



Une approche multi-agents pour la composition de services Web fondée sur la confiance et les réseaux sociaux

Amine Louati

► To cite this version:

Amine Louati. Une approche multi-agents pour la composition de services Web fondée sur la confiance et les réseaux sociaux. Intelligence artificielle [cs.AI]. Université Paris Dauphine - Paris IX, 2015. Français. NNT : 2015PA090035 . tel-01236284

HAL Id: tel-01236284

<https://theses.hal.science/tel-01236284>

Submitted on 10 Dec 2015

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

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

Université Paris-Dauphine
ÉCOLE DOCTORALE DE DAUPHINE
LABORATOIRE LAMSADE

THÈSE

pour obtenir le grade de
DOCTEUR DE L'UNIVERSITÉ PARIS-DAUPHINE
Spécialité : INFORMATIQUE

**Une approche multi-agents pour la composition
de services Web fondée sur la confiance et les
réseaux sociaux**

présentée par
Amine LOUATI

Soutenue publiquement le 13 Octobre 2015 devant le jury :

Suzanne	PINSON	Directeur de thèse	Université Paris-Dauphine, France
Joyce	EL HADDAD	Co-encadrant	Université Paris-Dauphine, France
Laurent	VERCOUTER	Rapporteur	INSA de Rouen, France
René	MANDIAU	Rapporteur	Université de Valenciennes et du Hainaut-Cambrésis, France
Salima	BENBERNOU	Examineur	Université Paris Descartes, France
Amal	El FALLAH SEGHROUCHNI	Examineur	Université Pierre et Marie Curie, France

Remerciements

Il me sera très difficile de remercier tout le monde car c'est grâce à l'aide de nombreuses personnes que j'ai pu mener cette thèse à son terme. J'aimerais tout d'abord remercier et exprimer toute ma reconnaissance à mon encadrante, Joyce EL HADDAD. Elle m'a initié à la recherche dans un domaine qui m'a toujours motivé. Meticuleuse et perfectionniste, toujours disponible, elle m'a prodigué des conseils inestimables, dans tous les domaines, tout au long de ma thèse. Je tiens à lui exprimer toute ma gratitude.

Je remercie ma directrice de thèse, Suzanne PINSON, pour m'avoir appris à être plus autonome tout au long de ce travail de recherche. Ses commentaires et ses critiques sur mon travail m'ont chaque fois permis de m'orienter dans les bonnes directions. Je suis ravi d'avoir travaillé en sa compagnie car outre son appui scientifique, elle a toujours été là pour me soutenir et me conseiller au cours de l'élaboration de cette thèse. Je suis très fier de la formation de chercheur acquise sous son encadrement.

Je souhaiterais exprimer ma gratitude à monsieur Laurent VERCOUTER de s'être rendu disponible pour la pré-soutenance, de m'avoir prodigué maints remarques, conseils bibliographiques, et d'avoir accepté la fonction de rapporteur. De même, je suis particulièrement reconnaissant à monsieur René MANDIAU de l'intérêt qu'il a manifesté à l'égard de ce travail en s'engageant à être rapporteur. Je tiens également à remercier madame Amal EL FALLAH SEGHRUCHNI et madame Salima BENBERNOU pour l'honneur qu'elles m'ont fait d'être dans mon jury de thèse et pour le temps qu'elles ont consacré pour examiner mon travail de recherche. Pour tout cela je les remercie.

Je remercie toutes les personnes formidables que j'ai rencontrées par le biais du LAMSADE. Merci pour votre support et vos encouragements. Je pense particulièrement à notre directeur, Alexis TSOUKIAS et à tous les professeurs avec lesquels j'ai travaillé et échangé. Les nombreuses discussions que j'ai pu avoir avec chacun m'ont beaucoup apporté. Merci donc à Ali Ridah MAHJOUB, Tristan CAZENAVE, Mohamed Ali ALOULOU, Dario COLAZO, Stefano MORETTI, Elsa NEGRE et Gabriella PIGOZZI. Je tiens également à remercier tout le personnel administratif et en particulier

Juliette, Mireille, Nathalie, Hawa, Katerina et Mehdi pour leur gentillesse, disponibilité et leur soutien logistique. Je voudrais exprimer particulièrement toute mon amitié à Olivier pour sa compétence et son humour.

Je remercie tous les membres et ex-membres du LAMSADE pour le climat sympathique dans lequel ils m'ont permis de travailler. Les nombreuses discussions que j'ai pu avoir avec chacun m'ont beaucoup apporté. Merci donc à ceux qui m'ont entouré et m'ont conseillé, ainsi que tous les thésards pour m'avoir supporté tous les jours depuis plusieurs années, même si ce n'est pas encore fini ou a juste commencé ! : Tom, Nathanaël, Mohamed Lamine, Rafael, Marek, Justin, Anaëlle, Fabian, Saeed, Satya, Mayassa, Thomas, Yassine, Raja, Mohamed Khalil, Youcef, Lynda, Ian et Nicolas. J'adresse aussi mes remerciements à nos docteurs : Irene, Florian, Dalal, Edouard, Renaud, Raouia, Amal, Souhila, Abdallah, Mohamed, Lydia, Lyes et Liangliang. Ils m'ont beaucoup aidé et sont devenus des amis à qui je souhaite tout le courage qu'ils m'ont apporté. Toutes mes gratitudes à Myriam qui a pris le temps de lire les premières pages de cette thèse.

Mes remerciements vont aussi à ma famille et mes amis qui, avec cette question récurrente, « quand est-ce que tu la soutiens cette thèse ? », bien qu'angoissante en période fréquente de doutes, m'ont permis de ne jamais dévier de mon objectif final. Malgré mon éloignement depuis de (trop) nombreuses années, leur intelligence, leur confiance, leur tendresse, leur amour me portent et me guident tous les jours. Merci pour avoir fait de moi ce que je suis aujourd'hui. Est-ce un bon endroit pour dire ce genre de choses ? Je n'en connais en tous cas pas de mauvais. Je vous aime.

Ces remerciements ne peuvent s'achever, sans une pensée pour ma première fan (et correctrice des fautes d'orthographe de cette thèse !) : ma mère. Sa présence et ses encouragements sont pour moi les piliers fondateurs de ce que je suis et de ce que je fais.

Résumé

Avec le nombre croissant de services Web publiés y compris ceux qui sont générés dans *les réseaux sociaux*, nombreuses *approches de découverte et de sélection de services Web* ont été proposées dans la littérature. Ces dernières utilisent souvent des techniques d'appariement basées sur les annuaires centralisés (e.g. UDDI, ebXML) qui ne considèrent que les attributs de qualité de service (QoS). Dans de nombreux cas, cela est insuffisant pour évaluer la fiabilité des fournisseurs de services et souvent échoue à se conformer aux exigences non fonctionnelles des demandeurs de services. Ceci dû au fait que les annuaires sont incapables (i) de prendre en considération les dimensions sociales et (ii) de capitaliser sur les informations résultant des expériences précédentes. Comment intégrer cet aspect social dans la découverte et la sélection de services est un nouveau défi à relever par la communauté de services Web.

Pour relever ce défi, nous proposons l'utilisation des *systèmes multi-agents* car ils ont démontré leur capacité à modéliser des processus distribués tout en utilisant l'intelligence, la représentation de connaissances et le raisonnement ainsi que les métaphores sociales comme *la confiance*. Chaque utilisateur dans le réseau social est associé à un agent autonome non malveillant qui est équipé d'un ensemble de services avec leurs valeurs de QoS. Plus précisément, dans cette thèse, nous améliorons la découverte et la sélection de services en intégrant *une vue sociétale* dans la modélisation de la confiance. Basés sur les relations entre les agents, leurs expériences antérieures et les informations extraites des réseaux sociaux, nous définissons notre modèle de confiance en tant que concept compositionnel qui comprend une composante *sociale*, une composante *d'expertise* et une composante *de recommandation*. La composante sociale juge s'il est intéressant de suivre un fournisseur avant d'utiliser ses services (à savoir, la confiance en la crédibilité sociale). La composante d'expertise estime si un service se comporte bien et comme prévu (à savoir, la confiance en l'expertise). La composante de recommandation vérifie si un agent est pertinent ou pas et si l'on peut compter sur ses recommandations (à savoir, la confiance en la recommandation). Nous proposons ensuite un algorithme distribué pour la découverte de services utilisant la confiance

entre les agents ainsi que les systèmes de références dans les réseaux sociaux.

Cependant, lors de la recherche d'un service dans un réseau social, les agents (le demandeur de services et les fournisseurs de services) peuvent ne pas avoir d'interaction directe ou d'expérience commune. Cela exige une méthode pour déduire la confiance entre eux. Nous proposons un modèle probabiliste pour calculer la confiance entre les agents non adjacents tout en tenant compte des rôles des agents intermédiaires. Dans le cas où aucun service unitaire n'est capable de satisfaire les fonctionnalités requises par un demandeur, il faut pouvoir combiner les services découverts pour répondre à son besoin. Comme plusieurs agents peuvent offrir différents services avec différentes valeurs de QoS pour les fonctionnalités requises, la coopération parmi eux est essentielle pour réaliser toutes *les compositions de services* possibles. Nous présentons dans cette thèse un processus original de *formation de coalitions fondé sur la confiance* pour la composition de services dans les réseaux sociaux. Pour ce faire, nous introduisons une nouvelle composante de confiance appelée *confiance en la coopération* qui permet aux fournisseurs de décider avec qui interagir au cours de la sélection des membres pour compléter la coalition. Notre processus de formation de coalitions engageant des agents égoïstes est *incrémentale, dynamique et recouvrant*.

Les résultats expérimentaux montrent que notre approche multi-agents est efficace, plus performante que les approches existantes et peut offrir des résultats plus dignes de confiance à faible coût de communication.

Mots clés : Découverte, sélection et composition de services, système multi-agents, modèle de confiance, systèmes de références, réseaux sociaux, formation de coalitions.

Table des matières

1	Introduction générale	1
2	Notions préliminaires	11
2.1	Architecture Orientée Services	12
2.1.1	Description des entités d'interaction	12
2.1.2	Composition de services Web	15
2.2	Réseaux Sociaux	17
2.2.1	Représentation d'un réseau social	18
2.2.2	Types de réseaux sociaux	20
2.2.3	Domaines d'utilisation des réseaux sociaux	23
2.3	Systèmes multi-agents	25
2.4	Terminologie et concepts	28
2.5	Conclusion	30
3	Découverte et sélection de services Web : État de l'art	31
3.1	Catégorisation des approches existantes	32
3.2	Approches basées sur la réputation	33
3.3	Approches basées sur la recommandation	36
3.3.1	Recommandation basée sur le filtrage par le contenu	36
3.3.2	Recommandation basée sur le filtrage collaboratif	37
3.3.3	Recommandation hybride	39
3.4	Réseaux Pair à Pair (P2P)	40
3.5	Systèmes de références	41
3.6	Conclusion	44
4	La confiance : définition, topologie et modélisation	47
4.1	Introduction	48

4.1.1	Définition de la notion de confiance	48
4.1.2	Caractérisation de la confiance	50
4.2	Sources d'information pour la confiance	52
4.3	Topologie du modèle de la confiance	55
4.3.1	La confiance en l'expertise	55
4.3.2	La confiance en la crédibilité sociale	57
4.3.3	La confiance dans les systèmes de recommandation	59
4.3.4	Discussion	61
4.4	Confiance en la crédibilité sociale	64
4.4.1	La position sociale (SPo)	64
4.4.2	La proximité sociale (SPr)	67
4.4.3	La similarité sociale (SSi)	68
4.4.4	Calcul de la confiance en la crédibilité sociale	74
4.5	Confiance en l'expertise	75
4.5.1	La spécialisation (Sp)	77
4.5.2	La disponibilité (Di)	77
4.5.3	La qualité (Eval)	78
4.5.4	Calcul de la confiance en l'expertise	81
4.6	Confiance en la recommandation	82
4.7	Conclusion	84
5	Approche multi-agents pour la découverte et la sélection de services	87
5.1	Méthodologie de conception	88
5.1.1	Choix de la méthodologie de conception et les hypothèses retenues	89
5.1.2	Définition des rôles des agents	90
5.1.3	Modèle d'organisation sociale : systèmes de références	92
5.1.4	Définition des interactions entre agents	94
5.2	Architecture d'un agent cognitif basée sur la confiance	98
5.2.1	Description de l'architecture	98
5.2.2	Protocole de propagation	103
5.2.3	Comportement de l'agent	105
5.3	Algorithme distribué pour la découverte de services Web dans les réseaux sociaux	106
5.3.1	Algorithme distribué basé sur la confiance pour la découverte de services dans les réseaux sociaux	108

5.3.2	Algorithme de propagation de confiance et modèle probabiliste et contextuel de calcul	114
5.3.3	Sélection de services	119
5.4	Conclusion	119
6	Composition de services Web fondée sur la confiance et la formation de coalitions	121
6.1	Approches multi-agents pour la composition de services	122
6.2	Modèle multi-agents	127
6.2.1	Définition des concepts	127
6.2.2	Modèle multi-agents basé sur un broker	129
6.3	Description du processus de formation de coalitions	130
6.3.1	Phase de génération de coalitions initiales	131
6.3.2	Phase de sélection des membres	133
6.3.3	Phase de sélection de la meilleure coalition	139
6.4	Conclusion	139
7	Mise en œuvre et validation expérimentale	141
7.1	Choix techniques et implémentation	142
7.1.1	La plateforme JADE et les agents développés	143
7.1.2	Choix du langage de stockage	144
7.2	Description des données	146
7.2.1	Description des jeux de données simulées	147
7.2.2	Description des jeux de données réelles	147
7.3	Évaluations empiriques et résultats	150
7.3.1	Implémentation	152
7.3.2	Évaluation de la performance de notre approche de découverte de services	154
7.3.3	Évaluation de la performance de notre approche de sélection de services	161
7.3.4	Évaluation de la performance de notre approche de composition de services	166
7.4	Conclusion	169
8	Conclusion générale	171
	Bibliographie	198

Table des figures

1.1	Le réseau social multi-relationnel (RSMR) de l'agent Alice a_0	7
2.1	Déploiement, découverte et invocation de services Web	14
2.2	Visualisation du réseau des personnages du roman « les Misérables » à l'aide de Gephi : la taille des sommets est proportionnelle à leur degrés et la couleur indique l'appartenance des sommets aux différentes communautés générées par un algorithme de partitionnement	19
2.3	Un réseau égocentrique avec des sommets étiquetés [McAu 12]	21
2.4	Trois exemples de réseaux hétérogènes : un graphe multimodal d'auteur-conférence (a), un réseau multi-relationnel (b) et un réseau multiplexe (c)	22
3.1	Un exemple d'une chaîne de références.	42
3.2	Un graphe de références généré à partir d'une requête [Sing 01].	43
4.1	Processus de calcul des mesures de la confiance en la crédibilité sociale	65
4.2	Position sociale de l'agent a_4 et l'agent a_8 dans le graphe G	66
4.3	Proximité sociale entre l'agent a_0 et l'agent a_9 dans le graphe G	68
4.4	Similarité de voisinage entre l'agent a_0 et les agents a_1 et a_8 dans le graphe G	69
4.5	Profils de deux agents dans le réseau social	73
4.6	Plusieurs fonctions de pondération pour différentes valeurs de σ [Leti 08]	80
5.1	Comparaison de deux systèmes de références	93
5.2	Automate de contextualisation des interactions	97
5.3	Architecture d'agent cognitif orientée services et basée sur la confiance	101
5.4	Diagramme AUMML d'interaction du protocole de propagation	104
5.5	Diagramme d'activité d'un agent	105

5.6	Approche multi-agent de découverte et de sélection de services basée sur la confiance	107
5.7	L'évolution du seuil de la confiance en fonction de la distance	112
5.8	Réseau social de confiance	113
5.9	Exemple de chaîne, dans laquelle chaque agent est caractérisé par son identifiant et son rôle.	115
5.10	Structure séquentielle de propagation de confiance.	117
6.1	Notre modèle multi-agents basé sur un broker	129
6.2	(a) Réseau social et (b) Réseau social de confiance : représentation plane	131
6.3	Réseau social de confiance : représentation multi-couche.	132
6.4	Protocole AUML de sélection d'un membre	137
7.1	Les réseaux « random », « scale-free » et « small-world » [Xiao 10a]. . .	148
7.2	Le réseau Enron Email (Visualisation à l'aide de Gephi).	149
7.3	Le réseau Facebook (Visualisation à l'aide de Gephi).	151
7.4	Temps de calcul de la confiance en la crédibilité sociale en fonction de la taille du réseau	154
7.5	Effet de la taille du réseau social sur le coût de la communication. . . .	155
7.6	Performance de l'approche de découverte : Efficacité.	157
7.7	Performance de l'approche de découverte : Efficience.	158
7.8	Valeur de la précision en fonction de la taille du réseau.	160
7.9	Valeur du rappel en fonction de la taille du réseau.	160
7.10	Performance de l'approche de sélection : Utilité	163
7.11	Performance de l'approche de sélection : Satisfaction des demandeurs .	165
7.12	Nombre moyen de messages échangés par scénario de test.	167
7.13	Pourcentage de coalitions générées par scénario de test	168
7.14	La fréquence moyenne d'abandon par scénario de test	169

Liste des tableaux

4.1	Tableau comparatif des approches de découverte et de sélection de services basées sur la confiance	64
4.2	Répartition des valeurs des champs de l'item "emploi" dans les profils des accointances d'Alice	74
5.1	Table d'interaction personnelle (PIT_k) d'un agent a_k	99
5.2	Table d'Histoire (HT_k) d'un agent a_k	100
7.1	Définition de scénarios de test	166

Chapitre 1

Introduction générale

Le paradigme de l'informatique orientée services a émergé au cours des dernières années et s'est imposé graduellement dans le cadre de la promotion des applications Web complexes conçues à partir de simples composants distribués appelés *services Web* [Papa 08]. Un service Web est défini par le W3C¹ comme un système logiciel conçu pour permettre une interaction interopérable de machine à machine via un réseau. Parmi les exemples courants de services Web, citons le service de réservation de chambres d'hôtel, le service de réservation de billets de train, les services d'achats en ligne, etc. Ces services Web sont publiés dans des répertoires centralisés appelés *annuaires de services*. Les plus connus sont les annuaires UDDI² (Universal Description, Discovery and Integration) et ebXML³ (Electronic Business using eXtensible Markup Language) utilisés pour publier la description fonctionnelle des services Web dans des interfaces publiques WSDL⁴ (Web Services Description Language) et les indexer par mots-clés. Cependant, les annuaires présentent plusieurs inconvénients dus principalement au fait que les services Web sont traités comme des éléments isolés et indépendants qui ont seulement connaissance d'eux-mêmes, mais pas de leurs applications ou utilisateurs. En plus, la structure actuelle des annuaires n'autorise qu'une découverte fonctionnelle et/ou sémantique de services qui échoue souvent à satisfaire les exigences non fonctionnelles des demandeurs de services. Enfin, la plus grande limite des annuaires est qu'ils ne prennent pas en considération les dimensions sociales entre les demandeurs et les fournisseurs de services et les informations résultant de leurs expériences précédentes. Comment intégrer cet aspect social dans la découverte, la sé-

1. <http://www.w3.org/TR/ws-arch/>.

2. <http://uddi.xml.org/>.

3. <http://www.ebxml.org>.

4. <http://www.w3.org/TR/wsdl>.

lection et la composition de services est un nouveau défi à relever par la communauté de services Web.

La représentation de connaissances, la cognition et l'interaction, points faibles des services Web, sont des domaines qui ont largement été explorés par la recherche dans les *systèmes multi-agents (SMA)*. Dans l'optique *d'agentification des services Web*, plusieurs initiatives ont été menées au cours des dernières années [Amor 03, Char 13, Hend 01, Huhn 02, Huhn 03, Syca 03, Sree 04, Mull 05, Maxi 05a, Sial 11]. Pour mettre en œuvre la combinaison des technologies agent et services Web, ils ont développé des solutions à base d'agents pour la découverte, la sélection et la composition de services. Les agents, qu'ils soient coopératifs ou compétitifs, sont capables de réaliser des tâches complexes, supportent différentes formes d'interactions y compris la négociation et la coordination, disposent de capacités bien développées pour exprimer et raisonner sur les données sémantiques et enfin proposent des solutions flexibles qui s'adaptent aux besoins et aux préférences de leurs utilisateurs. En conséquence, les SMAs semblent adaptés à la modélisation des problématiques (de distribution, d'adaptation et de collaboration) sous-tendues par la découverte, la sélection et la composition de services. Les travaux que nous présentons dans cette thèse s'inscrivent dans le cadre des systèmes multi-agents et les services Web. Nous nous intéressons à l'élaboration et la mise en œuvre d'une approche multi-agents pour les problèmes de découverte, de sélection, et de composition de services.

Le plein essor du Web 2.0 a entraîné l'apparition de nouvelles formes de publication et de consommation de services à travers les applications et les plateformes sociales connues sous le nom de *réseaux sociaux*. Les réseaux sociaux sont des communautés d'individus (collègues, co-auteurs, ...), d'organisations (entreprises, unités de recherche, ...) ou n'importe quelle ensemble d'entités sociales (livres, blogs, ...) [Stan 94] qui interagissent en ligne et qui sont reliés par différents types de relations sociales telles que l'amitié, le partenariat ou la famille dans des contextes variés tels que le divertissement, la politique, la religion ou les affaires. De nombreuses études [Bras 11, Gros 05, Maam 11b, Al S 12] ont démontré la puissance et le potentiel attractif des réseaux sociaux aussi bien pour les individus ordinaires que pour les professionnels. Des exemples comme Facebook⁵, LinkedIn⁶ et Friendster⁷ montrent clairement la volonté des individus à vouloir utiliser leurs réseaux sociaux pour publier leurs propres services (co-voiturage, aide aux devoirs, garde d'enfants ...) ou pour en trouver d'autres. Par exemple, lorsque l'on recherche un service pour garde d'enfants, une bonne stratégie consiste à consulter le réseau social du demandeur de services afin

5. <https://www.facebook.com/>

6. <http://www.linkedin.com/>

7. <http://www.friendster.com>

de découvrir des fournisseurs de services de garde d'enfants. A l'instar des individus, le monde de l'entreprise a compris aussi l'intérêt de l'utilisation des réseaux sociaux comme moyen de promotion, d'échange et de partage de services Web. Cette nouvelle tendance a engendré une augmentation exponentielle du nombre de services disponibles donnant naissance ainsi à un nouveau défi pour la communauté « services Web » ; *la découverte, la sélection et la composition de services à partir des réseaux sociaux*.

Un réseau social peut être considéré comme un SMA dans lequel les utilisateurs sont associés à des agents sociaux, non malveillants et équipés d'un ensemble de services avec les valeurs de leurs qualité de service (QoS). Ils agissent au nom de leurs utilisateurs de manière autonome afin de répondre à leurs besoins. C'est dans cette perspective sociale que certaines approches utilisent des mécanismes comme *le classement, la confiance, la recommandation ou la réputation* pour réaliser *une découverte et une sélection sociales de services à partir du Web*. Les systèmes les plus connus sont : le système de classement de pages de Google [Page 99], le système de recommandation et de réputation d'Amazon et de E-Bay (par exemple, [Resn 02]), et le système basé sur la confiance/réputation de Epinions (par exemple, [Mass 05]).

La confiance est un phénomène social complexe qui est à la base de toutes les interactions sociales [Abdu 00]. Dans les SMAs, elle a un rôle déterminant dans la prise de décision d'un agent lorsqu'il veut interagir avec d'autres [Cast 98, Falc 04]. Un agent doit avoir suffisamment confiance en un autre agent pour engager une interaction et déléguer une tâche. Dans un contexte orienté services, la confiance est généralement liée à la description non fonctionnelle d'un service exprimée en fonction de ses attributs de QoS [Li 12, Vu 05, Wang 06b, Xu 07]. Par exemple, si un agent Alice fait confiance à un autre agent Bob cela veut dire qu'Alice prévoit que le service fourni par Bob respecte ses QoS annoncées. Toutefois, comme démontré dans de nombreuses études [Al S 12, Li 09, Maar 10, Wang 07b, Bill 07, Bans 10, Da S 12], un processus de découverte, de sélection ou de composition de services basé sur les attributs de QoS ne serait pas efficace pour évaluer la fiabilité des fournisseurs et faire la distinction entre les bons et les mauvais services. Ainsi, un modèle computationnel de confiance doit s'inspirer de la façon dont la confiance fonctionne entre les individus dans la société. Pour ce faire, nous proposons un modèle de confiance basé sur les propriétés sociales de la confiance dans le monde réel en suivant les principes de la sociologie et de la psychologie sociale [Bers 98, Luhm 12, Rowl 79, Fisk 13]. Nous la définissons comme un concept compositionnel bâti sur trois composantes : (i) *la confiance en la crédibilité sociale*, (ii) *la confiance en l'expertise* et (iii) *la confiance en la recommandation*. Une crédibilité sociale élevée indique la connaissance des fournisseurs et des recommandeurs pertinents. Une grande expertise fait référence à des services avec une bonne qualité de réalisation. Un degré de recommandation élevé témoigne de la fiabilité des recomman-

dations obtenues. Plus précisément, la confiance en la crédibilité sociale évalue le niveau de confiance qu'un agent peut accorder à un autre agent avant d'utiliser ses services. Basées sur des techniques d'analyse des réseaux sociaux appliquées à un Réseau Social Multi-Relationnel (RSMR) et les informations extraites, trois mesures sont calculées : (1) la position sociale qui donne une indication sur la puissance sociale d'un agent, (2) la proximité sociale dans le graphe qui évalue le coût moyen d'un chemin entre deux agents, et (3) la similarité sociale entre deux agents qui repose sur la comparaison de leurs profils et de leurs voisinages. Nous pensons que la similarité sociale entre deux agents ne se limite pas à la similarité de leurs profils (la confiance et la similarité de profils sont corrélées comme il a été démontré par [Zieg 04]), mais elle peut également inclure la similarité de leurs voisinages. La confiance en l'expertise évalue le niveau de confiance que le demandeur porte à la qualité des services offerts par un agent. Fondées sur les attributs de qualité de services, trois mesures sont calculées : (1) la spécialisation qui indique le pourcentage des utilisations réussies dans le passé d'un service par rapport à tous les services offerts par le même fournisseur, (2) la disponibilité qui estime la probabilité qu'un service est opérationnel au moment de l'invocation, et (3) la qualité qui évalue la qualité d'exécution d'un service basée sur les notes attribuées par le demandeur de services après chaque exécution réussie.

