Demazeau. Bayesian dynamic trust model. In M. Pěchouček, P. Petta, and L. Z. Varga, editors, Proceedings of the Central and Eastern European Conference on Multi-Agent Systems (CEEMAS'05), volume LNAI 3690, pages 480–489, Bu-

dapest, Hungary, 2005. Springer-Verlag, Berlin, Germany. 2, 46, 47, 49, 63, 66

[MDW94] J.-J. Ch. Meyer, F.P.M. Dignum, and R.J. Wieringa. The paradoxes of deontic logic revisited : a computer science perspective. Technical Report UU-CS-1994-38, Utrecht University, Utrecht, The Netherlands, 94. 22

[Mer04] Merriam. Merriam webster online, 2004. http ://www.m-w.com. 32

[MHM02] L. Mui, A. Halberstadt, and M. Mohtashemi. Notions of reputation in multi-agent systems : A review. In C. Castelfranchi and W.L. Johnson, editors, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'02), pages 280–287, Bologna, Italy, July 2002. ACM Press, New York, NY, United States of America. 31, 35, 38, 41, 52, 111

[MKN98] A. Melek, V. Keong, and K. Nemani. Levels of trust. the commitment-trust theory of relationship marketing. Journal of Marketing, 58 :20–38, October 1998. 48

[MT95] Y. Moses and M. Tennenholtz. Artificial social systems. Computers and Artificial Intelligence, pages 533–562, 1995. 21

[NAI99a] NAI. Introduction to Cryptography, Network Associates International, chapter Notions Élémentaires de Cryptographie – Signatures Numériques. 1999. http ://www.pgpi.org/doc/pgpintro/#p12. 103

[NAI99b] NAI. PGP documentations, Network Associates International, 1999. http ://www.pgpi.org/doc/7.0/. 55

[NP04] B. Neville and J. Pitt. A simulation study of social agents in agent mediated e-commerce. In R. Falcone, K.S. Barber, J. Sabater, and M. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'04), pages 83–91, New York, NY, United States of America, July 2004. 146

[OMG03] OMG. Unified Modeling Language, v1.5. Technical report, Object Management Group, March 2003. http ://www.omg.org/cgi-bin/doc ?formal/03-03-01.pdf. 81, 93

[OnS03] OnSale. OnSale auction website, 2003. http ://www.onsale.com. 56, 67

[OP05] Open-PGP. Open-pgp web of trust, 2005. http ://en.wikipedia.org/wiki/Web_of_trust. 50, 55, 66, 67

[Ors98] E. Orstom. A behavioral approach to the rational-choice theory of collective action. American Political Science Review, 92(1) :1–22, March 1998. 35

[Pag00] A. Pagden. Trust. Making and Breaking Cooperative Relations, chapter The Destruction of Trust and its Consequences in the Case of Eighteenth Century Naples, pages 127–141. Basil Blackwell, New York, NY, United States of America, 2000. (electronic edition, Department of Sociology, University of Oxford, Oxford, United Kingdom). 32

[PANS98] E. Plaza, J. Lluìs Arcos, P. Noriega, and C. Sierra. Competing agents in agent-mediated institutions. Personal Technologies, 2(3) :212–220, September 1998. 25

[PCL87] H.E. Pattison, D.D. Corkill, and V.R. Lesser. Instantiating description of organizational structures. Distributed Artificial Intelligence, Research Notes in AI, 1987. 2

[PD92] G. B. Pollock and L. A. Dugatkin. Reciprocity and the evolution of reputation. Journal of Theoretical Biology, 159 :25–37, 1992. 32

[Pea88] J. Pearl. Probabilistic reasoning in intelligent systems : networks of plausible inference. Morgan Kaufmann Publishers Inc., San Francisco, CA, United States of America, 1988. 61, 64