Dans un contexte de réseau social, lorsqu'un agent demandeur est à la recherche d'un service, il soumet sa requête à ses accointances sociales (dans notre cas, ce sont ses voisins dans le réseau social) et plus particulièrement à ceux qui sont dignes de confiance. En l'absence d'une entité centrale de contrôle, les agents par leur perception partielle de l'environnement, ont intérêt à collaborer en échangeant leurs propres expériences ou en recommandant des références auxquelles ils font confiance. La question qui se pose est de savoir comment propager la requête et également les valeurs de confiance entre les agents en tenant compte de la topologie du réseau social. Les *systèmes de références*⁸ [Yu 03, Yolu 03, Liu 05] sont bien adaptés à notre contexte de SMA à grande échelle car ils respectent non seulement la topologie de notre réseau social mais aussi ils reposent sur la relation de confiance entre les agents dans la propagation de la recherche. En l'absence d'expériences et de liens directs, les systèmes de références agissent à la fois comme un mécanisme social de découverte de services et aussi comme un système de recommandation. Une découverte de services basée sur un système de références favorisent la coopération des agents pour localiser des fournisseurs de services pertinents. Dans [Yu 03] et [Yolu 03], un système de références est utilisé pour la recherche de services dans les réseaux sociaux, mais ces approches présentent la limite suivante : elles supposent une prise de décision centralisée dans laquelle le demandeur

8. Les systèmes de références sont des organisations sociales qui assurent une relation étroite entre les agents et les relient sous forme de chaîne.

de services rassemble toutes les références possibles et décide de poursuivre la recherche en communiquant avec certaines des références proposées.

Compte tenu de la nature distribuée du réseau social et donc, de la perception partielle d'un agent, il n'est pas possible de recueillir et de contrôler toutes les informations en un seul agent et faire par la suite une évaluation significative de la confiance. Contrairement aux approches dans [Yu 03, Yolu 03], nous avons distribué la prise de décision associée à la diffusion de la recherche en propageant la requête ainsi que les valeurs de confiance entre les agents intermédiaires jusqu'au demandeur de services.

Dans le cas où aucun service Web atomique ne pourrait satisfaire les besoins du demandeur de services, il devrait être possible de composer plusieurs services Web pour y parvenir. La composition de services Web est le processus qui crée une application à valeur ajoutée en combinant les services Web existants dans un service composite qui a le potentiel de réduire l'effort et le temps de développement [Ponn 02]. Le problème de composition de services a suscité l'intérêt de nombreux travaux de recherche (par exemple, [Agga 04, Bena 03, Casa 01, Char 13, Siri 04, Medj 14]). Cependant, les techniques proposées échouent souvent à se conformer aux exigences non fonctionnelles des demandeurs. Ceci car elles ne prennent pas en considération les dimensions sociales telles que la confiance entre les demandeurs et les fournisseurs et les informations issues de leurs expériences précédentes. Comme les agents ont démontré la capacité de promouvoir la représentation de connaissances et les interactions ainsi que des métaphores sociales comme la confiance, nous proposons l'utilisation d'une approche multi-agents pour réaliser *une composition de services Web fondée sur la confiance*. Nos agents sont autonomes, égoïstes et équipés d'un ensemble de services avec les valeurs de leurs attributs de QoS. Étant donnée que plusieurs agents peuvent fournir les mêmes services avec différentes valeurs de QoS, la coopération entre eux est essentielle pour déterminer toutes les compositions de services possibles. *La formation de coalitions* est bien adaptée pour modéliser la coopération entre les agents égoïstes. Dans cette thèse, nous proposons un nouveau processus incrémental, dynamique et recouvrant de formation de coalitions dans lequel les agents coopèrent afin de fournir le service composite satisfaisant les besoins du demandeur. En termes de prise de décision, nous introduisons une nouvelle composante de confiance appelée *la confiance en la coopération* permettant aux membres d'une coalition d'évaluer la fiabilité d'un candidat afin d'accepter son adhésion.

Motivations

Ces recherches ont été motivées par un certain nombre de préoccupations que nous avons exprimées au sujet de la découverte, la sélection et la composition de services dans les réseaux sociaux.

En pratique, la prise de décision concernant des besoins critiques (par exemple, garde d'enfants) ou complexes (par exemple, voyage touristique) a toujours été une tâche difficile. Les modèles de décision classiques pour la découverte, la sélection et la composition de services Web sont des modèles automatiques basés sur la description fonctionnelle et non fonctionnelle des services publiés dans des annuaires centralisés. Toutefois, ces modèles, s'appuyant uniquement sur les attributs de QoS, ont peu de chances d'être efficaces au vu des attentes des utilisateurs. De plus, la mise en place à grande échelle des plateformes sociales comme les réseaux sociaux a donné naissance à de nouveaux enjeux et à de nouvelles perspectives. Les plus importants d'entre eux étant la gestion de l'augmentation considérable du nombre de services offerts et l'adaptation à la nouvelle tendance sociétale : les utilisateurs préfèrent recevoir des services de la part des fournisseurs qu'ils connaissent et qui sont dignes de confiance (c'est à dire, ceux avec lesquels ils ont interagi auparavant) plutôt qu'à partir des annuaires.

L'interaction, la sociabilité et le raisonnement, lacunes des services Web, sont par ailleurs les caractéristiques des agents. Les recherches menées sur la pertinence du mariage des SMAs avec les services Web nous procurent une motivation supplémentaire pour utiliser les agents. C'est ainsi que l'intégration de la dimension sociale par l'intermédiaire des agents est l'un des défis les plus importants auxquels est actuellement confrontée la communauté « services Web ».

Pour répondre au mieux aux besoins des utilisateurs, ces nouvelles formes de découverte, de sélection, et de composition sociales devraient prendre en considération tous les types de services disponibles y compris les services humains (par exemple, co-voiturage et garde d'enfants), les interactions sociales reflétant une certaine relation de confiance et aussi les données sémantiques (comme les profils et les préférences des utilisateurs) et structurelles (comme la position et la proximité sociale des utilisateurs) disponibles dans les réseaux sociaux. Ainsi, et afin d'ancrer notre recherche dans la réalité, nous allons introduire un exemple d'application qui sert de fil conducteur pour la présentation de notre modèle de découverte, de sélection et de composition de services dans les réseaux sociaux.

Scénario de motivation

L'agent Alice, voyage souvent à l'étranger dans le cadre de son travail pour assister à des conférences. Pour ce faire, elle doit préparer son voyage et réserver à l'avance pour profiter des meilleurs tarifs. La routine de son voyage comprend : le transport du domicile à l'aéroport, un transport pour la ville de destination, le transport de l'aéroport à l'hôtel, l'hébergement en hôtel et les visites touristiques pendant le temps libre. Aussi, Alice doit trouver quelqu'un pour garder son enfant durant son absence.

Pour les services de réservation de voyage, il existe un grand nombre de sites spécialisés dans la vente de billets d'avion en ligne. La majorité de ces sites proposent des offres fonctionnellement équivalentes et souvent avec les mêmes qualités de services. Quant au service de garde d'enfants qui est un besoin critique, il est offert par des fournisseurs humains non pris en compte par les annuaires de services.

Ainsi, d'un côté, la préparation du voyage s'annonce fastidieuse et non évidente et, de l'autre côté, la découverte d'un service de garde d'enfants est non faisable à partir des sources classiques. L'idée est donc de rechercher une autre source plus riche, intégrant le contexte social, et compréhensible par le grand public et qui inclut tout type de service, entre autre, les services humains comme la garde d'enfants. C'est dans cette approche comme nous l'avons indiqué précédemment que les réseaux sociaux se présentent comme les outils les plus appropriés pour remplir ces exigences.

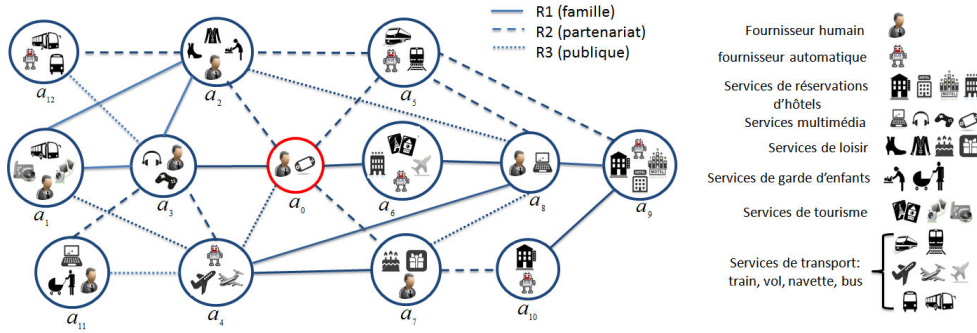


FIGURE 1.1 – Le réseau social multi-relational (RSMR) de l'agent Alice a_0

Dans ce cas, Alice modélisée par l'agent a_0 dans la figure 1.1 soumet sa requête à son réseau social et plus précisément à l'ensemble de ses accointances en spécifiant les fonctionnalités requises. Dans de telles conditions, la sélection d'un service ne se limite pas à un simple appariement entre les fonctionnalités requises et celles des services découverts mais inclut aussi l'aspect social exprimé par le degré de confiance entre le demandeur et le fournisseur de service. La question de la confiance est une préoccupation très importante du point de vue du demandeur de services car elle est la base des

interactions sociales, en particulier dans les applications Web, lorsque le demandeur de services doit choisir un fournisseur de services parmi plusieurs inconnus.

De plus, si Alice n'arrive pas à trouver localement tous les services requis, est ce qu'il existe un moyen qui lui permet de propager la recherche dans le réseau social afin de découvrir les bons fournisseurs dignes de confiance. Bien que certains agents n'aient pas une expertise requise, l'utilisation des réseaux sociaux offre une approche prometteuse pour faire des recommandations à d'autres agents pertinents pouvant satisfaire les besoins d'Alice. Cependant, lorsqu'un agent compte sur un autre agent, il y aura toujours un pourcentage considérable de risque associé à l'utilisation des services recommandés. La qualité d'un service offert ou recommandé peut être améliorée par l'évaluation de la crédibilité sociale du fournisseur ou de la qualité de recommandation du recommandeur en se basant sur les informations disponibles dans le réseau et les anciennes expériences. Une fois qu'Alice découvre la liste des fournisseurs dignes de confiance dans son réseau social, la sélection de l'ensemble de fournisseurs offrant le meilleur service composite représente un autre défi pour elle. Une des difficultés provient de la nature combinatoire du problème car à l'issue de la phase de découverte, chaque fonctionnalité requise peut être instanciée par un ou plusieurs fournisseurs. De plus, comme les fournisseurs sont autonomes, ils peuvent accepter ou non de coopérer avec d'autres dans une composition de services. Ces derniers, doivent s'organiser dans des groupes en fonction de leurs préférences de coopération pour répondre au mieux aux besoins d'Alice.

Dans cette thèse, nous démontrons, à travers une évaluation expérimentale, que nos approches multi-agents de découverte, de sélection et de composition de services fondées sur la confiance dans les réseaux sociaux sont plus efficaces que les approches traditionnelles.

Organisation du manuscrit

Cette thèse est divisée en sept chapitres :

- Le premier chapitre est consacré à la présentation des notions préliminaires sur l'architecture orientée services, les systèmes multi-agents et les réseaux sociaux. Nous décrivons les entités d'interaction dans une architecture orientée services puis nous décrivons la notion de composition de services Web et ses principales approches. Nous définissons aussi la terminologie utilisée et les concepts fonda-

mentaux nécessaires au positionnement de nos travaux et à la compréhension de la thèse.

- Dans le deuxième chapitre, nous catégorisons les approches existantes de découverte et de sélection de services. Nous analysons les différentes classes d'approches notamment celles intégrant l'aspect social. Nous étudions alors les travaux sur la coordination multi-agents dans les environnements distribués comme les réseaux sociaux à partir desquels nous pourrions esquisser une proposition satisfaisant les contraintes de notre problématique.
- Nous décrivons dans le troisième chapitre notre modèle de confiance. La première partie du chapitre concerne la définition de la notion de confiance à partir d'une revue de la littérature en sciences sociales. Cela va nous permettre d'identifier les sources d'information utilisées dans le calcul de la confiance et de définir la topologie et les caractéristiques du modèle proposé. La deuxième partie décrit formellement les différentes composantes de notre modèle de confiance à savoir la confiance en la crédibilité sociale, la confiance en l'expertise et la confiance en la recommandation.
- Nous montrons dans le quatrième chapitre comment ce modèle de confiance guide les interactions entre les agents et intervient dans leurs prises de décision durant le processus de découverte et de sélection de services. Dans la première partie de ce chapitre, nous présentons la méthodologie de conception en précisant les rôles des agents, le modèle d'organisation social et les actes de langages utilisés. Dans la deuxième partie, nous décrivons la structure interne de notre architecture basée sur la confiance combinant technologies agents cognitifs et services Web. Dans la dernière partie, nous présentons notre algorithme distribué basé sur la confiance pour la découverte de services dans les réseaux sociaux. Pour estimer la confiance entre les agents non adjacents, nous introduisons un nouveau modèle probabiliste et contextuel qui prend en considération le rôle des agents intermédiaires dans le calcul.
- Nous présentons dans le cinquième chapitre un processus original de formation de coalitions pour une composition de services dans les réseaux sociaux. Le processus de formation de coalitions est incrémental, dynamique et recouvrant. Les agents égoïstes coopèrent pour fournir des services composites qui répondent aux besoins complexes de l'utilisateur en se basant sur une prise de décision décentralisée guidée par la confiance.
- Le sixième chapitre présente les expérimentations que nous avons mises en place pour évaluer notre approche multi-agents de découverte, de sélection et de com-

position de services fondée sur la confiance. Nous présentons la plateforme expérimentale développée, les jeux de données réels et simulés utilisés, les méthodologies d'expérimentation appliquées et nous présentons et analysons les résultats obtenus.

- Enfin, nous concluons cette thèse par une synthèse des plus importantes contributions et la présentation des travaux en perspective.

Chapitre 2

Notions préliminaires

Sommaire

2.1	Architecture Orientée Services	12
2.1.1	Description des entités d'interaction	12
2.1.2	Composition de services Web	15
2.2	Réseaux Sociaux	17
2.2.1	Représentation d'un réseau social	18
2.2.2	Types de réseaux sociaux	20
	Réseaux complet, partiel et égocentrique	20
	Réseaux unimodal, multimodal et d'affiliation	21
	Réseaux multiplexes et multi-relationnel	22
2.2.3	Domaines d'utilisation des réseaux sociaux	23
2.3	Systèmes multi-agents	25
2.4	Terminologie et concepts	28
2.5	Conclusion	30

Ce chapitre est consacré à la présentation de quelques notions préliminaires sur l'architecture orientée services, les réseaux sociaux et les systèmes multi-agents. En particulier, dans la première section nous présentons les entités d'interaction dans une architecture orientée services puis nous décrivons la notion de composition de services Web et ses principales approches. La deuxième section est dédiée à la description des réseaux sociaux, leurs différents types de représentation et leurs principaux domaines

d'application. Nous donnons dans la troisième section un aperçu des systèmes multi-agents et l'avantage de l'introduction du paradigme agent dans l'architecture orientée services. Enfin, nous présentons dans la quatrième section la terminologie utilisée et les concepts fondamentaux nécessaires au positionnement de nos travaux et à la compréhension de la thèse.

2.1 Architecture Orientée Services

L'architecture orientée services (SOA) est une architecture logicielle s'appuyant sur un ensemble de composants simples appelés *services Web*. L'objectif d'une architecture orientée services est de décomposer une fonctionnalité complexe en un ensemble de fonctionnalités basiques, fournies par des services Web et de décrire finement le schéma d'interaction entre ces services Web. SOA implique quatre entités d'interaction : les services Web, les fournisseurs de services, les annuaires de service et les demandeurs de services auxquels nous rajoutons les attributs de QoS.

2.1.1 Description des entités d'interaction

Service Web. Un service Web selon le consortium W3C¹ est une application ou un composant logiciel conçu pour permettre une interaction interopérable de machine à machine via un réseau. Identifié par un URI (Uniform Resource Identifier), il dispose d'une interface décrite dans un format exploitable par des machines. Il peut également interagir directement avec d'autres services Web à travers le langage XML et en utilisant des protocoles Internet (dans la plupart des cas, le protocole utilisé est SOAP²). Les services Web sont décrits dans un langage standard appelé WSDL³. Cette description spécifie la fonctionnalité du service, les types de données employés dans les messages, le protocole de transport utilisé, ainsi que des informations pour localiser le service. Grâce à la description de services, les demandeurs s'abstraient des techniques d'implémentation des services, comme par exemple les langages de programmation utilisés ou leurs plateformes d'exécution.

Dans cette thèse, nous considérons un autre type de services appelé *service humain* tels que le co-voiturage, la garde d'enfants et la réparation de voitures. Nous supposons

1. World Wide Web Consortium est un organisme de normalisation des standards du Web.

2. http://www.w3.org/TR/#tr_SOAP

3. <http://www.w3.org/TR/wsdl>.

que ces services sont similaires aux services Web automatiques et possèdent la même description c'est à dire, la fonctionnalité du service, les types de données, ainsi que des informations pour localiser le service. Par conséquent, les termes service et service Web sont synonymes dans notre modèle.

Fournisseur de services. Un fournisseur de services est une entité qui fournit le service et le rend disponible pour les demandeurs. Un fournisseur de services peut être une entreprise, un organisme gouvernemental, une institution académique, etc. Un fournisseur peut fournir un ou plusieurs services. Cependant, un service est fourni par un fournisseur unique. Les fournisseurs ont des identités connues du public. Le fournisseur est propriétaire du service. Il peut ou non gérer effectivement le service. Dans notre modèle, les termes fournisseur de services et fournisseur représente le même type d'entité.

Demandeur de services. Un demandeur de services est une entité (par exemple, une application Web ou un autre service Web) qui invoque un service Web. Un demandeur de services humain peut également invoquer un service Web qui est représenté par un agent intelligent dans le réseau social. L'agent est donc responsable de toutes les interactions du demandeur de services et de la gestion de ses exigences fonctionnelles et non fonctionnelles. Nous utilisons également les termes *consommateur* et *demandeur* d'une manière interchangeable pour désigner un demandeur de services.

Annuaire de services Web. Un annuaire de services Web est un répertoire de recherches qui supporte l'enregistrement de descriptions des services Web disponibles pour faciliter leur découverte. C'est l'endroit où les fournisseurs peuvent publier et décrire la spécification technique concernant le contenu et l'interface de leurs services. Le but des annuaires est la mise en place d'une plateforme industrielle standardisée décrivant les moyens d'accès à un service à partir d'une requête portant sur son contenu ou sur son fournisseur.

Aujourd'hui, l'annuaire UDDI⁴ (Universal Description, Discovery and Integration) est considéré comme la norme de découverte de services la plus couramment utilisée. Il contient les informations sur les entreprises et les services qu'elles ont développés et publiés. Il est structuré en pages blanches (recensent des informations sur l'identité des fournisseurs), pages jaunes (comprennent les descriptions au format WSDL des services web déployés par les fournisseurs) et pages vertes (fournissent des informations techniques détaillées sur le services fournis). L'annuaire UDDI permet également aux développeurs de découvrir régulièrement des services ou des interfaces d'un certain type ou une fonction particulière. Il définit également un ensemble de "Application

4. [http ://uddi.xml.org/](http://uddi.xml.org/).

Programming Interfaces" (API) que les développeurs peuvent utiliser pour interagir avec les données UDDI directement. Cependant, UDDI présente deux grandes limites : la première est qu'il renvoie des résultats partiels et parfois peu pertinents pour une recherche par mot clé, et la seconde, est qu'il ne prend pas en considération l'aspect non fonctionnel des services telles que les valeurs de QoS, la confiance ou la réputation. Par conséquent, UDDI est essentiellement une plateforme qui prend en charge la recherche basée sur la catégorie.

Les entités décrites ci-dessus interagissent comme le montre la figure 2.1. Une fois créé, le service est **publié** dans un annuaire de services. Ce sont les fournisseurs qui se chargent de cette opération de publication en utilisant un fichier de description WSDL³ spécifique à chaque service Web. Puis un demandeur de services ayant des besoins spécifiques va **localiser** un service correspondant à ses besoins à l'aide d'un annuaire spécialisé. Enfin, une fois le service trouvé, le demandeur va **invoker** le service : une communication va être mise en place entre le demandeur et le service Web.

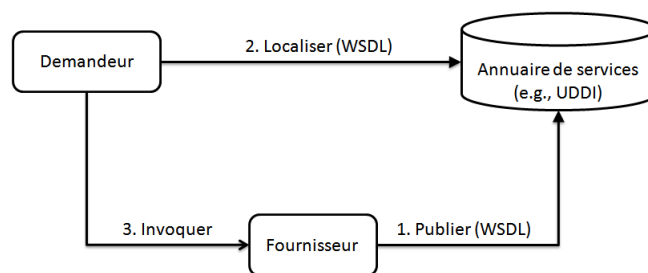


FIGURE 2.1 – Déploiement, découverte et invocation de services Web

Qualité de service (QoS). La qualité de service porte sur les exigences non fonctionnelles des services. Elle est définie comme un ensemble d'attributs perceptibles et compréhensibles avec des paramètres quantifiables décrivant le comportement du service au moment de l'exécution qui peuvent être subjectifs ou objectifs [Zhu 14]. Ces attributs doivent être :

- **Perceptibles par le demandeur** : Si le demandeur ne peut pas voir l'effet d'un paramètre mais peut le mesurer alors ce n'est pas de la qualité de service. Il y a toute une catégorie de paramètres de qualité que le demandeur ne peut pas voir dans le système, tels que les états des connexions, le passage à l'échelle etc. Ces paramètres internes appartiennent à une autre famille d'indicateurs qui évaluent la performance du système.
- **Compréhensibles par le demandeur** : Il est inutile de mentionner le risque d'une panne téléphonique estimée à 1 seul appel tous les 100000 appels à un

demandeur faisant un appel par jour ou de dire que le service sera disponible à 99,5% à une petite entreprise utilisant deux lignes téléphoniques. Cependant, dire que le service sera hors service pendant 2 jours par an a plus de signification (et pourrait être assez conséquent pour envoyer le client à un autre fournisseur indépendamment du prix).

- **Attributs subjectifs et objectifs** : Les attributs objectifs sont définis et mesurés d'une façon exacte et propre au service concerné, et peuvent être vérifiés par le demandeur. Les attributs subjectifs sont définis et mesurés par le fournisseur en fonction des opinions attribuées par les demandeurs après chaque exécution réussie. Un exemple d'une QoS objective est la disponibilité tandis qu'un exemple d'une QoS subjective est la qualité d'image d'un téléviseur. Les attributs subjectifs peuvent être approchés par des méthodes telles que l'échantillonnage des consommateurs.
- **Contrat demandeur-fournisseur** : Le contrat entre le demandeur et le fournisseur ne peut inclure que les attributs objectifs et les deux parties peuvent les mesurer.

2.1.2 Composition de services Web

La composition de services Web est considérée comme l'une des motivations les plus importantes du paradigme SOA. C'est un ensemble ordonnancé de services dont l'invocation et l'exécution fournissent un résultat qui répond à un besoin. D'une manière générale, si aucun service Web unitaire n'est capable de satisfaire les fonctionnalités requises par un demandeur, il devrait être possible de combiner les services existants pour répondre au besoin de l'utilisateur. Cette tendance sur la composition des services a suscité l'intérêt d'un nombre considérable de travaux à la fois dans le milieu académique qu'industriel. Plusieurs initiatives ont fourni des plateformes et des langages facilitant l'automatisation de la composition. En particulier, Business Process Execution Language for Web service (BPEL4WS)⁵ et DAML-S ServiceModel qui se focalisent sur la représentation des compositions de services où le flux de contrôle et les liaisons entre les services sont connus à priori. Selon les travaux effectués dans le champ de la composition de services Web (voir l'étude de [Dust 05] par exemple), on peut classer la composition selon deux principaux points de vue :

- **Approche « statique »** qui considère une composition de services comme un artefact prédéfini. Cet artefact décrit l'ordonnancement des services qui doivent

5. <http://www.ibm.com/developerworks/webservices/library/ws-spec.html>

être invoqués. Cette approche est aussi appelée approche « workflow » dans la mesure où la composition de services est vue comme un processus métier dont certaines activités sont réalisées en invoquant des services. Cette approche est plus adaptée aux applications stables dont la logique métier est complètement définie au moment de la conception et elle ne permet pas de prendre en compte les spécificités de la requête initiale. Cependant, l'utilisation de cette approche est inappropriée aux environnements à large échelle tels que le web ou le réseau social où il faut envisager des composants capables de s'adapter automatiquement aux changements imprévisibles.

- **Approche « dynamique »** dans laquelle une composition de services est un processus initié par une requête qui consiste à découvrir, sélectionner et assembler à la volée les services satisfaisant la requête. En conséquence, dans le contexte d'environnements dynamiques tels que le web ou l'informatique pervasive, où les services changent et les besoins des utilisateurs varient et ne peuvent pas tous être prévus à l'avance par un expert, la découverte des services doit se faire en temps réel. La composition de ces services, au lieu d'être préalablement fournie à travers un schéma de composition, elle doit s'effectuer de manière dynamique en fonction des besoins énoncés par l'utilisateur, des caractéristiques fonctionnelles des services découverts et des interactions résultantes de ceux-ci. L'approche dynamique répond à des besoins de flexibilité et d'adaptation. Elle permet en effet de construire une nouvelle composition adaptée à chaque requête en tenant compte des services disponibles au moment de la composition. Utilisée principalement dans le contexte des services web, elle devient difficile à mettre en œuvre si l'espace de recherche est important et si la requête initiale exprime un besoin complexe.

La composition de services Web a pour but de produire une description spécifiant une séquence d'appels à des services ainsi que la façon dont ces services sont liés entre eux. Cette opération se déroule en trois étapes :

- 1) Les services Web sont découverts à partir d'un annuaire UDDI en fonction des besoins fonctionnels définis par le demandeur de services ;
- 2) La sélection est effectuée en utilisant les propriétés non fonctionnelles comme les valeurs de QoS, la réputation ou la confiance ;
- 3) L'exécution du service composite, c'est-à-dire, l'enchaînement des appels aux services sélectionnés est créée.

Après la génération de la description, une composition de services concrète est prête pour l'exécution. L'exécution d'une composition de services est effectuée par la coordination de tous les services composants par échange de messages de sorte qu'ils soient en mesure de collaborer afin de fournir la fonctionnalité requise par le demandeur.

L'orchestration et la chorégraphie sont les deux principaux procédés de coordination pour modéliser les interactions entre les services au cours de l'exécution d'une composition de services.

- **Orchestration de Services Web.** L'orchestration de services présente la perspective du modèle de coordination centralisée. La mise en place du processus d'activité (par exemple, WS-BPEL⁶), la logique métier et l'ordre d'exécution sont contrôlés et exprimés du point de vue d'une seule partie. Le processus de gestion est exécuté par un moteur d'exécution centralisé qui coordonne une série d'invocations à tous les services composants.
- **Chorégraphie de services Web.** La chorégraphie de services se réfère à une coordination décentralisée et coopérative entre les services c'est à dire, la coordination est répartie entre tous les services composants. En conséquence, chaque service agit comme un pair et peut interagir directement avec un autre. L'exécution de la composition de services s'effectue donc comme une collaboration "peer-to-peer" par un échange de messages ordonné entre les services composants. La chorégraphie est souvent utilisée pour les applications B2B (Business to Business) et elle est supportée par le standard WS-CDL (Web Service Choreography Definition Language)⁷.

2.2 Réseaux Sociaux

Selon Garton, un réseau social peut se définir comme « *un ensemble d'individus, d'organisations ou d'entités entretenant des relations sociales fondées sur l'amitié, le travail collaboratif, l'échange d'information* » [Gart 97].

Dans cette section, nous commençons par donner les éléments de base (i.e. sommet, relation) ainsi que les notions utilisées par la théorie des graphes pour représenter un réseau social. Puis, Nous faisons un inventaire des différents types de réseaux rencontrés dans la littérature. Enfin, nous présentons les principaux domaines d'utilisation des réseaux sociaux et nous montrons leurs intérêts pour une découverte et une sélection de services basées sur la confiance.

6. <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>

7. <http://www.w3.org/TR/ws-cdl-10/>

2.2.1 Représentation d'un réseau social

La première personne à avoir représenté un réseau social est Jacob Levy Moreno au début des années 1930 [More 33]. Son objectif étant de visualiser et de cartographier les modèles d'interactions sociales. Pour ce faire, il a représenté les personnes par des points et les relations entre elles par des flèches. Ces points constituent tous les éléments d'un échange qui « agissent » alors que les flèches représentent les différentes façons dont les points « interagissent ». Cette représentation est depuis désignée par le terme *sociogramme*, mais nous parlons également de toiles en raison de leur aspect en toile d'araignée. Cette forme de visualisation, aussi peu innovante qu'elle puisse paraître de nos jours, fût un premier outil d'identification rapide des caractéristiques d'un réseau social. Moreno a ainsi introduit le concept d'étoile pour désigner les personnes ayant le plus de relations dans un réseau social, en référence à l'étoile formée par un point et ses connections. Les mathématiciens ont fait le rapprochement entre les représentations sociogrammes et la théorie des graphes.

[Cart 77] sont les premiers à avoir appliqué la théorie des graphes à l'analyse des réseaux sociaux. Le graphe est devenu par la suite la représentation adoptée par toutes les sciences manipulant l'analyse des réseaux sociaux dont la sociologie, les mathématiques et l'informatique.

Dans ce qui suit, nous décrivons en détail les principaux éléments d'un réseau social et les notions utilisées par la théorie des graphes pour les réseaux sociaux.

Sommets. Les sommets, aussi appelés nœuds, agents, entités ou acteurs peuvent représenter plusieurs objets. Souvent, ils représentent des personnes ou des structures sociales telles que les groupes de travail, les équipes, les organisations, les institutions, les états, ou même les pays. Souvent, ils correspondent aux blocs de construction des plateformes de médias sociaux tels que les pages Web, les individus dans les sites de réseaux sociaux ou les auteurs dans les blogs. Les sommets peuvent être caractérisé par un ensemble d'attributs qualitatifs ou quantitatifs. Ces attributs peuvent décrire les informations personnelles d'un individu (âge, sexe, profession) ou d'autres caractéristiques telles que les coordonnées géographiques d'une ville. Dans les outils de visualisation de réseaux sociaux tels que NodeXL⁸ et Gephi⁹, les modalités des attributs peuvent être associés à des propriétés visuelles telles que la taille, la couleur ou l'opacité des sommets. Cette association est très utile pour la compréhension et l'interprétation visuelle (voir Fig. 2.2) des résultats obtenus suite à l'application des mesures d'analyse des réseaux sociaux (ex. degré, détection de communautés, ...). Par exemple dans un

8. <http://nodexlgraphgallery.org/Pages/Default.aspx>

9. <http://gephi.github.io/>

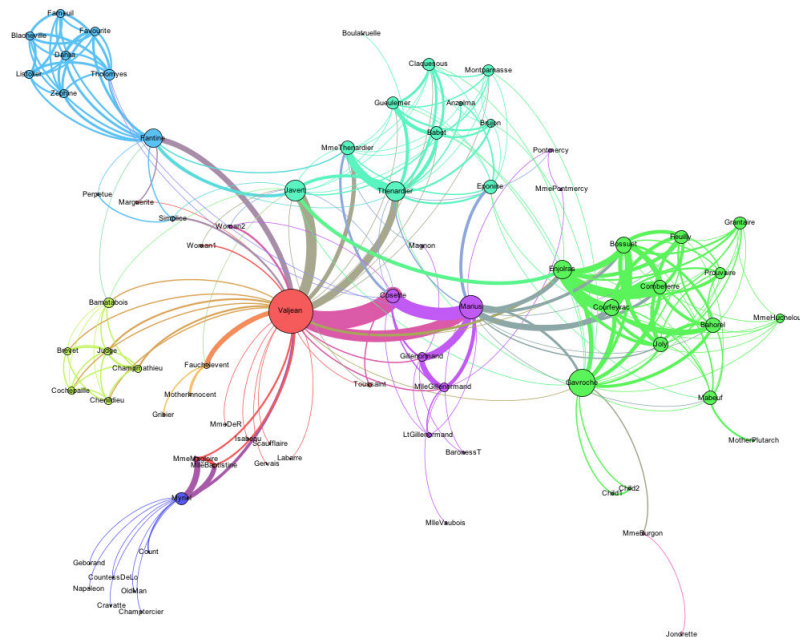


FIGURE 2.2 – Visualisation du réseau des personnages du roman « les Misérables » à l'aide de Gephi : la taille des sommets est proportionnelle à leur degrés et la couleur indique l'appartenance des sommets aux différentes communautés générées par un algorithme de partitionnement

contexte d'entreprise, cela nous permet de découvrir les rôles cachés des personnes (par exemple : qui détient l'information, qui est le responsable, qui est l'expert) et les interactions émergentes (par exemple : qui collabore avec qui).

Arêtes. Les arêtes, aussi connues sous le nom de liens, d'arcs, de connexions ou de relations sont les blocs de construction de réseaux reliant les sommets ensemble. Les arêtes peuvent représenter différents types de relations comme la proximité, la collaboration, la parenté, l'amitié, le partenariat, les citations, les investissements et les transactions. Les arêtes peuvent être orientées ou non orientées. Les arêtes **orientées** (également appelées arcs ou arêtes asymétriques) possèdent une origine et une destination : un utilisateur de Twitter suit un autre utilisateur ou un email envoyé à un destinataire ou une page Web qui réfère une autre. Une arête **non orientée** (appelée aussi arête symétrique) représente une relation mutuelle et réciproque entre deux sommets : un couple est marié, deux utilisateurs de Facebook sont des amis ou deux individus sont membres de la même organisation. Les arêtes peuvent être annotées de différentes façons. Une arête est **étiquetée** lorsqu'on lui attribue un label indiquant le type de la relation (ex. amitié, famille, partenariat,...). Une arête est **pondérée** lorsqu'on lui attribue un

poids indiquant la force ou la fréquence de la relation. Par exemple, une arête pondérée entre deux utilisateurs de Facebook peut indiquer le nombre de commentaires sur les photos échangées ou la durée d'une amitié.

2.2.2 Types de réseaux sociaux

Les réseaux sociaux varient en taille à partir de quelques individus (e.g. réseau d'entreprise) au centaines de milliers voir des millions d'individus (e.g. Facebook). Ils diffèrent aussi dans les types de sommets qu'ils comprennent, de la nature des arêtes qui les relient, et les façons dont ils sont formés. Dans cette section, nous présentons quelques distinctions identifiées dans la littérature pour décrire les différents types de réseaux. Ces distinctions affectent les métriques utilisées dans l'analyse ainsi que leur interprétation.

Réseaux complet, partiel et égocentrique

Il est souvent utile de considérer les réseaux sociaux d'un point de vue centré utilisateur. Dans la littérature [McCa 02], le sommet, centre d'intérêt, est appelé « *ego* ». Cependant, les sommets auxquels elle/il est connecté(e) sont appelés « *alters* ». Les « *alters* » peuvent être des membres de la famille, des amis, des collègues de travail/école, etc. Certains réseaux égocentriques ne comprennent que les individus qui sont connectés à un « *ego* » spécifique. Par exemple, un réseau formé seulement par mes amis Facebook serait un réseau égocentrique parce que par définition je suis relié à tous les sommets comme illustré par la figure 2.3. Autres réseaux égocentriques et leurs « sous-graphes » associés peuvent s'étendre à partir d'un ego atteignant non seulement les amis, mais aussi les amis d'amis. Plus généralement, les réseaux égocentriques peuvent s'étendre sur un certain nombre de degrés de « *ego* ». Le réseau égocentrique *1-degré* est le réseau de base qui se compose d'un « *ego* » et de ses « *alters* ». Le réseau égocentrique *1.5-degré* étend le réseau 1-degré en incluant les liens entre tous les « *alters* ». Par exemple, un réseau Facebook 1.5-degré caractériserait qui de vos amis se connaissent. Le réseau égocentrique *2-degrés* étend le réseau 1.5-degré en incluant tous les propres « *alters* » des « *alters* » (c'est à dire, les amis d'amis), dont certains ne sont pas connectés à « *ego* ». Ces trois réseaux égocentriques nous permet d'élargir l'espace d'analyse mais toujours dans un voisinage proche d'un individu particulier dans un réseau social. Les réseaux de degré supérieur existent, mais il sont utilisés rarement dans la pratique car ils peuvent rapidement devenir insolubles.

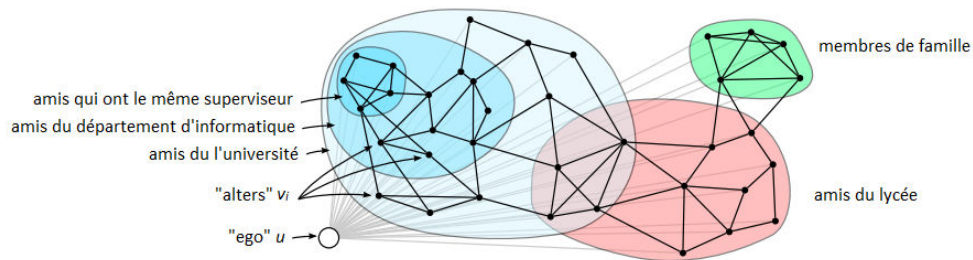


FIGURE 2.3 – Un réseau égocentrique avec des sommets étiquetés [McAu 12]

Un réseau *complet* contient tous les sommets d'intérêts et les arêtes entre eux. Tous les « ego » sont traités de manière égale. Un réseau complet est souvent créé lorsqu'un seul système comme une plateforme sociale agit comme un hub pour un ensemble d'individus/groupes connectés. Par exemple, le réseau Twitter inclut tous les utilisateurs et les liens entre eux. Dans la pratique, il n'est pas toujours possible (ou particulièrement perspicace) d'analyser un réseau complet. Au lieu de cela, les analystes créent un réseau partiel en sélectionnant un échantillon ou une partie du réseau. Le réseau *partiel* n'est pas égocentrique mais plutôt centré sur une thématique ou un sujet particulier. Autres types de réseaux partiels sont créés pour inclure un sous-groupe d'utilisateurs (par exemple, tous les participants à la conférence), les individus et leurs liens dans un intervalle de temps spécifié ou les personnes qui ont certaines caractéristiques (par exemple, les PDG des sociétés de CAC 40).

Réseaux unimodal, multimodal et d'affiliation

Jusqu'à présent, nous n'avons considéré que les réseaux qui relient des entités d'une seule communauté. Ces réseaux standards sont appelés réseaux *unimodaux* parce qu'ils comprennent un seul type (c'est à dire, mode) de sommet. Ils connectent les utilisateurs aux utilisateurs ou les documents aux documents, mais ils ne comprennent pas à la fois les utilisateurs et les documents. Cependant, les réseaux peuvent être hétérogènes en incluant différents types de sommets donnant naissance aux réseaux *multimodaux*. Par exemple, un réseau peut lier les utilisateurs aux forums de discussion et aux blog qu'ils ont commenté. Chaque sommet du graphe représente un utilisateur, un forum ou un article de blog que nous pouvons distinguer visuellement par différentes couleurs ou formes. YouTube peut être considéré comme un réseau multimodal dont les sommets représentent les vidéos, les utilisateurs et les mots clés. Dans de nombreux cas, ces réseaux multimodaux doivent être transformés en réseaux unimodaux simples pour effectuer une analyse de réseau significative, comme la plupart des mesures sont conçues

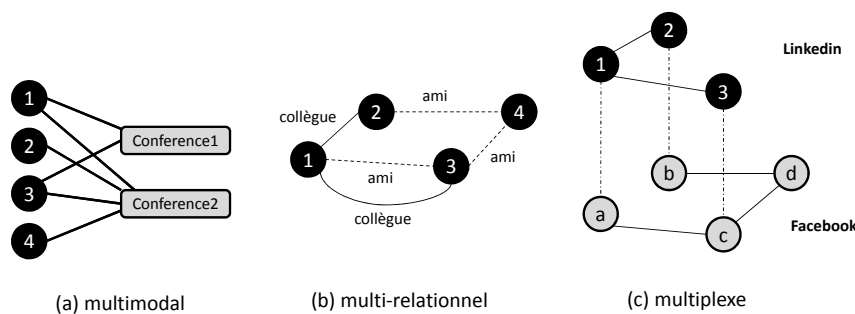


FIGURE 2.4 – Trois exemples de réseaux hétérogènes : un graphe multimodal d’auteur-conférence (a), un réseau multi-relationnel (b) et un réseau multiplexe (c)

pour les réseaux unimodaux.

Le réseau multimodal le plus utilisé est le réseau bimodal c’est à dire, contenant deux types de sommets. Les données de ces réseaux incluent souvent des individus et un événement, une activité ou un contenu dont ils sont affiliés, créant un réseau d’affiliation. Par exemple, un réseau d’affiliation peut lier les auteurs aux conférences. Chaque sommet dans le graphe représente un auteur ou une conférence ce qui pourrait être distingué visuellement par différentes couleurs ou formes (voir Fig. 2.4(a)). Dans ce réseau, les utilisateurs ne sont pas connectés directement les uns aux autres. De même, les conférences ne sont pas connectées directement les unes aux autres. Les réseaux d’affiliation sont exploités par de nombreux systèmes de recommandations qui recommandent les articles d’intérêt, tels que la fonction d’Amazon » *les clients ayant acheté cet article ont également acheté ces articles* « .

Réseaux multiplexes et multi-relationnel

Bien qu’il soit commun pour deux individus d’être connectés de différentes manières (par exemple, en échangeant des appels téléphoniques, des emails, travaillant

dans la même entreprise ou appartenant à la même famille), la plupart des réseaux considèrent un seul type de lien ou de relation. Cependant, il est possible d'envisager des réseaux avec plusieurs types de relations. Dans la littérature des réseaux sociaux [Gome 12, Li 14b], on peut représenter chaque type de relation entre les individus séparément dans une dimension ou une couche. Ce réseau social appelé réseau *multiplexe* est donc caractérisé par la superposition de ces couches toutes définies sur le même ensemble de nœuds. La multiplexité joue un rôle important dans l'organisation des réseaux de grande taille. Par exemple, l'existence de différents types de relations entre les individus explique le chevauchement des structures communautaires observées dans les réseaux sociaux où les nœuds peuvent appartenir à plusieurs communautés associées chacune à un type d'interaction différent. Les réseaux sociaux émergents de différents types de médias sociaux ou de sites des réseaux sociaux sont de bons exemples de réseaux multiplexes. Comme il ressort de la figure 2.4(c), les constituants principaux de ce réseau sont deux couches qui ne diffèrent pas des réseaux traditionnels et les correspondances indiquent quels nœuds dans les différentes couches correspondent aux mêmes individus. Il existe une autre manière pour représenter des réseaux avec plusieurs types de relations appelés réseaux *multi-relationnel* [Szel 10]. Contrairement aux réseaux multiplexes qui se rapportent à des graphes multi-couches définis sur le même ensemble de nœuds où chaque couche représente une relation différente, dans les réseaux multi-relationnel il n'existe qu'une seule couche donc un seul graphe mais qui contient les différents types de relations. Par exemple, le réseau que montre la figure 2.4(b) comprend deux types de relations, amitié et collègue. Un réseau social multi-relationnel (RSMR) est un graphe, où les nœuds représentent des agents et les arêtes entre deux nœuds indiquent des relations sociales symétriques entre les deux agents.

2.2.3 Domaines d'utilisation des réseaux sociaux

Les réseaux sociaux jouent un rôle important dans de nombreux domaines liés aussi bien à l'individu qu'à l'organisation, en particulier dans la recherche d'informations, les comportements de prise de décision ou encore la promotion sociale et économique. De ce fait, l'étude de leur utilisation a été rapportée dans différents domaines d'application tels que *les sciences sociales, l'intelligence artificielle et les sciences de l'entreprise*.

- Dans le domaine des sciences sociales, Jason Ethier a mentionné dans une étude sur l'analyse des réseaux sociaux moderne¹⁰ que « *les réseaux sociaux nous aident à mieux comprendre comment et pourquoi nous interagissons les uns avec les*

10. <http://www.scribd.com/doc/11171859/Current-Research-in-Social-Network-Theory#scribd>

autres, ainsi que la façon dont la technologie peut modifier ces interactions ». Raab et Milward ont classé les réseaux sociaux en deux catégories, les réseaux lumineux et les réseaux sombres [Raab 03]. Les réseaux lumineux s'inscrivent dans un contexte où les résultats de la socialisation s'avèrent bénéfiques pour les individus, les groupes, les entreprises et la société en général. Au contraire, les réseaux sombres (ex., les réseaux terroristes comme Al-Qaïda, les réseaux de trafic de drogues, et les réseaux de contrebande d'armes ou de diamant) se produisent dans un cadre illégal et dont les résultats de la socialisation sont nuisibles pour les individus, les groupes, et le bien-être social. La recherche révèle que les réseaux sociaux opèrent à plusieurs niveaux : du cadre familial jusqu'au niveau des nations. Ils jouent un rôle important dans la résolution de problèmes et dans la gestion d'organisations et déterminent comment les individus réussissent à atteindre leurs objectifs [Mood 03, Jack 96].

- Dans le domaine de l'intelligence artificielle distribuée, les réseaux sociaux permettent la spécification de la coordination, la coopération et les mécanismes de négociation entre agents. Castelfranchi note qu'un agent au sein d'une organisation peut être aidé ou affaibli, favorisé ou menacé et il peut rivaliser ou coopérer [Cast 95]. Un agent n'est pas une entité indépendante, mais il évolue dans une société et interagit avec des agents de sa propre société ou appartenant à d'autres sociétés. Pour évaluer la réputation d'un agent dans une société, des travaux comme ceux de Sierra et Debenham et Pujol et al. s'appuient sur l'analyse de la topologie des réseaux sociaux et les positions que l'agent détient dans ces réseaux au fil du temps [Pujo 02, Sier 09].
- Dans le domaine des sciences de l'entreprise, le rapport DEMOS¹¹ a encouragé les entreprises à autoriser leurs employés à utiliser des applications sociales pour mettre en place et consolider leurs réseaux de contacts (i.e., les collègues, les clients et les fournisseurs). Cela devrait avoir un impact positif sur la productivité, le développement des affaires et la collégialité sous réserve d'une utilisation correcte de ces applications. Les organisations peuvent également capitaliser sur les applications sociales pour attirer de nouveaux clients, identifier les fournisseurs appropriés et être conscients des concurrents [Badr 10, Badr 11]. McDonald et Ackerman ont utilisé les réseaux sociaux pour recommander des profils sur la base de l'expertise offerte et requise pour une éventuelle collaboration [McDo 00, Mcdo 03]. Une autre utilisation des réseaux sociaux dans le domaine de l'entreprise a été soulevée par [Zhan 07, Fone 96, Zhan 05, Yu 03] pour la recherche d'expertise. Ces travaux tentent de tirer profit du réseau social au

11. <http://news.bbc.co.uk/2/hi/business/7695716.stm>

sein d'une organisation pour aider les individus à trouver des experts appropriés capables de résoudre un problème spécifique.

Toutes ces formes d'utilisations montrent l'intérêt des réseaux sociaux pour la découverte, sélection et composition des services Web comme moyen de promotion, d'échange et de partage.

Des travaux récents ont combiné les réseaux sociaux avec les services Web pour pallier les limites inhérentes aux annuaires UDDI au niveau de la publication centralisée des services. Par exemple, Maaradji et al. ont proposé un compositeur sociale qui repose sur les interactions entre les utilisateurs et les services Web pour déterminer les prochaines actions à prendre [Maar 10]. Xie et al. ont proposé une plateforme pour une composition sémantique de services basée sur les réseaux sociaux [Xie 08]. La confiance entre les fournisseurs de services, les consommateurs de services et les services eux-mêmes est l'élément social pris en compte dans cette composition. Maamar et al. [Maam 10] ont utilisé la représentation d'un réseau social pour organiser les interactions entre les services Web afin de rendre le processus de découverte de service plus efficace.

2.3 Systèmes multi-agents

Dans cette section, nous allons nous intéresser dans un premier temps au paradigme agent comme étant une entité autonome qui peut s'associer à un service Web en le dotant de ses capacités. Puis, nous allons donner un aperçu des systèmes multi-agents dans un contexte de composition de service Web basée sur la confiance et les réseaux sociaux.

L'évolution des environnements orientés services a suscité de nouveaux défis pour la communauté « services Web » sur la manière de considérer la grande quantité d'information, les interactions complexes et l'hétérogénéité. Plusieurs recherches [Amor 03, Hend 01, Huhn 02, Huhn 03, Syca 01, Sree 04, Maxi 05b] ont étudié l'avantage de l'introduction du paradigme agent dans l'architecture orientée services afin de doter les services Web de nouvelles capacités comme la perception, la cognition, l'interaction, la sociabilité et la prise de décision. Hendler a prouvé dans [Hend 01] que l'état actuel des services Web a atteint ses limites. Malgré les efforts déployés au niveau de la publication en introduisant l'aspect sémantique dans la description des attributs de QoS, les moteurs de recherche échouent souvent à trouver le service Web approprié. Pour combler ces lacunes, l'auteur propose d'intégrer le paradigme agent dans l'architecture du services Web. [Syca 01] a proposé un autre modèle qui combine l'agent avec le service

Web. Dans ce modèle, Sycara exploite des propriétés de l'agent telles que l'interaction et la gestion de la confiance dans le développement de la technologie des services Web pour améliorer la découverte de services. Elle présente une analyse des différents types d'agents médiateurs, les infrastructures des systèmes multi-agents et leur application pour les services Web.