[PFCd04] P. Pasquier, R. A. Flores, and B. Chaib-draa. Modelling flexible social commitments and their enforcement. In Proceedings of Engineering Societies in the Agents' World (ESAW'04), volume 3451 of Lecture Notes in Computer Science, pages 139–151, Toulouse, France, October 2004. 12, 13, 14, 15, 27

[Qué01] L. Quéré. La structure cognitive et normative de la confiance. Réseaux, 19(108) :125–152, 2001. 32, 33, 34, 35, 37, 45, 48

[RG91] A.S. Rao and M.P. Georgeff. Modeling rational agents within a BDI-architecture. In J. Allen, R. Fikes, and E. Sandewall,

editors, Proceedings of Principles of Knowledge Representation and Reasoning (KR&R'91), pages 473–484, Cambridge, MA, United States of America, 1991. Morgan Kaufmann Publishers, San Mateo, CA, United States of America.

[Rou00] J. Rouchier. La confiance à travers l'échange. Accès aux pâturages au Nord-Cameroun et échanges non-marchands : des simulations dans des systèmes multi-agents. PhD thesis, Université d'Orléans, Orléans, France, mai 2000. 2, 32, 34, 35, 36, 45, 48

[Rou01] J. Rouchier. Est-il possible d'utiliser une définition positive de la confiance dans les interactions entre agents ? In Actes de la Journée "Modèles Formels de l'Interaction", Groupe de Recherche "Information, Intelligence et Interaction" (GdR I3), Lille, France, mai 2001. 48

[RPBF05] M. Rehák, M. Pěchouček, P. Benda, and L. Foltýn. Trust in coalition environment : Fuzzy number approach. In C. Castelfranchi, S. Barber, J. Sabater, and M. P. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'05), pages 132–144, Utrecht, The Netherlands, July 2005.

[RS03] S.D. Ramchurn and C. Sierra. A computational trust model for multi-agent interactions based on confidence and reputation. In R. Falcone, S. K. Barber, L. Korba, and M. P. Singh, editors, Proceedings of the Workshop on "Trust, Privacy, Deception, and Fraud in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'03), pages 69–75, Melbourne, Australia, July 2003. 48

[Sab02] J. Sabater i Mir. Trust and Reputation for Agent Societies. PhD thesis, Artificial Intelligence Research Institute, Universitat Autònoma de Barcelona, Barcelona, Spain, July 2002. 2, 45, 46, 47, 64, 66

[Sal02] M. Sallé. Electronic contract framework for contractual agents. In Proceedings of the Conference of the Canadian Society for Computational Studies of Intelligence on Advances in Artificial Intelligence, pages 349–353, London, United Kingdom, 2002. Springer-Verlag. 23

[SCCD94] J.S. Sichman, R. Conte, C. Castelfranchi, and Y. Demazeau. A social reasoning mechanism based on dependence networks. In A. G. Cohn, editor, Proceedings of the European Conference on Artificial Intelligence (ECAI'94), pages 188–192, Chichester, United Kingdom, August 1994. John Wiley & Sons. 45

[SCJ97] F. Santos, J. Carmo, and A. Jones. Action concepts for describing organised interaction. In R. A. Sprague, editor, Proceedings of the Hawai International Conference on System Sciences (HICSS-30), pages 373–382, Maui, Hawaii, United States of America, 1997. 25

[Sea69] J. R. Searle. Speech Acts : an essay in the philosophy of language. Cambridge University Press, 1969. 10, 11

[SF99] M. Schillo and Petra Funk. Learning form and about other agents in terms of social metaphors. In J. M. Vidal and S. Sen, editors, Proceedings of the Workshop "Agents Learning About, From and With other Agents" at International Joint Conference on Artificial Intelligence (IJCAI'99), 1999. 61, 66, 67

[SFJ02] J.-M. Seigneur, S. Farrell, and Ch. Jensen. Secure ubiquitous computing based on entity recognition. In Proceedings of the Workshop on "Security" at Ubiquitous Computing (UbiComp'02), Göteborg, Sweden, 2002. 2