Plus tard, Huhns a introduit dans [Huhn 02, Huhn 03] une nouvelle vision concernant la mise en œuvre des services Web dans laquelle l'architecture orientée objet est substituée par une architecture orientée agent. Grâce à cette encapsulation c'est à dire, envelopper chaque service Web dans un agent logiciel, les services Web acquièrent le comportement agent comme l'autonomie, la flexibilité et l'adaptation et montrent la volonté de devenir interactif. Plus tard, Maximilien et Singh ont considéré les services Web comme des agents afin d'augmenter leurs styles d'interaction à un niveau plus élevé qui inclut des interactions entre les agents consommateurs et les agents fournisseurs. De plus, les agents peuvent assurer la médiation, la négociation, la communication et la collaboration entre les parties concernées, à savoir, les services Web et les consommateurs, en utilisant des interactions de confiance [Maxi 05b]. Selon ces travaux, l'utilisation des agents améliore le comportement des services Web de plusieurs façons :

- Un service Web est un composant isolé, passif qui a seulement connaissance de lui-même, mais pas de ses consommateurs ou d'autres services (par exemple, les services avec lesquels il a interagi dans des compositions antérieures). Cependant, les agents sont conscients de leurs propres états et aussi des autres agents et de leurs capacités. Cette prise de conscience est possible grâce aux interactions et aux échanges d'informations stockées dans des modèles de croyances. Sans cette prise de conscience, un service Web est incapable de tirer parti de ses anciennes interactions, des nouvelles caractéristiques de son environnement et il ne peut pas personnaliser et adapter son service en fournissant par exemple, des services améliorés aux consommateurs réguliers.
- Les services Web à la différence des agents ne sont pas conçus pour utiliser et réconcilier les ontologies. Si le consommateur et le fournisseur utilisent des ontologies différentes, alors le résultat d'invocation d'un service peut être incompréhensible pour le consommateur. Cependant, les agents peuvent résoudre de telles hétérogénéités.
- Modéliser des capacités de collaboration au niveau des services nécessite qu'ils soient dotés de propriétés d'autonomie. Cette propriété, qui fait défaut aux services Web, est une caractéristique intrinsèque à l'agent qui se réfère généralement à l'autonomie sociale où l'agent est conscient de ses accointances, mais exerce

néanmoins son indépendance dans la mesure où il possède ses propres buts et motivations.

- Les agents sont des entités sociales, la coopération entre eux se fait à travers des mécanismes sociaux comme la confiance, la motivation ou l'attraction et se manifeste à travers plusieurs modèles organisationnels comme les teamworks, les coalitions, les hiérarchies et les sociétés [Horl 05] proposant des solutions flexibles, plus complets et de meilleure qualité. Cependant, les standards actuels de services Web fournissent un schémas statiques pour composer les fonctionnalités.

Par analogie avec les sciences sociales, le terme de société est utilisé pour désigner un ensemble d'agents constituant un système multi-agents. C'est ici que réside pour nous le principal intérêt du paradigme multi-agents : ne pas concevoir uniquement des entités intelligentes, mais aussi mettre ces entités en relation de manière « intelligente ». Cette approche transparaît également à travers trois autres concepts communément employés dans les SMAs : *les interactions, l'organisation et l'environnement*. Demazeau a défini le système multi-agents selon cinq aspects : l'agent, l'environnement, l'interaction, l'organisation et l'utilisateur [Dema 95]. L'environnement est un support logique et physique qui permet aux agents d'agir, d'interagir, et de s'organiser. Un environnement est ainsi un médium d'interaction (messages, signaux, traces,...), un lieu où les actions individuelles ou collectives sont réalisées, où des réactions sont perçues, un moyen de structuration des agents (relation de proximité, topologie spatiale ou temporelle,...), une source de données pour le système et un espace où les ressources sont disponibles. Dans cette thèse, le réseau social du demandeur de services représente l'environnement de notre système multi-agents. Le terme interaction désigne l'enchaînement d'échanges d'informations ou d'influences ayant lieu entre les agents. La mise en place de ces interactions permet aux agents de résoudre un problème grâce à leur capacités respectives. L'interaction est l'un des aspects les plus importants de l'intelligence artificielle distribuée dans la mesure où elle permet de relier des agents pour qu'ils constituent un système. Dans un contexte social comme le notre, les interactions entre les agents sont entraînées par des mécanismes sociaux comme la confiance et la réputation. Mais ce concept ne permet de mettre en relation qu'un sous-ensemble des agents composant le système (i.e., ceux qui interagissent). Il est alors nécessaire d'utiliser un autre concept plus général pour exprimer les relations liant les membres d'un groupe d'agents : l'organisation. Ce concept a pour objectif principal de contrôler et de coordonner l'ensemble des interactions présentes entre les agents de la société, afin de structurer les activités des agents et permettre ainsi au système de se comporter comme un tout efficace et cohérent, capable de conduire à une solution à un problème. C'est le cas d'une composition de services dans laquelle les agents s'organisent dans

un groupe (teamwork, coalition, société, hiérarchie) pour satisfaire la requête complexe d'un demandeur de services.

Pour résumer, le fondement du paradigme multi-agents est évidemment le concept d'agents qui définit les caractéristiques des composants de base d'un système. Les agents, qu'ils soient coopératifs ou compétitifs, sont capables de réaliser des tâches complexes et supportent différentes formes d'interactions y compris la négociation et la coordination. Par ailleurs, ils disposent de capacités bien développées pour raisonner sur les données sémantiques et proposent des solutions flexibles qui s'adaptent aux besoins et aux préférences de leurs utilisateurs. En conséquence, les systèmes multi-agents semblent adaptés à la modélisation des problématiques (de distribution, d'adaptation et de collaboration) sous-tendues par la découverte, la sélection et la composition de services dans les réseaux sociaux. Les travaux que nous présentons dans cette thèse s'inscrivent dans le cadre des systèmes multi-agents et des services Web. Tout au long de cette thèse, nous allons décrire les interactions, l'organisation et l'environnement de notre système multi-agents dans le but de résoudre les problématiques de découverte, de sélection et de composition de services.

2.4 Terminologie et concepts

Dans cette section, nous définissons les principaux concepts utilisés dans cette thèse, c'est à dire, les réseaux sociaux, les services et la requête du demandeur de services.

Réseaux Sociaux. Nous considérons le réseau social multi-relationnel (RSMR) qui prend en considération la sémantique de la relation reliant deux nœuds. Les relations peuvent être de différents types. Par exemple, si l'on considère deux types, R_1 peut être une relation d'amitié et R_2 peut être une relation de partenariat. Plus formellement, un réseau social multi-relationnel est défini comme suit :

Definition 2.1 (*Réseau social multi-relationnel*) Soit $V = \{a_1, a_2, \dots, a_s\}$ un ensemble d'agents et soit $R = \{R_1, R_2, \dots, R_r\}$ un ensemble de types de relations symétriques défini sur V , un réseau social multi-relationnel (RSMR) est un graphe connecté non orienté $G = \langle V, E_1, E_2, \dots, E_r \rangle$ où $E_i \subseteq V \times V \forall i \in \{1, \dots, r\}$ est l'ensemble des arêtes par rapport à la i -ème relation, i.e., une arête $(a_k, a_j) \in E_i$ représente une relation sociale de type R_i entre a_k et a_j et $\forall i \neq i', E_i \cap E_{i'} = \emptyset$.

Dans un RSMR, la notion de voisinage d'un agent peut être exprimée comme suit :

Definition 2.2 (*Voisinage*) Étant donné un graphe $G = \langle V, E_1, E_2, \dots, E_r \rangle$ qui représente un RSMR, le voisinage d'un agent a_k par rapport à un type de relation $R_i \in \{R_1, R_2, \dots, R_r\}$, noté $N_{R_i}(a_k)$, est défini par $N_{R_i}(a_k) = \{a_j \in V \mid (a_k, a_j) \in E_i\}$.

Par exemple, le voisinage de l'agent a_0 dans la figure 1.1 du chapitre 1 selon la relation "partenariat" est $N_{\text{partenariat}}(a_0) = \{a_2, a_5, a_7\}$. Dans un RSMR, chaque agent a_k est doté d'une vue locale, il interagit avec un sous-ensemble d'agents noté SA_k représentant ses accointances sociales : $SA_k = \bigcup_{R_i \in R} N_{R_i}(a_k)$. Par exemple, dans la figure 1.1, $SA_0 = \{a_2, a_3, a_4, a_5, a_6, a_7\}$ est l'ensemble des accointances sociales de l'agent a_0 .

Services. Un service est décrit en termes d'une fonctionnalité, des entrées, des sorties et des valeurs d'attributs de QoS.

Definition 2.3 (*Service*) Un service s est un n -uplet $\{in, out, f, q^1, \dots, q^d\}$ où in est un ensemble d'entrées requises pour utiliser le service, out est un ensemble de sorties prévues à la fin de l'exécution du service, f est une interface sémantique décrivant la fonctionnalité offerte, et $q^1, \dots, q^d$ sont les valeurs des d attributs de qualité de service.

Par exemple, considérons le service d'achat de billet d'avion offert par l'agent a_4 dans la figure 1.1. Ce service est décrit comme suit : $in = \{departure, arrival, check_in, check_out\}$, $out = \{online_invoice, price\}$, $f = \{Flight_Booking\}$ et $\{q^1 = \{availability = 0.8\}, q^2 = \{Reliability = 0.9\}\}$.

Requête de l'utilisateur. Un utilisateur communique ses besoins dans une requête en exprimant ses préférences sur les fonctionnalités demandés.

Definition 2.4 (*Requête*) Une requête Q est un 5-uplet $(F, U, \alpha, \beta, \mu)$ où $F = \{f_1, f_2, \dots, f_n\}$ est l'ensemble des fonctionnalités requises, $U : R \mapsto]0, 1]$ est une fonction d'utilité exprimant les préférences de l'utilisateur pour les types de relations, $\alpha \in [0, 1]$ est le seuil de la crédibilité sociale, $\beta \in [0, 1]$ est le seuil de confiance en la recommandation et, $\mu \in [0, 1]$ est le seuil de confiance en l'expertise.

Pour illustrer la requête de l'utilisateur, considérons le scénario de motivation décrit dans la section 1. Supposons qu'Alice, le demandeur de services, réside à Paris et a l'intention d'assister à une conférence à Berlin se déroulant entre le 5 mai et le

8 mai. Elle requiert un transport pour la ville de destination, un hébergement et un service de garde d'enfants. Dans ce cas, $F = \{f_1 = \textit{transport}, f_2 = \textit{hebergement}, f_3 = \textit{garde_d'enfant}\}$. Pour la garde d'enfants, Alice préfère confier son enfant à quelqu'un de la famille plus qu'à un partenaire et encore moins à une connaissance publique. De même, elle préfère voyager avec un membre de la famille plutôt qu'avec un partenaire et encore moins par le transport public. Ainsi, ses préférences pour les types de relations sont exprimées comme suit : $R_{famille} \succ R_{parenariat} \succ R_{publique}$. Ces préférences sont modélisés dans la requête par la fonction d'utilité tel que : $U(R_{famille}) = a$, $U(R_{parenariat}) = b$ et $U(R_{publique}) = c$ avec $a \geq b \geq c$. Par exemple, si Alice préfère la première relation deux fois plus que la seconde alors l'utilité de la première relation sera le double de la seconde (i.e., $a=2b$). Ceci permet de favoriser dans le graphe les chemins composés d'un nombre minimal de types de relations différentes compte tenu des préférences du demandeur de services. Si nous considérons $a = 1$, alors les préférences d'Alice pour les types de relation deviendront $U(R_{famille}) = 1$, $U(R_{parenariat}) = \frac{1}{2}$ et $U(R_{publique}) = \frac{1}{4}$. Au niveau de la confiance, Alice exige que les agents participants ont au moins une crédibilité sociale égale à 0.7 et que la qualité de leurs recommandations est supérieure à 0.6. Pour la confiance des services, Alice préfère ceux fournis par des fournisseurs dont l'expertise est au dessus de 0.65. Dans la requête, ces préférences seront modélisées comme suit : $\alpha = 0.7$, $\beta = 0.65$ et $\mu = 0.6$.

2.5 Conclusion

Nous avons décrit dans ce chapitre les trois notions utilisées dans cette thèse à savoir, l'architecture orienté services, les réseaux sociaux et les systèmes multi-agents. Dans la première section, nous avons présenté les entités d'interaction dans une architecture orientée service puis, nous avons décrit la notion de composition de services Web avec ses deux principaux types d'approches. Dans la deuxième section, nous nous sommes focalisés sur la notion de réseau social. À cet égard, nous avons présenté ses éléments de base (i.e. sommet et relation), ses différents types de représentation et ses principaux domaines d'application. Cela nous a permis de montrer l'intérêt des réseaux sociaux pour la découverte, la sélection et la composition de service. Nous avons donné dans la troisième section un aperçu sur les agents et les systèmes multi-agents en précisant l'avantage de l'introduction du paradigme agent dans l'architecture orientée services. Enfin, nous avons présenté dans la dernière section la terminologie utilisée et les concepts fondamentaux nécessaires au positionnement de nos travaux et à la compréhension de la thèse.

Chapitre 3

Découverte et sélection de services Web : État de l'art

Sommaire

3.1	Catégorisation des approches existantes	32
3.2	Approches basées sur la réputation	33
3.3	Approches basées sur la recommandation	36
3.3.1	Recommandation basée sur le filtrage par le contenu	36
3.3.2	Recommandation basée sur le filtrage collaboratif	37
3.3.3	Recommandation hybride	39
3.4	Réseaux Pair à Pair (P2P)	40
3.5	Systèmes de références	41
3.6	Conclusion	44

Les problèmes de découverte et de sélection de services ont été étudiés dans plusieurs domaines de recherche et apparaissent sous différentes formulations qui ne posent pas nécessairement les mêmes contraintes, les mêmes hypothèses et ne nécessitent pas les mêmes techniques et méthodes de résolution. Nous présentons dans ce chapitre une revue de littérature sur la problématique de la découverte et de la sélection sociales de services. Cette étude bibliographique est effectuée à travers une catégorisation distinguant quatre catégories d'approches à savoir les approches basées sur la réputation, les approches basées sur la recommandation, les réseaux pair à pair et les systèmes de références. Nous allons examiner chacune de ces catégories en décrivant les principaux travaux existants et en précisant à chaque fois leurs avantages et leurs limites.

L'objectif de ce chapitre est de bien définir le cadre de recherche ainsi que le domaine d'application de cette thèse afin de mettre en valeur notre contribution.

3.1 Catégorisation des approches existantes

Les techniques de découverte et de sélection de services traitent la façon dont les gens prennent des décisions dans la vie de tous les jours lorsque les informations concrètes concernant les services désirés ne sont pas disponibles. Dans de tels cas, les utilisateurs s'appuient sur l'aide de leurs amis, les amis-des-amis, les experts, les leaders et les opinions publiées. Étant donné la nature de l'environnement du SOC, les approches sociales peuvent être utilisées dans la sélection des services Web [Bill 09]. La dimension sociale est appliquée lorsque les gens achètent un produit ou un service à partir d'un site Web, comme Amazon. Par exemple, lors de la recherche d'un livre, un utilisateur peut bénéficier des notes attribuées par d'autres utilisateurs ou peut recevoir des recommandations. Comme Sreenath et Singh [Sree 04] et Singh et Huhns [Sing 05], nous distinguons quatre catégories d'approches sociales de découverte et de sélection de services :

- 1) *Les approches basées sur la réputation* regroupent les notes attribuées par les consommateurs suite à l'utilisation d'un service au niveau d'une autorité centrale. Ces notes sont agrégées à l'aide d'une fonction d'agrégation en une seule valeur qui estime soit la fiabilité des fournisseurs soit, la qualité de leurs services offerts. Cette valeur qui représente la réputation influence la décision des autres consommateurs lors de la sélection de services.
- 2) *Les approches basées sur la recommandation* sont utilisées généralement pour prédire des besoins personnalisés d'un utilisateur en se basant sur la description de son profil et/ou les relations qu'il a avec d'autres utilisateurs ayant des comportements ou des profils similaires. Cela permet d'aider les utilisateurs qui n'ont pas suffisamment d'expérience ou la compétence nécessaire pour évaluer le nombre, potentiellement important, d'alternatives offertes par un site par exemple.
- 3) *Réseaux Pair à Pair (P2P)* distribuent le traitement et assurent l'autonomie des utilisateurs. Cela facilite le partage et l'échange de ressources dans un environnement où la masse de données disponibles est importante, tel est le cas de l'environnement orienté services. Cela permet de propager la recherche dans le réseau afin de répondre au mieux aux besoins de l'utilisateur.

- 4) *Systèmes de références* captent la manière dont les utilisateurs s'entraident les uns les autres pour trouver des références dignes de confiance. Cette dimension sociale est une motivation importante pour les systèmes de références. Cela permet de prendre en considération la confiance des services ou des fournisseurs de services ce qui améliore le résultat de la découverte et de la sélection de services par la suite.

Dans ce chapitre, nous n'allons pas effectuer une revue exhaustive de la littérature traitant la problématique de découverte et de sélection de services, mais nous allons plutôt présenter les approches les plus pertinentes compte tenu de notre catégorisation. Au cours de la description de chaque catégorie, nous essayons de mettre en valeur ses avantages et dégager ses points faibles.

3.2 Approches basées sur la réputation

Parmi les approches existantes relatives à la découverte et à la sélection sociale, une approche dominante consiste à utiliser les systèmes de réputation. La définition de base de la réputation est celle énoncée par Sabater et Sierra [Saba 02] *"la réputation est une opinion ou un point de vue de quelqu'un à propos de quelque chose"*. Aberer a défini dans [Aber 01] la réputation comme *"une mesure dérivée de la connaissance directe ou indirecte issues des interactions antérieures entre les agents et est utilisée pour évaluer le niveau de confiance qu'un agent a en un autre agent"*. Selon [Abdu 00], *"la réputation est une estimation qui prédit le comportement futur d'un agent basée sur des opinions ou des observations de son comportement antérieur"*. Chaque opinion représente une impression subjective reflétant la façon dont l'agent consommateur apprécie/regrette l'interaction avec l'agent fournisseur. Ces impressions individuelles sont les briques qui, adéquatement combinées, peuvent être utilisées pour construire la réputation d'autres agents.

Dans les systèmes de réputation centralisés, une seule autorité centrale recueille les informations sur les utilisateurs du système ou de la communauté en se basant sur les notes des consommateurs. L'autorité centrale établit pour chaque agent un seul score de réputation et met l'ensemble des scores calculés à la disposition du public [Pujo 02]. La réputation est donc une mesure sociale globale et partagée qui résulte d'une évaluation collective (c'est à dire, à partir de plusieurs sources d'information) et qui est perçue de la même façon par tous les utilisateurs du système. Contrairement à la confiance qui est une mesure sociale locale résultant d'un raisonnement individuel et personnalisé qui dépend des croyances de l'agent évaluateur. Dans ce cas, les agents peuvent avoir

des opinions complètement différentes sur un même agent. Les travaux de recherche menés autour de la découverte et la sélection de services Web utilisant la confiance seront détaillés dans le chapitre 4.

En commerce électronique, la réputation d'un utilisateur est le résultat de l'agrégation de toutes les impressions des autres utilisateurs qui interagissent avec lui/elle dans le passé. Des sites tels que eBay¹, Amazon² et OnSale³ sont des maisons d'enchères fermées en ligne où les utilisateurs achètent et vendent des biens. Chaque fois qu'une nouvelle transaction est terminée, l'acheteur évalue le vendeur. Ces notes sont utilisées pour construire la réputation d'un vendeur. D'autres sites, tels que Epinions⁴, permettent aux utilisateurs d'évaluer tout produit ou service sans qu'il y ait de contraintes sur les scores attribués. Les utilisateurs peuvent évaluer des services sans les avoir utilisés ou consommés et peuvent aussi décider de bloquer ou de promouvoir des fournisseurs en les qualifiant comme dignes de confiance. Cependant, il n'y a pas de mécanisme pour affiner ces évaluations.

Sporas [Zach 05] est une version évoluée de ce type de modèles de réputation centralisés. Sporas introduit la notion *de fiabilité* de la réputation et il est plus robuste quant aux changements qui surviennent dans le comportement d'un utilisateur contrairement à Amazon qui se base sur la moyenne de toutes les notes attribuées à cet utilisateur. Cependant, un seul score global peut ne pas convenir à tous les agents. Les agents peuvent avoir des besoins différents ainsi, ils peuvent juger différemment la même situation selon différents aspects. Par exemple, un retard de livraison pourrait être un désastre pour un agent donné et, par conséquent, le score attribué serait négatif, tandis que dans d'autres contextes, il peut ne pas être un problème tant que la qualité obtenue est bonne et le score peut être positif. Les agents peuvent ensuite utiliser les scores des autres pour prendre des décisions de sélection. En général, les agents ayant une réputation élevée sont préférés et sont susceptibles de produire des résultats favorables [Josa 08].

Dans un environnement orienté service, la majorité des systèmes de réputation sont développés à l'aide d'une solution multi-agents où les agents sont coopératifs et ont la volonté de partager leurs opinions sur les autres. Ces opinions sont utilisées pour estimer la fiabilité des fournisseurs ainsi que la qualité des services qu'ils offrent [Huyn 06, Josa 08, Josa 07, Resn 02, Saba 02, Sier 09]. Les deux approches de [Huyn 06, Saba 02] ont développé deux modèles de réputation décentralisés appelés respectivement *Fire* et *Regret* où chaque agent est capable d'évaluer la réputation d'autrui par lui-même.

-
1. <http://www.eBay.com/>
 2. <http://www.amazon.com/>
 3. <http://www.onsale.com/>
 4. <http://www.epinions.com/>

Pour établir la réputation d'un agent, les deux modèles catégorisent les informations disponibles selon plusieurs dimensions en fonction de la source.

Regret par exemple, prend en compte trois dimensions : **la dimension individuelle** modélise l'interaction directe entre deux agents, ce qui représente l'expérience personnelle, **la dimension sociale** intègre la relation du groupe lorsque les interaction directes font défaut en prenant en considération ce que les membres du groupe pensent de l'agent en cours d'évaluation, et **la dimension ontologique** qui ajoute la possibilité d'enrichir la réputation avec d'autres aspects connexes pour établir une autre plus complète.

Tandis que *Fire* a défini quatre sources d'information pour établir la réputation d'un agent qui sont : **la relation de confiance** basée sur l'interaction directe entre deux agents, **la confiance liée au rôle** basée sur d'autres croyances mutuelles autre que l'interaction directe, **la réputation témoin** basée sur le comportement de l'agent envers les autres agents et **la réputation certifiée** fournie par une partie tierce.

Essentiellement, un système de réputation vise à évaluer le comportement d'un agent en se basant sur la collecte, l'intégration et l'agrégation d'opinions exprimées par d'autres agents du système [Resn 00]. Dans le travail de Yu et Singh [Yu 00], l'idée d'agréger les opinions d'autres agents pour bâtir une réputation est appliquée. Ils visent également à éviter l'interaction avec les participants non pertinents et formalisent la génération et la propagation de la réputation dans les environnements SOC. Les scores des agents peuvent être explicites ou implicites selon la méthode de la collecte. Bien que la méthode explicite capte mieux la satisfaction des utilisateurs, elle est plus coûteuse par manque de coopération [Krit 07]. Resnick et Zeckhauser [Resn 02] ont remarqué que certaines personnes montrent une réticence à fournir des opinions et qu'il est difficile d'obtenir des scores négatifs avec un niveau d'honnêteté élevé. Une recherche utilisant des données eBay a révélé que seulement 50% des transactions sont accompagnées de notes [Cont 03].

Par ailleurs, Wang et Vassileva [Wang 07b] ont identifié d'autres problèmes dans les systèmes de réputation :

- 1) Les gens sont généralement réticents à donner des notes négatives, car ils peuvent percevoir les notes attribuées mutuellement et ils ont peur de la vengeance.
- 2) Les agents peuvent changer leur identité, surtout s'ils obtiennent une mauvaise réputation.
- 3) La réputation d'un agent est représentée par une valeur numérique unique, ce qui rend difficile la distinction des différents aspects de la réputation comme la sécurité ou la confidentialité.

- 4) Le calcul de la réputation traite de façon égale toutes les notes et ne prend pas en considération la crédibilité des évaluateurs.

On compte également les approches qui analysent le comportement d'un service Web en terme d'attributs de QoS pour établir sa réputation. La collecte des valeurs est réalisée d'une façon *régulière et objective* par l'intermédiaire d'une partie tierce indépendante [Li 12]. Pour estimer la réputation, la majorité des approches [Day 04, Kale 04, Lala 12, Vu 05, Yu 06] définissent a priori un ensemble fixe d'attributs QoS. La raison de ce choix (un ensemble fixe d'attribut de QoS) n'a jamais été étudié de manière adéquate autre le fait d'indiquer que les études précédentes étaient souvent traitées avec cet ensemble, bien que les attributs repris semblaient raisonnables pour établir la réputation d'un service. Les attributs les plus utilisés sont : la disponibilité, la précision, le temps d'exécution, le coût et la bande passante. Il serait possible, d'ajouter d'autres attributs à cette liste, un attribut représentant l'évaluation globale par exemple, ou d'autres types d'attributs spécifiques au contexte d'application [Bill 09].

3.3 Approches basées sur la recommandation

La découverte et la sélection de services ont également fait l'objet de plusieurs travaux et propositions dans le domaine des systèmes de recommandation. Par exemple, de nombreux sites Web de commerce électronique, comme eBay, Amazon et EPINIONS adoptent des systèmes de recommandation pour alléger la charge de la découverte de services [Kart 05].

Dans la littérature, nous distinguons plusieurs types de systèmes de recommandation, les plus utilisées sont : les systèmes basés sur le filtrage par le contenu, les systèmes basées sur le filtrage collaboratif et les systèmes hybrides. Chacun de ces systèmes a des avantages et des inconvénients [Adom 05b], cependant, leur point commun est qu'ils effectuent des recommandations fondées sur des connaissances concernant les utilisateurs et/ou les services.

3.3.1 Recommandation basée sur le filtrage par le contenu

Un système de recommandations basé sur le filtrage par le contenu recommande des services qui correspondent au profil de l'utilisateur. Le profil est construit à partir du contenu des services que l'utilisateur a noté dans le passé et sur les intérêts qu'il a explicitement définis. Le filtrage basé sur le contenu est composé de deux étapes ; tout

d'abord il analyse les services notés par le même utilisateur et exploite leurs contenus ainsi que les notes fournies pour construire le profil utilisateur ensuite, il positionne ce profil par rapport à un service que l'utilisateur n'a pas encore évalué pour décider de sa pertinence. Dans un système de recommandation basé sur le filtrage par le contenu, la note $R(u, s)$ d'un service s attribuée par un utilisateur u est généralement déduite à partir de l'ensemble des notes $\{R(u, s')\}$ attribuées par le même utilisateur u à d'autres services s' *similaires* à s en termes de contenu. Par exemple, dans un scénario de recommandations de films, en vue de recommander un film à l'utilisateur u , le système de recommandation basé sur le filtrage par le contenu doit déterminer les préférences de l'utilisateur en analysant son profil dans le but de trouver les points communs dans le contenu des différents films auxquels il a attribué une note élevée dans le passé. Ensuite, seuls les films qui disposent d'un degré de similarité élevé avec le profil de l'utilisateur, seront recommandés [Adom 05a]. Il faut noter que le filtrage par le contenu ne pourrait s'appliquer qu'à la découverte de services. Cependant, il est incapable de s'adresser au problème de la sélection de services.

Les systèmes de recommandation basés sur le filtrage par le contenu présentent quelques inconvénients. En général, plus le nombre d'items augmente, plus le nombre d'items dans une seule catégorie augmente, ce qui diminue l'efficacité de la recommandation [Li 03]. Le deuxième problème que rencontre le filtrage par le contenu est la sur-spécification. La sur-spécification consiste à limiter la recommandation à des items similaires ayant un score élevé vis à vis du profil de l'utilisateur. Dans certains cas, il faut faire attention à l'évolution des intérêts de l'utilisateur et les prendre en compte. Enfin, pour produire des recommandations précises, l'utilisateur doit fournir un « feedback » sur les suggestions retournées, une opération que certains utilisateurs n'aiment pas faire.

3.3.2 Recommandation basée sur le filtrage collaboratif

Le filtrage collaboratif ou social de l'information (FC) est l'une des techniques les plus utilisées dans les systèmes de recommandation pour suggérer des services/produits pertinents aux utilisateurs. Les recommandations suggérées se basent sur les relations sociales entre les utilisateurs et sur leurs jugements subjectifs [Bree 98, Sarw 00]. Un système de FC typique est un système centralisé qui est largement utilisé dans les sites de l'e-commerce tels que Amazon. Un nœud centralisé est responsable de la collecte des notes des utilisateurs et de leur stockage dans une matrice avec une ligne pour chaque utilisateur et une colonne pour chaque produit. Un utilisateur reçoit des recommandations basées sur les notes d'autres utilisateurs qui lui sont semblables. Par exemple,

si Alice et Bob ont acheté les mêmes livres A, B, C, et D, ils peuvent être considérés comme similaires. Maintenant, si Alice a également acheté le livre E, un système de FC peut recommander à Bob d'acheter ce livre E. Selon Massa et Bhattacharjee [Mass 04], l'algorithme standard de filtrage collaboratif comporte trois étapes :

- 1) Calculer la similarité entre un utilisateur donné et tous les autres utilisateurs en se basant sur la similarité des notes qu'ils ont attribuées aux mêmes produits utilisés. Le coefficient de corrélation de Pearson est la mesure la plus utilisée pour évaluer la similarité entre deux utilisateurs.
- 2) Pour cet utilisateur, une note d'un produit qu'il/elle n'a pas encore noté est prédite à partir des notes d'autres utilisateurs qui ont noté le même produit. Cette note prédite est pondérée par la valeur de la similarité calculée dans la première étape.
- 3) Les produits avec les notes de prédiction les plus élevées sont recommandés à l'utilisateur.

L'utilisation du filtrage collaboratif dans la découverte et la sélection de services Web est une tendance émergente. Par exemple, Sellami et al. [Sell 09] ont proposé une méthodologie de recommandation pour la découverte de services Web dans des annuaires distribués. Basée sur une requête et un profil utilisateur, la méthodologie peut trouver un annuaire approprié parmi des annuaires répartis en le faisant correspondre avec le profil de l'utilisateur. Puis, elle exécute la requête dans l'annuaire sélectionné pour recommander des services Web appropriés. Manikrao et Prabhakar [Mani 05] ont proposé une plateforme pour la sélection de services Web qui combine un système de recommandation avec un appariement sémantique des exigences du service.

L'approche est basée sur les « feedback » des utilisateurs et une technique de filtrage collaboratif afin d'aider les utilisateurs à sélectionner un service Web à partir d'un ensemble de services similaires. Lorsque l'utilisateur utilise un service Web, le système lui demande d'évaluer le service utilisé. Cependant, une recherche précédente a montré que les utilisateurs sont réticents à fournir des évaluations explicites [Clay 01].

Les systèmes de FC traditionnels ignorent les relations de confiance entre les utilisateurs. Cependant, les utilisateurs préfèrent recevoir des recommandations de la part des pairs dignes de confiance [Mass 04]. Par conséquent, les recommandations peuvent être directement fournies par des pairs en qui l'utilisateur a confiance, ou indirectement par d'autres pairs en qui les pairs dignes de confiance font confiance à l'aide d'un mécanisme de propagation de la confiance [Zarg 09]. L'inconvénient majeur des systèmes FC est qu'ils sont incapables de différencier entre les voisins ayant des goûts similaires

comme des amis ou des étrangers. Bien que les FC utilisent les voisins pour faire des recommandations, ils sont incapables de démontrer comment les individus recherchent des informations à l'aide de leurs amis comme dans les réseaux sociaux.

3.3.3 Recommandation hybride

Une autre famille des systèmes de recommandation est l'approche hybride. Il ne s'agit pas d'une approche indépendante, mais plutôt d'une combinaison de plusieurs approches de recommandation basées sur le filtrage par le contenu et sur le filtrage collaboratif. Pour prédire une note, dans un système de recommandation hybride, chaque approche calcule une valeur de similarité entre les deux utilisateurs (score local). Le système de recommandation hybride combine les scores pour calculer un score final. La différence entre les techniques d'hybridation réside dans la stratégie choisie pour combiner les scores locaux. L'hybridation peut être appliquée au niveau des notes, ou bien au niveau de la fonction de prédiction. Robin Burke résumait les techniques d'hybridation en sept techniques [Burk 02, Burk 07]. Parmi les sept techniques, trois exigent l'existence d'une approche à base d'items parmi les approches hybridées. Dans les environnements orientés services, Balabanovic a proposé un système de recommandation hybride de type méta-niveau⁵, appelé Fab, pour le filtrage du Web [Bala 97]. Dans Fab, les agents de sélection spécifiques à un utilisateur effectuent un filtrage basé sur le contenu en utilisant la méthode de Rocchio afin de maintenir un modèle utilisateur en vecteur de termes qui décrit son centre d'intérêt. Zheng et al. ont proposé WSRec qui est une méthode de filtrage collaboratif hybride pour la recommandation des services Web. Afin de prédire les valeurs de QoS des services candidats, les auteurs exploitent l'historique des valeurs de QoS relevées après utilisation [Zhen 09]. Chiu, et al. ont proposé un modèle de recommandation quantitative hybride qui utilise l'utilité, la réputation, l'impression, la fiabilité, l'attitude des risques et la persuasion [Chiu 09].

D'une manière générale, les systèmes de recommandations sont vulnérables au démarrage à froid et au manque de données. Ces problèmes ont une cause semblable, si le chevauchement entre les notes disponibles est trop petit, le système ne peut pas définir un ensemble d'utilisateurs similaires fiable afin de générer des recommandations. De plus, la manipulation des notes est un autre problème ; néanmoins, quelques approches (par exemple, [Ping 05]) préviennent des agents malveillants. L'idée est d'ignorer les notes qui sont en dessous d'un certain seuil de qualité, ou reçues d'un agent dont la crédibilité est en dessous d'un certain seuil.

5. Un système de recommandation hybride méta-niveau est la combinaison de deux techniques de recommandation où le résultat généré par une est l'entrée de l'autre.

3.4 Réseaux Pair à Pair (P2P)

L'informatique s'est d'abord développée autour du concept de l'ordinateur central tout puissant : les utilisateurs ne communiquent entre eux qu'à travers une machine centrale. Mais avec l'émergence de l'Internet, il devient quasiment impossible de suivre la cadence de l'essor des ressources et des services disponibles d'où la naissance de la notion *de partage et de répartition des données*. C'est dans ce contexte de partage que le concept des *réseaux Pair à Pair (P2P)* est mis en place. Dans un environnement orienté services, au lieu d'utiliser les annuaires centralisés (par exemple, UDDI, ebXML) dans la découverte et la sélection de services, les réseaux Pair à Pair transforment cette fonction d'annuaire à des milliers d'ordinateurs à travers le monde.

La particularité d'un tel système est qu'il est social ; les utilisateurs sont des acteurs actifs du réseau, ils peuvent remplir à la fois le rôle d'un client (consommateur de services) et celui d'un serveur (fournisseur de services). Un réseau Pair à Pair a la particularité d'exiger une répartition distribuée de la description des services entre les pairs sans aucun contrôle centralisé ni organisation hiérarchique et où le comportement global émerge des interactions locales. En quelques années, les réseaux P2P sont devenus, une application majeure de l'Internet en permettant à des millions d'utilisateurs de partager rapidement et sans coût d'infrastructure de grandes quantités de données et de services. C'est pour cela que la majorité des acteurs industriels ont cru que les réseaux P2P reflètent mieux la société que d'autres types d'architectures informatiques [Clar 01]. Par conséquent, la découverte d'un service se réduit à l'interrogation du réseau des pairs pour le type de fonctionnalité requise. Dans la littérature, nous distinguons deux familles de réseaux P2P :

- **Réseaux P2P structurés.** Ils ont la particularité d'organiser le réseau P2P en une topologie routable, c'est-à-dire que chaque pair dispose d'un identifiant permettant de le localiser en suivant un chemin déterministe parmi les pairs, et nécessitant un minimum de messages. La principale limite de cette approche est que le placement des services au niveau des pairs est strictement contrôlé et l'évolution du réseau, par conséquent, entraîne une charge supplémentaire. Cette contrainte limite la robustesse du réseau et de l'autonomie des pairs. Les modèles P2P structurés sont bien adaptés pour mettre en place des systèmes où le contrôle du placement des services est prioritaire, comme le stockage distribué des fichiers. Cependant, ils ne sont pas de bons modèles pour les systèmes avec des membres très dynamiques. Les deux modèles de réseaux P2P structurés les plus utilisés sont *CAN* [Ratn 01] et *CHORD* [Stoi 01].
- **Réseaux P2P non structurés.** Les réseaux P2P non structurés comme *GNU-*

tella [Kan 01] sont caractérisés par une absence totale de contraintes au niveau de la répartition des ressources et la croissance du réseau. Ils s'intéressent plutôt à la décentralisation complète de la prise de décision et du calcul. Ils requièrent uniquement des procédures de maintenance locales et sont topologiquement robuste face à l'évolution du système. Ces modèles sont bons pour la construction des systèmes très dynamiques où l'anonymat et la réduction de la charge sont demandés. Pour rejoindre le réseau, un pair doit connaître l'adresse d'un autre pair déjà connecté et qui sert alors de nœud d'insertion. Par ce nœud, le pair entrant découvre progressivement d'autres nœuds du réseau et établit des liens avec eux, selon un algorithme propre au réseau, pour ensuite y propager les messages. Les requêtes sont ainsi propagées par inondation, chaque pair propageant la requête à tous ses voisins, ce qui implique un grand nombre de messages, ou en suivant un chemin aléatoire, chaque pair sélectionnant un de ses voisins pour propager la requête.

La recherche dans les réseaux P2P est orientée vers la découverte de services plutôt que vers la sélection de services. La recherche peut donner plusieurs services et il n'y a aucun moyen pour les classer afin de sélectionner un service parmi eux (le meilleur service). L'inconvénient majeur de cette approche est que la confiance n'est pas associée aux suggestions faites par les pairs. Pour résoudre cela, Aberer et Despotovic ont proposé dans [Aber 01] une approche qui traite le problème de la gestion de la confiance en se basant sur la réputation. Leur modèle est basé sur la confiance binaire, à savoir, un agent est soit digne de confiance ou non. Dans le cas d'un comportement malhonnête, les agents peuvent transmettre leurs plaintes à d'autres agents. Pour ce faire, ils utilisent une structure de données spéciale, appelé P-Grid, pour stocker les plaintes dans un réseau P2P. Afin d'évaluer la confiance d'un autre agent B, un agent A recherche au niveau des feuilles de la P-Grid les plaintes qui correspondent à l'agent B. Plus tard, Damiani et al. ont proposé dans [Dami 02] une approche qui rajoute à la réputation des agents la réputation des ressources, fournissant des résultats plus satisfaisants et surmontant les limites des solutions qui ne considèrent pas le contexte. Cette approche permet au demandeur d'évaluer la confiance du fournisseur ainsi que la fiabilité de la ressource qu'il offre avant de la télécharger.

3.5 Systèmes de références

Un système de références est un réseau P2P distribué où les pairs fournissent non seulement des services, mais peuvent également référer les demandeurs de services à

d'autres pairs pertinents [Sing 01]. Dans ces systèmes, chaque pair est représenté par un agent. Celui-ci dispose d'une perception restreinte du réseau et maintient une liste dynamique d'agents, qui sont ses accointances sociales i.e., voisins immédiats dans le réseau, dignes de confiance à partir desquels il peut obtenir de bons services ou des références pertinentes.

Au cours d'un processus de découverte de services basé sur un système de références, les agents coopèrent en donnant, poursuivant et évaluant des références [Yu 03]. Les agents qui offrent un bon service sont appelés *fournisseurs de services*. Par contre, ceux qui font référence à d'autres agents sont considérées comme des *témoins* et enfin les agents recommandés sont des *références dignes de confiance* [Artz 07]. Un exemple d'une chaîne de référence est représenté dans la figure 3.1 ci-dessous. Lorsqu'un agent A envoie une demande de service à l'agent B, l'agent B recommande l'agent C, et ensuite l'agent C recommande l'agent D, et finalement l'agent D recommande l'agent E, l'agent qui fournit le service. Les agents B, C, et D sont tous des références. L'agent demandeur A calcule ensuite la confiance qu'il accorde à l'agent E sur la base d'une combinaison linéaire de l'expertise et de la sociabilité des différents agents intermédiaires dans la chaîne de références qui les relie.

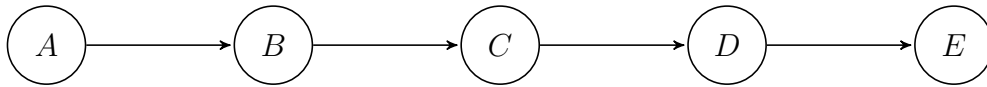


FIGURE 3.1 – Un exemple d'une chaîne de références.

Dans un environnement orienté services lorsqu'un agent a besoin d'un service, il soumet sa requête à ses accointances. Une accointance peut, soit directement offrir le service demandé, soit fournir des références d'autres agents auxquels elle fait confiance. Les références sont en quelque sorte des recommandations qui agissent comme un relais permettant la propagation de la recherche dans le réseau. Chaque agent conserve un modèle pour tous ses accointances, qui décrit leurs expertises (c'est à dire, la qualité des services qu'ils fournissent) et leur sociabilité (c'est à dire, la qualité des références qu'ils fournissent). En se basant sur ces informations, l'agent demandeur décide de le sélectionner en tant que fournisseur de services ou/et de suivre les références qu'il propose. Ce mécanisme de propagation de la recherche est le principe des systèmes de références au cours duquel le demandeur collecte toutes les références possibles et décide localement de poursuivre la recherche en contactant certaines des références proposées ou d'arrêter la recherche.

Si à la fin de la recherche plusieurs services appropriés sont découverts, le demandeur dispose des compétences nécessaires lui permettant de sélectionner la meilleure offre. Une fois le service utilisé, il l'évalue. En se basant sur cette évaluation, il met à jour les valeurs de confiance du fournisseur de service sélectionné et des agents intermédiaires

(témoins) qui ont fourni des références menant à ce fournisseur de service en utilisant un mécanisme d'attribution de récompenses et de pénalités.

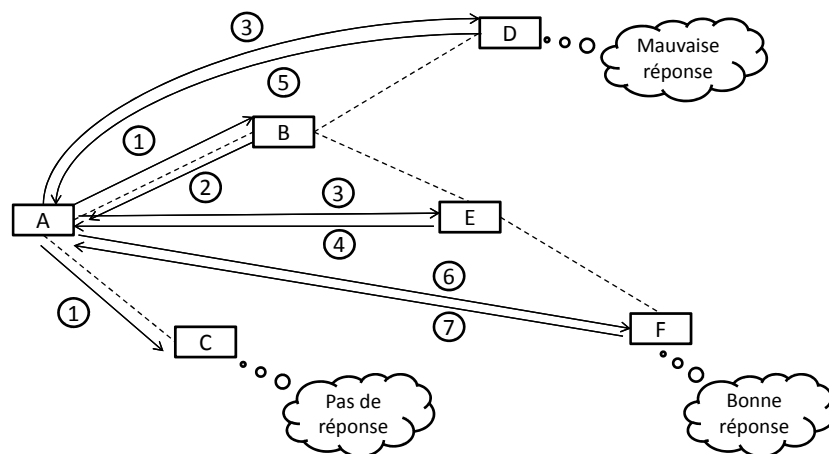


FIGURE 3.2 – Un graphe de références généré à partir d’une requête [Sing 01].

Dans [Sing 01, Yu 99, Yolu 03], les auteurs proposent une approche de localisation d'expertise basée sur un système de références. Comme les réseaux P2P, leur système de références assure une répartition distribuée de la description des services entre les agents sans aucun contrôle centralisé ni organisation hiérarchique. Cependant, la prise de décision associée à la propagation de la recherche est centralisée : quand un agent reçoit une demande d'expertise, soit il essaie de la faire correspondre avec ses propres fonctionnalités, soit il génère une liste de références à partir de la liste de contacts de son utilisateur, à savoir, ses connaissances sociales. Le demandeur de services rassemble toutes les références possibles, évalue les fournisseurs et les services qu'ils offrent et décide de poursuivre ou non la recherche en communiquant avec certaines des références proposées. La figure 3.2 représente un exemple de graphe de références généré à partir d'une requête. Les différentes arêtes montre les échanges effectués entre les agents durant le processus de propagation de la recherche. Comme illustré, l'agent A envoie une requête à ses voisins B et C. L'agent C ne répond pas par contre l'agent B renvoie deux références à D et E, et E renvoie à son tour une référence à F. Les lignes en

pointillé sur la figure indiquent les relations physiques entre les agents, et les cercles numérotés indiquent ce qui suit :

- 1) Requête initiale envoyée par A à B et C (C ignore la requête) ;
- 2) Références vers D et E donnée par B ;
- 3) Requête envoyée par A à D et E ;
- 4) Référence vers F donnée par E ;
- 5) Réponse de D à A ;
- 6) Requête envoyée par A à F ;
- 7) Réponse de F à A.

Comme nous pouvons le constater à travers cet exemple, la décision de propagation de la recherche revient à l'agent demandeur qui a initialisé le processus. L'approche présentée dans [Yu 03] utilise le même mécanisme de propagation qui se base sur une prise de décision centralisée. Cependant, elle évalue la confiance qu'un agent demandeur accorde à un autre agent en se basant sur un graphe de références pondéré. La confiance est définie comme une agrégation de deux mesures à savoir la sociabilité et l'expertise. Abdul-Rahman et Hailes ont classifié les agents dans leur travail [Abdu 00] selon quatre degrés de confiance : très digne de confiance, digne de confiance, peu digne de confiance et non digne de confiance. Cela permet aux agents d'affiner progressivement leur évaluation des références recommandées par un autre agent. Compte tenu des opinions subjectives des témoins dignes de confiance, l'agent demandeur décide de poursuivre la référence qu'ils recommandent ou pas.

3.6 Conclusion

Dans la littérature, plusieurs travaux se sont déjà penchés sur le problème de la découverte et de la sélection sociales de services. Dans ce chapitre, nous avons passé en revue les principaux travaux du domaine à travers une classification distinguant quatre catégories.

La première catégorie regroupe les travaux qui utilisent les systèmes de réputation pour permettre aux agents d'évaluer la fiabilité des fournisseurs ainsi que les services qu'ils offrent. L'évaluation d'un fournisseur est une note globale, publique et accessible

par tout le monde. Elle se fait d'une manière collective en se basant sur les notes attribuées par les agents qui ont interagi avec ce fournisseur dans le passé.

La deuxième catégorie recueille les travaux qui utilisent des techniques de recommandation pour prédire des besoins personnalisés d'un agent en se basant sur la description de son profil et/ou des relations qu'il a avec d'autres agents ayant des comportements ou des profils similaires. Cela permet d'aider les agents à découvrir des services pertinents qui sont plus susceptibles d'être sélectionnés par la suite.

La troisième catégorie porte sur les travaux basés sur les systèmes pair-à-pair. La particularité d'un tel système est qu'il est social, il distribue le contrôle et assure l'autonomie des agents, ce qui facilite le partage et l'échange de ressources. Cela permet de propager la recherche dans le système afin de répondre au mieux aux besoins de l'agent.

Enfin, la quatrième catégorie concerne les travaux qui se basent sur les systèmes de références. L'avantage de ces systèmes réside dans le fait qu'ils sont bien adaptés aux environnements sociaux comme les réseaux sociaux, où les connaissances sont complètement décentralisées. De plus, pour guider les agents dans leur interaction, les systèmes de références utilisent un mécanisme fondé sur la confiance.

Après analyse de ces différents travaux et les limites qui en découlent, nous proposons une approche multi-agents pour la découverte et la sélection de services dans les réseaux sociaux intégrant :

- La conception d'un modèle de confiance bâti sur les propriétés sociales de la confiance qui permet aux agents de raisonner localement sur la fiabilité d'autres agents avant d'interagir avec eux. Contrairement aux systèmes de réputation, notre modèle de confiance doit être personnalisé en se basant sur les croyances de l'agent évaluateur.
- La proposition d'un autre mécanisme de recherche qui va aller au-delà des annuaires centralisée (UDDI) afin d'améliorer les résultats de la recherche par rapport aux besoins et préférences des agents. En effet, le nombre croissant des services publiés dans les réseaux sociaux limite la portée des implémentations centralisées des annuaires UDDI causant un manque d'évolutivité et introduisant un point de défaillance dans le système.
- La distribution de la prise de décision des systèmes de références actuels, ce qui rend les agents plus autonomes. Dans ce cas, les agents décident localement de continuer ou non la recherche sans revenir à l'agent demandeur.

C'est sur ces points que porte notre travail dans les deux prochains chapitres. Nous avons pour objectif de proposer un modèle de confiance compositionnel établi à partir

de plusieurs sources d'informations et applicable dans un contexte de résolution distribuée de problèmes, dans lequel le but global à satisfaire (i.e. les besoins de l'utilisateur) est présumé. Plus précisément, nous voulons proposer une approche de découverte et de sélection de services fondée sur des agents cognitifs capables de raisonner sur les besoins de l'utilisateur et d'interagir avec d'autres agents dignes de confiance.

Chapitre 4

La confiance : définition, topologie et modélisation

Sommaire

4.1	Introduction	48
4.1.1	Définition de la notion de confiance	48
4.1.2	Caractérisation de la confiance	50
4.2	Sources d'information pour la confiance	52
4.3	Topologie du modèle de la confiance	55
4.3.1	La confiance en l'expertise	55
4.3.2	La confiance en la crédibilité sociale	57
4.3.3	La confiance dans les systèmes de recommandation	59
4.3.4	Discussion	61
	Critères de comparaison	61
	Comparaison des approches	62
4.4	Confiance en la crédibilité sociale	64
4.4.1	La position sociale (SPo)	64
4.4.2	La proximité sociale (SPr)	67
4.4.3	La similarité sociale (SSi)	68
	La similarité de voisinage (NS)	68
	La similarité de profil (PS)	70
4.4.4	Calcul de la confiance en la crédibilité sociale	74

4.5	Confiance en l'expertise	75
4.5.1	La spécialisation (Sp)	77
4.5.2	La disponibilité (Di)	77
4.5.3	La qualité (Eval)	78
4.5.4	Calcul de la confiance en l'expertise	81
4.6	Confiance en la recommandation	82
4.7	Conclusion	84

Ce chapitre est dédié à la description de notre modèle de confiance. Tout d'abord, nous allons examiner et définir la notion de confiance à partir d'une revue de la littérature en sciences sociales. Notre objectif est de fournir un modèle de confiance bâti sur les propriétés sociales de la confiance dans le monde réel. Puis, nous identifions les différentes sources d'information conduisant à l'émergence et au calcul de la confiance. Par la suite, nous détaillons la topologie et les caractéristiques sociales de notre modèle de confiance en le situant par rapport aux travaux effectués dans les environnements orientés services. Enfin, nous allons décrire formellement les différentes composantes de notre modèle de confiance à savoir la confiance en la crédibilité sociale, la confiance en l'expertise et la confiance en la recommandation.