[SFR99] M. Schillo, P. Funk, and M. Rovatsos. Who can you trust : Dealing with deception. In C. Castelfranchi, Y. Tan, R. Falcone, and B. S. Firozabadi, editors, Proceedings of the Workshop on "Deception, Fraud, and Trust in Agent Societies" at Autonomous Agents (AA'99), pages 81–94, Seattle, WA, United States of America, May 1999. 61, 66

[SFR00] M. Schillo, P. Funk, and M. Rovatsos. Using trust for detecting deceitful agents in artificial societies. Applied Artificial Intelligence, 14(8) :825–849, 2000. 146

[Sha87] S. P. Shapiro. The social control of impersonal trust. American Journal of Sociology, 93 :623–658, 1987. 32, 33

[SHT73] B.R. Schlenker, B. Helm, and J.T. Tedeschi. The effects of personality and situational variables on behavioral trust. Journal of Personality and Social Psychology, 25(3) :419–427, 1973. 48

[Sim95] M. R. Simmons. Recognizing the elements of fraud, 1995. http ://users.aol.com/marksimms/mrweb/fraudwww.htm. 25

[Sin91] M. P. Singh. Social and psychological commitments in multi-agent systems. In Proceedings of the AAAI Fall Symposium on Knowledge and Action at Social and Organizational Levels (longer version), pages 1–5, Monterey, CA, United States of America, November 1991. 10, 11, 28, 69

[Sin98] M. P. Singh. Agent communication languages : Rethinking the principles. IEEE Computer, 31(12) :40–47, December 1998. 10

[Sin99] M. P. Singh. An ontology for commitments in multi-agent systems : Towards a unification of normative concepts. AI and Law, 7 :97–113, 1999. 24

[Sin00] Munindar P. Singh. A social semantics for agent communication languages. In F. Dignum and M. Greaves, editors, Proceedings of the Workshop on "Agent Communication Languages" at the International Joint Conference on Artificial Intelligence (IJCAI'99), Issues in Agent Communication, pages 31–45. Springer-Verlag, Heidelberg, Germany, 2000. 10, 11, 12

[Smi80] R. G. Smith. The contract net protocol : High-level communication and control in a distributed problem solver. IEEE Transactions on Computers, 29(12) :1104–1113, 1980. 10

[SN85] J.E. Swan and J.J. Nolan. Gaining customer trust : A conceptual guide for the salesperson. Journal of Personal Selling & Sales Management, pages 39–48, November 1985. 48

[SS02] S. Sen and N. Sajja. Robustness of reputation-based trust : boolean case. In C. Castelfranchi and W.L. Johnson, editors, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'02), volume 1, pages 288–293, Bologna, Italy, July 2002. ACM Press, New York, NY, United States of America. 61, 66, 67

[ST95] Y. Shoham and M. Tennenholtz. On social laws for artificial agent societies : Off-line design. Artificial Intelligence, 73(1–2) :231–252, 1995. 21

[Str02] T. Stratulat. Systèmes d'agents normatifs : concepts et outils logiques. PhD thesis, Université de Caen, Caen, France, décembre 2002. 23

[SW49] Claude E. Shannon and W. Weaver. The Mathematical Theory of Communication. University of Illinois Press, Urbana, IL, United States of America, 1949. 10

[TBT95] R. Tuomela and M. Bonnevier-Tuomela. Norms and agreement. European Journal of Law, Philosophy and Computer Science 5, 41–46, 1995. 19, 20, 21, 35, 36, 174

[Tes04] Testbed. Trust Competition Testbed, 2004. http ://www.art-testbed.net. 175

[Tuo95] R. Tuomela. The Importance of Us : A Philosophical Study of Basic Social Norms. Stanford University Press, Stanford, CA, United States of America, 1995. 18, 19, 174

[Val95] A. Valente. Legal Knowledge and Engineering. IOS Press, Amsterdam, The Netherlands, 1995. 22