4.1 Introduction

La confiance est un phénomène social complexe à multiples facettes [Abdu 00, Saba 02, Al S 12, Saba 13, Huyn 04]. Elle est considérée comme la base de toutes les interactions sociales. Ainsi, un modèle artificiel de confiance doit être basé sur la façon dont fonctionne la confiance entre les personnes au sein de la société. À cette fin, une étude a été menée en sciences sociales pour identifier les caractéristiques de la confiance pertinentes pour notre contexte. Ainsi, nous allons identifier la notion de confiance dans notre domaine d'application de découverte et sélection de services dans les réseaux sociaux.

4.1.1 Définition de la notion de confiance

Mcknight et Chervany ont souligné dans [McKn 01] que la confiance est un concept complexe et pluridisciplinaire et que la recherche de sa définition exige un regard critique. Par conséquent, elle ne peut être bien définie en l'absence d'interaction avec

les sciences sociales [Ivan 08]. De nombreuses études ont été menées à cet effet dans les domaines de la sociologie [Seli 00, Dunn 84, Cole 98, Misz 96, Rowl 79, Mckn 96], l'épistémologie [Hard 93], la psychologie [Bers 98, Cast 98, Fisk 13, Josa 07, Nord 10], l'économie [Adle 01, Ivan 08, Luhm 12] et l'informatique [Abdu 00, Gamb 88, Huyn 04, Golb 05, Mass 06, Saba 02, Saba 13].

Dans cette section nous allons étudier la notion de confiance à partir de différents points de vue sociaux. Une des premières définitions rencontrée dans le domaine de la sociologie est celle énoncée par [Dunn 84]. Celui-ci qualifie la confiance comme une pré-condition clé pour l'existence des relations sociales. Elle représente la capacité d'engagement de quelqu'un à répondre aux demandes des autres. Seligman a défini la confiance comme étant une forme émergente des interactions humaines spécifique à une organisation sociale [Seli 00]. La question de la confiance est abordée dans l'organisation sociale uniquement lorsque les individus ne remplissent pas leur part de responsabilité conformément aux attentes. Coleman a proposé une définition en quatre parties de la confiance [Cole 98]. Il s'agit d'une action qui implique un transfert volontaire d'une ressource (par exemple, physique, financière, intellectuelle, ou temporelle) de la part du "trustor", celui qui évalue la confiance, au "trustee", celui qui est évalué, sans engagement réel de ce dernier. Mcknight et Chervany ont identifié trois conséquences de base que la confiance instaure dans la vie des individus : elle rend la vie sociale prévisible, crée un sentiment de communauté, et il devient plus facile pour les individus de travailler ensemble [Mckn 96].

D'un point de vue économique, la confiance est souvent perçue comme étant un critère qui assure non seulement la fiabilité mais qui réduit aussi les coûts des transactions [Ivan 08]. Dans ce contexte, la confiance fournit une sécurité minimale avant de prendre une action en dépit des incertitudes et des informations incomplètes. Selon Adler [Adle 01], pour réduire les coûts des transactions, nous pouvons remplacer les contrats entre les partenaires économiques et les assurances inhérentes aux risques par la confiance mutuelle. Aussi, dans la mesure où la connaissance prend une forme tacite, la confiance est une condition préalable et essentielle pour le transfert efficace des connaissances. Cela rend la confiance un mécanisme de plus en plus attractif pour les agents économiques.

D'un point de vue psychologique, la confiance est la croyance d'un agent en un autre, en fonction de leurs interactions précédentes, dans la mesure où l'action future effectuée par le dernier conduira à un résultat attendu [Liu 14]. Les personnes dont les relations sont marquées par des niveaux élevés de confiance sociale sont plus enclines à échanger des informations et à agir avec bienveillance envers les autres que ceux dont on n'a pas confiance dans leurs relations. Castelfranchi et Falcone définissent la confiance comme une attitude complexe d'un agent cognitif envers un autre quant au

comportement/action convenable que le second s'engage à effectuer pour l'accomplissement du but du premier [Cast 98]. Elle représente l'état mental d'un agent guidant son comportement et sa prise de décision lorsqu'il veut interagir et déléguer des tâches. Cette confiance entre deux agents peut être vue comme une mesure dans l'intervalle $[0,1]$ selon laquelle, l'un des agents, le "trustor" s'attend à ce que l'autre agent, le "trustee", effectue pour lui une action favorable, ou conduira le "trustee" à se comporter de manière appropriée avec le "trustor" [Josa 07, Nord 10]. La confiance n'est pas une propriété objective qui caractérise le "trustor" mais un degré de croyance subjective basée sur l'expérience au sujet de la fiabilité du "trustee". En ce sens, le raisonnement sur la confiance a une forme plus inductive que déductive.

Les sciences informatiques, en particulier le domaine des systèmes multi-agents et les systèmes en ligne n'ont pas fait exception et ont contribué à la définition de la confiance. Par exemple, [Gamb 88] a défini la confiance comme étant la condition nécessaire qui provoque la coopération entre les agents. Il s'agit d'une probabilité par laquelle un agent A prévoit qu'un autre agent B, effectuera une action donnée dont son bien-être dépend. De même, [Golb 05] a défini la confiance en un agent comme un engagement à adopter fondé sur la conviction que les actions futures de cet agent conduiront à un bon résultat. Selon Abdul-Rahman et Hailes [Abdu 00], la confiance est un phénomène social complexe qui est la base des interactions dans toute société y compris les communautés virtuelles des systèmes en ligne. Par ailleurs, [Mass 06] a utilisé la confiance dans les systèmes en ligne pour indiquer les différents types de relations sociales entre deux individus, telles que l'amitié, l'appréciation et l'intérêt. Dans ce contexte, la confiance est l'opinion explicite d'un individu au sujet d'un autre individu en ce qui concerne la qualité perçue d'une certaine caractéristique du dernier.

Dans cette thèse, la confiance est identifiée comme étant une relation sociale résultant d'un raisonnement individuel et personnalisé. Elle est utilisée pour exprimer un degré de croyance selon lequel le "trustor" prévoit le comportement du "trustee" d'effectuer ou de recommander un service offrant une certaine qualité (QoS). Sur le plan décisionnel, la confiance est un mécanisme qui permet à l'agent de raisonner sur la fiabilité de chacune de ses accointances sociales et de guider donc son comportement et sa prise de décision lorsqu'il veut interagir avec celles-ci.

4.1.2 Caractérisation de la confiance

Pour découvrir les caractéristiques du « monde réel » nous permettant de modéliser la confiance, nous nous sommes tournés vers les sciences sociales. L'intégration de l'aspect social constitue une base solide et pratique dans la modélisation de la confiance. Bien

que les êtres humains soient en mesure de former une seule opinion globale sur le niveau de la confiance qu'il attribuent aux autres avant d'interagir avec eux, une seule dimension ne peut pas refléter les caractéristiques de la confiance. En particulier dans les application SMA [Abdu 00, Huyn 04, Saba 02] où la prise de décision basée sur la confiance est une tâche complexe car l'évaluation globale d'un agent est obtenue par la combinaison de différentes mesures. Ainsi, il est plus judicieux et plus pertinent de définir la confiance comme un concept compositionnel. Dans cette thèse, la confiance en un agent est estimée en fonction de *son expertise* en se fondant sur l'historique interactionnel exprimé par les notes et les opinions subjectives fournies par d'autres agents [Lala 12, Li 12, Mass 05]. Elle est aussi évaluée en fonction de la *crédibilité sociale* de l'agent en se fondant sur son profil, ses relations sociales, son appartenance à une catégorie sociale ou organisationnelle et le rôle qu'il peut jouer dans la société [Bans 10, Liu 14, Maar 10, Sier 09, Huyn 04, Saba 02, Yan 08]. Elle est aussi calculée par rapport au degré de fiabilité des *recommandations* de l'agent en se basant sur la qualité des recommandations qu'il a fournies durant ses anciennes interactions [Golb 05, Hess 08, Mass 07].

Dans les réseaux sociaux, la question de la confiance devient de plus en plus importante. Dans [Walt 08], les auteurs ont identifié que la densité du réseau, la similarité des préférences entre les agents, et la rareté des connaissances sur la fiabilité des recommandations sont des facteurs cruciaux pour générer des recommandations axées sur la confiance. Dans cette même perspective, nous renforçons l'utilisation de l'information sociale en se fondant sur l'Analyse des Réseaux Sociaux (ARS). ARS sont des techniques sociologiques utilisées pour analyser les tendances des interactions et des relations entre les acteurs sociaux afin de déterminer les structures sociales sous-jacentes telles que les nœuds centraux qui agissent comme des influenceurs (Le terme « influenceur » fait référence au leader d'opinion opérant sur le Web et particulièrement au leader d'opinion actif au sein des blog-sphères), les groupes denses et les modes d'interaction entre les nœuds. Les informations extraites sont aussi bien sémantiques que structurelles.

Les informations structurelles comprennent l'analyse des chemins dans le graphe, la structure des voisinages et la position sociale. Par exemple, un demandeur de service météo accorde plus de confiance au service météorologique national qu'à un fournisseur de services privé qui peut exploiter les données personnelles des demandeurs de services à des fins commerciales.

Dans un réseau social, des agents comme le fournisseur de service météorologique national sont très sollicités et par la suite reliés à un grand nombre de demandeurs de services. Dans ce cas un tel fournisseur de services occupe une position centrale dans le réseau social et est considéré comme un fournisseur digne de confiance. La mesure

de degré est fondamentale dans un réseau social et participe au calcul de la confiance en la crédibilité sociale.

Les informations sémantiques incluent les profils de l'agent demandeur de services et des autres agents de son réseau social, leur rôles et la nature de leurs interactions. Les agents peuvent avoir plusieurs types de relations telles que l'amitié, le partenariat, la famille, etc. Selon les préférences du demandeur de services et le contexte de l'interaction, la nature de la relation peut influencer le calcul de la confiance. Dans l'exemple de garde d'enfants, le demandeur de services accordera naturellement plus de confiance à un membre de la famille qu'à un ami, encore moins à un collègue de travail pour s'occuper de son enfant. D'autre part, comme indiqué dans [Zieg 07, Zieg 04], il existe une forte corrélation entre la confiance et la similarité de profils : les gens préfèrent généralement les suggestions qui viennent de personnes ayant des profils similaires. Ainsi, plus la similitude entre deux utilisateurs est importante, plus la relation de confiance entre eux est élevée.

Avant de présenter la topologie de notre modèle de confiance, nous allons présenter dans la section suivante les différentes sources d'information que nous allons exploiter.

4.2 Sources d'information pour la confiance

Dans le contexte des SMA, la confiance représente la croyance d'un "trustor" (l'agent évaluateur de la confiance) envers le comportement attendu du "trustee" (l'agent évalué) si une interaction est établie entre eux [Cast 98]. Cette croyance est le résultat d'un raisonnement complexe et interne de l'agent qui se fonde sur plusieurs facteurs et critères issus de différentes sources d'information.

Des recherches tentent d'identifier les sources d'information [Adle 01, Huyn 06, Kram 99, Saba 13, Wang 13, Li 14a] qui conduisent à l'émergence de la confiance, y compris les facteurs psychologiques, sociaux et organisationnels qui influencent la prise de décision des agents quant à la fiabilité et la volonté des autres agents de se comporter comme prévu lors d'une interaction. Comme nous l'avons dit, un modèle de confiance qui n'utilise qu'une seule source d'information est incapable d'effectuer une évaluation significative de la confiance. Dans les SMAs, l'évaluation de la confiance peut être considérée comme une synthèse de toutes les croyances, les états mentaux et les valeurs connues par le "trustor" concernant le "trustee" [Saba 13].

La phase d'identification des sources d'information est donc importante pour comprendre le modèle computationnel de la confiance dans les applications informatiques

[Saba 05]. Dans cette thèse, nous utilisons de nombreuses sources d'information pour construire notre modèle de confiance. Bien que le nombre de sources qui fournissent des informations liées à la confiance peut varier d'une application à l'autre, nous considérons que la plupart des sources d'information peuvent être classées en quatre catégories principales comme suit :

- **L'expérience individuelle** : elle représente l'interaction directe entre le "trustor" et le "trustee". Basé sur ses propres expériences antérieures, le "trustor" estime le comportement attendu du "trustee" dans l'interaction en cours. À cet égard, l'expérience individuelle peut être considérée comme une source importante pour établir une confiance personnalisée fondée sur les connaissances [Lewi 95]. Dans notre modèle, ce type d'information est utilisé pour calculer la confiance en la recommandation et à la fois la confiance en l'expertise.
- **L'expérience communiquée** : Dans l'expérience individuelle, seuls les interactions directes entre le "trustor" et le "trustee" sont prises en considération. En l'absence de ces expériences directes, les expériences indirectes ou les opinions d'autres agents peuvent être considérées. En particulier, dans les environnements distribués les expériences individuelles sont généralement limitées et d'ordre local. En l'absence d'une partie tierce digne de confiance, l'historique interactionnel fournit au "trustor" des informations utiles pour estimer la confiance, la motivation et les intentions du "trustee". L'expérience communiquée ne doit pas être confondue avec la réputation. Le mécanisme permettant la transmission et la communication de ces informations est connu sous le nom du *système de références* souvent utilisé pour localiser des services grâce à des références dignes de confiance [Rein 86]. Les agents non adjacents interagissent indirectement via les agents intermédiaires afin de calculer la confiance entre eux.
- **Information sociales** : Outre l'historique des interactions entre les agents (utilisés dans les deux catégories précédentes), d'autres sources d'information peuvent être utilisées pour établir la confiance. Il s'agit des informations sociales que l'on peut extraire du réseau social. Selon [Gros 05], les réseaux sociaux sont une source riche pour la collecte d'information. Ces informations peuvent être classées en deux catégories ; sémantique et structurelle. L'information sémantique inclut les profils des agents et les différents types de relation entre eux. Les informations structurelles sont dérivées à partir des techniques d'analyse de réseaux sociaux. Par exemple, la position du "trustee" dans le graphe du réseau social, la proximité sociale entre le "trustor" et le "trustee" et la structure de leur voisinage. Le principal avantage de cette source d'information est que les données récupérées sont quantifiables et peuvent être utilisées pour évaluer *objectivement* la confiance

en la crédibilité sociale entre deux agents. Par exemple, pour rechercher un film, un individu préfère faire confiance à une personne qui partage les mêmes goûts cinématographiques et qui possède en plus une position centrale dans le réseau social.

- **Information certifiées fournies par le "trustee" :** Dans les catégories précédentes, le "trustor" doit recueillir par lui-même l'information nécessaire. Toutefois, le "trustee" peut également participer activement à l'évaluation de la confiance en présentant des arguments à propos de son expertise. Dans un environnement orienté services, de tels arguments sont les attributs de QoS (par exemple, la fiabilité, la facilité d'utilisation, la disponibilité) des différents services qu'il offre. Contrairement à l'expérience communiquée qui doit être collectée par le "trustor" depuis d'autres agents, le "trustee" maintient ses informations certifiées et les fournit sur demande pour gagner la confiance du "trustor". Comme nous l'avons mentionné précédemment, une partie du calcul de la confiance en l'expertise est fondée sur l'expérience individuelle. L'autre partie est fondée sur cette dernière source d'information.

Les sources d'information disponibles pour un agent dépendent de ses capacités sensorielles et les caractéristiques de l'environnement dans lequel il évolue. L'utilisation de plusieurs sources d'information, si elles sont prises en compte de manière intelligente par le modèle, peut augmenter la fiabilité des valeurs de confiance calculées, mais elles augmentent également la complexité du modèle. En outre, les scénarios qui permettent aux agents d'obtenir diverses informations exigent des agents intelligents et par conséquent, plus complexes.

Notre modèle de confiance intègre ces quatre sources d'information. Il est capable de fournir des mesures de confiance qui guident les agents dans leur prise de décision lorsqu'ils veulent interagir avec d'autres agents. Dans la littérature, il a été souligné que la volonté des individus de s'engager à faire confiance à d'autres est un processus qui dépend en grande partie de l'historique des interactions [Boon 91]. Cependant, dans certaines situations il est difficile d'accumuler suffisamment de connaissances concernant les agents potentiels avec lesquels le décideur veut interagir [Li 14a]. C'est pour cela que les informations sociales fournissent une base objective pour la prise de décision et améliore grandement la qualité du jugement à cet égard. Contrairement à d'autres modèles de confiance basés sur l'historique interactionnel, le "trustor" n'a pas à obtenir ce type d'information à partir des interactions passées. Ainsi, même en l'absence d'interactions directes ou indirectes le "trustor" est capable d'évaluer la confiance du "trustee" ce qui améliore la *robustesse* de notre modèle. Notre préoccupation est d'augmenter la précision, l'expressivité et la robustesse de notre modèle de confiance

afin d'améliorer la qualité des services sélectionnés. Il convient de noter que la composante de la confiance en l'expertise dépend des informations fournies par le "trustee" (informations certifiées) et, par conséquent, l'évaluation est sensible à toute inexactitude de ces informations. Dans cette thèse, nous n'allons pas traiter les problèmes liés aux mensonges et à l'imprécision. Ces derniers seront traités dans les travaux futurs.

Avant de détailler formellement chacune des composantes de notre modèle de confiance, nous allons présenter, dans la section suivante, une étude bibliographique faisant état de la topologie de la confiance.

4.3 Topologie du modèle de la confiance

Comme nous l'avons dit, la confiance est un concept compositionnel, complexe et à plusieurs facettes [Abdu 00, Saba 02] particulièrement dans les applications SMA où les décisions associées à l'interaction et à la délégation des tâches dépendent de plusieurs facteurs associés à la confiance. La prise de décision associée à la sélection d'un service Web dépend non seulement de la qualité de ce service (i.e., la confiance en l'expertise) mais aussi du degré de sociabilité de l'agent qu'il offre et également de la fiabilité des recommandations des agents intermédiaires conduisant vers lui.

L'objectif de cette section est de présenter la topologie de notre modèle de confiance et de décrire chacune de ses composantes ; (i) *la confiance en l'expertise (EC)*, (ii) *la confiance en la crédibilité sociale (SC)* et (iii) *la confiance en la recommandation (RC)*. À cet égard, nous avons réalisé une revue de la littérature pour dégager l'intérêt de chaque composante dans l'évaluation globale de la confiance et pour mettre en évidence notre apport en égard aux travaux existants.

4.3.1 La confiance en l'expertise

La confiance en l'expertise d'un agent est évaluée principalement en fonction de son comportement associé à la qualité des services qu'ils offre. Dans la littérature, plusieurs travaux [Abdu 00, Bill 07, Lala 12, Li 12, Vu 05, Xu 07, Wang 07b, Wang 09] s'appuient sur la description non fonctionnelle des services offerts pour établir les modèles de confiance. Principalement, ils utilisent les attributs de qualité de service (QoS) comme la fiabilité, la disponibilité et le temps d'exécution et/ou les scores de réputation fournis par d'autres consommateurs après une transaction réussie. Par exemple, Vu et al. [Vu 05] ont introduit une plateforme distribuée pour la sélection de services se basant sur les attributs de QoS. Pour estimer la confiance et la réputation des services, ils comparent

la qualité du service annoncée à la qualité du service rendue. Si la qualité de service rendue est aussi bonne que la qualité du service annoncée, le service est de bonne réputation. De même, Xu et *al.* ont présenté un algorithme de découverte basé sur les QoS et renforcé par la réputation pour l'appariement fonctionnel, le classement et la sélection de services visant à améliorer la qualité de la composition sous-jacente [Xu 07]. Dans [Wang 09], les auteurs ont proposé des mesures d'évaluation de confiance et une formule de calcul pour les agréger. En outre, ils ont proposé une approche basée sur la logique floue pour déterminer des rangs de réputation qui permettent de faire la distinction entre un nouveau et un ancien fournisseur de services. L'objectif est de motiver les nouveaux fournisseurs de services et de pénaliser les anciens fournisseurs de services qui disposent d'une mauvaise qualité de service.

D'autres approches tirent profit des retours d'expérience des consommateurs quant aux services qu'ils ont utilisés. [Bour 09] ont proposé un modèle de confiance centré fournisseur pour la composition des services Web. Dans ce modèle, la confiance d'un fournisseur est évaluée à l'aide d'une mesure de réputation basée sur les valeurs de QoS. Cette réputation est établie à partir des retours d'expériences récupérés après l'exécution du service et des commentaires subjectifs fournis par les consommateurs. Dans la même lignée, [Li 12] s'appuient sur les évaluations fournies par les consommateurs y compris les notes, les avis et les commentaires pertinents pour prédire la réputation d'un service.

Malik et Bouguettaya ont proposé une plateforme appelée RATEWeb visant à établir la confiance dans les environnements orientés services [Mali 09]. RATEWeb prend en charge un modèle coopératif dans lequel les consommateurs de services partagent leurs expériences par rapport aux services utilisés sous forme d'une note d'évaluation personnelle. Les différentes notes sont agrégées par une somme pondérée pour obtenir la réputation d'un fournisseur de services. Cela est utilisé pour réaliser une sélection et une composition de services digne de confiance. Abdul-Rahman et Hailes ont défini un modèle de confiance pour les communautés virtuelles basé exclusivement sur les croyances des agents. Ces croyances incluent des informations collectées depuis l'expérience directe, les informations de réputation et les recommandations fournies par d'autres agents [Abdu 00].

Billhard et *al.* ont examiné le problème de la sélection des fournisseurs de services en utilisant une approche fondée sur l'expérience. Dans cette approche [Bill 07], les consommateurs utilisent des mécanismes de confiance et de réputation leur permettant d'estimer les comportements attendus des fournisseurs à partir des anciennes expériences faites dans des situations similaires. Plus précisément, lorsqu'aucune expérience antérieure (i.e., une valeur de réputation) est disponible pour une paire particulière

fournisseur/service, les auteurs ont proposé une méthode pour inférer la valeur de la confiance basée sur les expériences passées avec des services similaires. Lalanne et *al.* ont proposé une approche pour l'utilisation de *la qualité d'expérience* (QoE) en tant que facteur de sélection de services, y compris une analyse des différents facteurs qui peuvent affecter la qualité perçue par le consommateur. Dans leur solution proposée [Lala 12], ils décrivent une méthodologie pour mesurer la QoE et établissent un modèle de corrélation entre la QoS et la QoE.

Dans cette thèse nous distinguons deux types d'interaction dans l'évaluation de la confiance d'un fournisseur :

- Interaction *agent/service* qui est exprimée par l'avis et l'opinion d'un consommateur qui a interagi dans le passé avec le service offert par le fournisseur.
- Interaction *agent/agent* qui est représentée par la relation sociale reliant le consommateur au fournisseur dans le réseau social indépendamment des services qu'il offre.

Dans les travaux mentionnés ci-dessus, seules les interactions de type *agent/service* sont prises en considération. Cependant, *l'aspect social*, qui est un enjeu important pour les fournisseurs humains, n'est pas pris en considération. Dans la section suivante, nous présentons les travaux existants qui évaluent la confiance des fournisseurs sur la base de la crédibilité de leurs comportements sociaux.

4.3.2 La confiance en la crédibilité sociale

Dans les environnements orientés services, si un agent A fait confiance à un autre agent B cela veut dire que A s'attend à ce que B délivre le service souhaité avec les qualités de services prévues. Cependant, comme il a été démontré dans [Al S 12], évaluer la confiance d'un agent en se basant uniquement sur les propriétés non fonctionnelles (telles que les attributs de QoS et la réputation) des services qu'il offre n'est pas suffisant pour s'assurer de l'efficacité des services découverts. D'autant plus que dans les applications SMA, la confiance entre deux agents est basée sur un autre aspect appelé la sociabilité [Cast 98, Saba 13]. Dans notre cas, cet aspect est un mécanisme utile pour les agents dans leurs prises de décision avant de s'engager dans une interaction. L'utilité est plus substantielle dans une telle situation : si un agent A n'a jamais interagi dans le passé avec les services offerts par un agent B, il est dans l'incapacité de savoir s'il est digne de confiance en se fondant sur un historique vide. Cependant, A peut évaluer

la confiance de B en termes de sa crédibilité sociale sans compter sur son historique ni sur les opinions des autres agents et ensuite, éviter la subjectivité sous-jacente. Il est judicieux d'intégrer une autre composante dans la modélisation de la confiance tenant compte des interactions de type agent/agent. Il s'agit de la composante sociale que nous allons exprimer à travers *la confiance en la crédibilité sociale*. Celle-ci s'appuie sur les connaissances tirées du réseau social de l'agent demandeur de service telles que les types de relations entre agents, leurs similarités sociales basées sur l'analyse de leurs profils et la comparaison de leurs voisinages et les rôles qu'ils peuvent jouer dans le processus de sélection de services. Cela nous permet d'étudier la crédibilité sociale d'un fournisseur avant même d'utiliser ses services. Par exemple, Da Silva et Zisman [Da S 12] ont intégré les interactions de type agent/agent dans leur modèle de confiance en considérant différents types de relations entre les agents. Ces types de relations sont définis par le biais d'un concept organisationnel appelé groupe. Chaque agent est caractérisé par trois groupes connexes à savoir le groupe des agents dignes de confiance, celui des agents non dignes de confiance, et celui des inconnus lui permettant de déterminer son niveau de confiance pour chaque agent. Ziegler et *al.* ont démontré dans [Zieg 07, Zieg 04] qu'il existe une forte corrélation entre la confiance et la similarité des profils : les gens préfèrent souvent recevoir des services de la part des personnes ayant des profils similaires. Ainsi, plus la similarité de profils entre deux utilisateurs est grande, plus la relation de confiance sociale entre eux est importante.

D'autres travaux [Bans 10, Liu 14, Maar 10, Sier 09] ont utilisé les techniques d'analyse de réseaux sociaux pour évaluer la confiance en la crédibilité sociale qu'un agent peut avoir à l'égard d'un autre agent. Bansal et *al.* [Bans 10] par exemple, ont évalué la confiance des agents en fonction de leur degré de centralité, car celui-ci donne une indication sur leur puissance sociale dans le réseau social, et ainsi sur leur degré de sociabilité. Cependant, cela est insuffisant pour réaliser une évaluation significative de la confiance dans les applications SMA car le modèle de confiance est trop simple étant donné qu'il est établi sur la base d'une seule mesure. Maraadji et *al.*, ont considéré la proximité sociale comme une mesure pour évaluer la confiance en la crédibilité sociale [Maar 10]. Sierra et Debenham ont utilisé la réputation pour évaluer la confiance des agents [Sier 09]. La réputation d'un agent est fondée sur un certain nombre de mesures d'analyse de réseaux sociaux (par exemple, son prestige et sa centralité) pour analyser sa capacité à faire circuler l'information en l'absence d'indépendance complète entre les opinions exprimées par les agents. Récemment, Liu et Wang ont présenté dans [Liu 14] un modèle pour la sélection des fournisseurs de service tenant compte des contraintes relatives aux nœuds adjacents et ceux de l'extrémité de la chaîne basé sur la confiance dans des réseaux sociaux complexes. Leur réseau social complexe considère trois concepts qui sont la confiance, les relations sociales et le rôle de recommandation

reflétant mieux les situations du monde réel.

Toutes les approches citées ci-dessus ne prennent pas en compte l'aspect sémantique des relations entre les agents et opèrent avec un réseau social mono-relationnel. Dans le présent travail, nous intégrons l'aspect sémantique (i.e., les profils des agents et les types des relations entre eux) dans la modélisation de la confiance afin d'améliorer son expressivité et rendre plus efficace la découverte et la sélection des services. Lorsqu'un agent est incapable d'offrir un service satisfaisant les besoins de l'agent demandeur, il peut l'aider en lui recommandant des agents susceptibles de fournir de bons services. Dans la suite, nous allons présenter quelques travaux analysant la confiance des recommandations fournies pour montrer l'intérêt que représente cette composante dans la modélisation de la confiance.

4.3.3 La confiance dans les systèmes de recommandation

Le but d'un système de recommandation basé sur la confiance est de générer des recommandations personnalisées en agrégeant des avis provenant de différents utilisateurs dans le réseau de confiance. Selon Golbeck [Golb 06a], les techniques de recommandation qui analysent les relations de confiance fournissent des résultats plus précis et hautement personnalisés par rapport à d'autres techniques de filtrage collaboratif (voir Sect. 3.3.2). Golbeck a proposé un site appelé FilmTrust qui génère des recommandations prédictives basées sur la confiance pour des films dans un réseau social. La confiance est fondée sur les connaissances extraites à partir des notes et des commentaires ajoutés par les utilisateurs dans le système. Hang et Singh ont proposé dans [Hang 10] une approche de recommandation fondée sur la confiance, qui fournit des recommandations personnalisées dans le réseau de confiance¹ de l'utilisateur. L'approche repose sur une mesure de similarité de sommets entre le réseau de confiance et un graphe de chemin de longueur trois. En se basant sur les scores de similarité calculés, l'approche produit des recommandations qui reflètent à la fois la structure et les valeurs de confiance des relations. Al-Sharawneh et Williams ont développé dans [ALS 09] une approche pour modéliser des systèmes constitués d'agents autonomes et interactifs. À travers un ensemble de simulations, ils ont étudié l'influence du rôle des agents, selon la stratégie « Follow the Leader » énoncé par Goldbaum [Gold 08], dans la formulation des recommandations. Massa et Aversani ont proposé un système de recommandation basé sur la confiance où il est possible de rechercher des utilisateurs

1. Les réseaux de confiance sont des graphes orientés et pondérés dont les nœuds représentent les agents, les liens représentent les relations de confiance et les poids représentent des valeurs de confiance.

dignes de confiance en appliquant la propagation de la confiance dans le réseau de confiance [Mass 07]. Heß [Hess 08] a généralisé le système de recommandation basé sur la confiance pour un seul item (tel que les films) vers les ressources liées, en particulier les documents. A cet effet, elle a construit un deuxième type de réseau, appelé réseau de références pour les documents. Les recommandations pour les documents sont généralement faites par des mesures de visibilité fondées sur les références qui considèrent qu'un document est pertinent s'il est souvent référencé par des documents importants. Ces mesures sont calculées dans un réseau en architecture multi-couches qui intègre plusieurs réseaux, tels que les réseaux de confiance et les réseaux de références pour les documents, permettant de combiner les informations présentes dans les différentes couches.

Les techniques de recommandation fondées sur la confiance fournissent des mécanismes de dissémination des valeurs de la confiance au sein du réseau social. La diffusion est le principe de base qui permet d'inférer la confiance entre des agents non adjacents [Nevi 15]. Cependant, comme les approches vues dans la section 4.3.1, ces techniques de recommandation dépendent entièrement des informations fournies par les consommateurs à la fin des transactions réussies. Dans le cas d'une désinformation ou manque d'information, l'estimation de la confiance pourrait être manipulée ou biaisée. En outre, comme l'a souligné [Sarw 01], la rareté des données dans les systèmes de recommandation demeure un problème non résolu, ce qui rend les techniques de recommandation moins efficaces. Néanmoins, le but ultime d'un système de recommandation est de fournir des recommandations dignes de confiance et de haute qualité qui peuvent être acceptées par les utilisateurs.

L'émergence des réseaux sociaux a attiré récemment l'attention de certaines études dans les systèmes de recommandation telles que [Ma 08, Ma 11] qui proposent d'intégrer l'information sociale pour améliorer la qualité des recommandations. D'un autre côté, comme mentionné en psychologie sociale [Bers 98, Fisk 13], dans la société réelle, un individu préfère les recommandations fournies par des amis de confiance. De plus, sur la base d'une étude statistique, [Sinh 01] et [Bedi 07] ont démontré qu'entre des recommandations fournies par des amis dignes de confiance et celles des systèmes de recommandation, les recommandations fournies par les amis dignes de confiance sont préférées en termes de qualité et d'utilité. C'est dans ce cadre s'inscrivent nos travaux. Les réseaux sociaux ont une structure distribuée dans laquelle les agents sont dotés d'une vision locale et de connaissances limitées. Ces connaissances sont utiles pour évaluer la confiance en la recommandation d'un agent. Dans la littérature, peu de travaux combinent différentes composantes dans la modélisation de la confiance. Indrajit et Sudip ont modélisé la confiance par un vecteur qui se compose de l'expérience du "trustor" par rapport au "trustee", la connaissance du "trustor" concernant le "trus-

tee" dans un contexte particulier, et les recommandations d'autres "trustee" [Indr 04]. L'objectif de ce modèle est de savoir comment aborder ces trois aspects indépendants de la confiance dans les évaluations. Le travail le plus similaire est celui de [Wang 13] dans lequel les auteurs ont proposé un modèle contextuel de confiance dans les réseaux sociaux. Selon les auteurs, la confiance entre deux agents dépend du contexte social ainsi que du contexte de l'interaction. Le contexte social est l'environnement social de l'agent dans le réseau social qui peut être divisé en contexte social indépendant, comme le facteur d'impact de son rôle dans l'organisation, ses préférences et son expertise du domaine, et en contexte social dépendant, comme sa confiance, ses relations sociales et le degré de son intimité sociale avec les agents adjacents. Bien que ce modèle de confiance combine les aspects de sociabilité, d'expertise et de recommandation dans l'évaluation, le processus de décision associé à l'inférence de la confiance entre deux agents distants est centralisé ce qui s'oppose à la nature des réseaux sociaux.

4.3.4 Discussion

Dans cette section, nous présentons un ensemble de critères pour évaluer les principales approches de la découverte et de la sélection de services fondées sur la confiance dans les réseaux sociaux. Nous comparons ensuite notre approche à ces approches selon les critères sélectionnés.

Critères de comparaison

Nous avons déterminé trois groupes de critères : la modélisation de confiance, la mise en œuvre et la performance de l'approche. Les critères sur la modélisation de confiance tiennent compte des trois composantes identifiées précédemment :

- l'expertise ;
- la crédibilité sociale ;
- la recommandation.

Les critères sur la mise en œuvre des approches sont :

- la prise de décision (centralisée, distribuée) ;

- le type d'interaction (agent/service, agent/agent) ;
- le démarrage à froid ;
- l'intégration de préférences des utilisateurs.

Tout d'abord, il s'agit de déterminer la nature de la prise de décision associée au processus de découverte de services. En effet, ce critère permet d'évaluer la complexité de la mise en place du système. Ensuite, l'idée est de préciser les types d'interactions utilisés dans l'évaluation de la confiance. Puis, savoir si le démarrage à froid est assuré ou non. Cela permet d'évaluer la robustesse du système. Enfin, il s'agit de déterminer si l'approche prend en considération les préférences du demandeur, centre du processus décisionnel, pour fournir des résultats personnalisés et adaptatifs. Il est ainsi intéressant d'étudier comment sont mises en œuvre les approches afin de déterminer les avantages et les inconvénients de chacune. Les critères sur les performances liées aux approches sont :

- le passage à l'échelle ;
- la flexibilité ;
- la robustesse.

Compte tenu de l'objectif lié à la découverte et la sélection des services dans les réseaux sociaux qui est la recherche « en ligne », donc souhaitée rapide, les performances constituent un aspect crucial, non seulement au niveau du temps d'exécution, mais également au niveau du passage à l'échelle, de la flexibilité et de la robustesse.

Comparaison des approches

La comparaison des travaux est récapitulée dans le Tableau 4.1), où un + (resp.-) signifie que l'approche a une influence positive (resp. négative) sur le critère précisé dans le titre des colonnes.

Concernant le critère de la modélisation de confiance, nous distinguons trois familles d'approches. La première famille contient les approches récentes de [Al S 12] et de [Wang 13] qui utilisent les trois composantes d'expertise, de crédibilité sociale et de recommandation dans la modélisation de la confiance comme c'est notre cas. La deuxième famille est formée par les approches de [Saba 02], de [Yu 03] et de [Huyn 04]

qui ne prennent pas en considération la composante de recommandation dans la modélisation de la confiance. La troisième famille est représentée par l'approche de [Golb 05] qui ignore la qualité d'expertise dans l'évaluation de la confiance d'un agent.

Dans la mise en œuvre, nous identifions deux classes d'approches pour la prise de décision. La première classe inclut les approches qui distribuent la prise de décision entre les agents comme notre approche et celles de [Saba 02], [Yu 03], [Huyn 04]. Ce type d'approche accorde une autonomie aux agents leur permettant d'être actifs et impliqués dans les processus de découverte et de sélection de services Web. La deuxième classe concerne les approches de [Golb 05], [Al S 12] et [Wang 13] qui centralise le processus décisionnel dans un seul agent qui est généralement le demandeur de services. Ce dernier a une connaissance globale des ressources disponibles dans le réseau et agit d'une manière individuelle dans la sélection de services Web. Cela induit un seul point de défaillance dans le système.

En ce qui concerne le type d'interaction, il s'avère que toutes les approches considèrent aussi bien les informations résultant des interactions de type agent/service que celles issues des interactions de type agent/agent à l'exception de [Golb 05] qui capitalise uniquement sur les interactions de type agent/agent dans le réseau social.

Pour le démarrage à froid, à l'exception de [Wang 13], toutes les approches y compris la notre introduisent une phase d'amorçage qui permet l'initialisation des attributs de QoS des nouveaux services par des valeurs non nulles. L'intérêt de cette phase est d'améliorer la robustesse du modèle de confiance.

Un autre critère qui devrait être pris en compte dans la mise en œuvre d'une approche de découverte ou de sélection de services est l'intégration des préférences du demandeur de services. Les approches [Yu 03] [Al S 12], [Wang 13] et notre approche considèrent les préférences d'un demandeur dans la formulation de la requête répondant ainsi à ses besoins personnalisés.

Enfin, concernant la performance, il est à souligner qu'une prise de décision distribuée permet un passage à l'échelle de l'approche qui est le cas de notre approche et celles de [Saba 02], [Yu 03], [Huyn 04].

Pour la robustesse, la première chose à noter est que les approches qui intègrent les trois composantes de confiance dans la modélisation sont robustes. L'avantage d'un modèle de confiance riche et expressif est qu'il protège contre la manipulation. De plus, les approches de [Saba 02] et [Huyn 04] sont aussi robustes car elles utilisent des sources de données variées capables de fournir des mesures de confiance fiables.

Approches	Critères			Modélisation		Mises en œuvre						Performances	
	expertise	crédibilité sociale	recommandation	prise de décision	type d'interaction	démarrage à froid	intégration de préférences	passage à l'échelle	robustesse				
[Golb 05]	-	+	+	distribuée	centralisée	agent/service	agent/agent	+	-	-	-		
[Saba 02]	+	+	-	+	-	+	+	+	-	+	+		
[Yu 03]	+	+	-	+	-	+	+	+	+	+	-		
[Huyn 04]	+	+	-	+	-	+	+	+	-	+	+		
[Al S 12]	+	+	+	-	+	+	+	+	+	-	+		
[Wang 13]	+	+	+	-	+	+	+	-	+	-	+		
notre approche [Loua 14]	+	+	+	+	-	+	+	+	+	+	+		

TABLE 4.1 – Tableau comparatif des approches de découverte et de sélection de services basées sur la confiance

4.4 Confiance en la crédibilité sociale

La confiance en la crédibilité sociale $SC(a_k, a_j)$ évalue le degré de sociabilité d'un agent a_j selon le point de vue d'un agent a_k . Elle est calculée sur la base des informations extraites du Réseau Social Multi-Relationnel (RSMR) (voir Sect. 2.4) du demandeur de services. Les informations extraites comportent la structure du graphe, les profils des agents contenant des données personnelles et des données sur leurs intérêts, la proximité sociale entre les agents et l'identification de tous les types de relations entre eux. Basées sur l'analyse du RSMR du demandeur de service et les informations extraites, trois mesures sont calculées comme indiqué dans la figure 4.1 : la position sociale (SPo), la proximité sociale (SPr) et la similarité sociale (SSi) qui est une agrégation de deux mesures ; la similarité de voisinage (NS) et la similarité de profil (PS). Nous croyons que la similitude entre deux agents ne se limite pas à la similarité de leurs profils, mais va au-delà et peut également inclure la similarité de leurs voisinages. Dans cette section, nous décrivons formellement les mesures de la position sociale, la proximité sociale et les deux mesures de la similarité sociale. Chaque mesure est illustrée par un exemple d'application.

4.4.1 La position sociale (SPo)

La position sociale d'un agent a_j est calculée habituellement (voir [Bans 10]) en fonction de son degré de centralité afin de donner une indication de sa puissance sociale dans le réseau social de l'agent a_k . Dans le contexte d'un réseau social multi-relationnel, elle dépend aussi des types de relations reliant a_j à ses accointances.

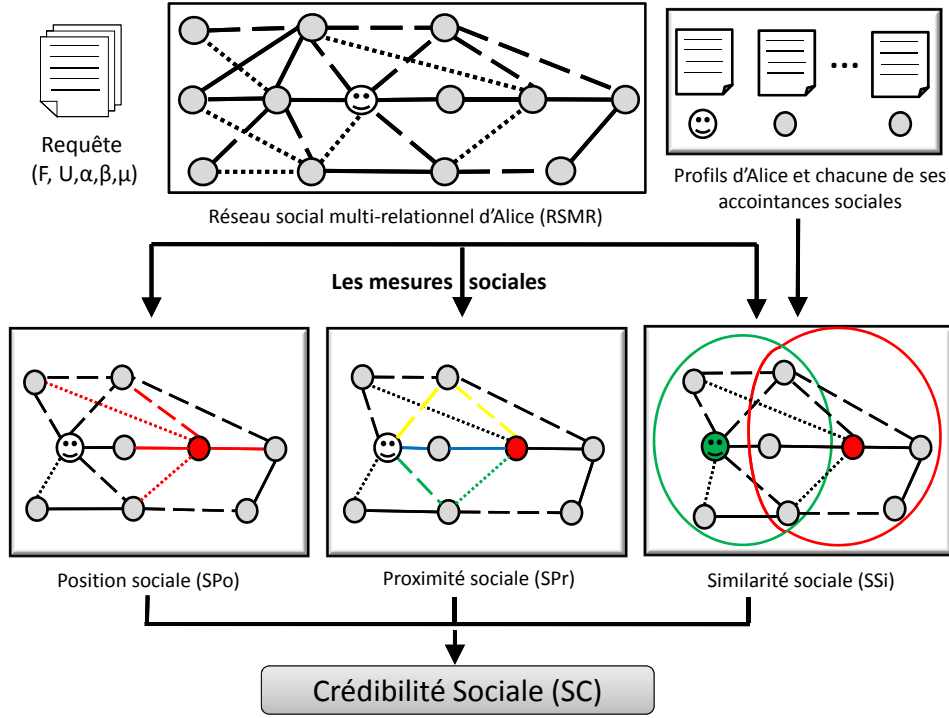


FIGURE 4.1 – Processus de calcul des mesures de la confiance en la crédibilité sociale

Definition 4.1 Soit $V = \{a_1, a_2, \dots, a_n\}$ un ensemble d'agents et soit $R = \{R_1, R_2, \dots, R_r\}$ un ensemble de types de relations, $G = \langle V, E_1, E_2, \dots, E_r : E_i \subseteq V \times V \forall i \in \{1, \dots, r\} \rangle$ est un graphe qui représente un RSMR où E_i est l'ensemble d'arêtes de type R_i . Soit $\rho : E \mapsto R$ une fonction qui associe à chaque arête la relation qu'elle représente, i.e., une arête $(a_{l-1}, a_l) \in E_i$ représente une relation sociale de type R_i entre a_{l-1} et a_l . Soit $U(\rho((a_{l-1}, a_l)))$ le coût de l'arête (a_{l-1}, a_l) selon les préférences du demandeur de services. Soit

$$b^i(a_j, a_l) = \begin{cases} 1 & \text{si } (a_k, a_j) \in E_i \\ 0 & \text{sinon} \end{cases}$$

La position sociale d'un agent a_j est définie comme suit :

$$SPo(a_j) = \sum_{i=1}^r \sum_{a_l \in SA_j} U(\rho((a_j, a_l))) \cdot b^i(a_j, a_l) \quad (4.1)$$

Pour illustrer cette mesure, considérons le graphe G du réseau social multi-relational de la figure 4.2. Les agents dans G sont liés par trois types de relations : R_1 représentant la relation de famille, R_2 représentant la relation de partenariat, et R_3 représentant

la relation d'une connaissance publique. Supposons que les préférences de a_0 , l'agent demandeur, sont telles que $U(R_1) = 1, U(R_2) = \frac{1}{2}, U(R_3) = \frac{1}{4}$.

Exemple 4.2 Nous souhaitons calculer la position sociale des fournisseurs a_4 et a_8 dans le graphe G . Dans ce graphe, tous les deux ont 6 relations. Cependant, en considérant l'aspect sémantique, l'agent a_4 possède 2 relations de type famille, 1 relation de type partenariat, et 3 relations de type connaissance publique. L'agent a_8 a 3 relations de type famille, 1 relation de type partenariat, et 2 relations de type connaissance publique. La position sociale de l'agent a_4 est donc $SPo(a_4) = 1 \times 2 + \frac{1}{2} \times 1 + \frac{1}{4} \times 3 = 3,25$ et la position sociale de l'agent a_8 est $SPo(a_8) = 1 \times 3 + \frac{1}{2} \times 1 + \frac{1}{4} \times 2 = 4$. À l'issue de ces résultats, l'agent a_8 a une position sociale supérieure à celle de a_4 .

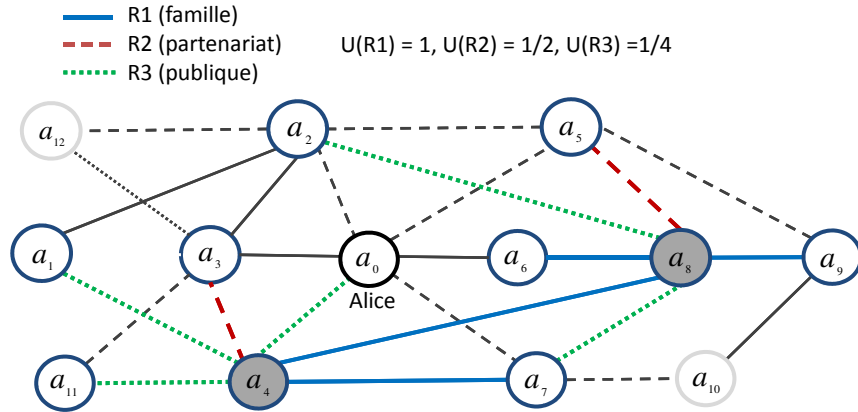


FIGURE 4.2 – Position sociale de l'agent a_4 et l'agent a_8 dans le graphe G

Un agent avec une position sociale élevée détient une position structurelle importante dans le réseau car il a un potentiel élevé pour faire circuler l'information [Sier 09], par sa forte connectivité avec les autres agents du réseau. Socialement, ces agents suscitent la confiance des agents qui préfèrent interagir avec eux [Barb 14]. Ainsi, nous pensons que les agents de forte position sociale ont une position de « leader » dans le réseau et sont considérés comme dignes de confiance.

4.4.2 La proximité sociale (SPr)

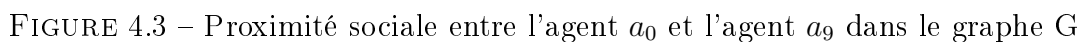
La proximité sociale est une distance métrique entre deux agents dans le graphe. Étant donné que notre réseau social est multi-relationnel, les relations formant le chemin séparant les deux agents dans le graphe peuvent être de différents types. Nous utilisons cette mesure pour calculer le coût moyen du chemin séparant l'agent a_k d'un agent a_j .

Definition 4.3 Soit $path = (a_k, \dots, a_j)$ un chemin de longueur d séparant l'agent a_k d'un agent a_j . Soit $U(\rho((a_{l-1}, a_l)))$ le coût de l'arête $(a_{l-1}, a_l) \in path$ selon les préférences du demandeur de services. La proximité sociale entre un agent a_k et un agent a_j est définie par :

$$SPr(a_k, a_j) = \frac{\sum_{l=1}^d U(\rho((a_{l-1}, a_l)))}{d} \quad (4.2)$$

Dans un réseau social mono-relationnel, le plus court chemin entre deux agents est le meilleur chemin pour la proximité sociale [Maar 10]. Cependant, dans notre cas, le plus court chemin n'est pas forcément le meilleur chemin si nous incluons l'aspect sémantique illustré par les différents types de relations. En effet, nous pensons que la proximité entre deux agents dépend non seulement de la longueur du chemin qui les sépare mais aussi des types de relations qui composent ce chemin. Dans certains cas, un chemin plus long peut avoir un meilleur coût s'il contient des relations préférées par l'utilisateur comme le montre cet exemple.

Exemple 4.4 Considérons le graphe G du RSMR de la figure 4.3 et les préférences de l'agent demandeur a_0 pour les types de relations tels que $U(R_1) = 1, U(R_2) = \frac{1}{2}, U(R_3) = \frac{1}{4}$. Nous voulons calculer la proximité sociale entre l'agent a_0 et l'agent a_9 . Plusieurs chemins lient ces deux agents a_0 et a_9 : $path_1 = (a_0, a_6, a_8, a_9)$, $path_2 = (a_0, a_5, a_8, a_9)$, $path_3 = (a_0, a_5, a_9)$, $path_4 = (a_0, a_7, a_8, a_9)$, and $path_5 = (a_0, a_7, a_{10}, a_9)$. Si nous ne prenons pas en compte les préférences de l'agent demandeur, il est clair que le meilleur chemin est $path_3$, qui est le chemin le plus court de longueur 2. Maintenant, si nous intégrons les préférences de l'utilisateur pour les types de relations, le coût du chemin $path_3$ est $1 = \frac{1}{2} + \frac{1}{2}$, et sa proximité sociale est $\frac{1}{2}$. Cependant, il existe des chemins de longueur 3 avec une meilleure proximité sociale : le coût du chemin $path_1$ est $3 = 1 + 1 + 1$ et sa proximité sociale est 1, c'est le meilleur chemin.



La similarité sociale entre deux agents est calculée à partir de la comparaison de leurs profils et de leurs voisinages respectifs dans le RSMR. $SSi(a_k, a_j)$ est une agrégation de deux mesures : similarité de voisinage (NS) et similarité de profil (PS).

Comme l’ont montré les études réalisées par Hang et Singh dans [Hang 10], la structure des liens dans un réseau social (par exemple, le nombre de voisins communs) influe sur la génération des recommandations dignes de confiance. Plus le nombre de voisins communs entre deux agents est élevé, plus le degré d’intimité sociale entre eux est important. C’est le cas de Facebook qui recommande de nouvelles connaissances en fonction du nombre d’amis communs entre les individus. De plus, l’un des principes fondamentaux de la psychologie sociale [Rowl 79] est qu’un agent peut faire plus confiance aux agents avec lesquels il a des relations sociales fortes que ceux avec qui il a

des relations sociales faibles. En se basant sur ce principe, nous définissons une mesure appelée la similarité de voisinage qui analyse la structure du graphe afin de déterminer la similarité entre les agents en fonction de la comparaison de leurs relations sociales.