[VCSB07] L. Vercouter, S. J. Casare, J. S. Sichman, and A. A. F. Brandão. An experience on reputation models interoperability using a functional ontology. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'07), Hyderabad, India, 2007. (in press). 175

[vdT94] Leendert W. N. van der Torre. Violated obligations in a defeasible deontic logic. In Proceedings of the European Conference on Artificial Intelligence (ECAI'94), pages 371–375, 1994. 22

[Ven72] Z. Vendler. Res Cogitans : A Study in Rational Psychology. Cornell University Press, Ithaca, NY, United States of America, 1972. 11

[VFC05] F. Viganò, N. Fornara, and M. Colombetti. An event driven approach to norms in artificial institutions. In O. Boissier, J. Padget, V. Dignum, G. Lindemann, E. Matson, S. Ossowski, J. S. Sichman, and J. Vázquez-Salceda, editors, Proceedings of the Workshop "Agents, Norms and Institutions for REgulated Multi-Agent Systems" (ANIREM) at Autonomous Agents and Multi-Agent Systems (AAMAS'05), volume 3913 of Lecture Notes in Computer Science, pages 142–154, Utrecht, The Netherlands, July 2005. Springer-Verlag, Berlin, Germany. 24

[von51] G.H. von Wright. Deontic logic. Mind, 60 :1–15, 1951. 22, 25

[von68] G.H. von Wright. An essay in deontic logic and the general theory of action. Acta Philosophica Fennica, 21, 1968. 22

[von81] G.H. von Wright. New studies in Deontic Logic : Norms, Actions, and the Foundations of Ethics, chapter On the Logic of Norms and Actions, pages 3–35. D. Reidel, 1981. 22

[von93] G.H. von Wright. On the logic and ontology of norms. Philosophical Logic, pages 89–107, 1993. 19, 20

[vRW05] W. van der Hoek, M. Roberts, and M. Wooldridge. Knowledge and social laws. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M.P. Singh, and M. Wooldridge, editors, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'05), volume 2, pages 674–681, Utrecht, The Netherlands, July 2005. ACM Press, New York, NY, United States of America. 21

[VS03] J. Vàzquez-Salceda. The role of norms and electronic institutions in multi-agent systems applied to complex domains. The HARMONIA framework. PhD thesis, Universitat Politècnica de Catalunya, Barcelona, Spain, April 2003. 18, 23

[VSD03] J. Vàzquez-Salceda and F. Dignum. Modelling electronic organizations. In V. Marik, J. Müller, and M. Pěchouček, editors, Proceeding of the Central and Eastern European conference on Multi-Agent Systems(CEEMAS'03), volume 2691 of Lecture notes in computer science, pages 584–593, Prague, Czech Republic, June 2003. Springer-Verlag, Berlin, Germany. 21

[VSDD05] J. Vàzquez-Salceda, V. Dignum, and F. Dignum. Organizing multi-agent systems. Journal of Autonomous Agents and Multi-Agent Systems (JAAMAS), 11(3) :307–360, November 2005. 23

[WK95] D. Walton and E. Krabbe. Commitment in Dialogue. SUNY Press, 1995. 12

[Wor05] WordNet, 2005. http ://wordnet.princeton.edu/perl/webwn. 34

[WsI04] A. Whitby, A. Jøsang, and J. Indulska. Filtering out unfair ratings in bayesian reputation systems. In R. Falcone, K.S. Barber, J. Sabater, and M. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'04), pages 106–117, New York, NY, United States of America, July 2004. 145

[WV03] Y. Wang and J. Vassileva. Bayesian network-based trust model in peer-to-peer networks. In R. Falcone, S. K. Barber, L. Korba,

and M. P. Singh, editors, Proceedings of the Workshop on "Trust, Privacy, Deception, and Fraud in Agent Societies" at Autonomous Agents and Multi-Agents Systems (AAMAS'03), pages 57–68, Melbourne, Australia, July 2003. 46, 47, 62, 66, 67

[YIO04] H. Yamamoto, K. Ishida, and T. Ohta. Trust formation in a C2C market : Effect of reputation management system. In R. Falcone, K.S. Barber, J. Sabater, and M. Singh, editors, Proceedings of the Workshop on "Trust in Agent Societies" at Autonomous Agents and Multi-Agent Systems (AAMAS'04), pages 126–136, New York, NY, United States of America, July 2004. 146

[ZMM99] G. Zacharia, A. Moukas, and P. Maes. Collaborative reputation mechanisms in electronic marketplaces. In Proceedings of the Hawaii International Conference on System Sciences (HICSS-32), volume 08, page 8026, Maui, Hawaii, United States of America, 1999. IEEE Computer Society, Washington, DC, United States of America. 46, 50, 57, 60, 66

École Nationale Supérieure des Mines de Saint-Étienne

N° d'ordre : 426 I.

Guillaume Muller

Using norms and reputations to detect and sanction contradictions – Contribution to the social control of interactions in open and decentralized multi-agent systems

Computer science

Multi-Agent System, Social Control, Trust, Reputation, Social Norm, Social Commitment

Abstract :
Open and Decentralized Multi-Agent Systems (ODMAS) are particularly vulnerable to the introduction of badly designed or malevolent agents. It is therefore necessary to control such systems. In this thesis, we propose the L.I.A.R. model, which enables agents to control their peers' interactions, thanks to a reputation model. Agents equiped with the L.I.A.R. model can first, represent interactions they perceive with the help of a social commitment model. They can also model the rules that each agent should follow thanks to a model of social norms. By comparing observed behaviours with the norms they know, agents are able to evaluate their peers and to estimate a reputation level to associate to each of them. Agents are then able to make a decision about the sanctions they wish to apply to their peers, based on these levels of reputation. Thanks to the complete integration of both steps: evaluation of the perceived behaviours and decision of the sanctions to apply, the L.I.A.R. model allows the agents to establish a fuly automatic social control of agents' interactions. Various experimentations have been conducted with this model in a peer-to-peer context in order to show how agents where able control their peers' interactions.

École Nationale Supérieure des Mines de Saint-Étienne

N° d'ordre : 426 I.

Guillaume MULLER

Utilisation de normes et de réputations pour détecter et sanctionner les contradictions – Contribution au contrôle social des interactions dans les systèmes multi-agents ouverts et décentralisés

Informatique

Système Multi-Agent, Contrôle Social, Confiance, Réputation, Norme Sociale, Engagement Social

Résumé :
Les Systèmes Multi-Agents Ouverts et Décentralisés (SMAOD) sont particulièrement vulnérables à l'introduction d'agents mal conçus ou malveillants. Il est donc nécessaire de contrôler ces systèmes. Dans cette thèse, nous proposons le modèle L.I.A.R., permettant aux agents eux-mêmes de mettre en place un contrôle des interactions des autres agents, à l'aide d'un modèle de réputation. Ce modèle permet d'abord aux agents de représenter les interactions qu'ils perçoivent grâce à des engagements sociaux, ainsi que de modéliser les règles que chaque agent doit respecter à l'aide de normes sociales. En comparant les comportements qu'ils ont observés aux normes dont ils ont connaissance, les agents sont capables d'évaluer leurs pairs et d'estimer les niveaux de réputation qu'ils leur associent. Ensuite, les agents peuvent décider des sanctions à appliquer en s'appuyant sur les niveaux de réputation ainsi estimés. Grâce à l'intégration des deux phases : évaluation des comportements et décision des sanctions à appliquer, le modèle L.I.A.R. permet de mettre en place un contrôle social des interactions entièrement automatisé. Diverses expérimentations ont été menées avec ce modèle dans le cadre d'un réseau pair-à-pair, afin de montrer comment les agents contrôlent les interactions de leurs pairs.