Definition 4.5 Pour une paire d'agents (a_k, a_j) et un type de relation $R_i \in R$ soit :

- $x_i = |N_{R_i}(a_k) \cap N_{R_i}(a_j)|$ le nombre d'agents en commun dans les deux voisinages
- $y_i = |N_{R_i}(a_k)| - x_i$ le nombre d'agents dans le voisinage de a_k à l'exception des agents en commun
- $z_i = |N_{R_i}(a_j)| - x_i$ le nombre d'agents dans le voisinage de a_j à l'exception des agents en commun

Soit $U(\rho((a_{l-1}, a_l)))$ le coût de l'arête (a_{l-1}, a_l) selon les préférences du demandeur de services. Soit $jac^i = \frac{y_i + z_i}{x_i + y_i + z_i}$ la distance de Jaccard [Kauf 05] et soit $\delta^i(a_k, a_j) = \frac{1}{1 + jac^i}$ la similarité entre a_k et a_j selon la relation R_i . La similarité de voisinage, $NS(a_k, a_j)$, est calculée comme suit :

$$NS(a_k, a_j) = \sum_{i=1}^{|R|} U(\rho((a_{l-1}, a_l))) \cdot \delta^i(a_k, a_j) \quad (4.3)$$

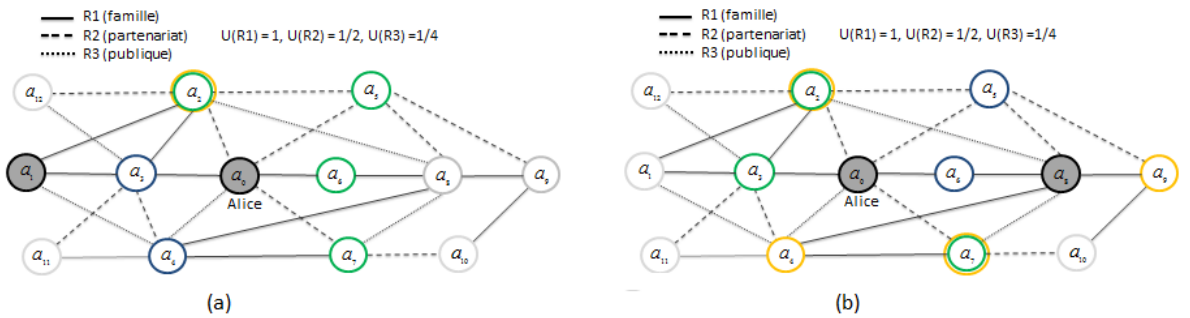


FIGURE 4.4 – Similarité de voisinage entre l'agent a_0 et les agents a_1 et a_8 dans le graphe G

Exemple 4.6 Considérons le graphe G du RSMR de la figure 4.4 dans lequel nous allons comparer la similarité de voisinage entre l'agent demandeur a_0 et les agents a_1 et

a_8 . Les ensembles de voisinages de a_0 selon les trois types de relations sont : $N_{R_1}(a_0) = \{a_3, a_6\}$, $N_{R_2}(a_0) = \{a_2, a_5, a_7\}$, $N_{R_3}(a_0) = \{a_4\}$. Les ensembles de voisinages de a_1 sont $N_{R_1}(a_1) = \{a_2, a_3\}$, $N_{R_2}(a_1) = \emptyset$, $N_{R_3}(a_1) = \{a_4\}$, et ceux de a_8 sont $N_{R_1}(a_8) = \{a_4, a_6, a_9\}$, $N_{R_2}(a_8) = \{a_5\}$, $N_{R_3}(a_8) = \{a_2, a_7\}$. Pour la paire d'agents (a_0, a_1) dans la figure 4.4(a), les distances de Jaccard sont égales à $dist^1 = \frac{2}{3}$, $dist^2 = \frac{3}{3}$, et $dist^3 = \frac{0}{1}$. Pour la paire d'agents (a_0, a_8) dans la figure 4.4(b), les distances de Jaccard sont égales à $dist^1 = \frac{3}{4}$, $dist^2 = \frac{2}{3}$, et $dist^3 = \frac{3}{3}$. En considérant les mêmes préférences pour les types de relations ($U(R_1) = 1, U(R_2) = \frac{1}{2}, U(R_3) = \frac{1}{4}$), la similarité de voisinage entre l'agent a_0 et l'agent a_1 est $NS(a_0, a_1) = 1 \times \frac{1}{1+\frac{2}{3}} + \frac{1}{2} \times \frac{1}{1+\frac{3}{3}} + \frac{1}{4} \times \frac{1}{1+\frac{0}{1}} = 1, 1$. La similarité de voisinage entre l'agent a_0 et l'agent a_8 est $NS(a_0, a_8) = 1 \times \frac{1}{1+\frac{3}{4}} + \frac{1}{2} \times \frac{1}{1+\frac{2}{3}} + \frac{1}{4} \times \frac{1}{1+\frac{3}{3}} = 0, 9$. Ces résultats montrent que le voisinage de l'agent a_0 est plus similaire à celui de l'agent a_1 qu'à celui de l'agent a_8 . Bien que les deux agents a_1 et a_8 ont le même nombre de voisins communs (i.e., 2) avec l'agent a_0 , ce nombre ne représente pas la même proportion par rapport à leur nombre total de voisins (pour l'agent a_1 ce nombre est égal à 2 parmi 3 voisins or pour l'agent a_8 il est égal à 2 parmi 6 voisins).

La similarité de profil (PS)

Dans les réseaux sociaux, le profil d'un agent n'est pas caractérisé uniquement par l'ensemble de ses accointances sociales, mais aussi par un ensemble de données décrivant ses informations personnelles (par exemple genre, adresse, travail ...) et ses centres d'intérêt (par exemple jeux, musique, films, ...). Un profil se décompose en un ensemble d'items structurés en un ensemble de champs. Un champ est représenté par un attribut nominal (également connu sous le nom d'attribut qualitatif) contenant une ou plusieurs valeurs (par exemple le genre = [femme], la musique préférée = [folk, jazz, pop]). Dans le premier cas, le champ est monovalué, dans le second cas, il est multivalué. Formellement, un profil peut être exprimé sous forme d'une grammaire en notation BNF (Backus Naur Form) :

- $\langle P \rangle ::= \langle item \rangle^+$
- $\langle item \rangle ::= \langle champ \rangle^+$
- $\langle champ \rangle ::= \langle valeur \rangle^*$

La liaison entre la confiance et la similarité de profil a été démontrée par Ziegler et al. dans [Zieg 04, Zieg 07]. À l'aide d'une étude expérimentale, ils ont prouvé l'existence d'une corrélation significative entre la confiance exprimée par les agents à l'aide

des recommandations générées dans le réseau social et la description de leurs centres d'intérêt respectifs. Plus la confiance entre deux agents dans un contexte donné est grande, plus ces agents sont similaires dans ce contexte. Prenons l'exemple d'une recherche documentaire, Alice, l'enseignante chercheuse, accorde plus de confiance aux propositions d'articles faites par un collègue de son domaine qu'à celles faites par un ami ou membre de la famille. Sur la base de ces résultats, nous définissons une mesure appelée la similarité de profil qui consiste à comparer les valeurs des champs de profils de deux agents afin de déterminer à quel point ces agents sont similaires.

Boriah et al. ont présenté dans [Bori 08] une évaluation de performance de 14 mesures de similarité opérant sur les données nominales. Pour ce faire, ils ont identifié les caractéristiques des données nominales puis, ont classé les mesures existantes en trois catégories en fonction de la façon dont elles utilisent les informations dans ces données. À travers cette évaluation basée sur la détection des valeurs aberrantes, ils ont montré l'utilité de chaque mesure en fonction de la description des données utilisées (la taille des données, la nombre d'attributs, la distribution du nombre de valeurs prises par chaque attribut...). Parmi les mesures évoquées, nous avons choisi la mesure de Burnaby [Burn 70] pour évaluer la similarité de profil. Ce choix de mesure est motivé par les raisons suivantes :

- Elle prend en compte la distribution de la fréquence d'apparition des différentes valeurs d'attributs dans l'ensemble de données.
- Elle possède une des meilleurs performances en terme de détection des valeurs aberrantes.

Il s'agit d'une mesure probabiliste utilisée à la fois pour l'égalité et l'inégalité d'une paire de valeur (X_k, Y_k) . Dans le cas d'une inégalité $(X_k \neq Y_k)$, la valeur de similarité attribuée n'est pas nulle. Celle-ci est calculée en fonction de la distribution de la fréquence d'apparition des autres valeurs de l'attribut dans le jeu de données. Une plus grande valeur de similarité est accordée lorsque cette inégalité se produit entre des valeurs rares. Plus précisément, la mesure de Burnaby est utilisée pour évaluer la similarité entre les champs monovalué de deux profils comme indiqué par la définition suivante.

Définition 4.7 (Burnaby) *Soit i un item d'un profil composé d'un ensemble Fd de champs. Soit X_r et Y_r les valeurs du r -ème champ monovalué Fd_r respectivement dans les profils d'un agent a_k et un agent a_j de son ensemble d'acointances sociales*

($a_j \in SA_k$). La valeur de la similarité des champs monovalués est définie par :

$$\text{Burnaby}(X_r, Y_r) = \begin{cases} 1 & \text{si } X_r = Y_r \\ \frac{\sum_{q \in \mathcal{A}_r} 2 \log(1 - \hat{p}_r(q))}{\log \frac{\hat{p}_r(X_r) \hat{p}_r(Y_r)}{(1 - \hat{p}_r(X_r))(1 - \hat{p}_r(Y_r))} + \sum_{q \in \mathcal{A}_r} 2 \log(1 - \hat{p}_r(q))} & \text{sinon} \end{cases} \quad (4.4)$$

avec

$$\hat{p}_r(x) = \frac{f_r(x)}{N}$$

où $\mathcal{A}_r$ désigne l'ensemble de toutes les valeurs possibles du champ Fd_r ; N est le nombre total d'acointances de l'agent a_k ($N = |SA_k|$) ; $f_r(x)$ est la distribution de la fréquence d'apparition des valeurs prises par un champ (i.e., le nombre de fois où le champ Fd_r prend la valeur x), et $\hat{p}_r(x) = \frac{f_r(x)}{N}$ est la probabilité qu'un champ Fd_r prend la valeur x .

Comme mentionné ci-dessus, un item est structuré en un ensemble de champs qui peuvent être multivalués. Inspiré de [Akco 11], nous présentons dans un premier temps une méthode de calcul de la similarité entre les items, puis nous introduirons ensuite la similarité entre les profils.

Définition 4.8 (Similarité d'items) Soit i un item d'un profil composé d'un ensemble Fd de champs. Soit $\mathcal{V}(Fd_r^{a_k})$ et $\mathcal{V}(Fd_r^{a_j})$ les valeurs prise par un champ multivalué Fd_r de l'item i dans les profils de a_k et a_j respectivement. Soit $B_r = \{\text{Burnaby}(X_{rm}, Y_{rn}) ; \forall X_{rm} \in \mathcal{V}(Fd_r^{a_k}) \text{ and } \forall Y_{rn} \in \mathcal{V}(Fd_r^{a_j})\}$ l'ensemble des valeurs de la similarité des champs monovalués calculées entre toutes les paires possibles de $\mathcal{V}(Fd_r^{a_k})$ et $\mathcal{V}(Fd_r^{a_j})$. Soit $\text{Max}B_r = \{bur_l \in B_r \mid \forall l, 1 \leq l \leq |\mathcal{V}(Fd_r^{a_k})|, bur_l = \text{Burnaby}(X_{rm}, Y_{rn})\}$ l'ensemble des $|\mathcal{V}(Fd_r^{a_k})|$ plus grandes valeurs dans B_r . La similarité entre les i -ème items de a_k et a_j est définie par :

$$S_i(a_k, a_j) = \frac{1}{|Fd|} \times \sum_{r=1}^{|Fd|} \frac{1}{|\mathcal{V}(Fd_r^{a_k})|} \sum_{l=1}^{|\text{Max}B_r|} bur_l \quad (4.5)$$

Dans l'exemple suivant (voir figure 4.5), nous illustrons la similarité des items.

Exemple 4.9 Soit l'item emploi composé des champs $\{\text{employeur}, \text{type}, \text{poste}\}$ pour la description de l'emploi dans les profils. Les valeurs des champs de l'agent demandeur a_0 (Alice) et une accointance a_1 (Bob dans cet exemple) sont données dans la

 Alice	Info générales Genre: F Date de naissance: 12/11/2012 Adresse: Paris Langues: Français, Anglais Emploi Employeur: societe1 Type: developpement Poste: ingénieur Centre d'intérêt lecture, voyage, sport
 Bob	Info générales Genre: M Date de naissance: 15/11/2012 Adresse: Paris Langues: Français, Chinois Emploi Employeur: societe1 Type: consultant Poste: manager Centre d'intérêt Jeux vidéo, films, football

FIGURE 4.5 – Profils de deux agents dans le réseau social

figure 4.5. Supposons que l'agent demandeur a_0 a 300 accointances dans son SA_0 (Dans les mesures précédentes, nous n'avons montré qu'une partie du réseau social d'Alice pour faciliter la visualisation), dans lesquels les valeurs des champs sont réparties comme montré dans le tableau 4.2. Puisque dans tous les profils, chaque item emp est composé de 3 champs, $\{employeur, type, poste\}$, alors $|Fd|$ est égale à 3. Tous les champs sont monovalués, alors $|\mathcal{V}(Fd_r^{a_0})| = 1$ (pour tout r). Pour le champ employeur, $\mathcal{V}(Fd_{employeur}^{a_0}) = societe1$ et $\mathcal{V}(Fd_{employeur}^{a_1}) = societe1$. Pour le champ type, $\mathcal{V}(Fd_{type}^{a_0}) = developpement$ et $\mathcal{V}(Fd_{type}^{a_1}) = consultant$. Pour le champ poste, $\mathcal{V}(Fd_{poste}^{a_0}) = ingénieur$ and $\mathcal{V}(Fd_{poste}^{a_1}) = manager$. Par conséquent, $B_{employeur} = Burnaby(societe1, societe1)$, $B_{type} = Burnaby(developpement, consultant)$, et $B_{poste} = Burnaby(ingénieur, manager)$. Puisque $MaxB_r$ est l'ensemble des $|\mathcal{V}(Fd_r^{a_0})|$ plus grandes valeurs dans B_r et $|\mathcal{V}(Fd_r^{a_0})| = 1$ (pour tout r), alors $MaxB_{employeur} = B_{employeur}$, $MaxB_{type} = B_{type}$, and $MaxB_{poste} = B_{poste}$. Ainsi, la similarité entre les items emp dans les profils de a_0 et a_1 est égale à :

$$S_{emp}(a_0, a_1) = \frac{1}{3} \times \left[\frac{1}{1} \times Burnaby(societe1, societe1) + \frac{1}{1} \times Burnaby(developpement, consultant) + \frac{1}{1} \times Burnaby(ingénieur, manager) \right].$$

Selon la définition 4.7, nous avons :

$$Burnaby(societe1, societe1) = 1$$

$$Burnaby(developpement, consultant) = \frac{2(\log(1 - \frac{120}{300}) + \log(1 - \frac{100}{300}) + \log(1 - \frac{80}{300}))}{\log(\frac{120}{300} * \frac{100}{300}) + 2(\log(1 - \frac{120}{300}) + \log(1 - \frac{100}{300}) + \log(1 - \frac{80}{300}))}$$

Emploi	300 profils			
Employeur	societe1 80	societe2 100	societe3 60	societe4 60
Type	developpement 120	consultant 100	audit 80	
Poste	ingenieur 130	manager 40	expert 30	assistant 100

TABLE 4.2 – Répartition des valeurs des champs de l'item "emploi" dans les profils des accointances d'Alice

$Burnaby(ingenieur, manager) = \frac{2(\log(1-\frac{130}{300})+\log(1-\frac{40}{300})+\log(1-\frac{30}{300})+\log(1-\frac{100}{300}))}{\log(\frac{\frac{130}{300} * \frac{40}{300}}{(1-\frac{130}{300})(1-\frac{40}{300})})+2(\log(1-\frac{130}{300})+\log(1-\frac{40}{300})+\log(1-\frac{30}{300})+\log(1-\frac{100}{300}))}$

Ainsi, $S_{emp}(a_0, a_1) = \frac{1}{3} \times [\frac{1}{1} \times 1 + \frac{1}{1} \times 0,69 + \frac{1}{1} \times 0,53] = 0,74$. Dans ce scénario, bien qu'il n'y ait qu'une seule valeur identique, la mesure de similarité est égal à 0,74.

Definition 4.10 Soit I l'ensemble des items dans chacun des deux profils. Soit β_i le poids attribué à chaque item i reflétant son importance dans la description du profil. On définit la similarité de profil, $PS(a_k, a_j)$ comme suit :

$$PS(a_k, a_j) = \frac{1}{|I|} \times \sum_{i \in I} \beta_i \cdot S_i(a_k, a_j) \quad (4.6)$$

Étant donné que $NS \in [0, 1]$ et $PS \in [0, 1]$, la similarité sociale est définie par leur produit :

$$SSi(a_k, a_j) = NS(a_k, a_j) \times PS(a_k, a_j) \quad (4.7)$$

4.4.4 Calcul de la confiance en la crédibilité sociale

Soit M_j un vecteur associé à chaque accointance $a_j \in SA_k$ tel que $M_j = (SPo(a_k), SPr(a_k, a_j), SSi(a_k, a_j))$. Soit $M = (M_{jt}, a_j \in SA_k \text{ avec } 1 \leq t \leq 3)$ une matrice dans laquelle chaque ligne M_j représente une accointance a_j et chaque colonne représente une mesure sociale (i.e., la première colonne pour SPo , la deuxième colonne pour SPr , et la troisième colonne pour SSi). Pour calculer la valeur de la confiance en la crédibilité sociale de chaque accointance a_j , nous utilisons la technique de pondération additive qui procède en deux phases :

- Phase de mise à l'échelle : elle vise à standardiser chaque valeur du vecteur $M_j \in [0, 1]$. Plus les valeurs sont élevées, plus la qualité est importante. Soit M_t^{max} la valeur maximale d'une mesure sociale (SPo , SPr ou SSi) dans la matrice M , i.e., $M_t^{max} = \max(M_{jt}), \forall a_j$. Soit M_t^{min} la valeur minimale d'une mesure sociale dans la matrice M , i.e., $M_t^{min} = \min(M_{jt}), \forall a_j$. Pour des mesures positives, les valeurs M_{jt} sont échelonnées en fonction de l'équation suivante.

$$M'_{jt} = \begin{cases} \frac{M_{jt} - M_t^{min}}{M_t^{max} - M_t^{min}} & \text{if } M_t^{max} - M_t^{min} \neq 0 \\ 1 & \text{if } M_t^{max} - M_t^{min} = 0 \end{cases} \quad (4.8)$$

En appliquant cette équation (Equa. 4.17) pour M , nous obtenons une nouvelle matrice $M' = (M'_{jt}, a_j \in SA_k \text{ and } 1 \leq t \leq 3)$, dans laquelle chaque ligne M'_j contient les valeurs standardisées des différentes mesures sociales d'une accointance a_j .

- Phase de pondération : elle vise à attribuer à chaque agent $a_j \in SA_k$ une valeur de confiance sociale. Soit $\lambda_t \in [0, 1]$ le poids de la t -ème mesure sociale tel que $\sum_{t=1}^3 \lambda_t = 1$. Le score global de la confiance sociale que l'agent a_k a envers un autre agent a_j , noté $SC(a_k, a_j)$, est donné par :

$$SC(a_k, a_j) = \sum_{t=1}^3 \lambda_t \cdot M'_{jt}(a_k, a_j) \quad (4.9)$$

4.5 Confiance en l'expertise

La confiance en la crédibilité sociale d'un agent détermine si celui-ci est digne de confiance d'un point de vue social. Cependant, elle ne permet pas d'évaluer l'expertise de l'agent en d'autres termes, la fiabilité et la qualité d'exécution des services qu'il offre. Selon [Geor 50, Wang 13], la confiance en un agent est aussi bien liée à sa sociabilité qu'à son expertise selon le contexte d'interaction. Nous faisons par exemple confiance à la capacité d'un professeur pour faire une recherche documentaire, mais il n'a pas notre confiance dans l'enseignement de C++ puisque il n'est pas expert dans ce domaine. Dans un environnement orienté services, Al-Sharawneh a montré que lorsqu'un agent interagit avec un autre agent socialement digne de confiance il faut prendre en considération une part de risque concernant la qualité des services qu'il offre [Al S 12]. Il en résulte qu'un modèle de confiance établi sur la base de la seule composante sociale n'est pas suffisant pour réaliser une découverte et une sélection de services efficaces. Pour mettre en œuvre cette notion, nous introduisons une autre composante dans la

modélisation de confiance appelée la confiance en l'expertise EC qui traite l'aspect non fonctionnel (c'est à dire, la qualité de service QoS) des services.

Comme il est indiqué dans [Leti 08], nous pensons dans cette thèse que la crédibilité sociale et l'expertise d'un agent sont corrélées. Cela sera vérifié dans notre évaluation empirique (voir section 7.3.2). Un agent avec un score de crédibilité sociale élevé a de forte chance de détenir une forte expertise et réciproquement. Cela sera vérifié dans notre évaluation empirique (voir Sect. 7.3.2). À la différence de la confiance en la crédibilité sociale, la confiance en l'expertise est une mesure spécifique associée à un seul service en fonction du contexte de l'interaction, c'est à dire, la requête du demandeur de services. Afin d'établir la confiance en l'expertise, nous définissons les attributs de QoS, qui décrivent la performance d'un service et comment ils peuvent être mesurés. Dans la littérature, de nombreux attributs de QoS ont été identifiés [Lee 03, Ran 03]. Ceux-ci peuvent être classés en deux catégories :

- Attributs objectifs (e.g., temps de réponse, disponibilité) – qui peuvent être mesurés et évalués d'une façon exacte. Ce type d'attribut est quantifiable par une valeur numérique et mesuré à l'aide d'une métrique.
- Attributs subjectifs (e.g., la qualité d'image ou du son, retour d'expérience) – qui expriment l'opinion personnelle des utilisateurs reflétant leur degré de satisfaction du service utilisé. Tout aspect perceptible et évaluable par les utilisateurs peut être considéré comme un attribut subjectif de QoS.

Letia et Pop distinguent les attributs de qualité objectifs des attributs de qualité subjectifs [Leti 08]. Les premiers permettent d'automatiser le calcul des mesures de confiance et réduisent l'intervention des utilisateurs au contraire des seconds qui, de plus, sont plus vulnérables à la manipulation des agents malveillants.

Dans ce travail, nous nous intéressons à ces deux catégories d'attributs de qualité (objectif et subjectif). Plus particulièrement, comme [Lala 12] dans chacune de ces catégories, nous allons choisir des attributs qui peuvent avoir une influence directe sur la qualité perçue par l'utilisateur. La confiance en l'expertise $EC(a_r, a_j, s_{jl})$ que le demandeur de services a_r accorde à un service s_{jl} offert par un fournisseur a_j est définie comme étant l'agrégation des deux attributs de QoS objectifs : la *spécialisation* (Sp), la *disponibilité* (Di) et un attribut de QoS subjectif : la *qualité* ($Eval$). Dans les sections suivantes, nous définissons chacun de ces attributs et les exigences désirées (sensibilité temporelle, démarrage à froid) en se basant sur les recherches effectuées dans ce domaine.

4.5.1 La spécialisation (Sp)

Dans un environnement orienté services l'expertise d'un agent peut être vue comme une notion de *spécialisation*. Comme nos agents peuvent offrir plusieurs services à la fois, nous définissons la spécialisation d'un agent a_j pour un service s_{jl} comme étant le rapport entre le nombre de fois où le service s_{jl} a été exécuté avec succès [Maar 10] et le nombre total des exécutions de l'agent a_j indépendamment du service utilisé.

Definition 4.11 Soit $Nb_{exec}(s_{jl})$ le nombre d'exécutions réussies de s_{jl} . Soit m_j le nombre de services offerts par l'agent a_j . La spécialisation d'un agent a_j pour un service s_{jl} est définie par :

$$Sp(s_{jl}) = \frac{Nb_{exec}(s_{jl})}{\sum_{t=1}^{m_j} Nb_{exec}(s_{jt})} \quad (4.10)$$

Notons que $\sum_{t=1}^{m_j} Sp(s_{jt}) = 1$. Cela signifie que plus le service s_{jl} est sollicité dans le réseau social, plus a_j est reconnu comme un spécialiste de ce service.

4.5.2 La disponibilité (Di)

La disponibilité est l'un des critères de qualité les plus étudiés dans la littérature étant donné sa perception directe par les utilisateurs. En effet, comme le montre [Cotr 02], un service indisponible est un service aussi mal perçu par les consommateurs qu'un service de mauvaise qualité. La disponibilité d'un service délivré est définie comme étant la probabilité de répondre avec succès et correctement [The 90]. Pratiquement, un service web devrait être prêt et opérationnel (i.e., disponible) pour une utilisation immédiate. Il existe différents types de disponibilité et de nombreuses façons de la calculer [Elsa 96]. La majorité des approches existantes utilisent la disponibilité opérationnelle qui mesure la disponibilité moyenne sur une période de temps.

Dans ce travail, nous définissons la disponibilité d'un service s_{jl} comme étant sa capacité à remplir les fonctions requises au moment de l'invocation. Formellement, elle correspond au ratio du nombre de ses exécutions réussies sur le nombre total d'invocations.

Definition 4.12 Soit $Nb_{exec}(s_{jl})$ le nombre d'exécutions réussies de s_{jl} . Soit $Nb_{invoc}(s_{jl})$ le nombre total d'invocations de s_{jl} . La disponibilité d'un agent a_j pour un service s_{jl} est définie par :

$$Di(s_{jl}) = \frac{Nb_{exec}(s_{jl})}{Nb_{invoc}(s_{jl})} \quad (4.11)$$

La disponibilité peut être considérée comme un facteur de réussite d'un service [Sure 00] car elle garantit un minimum de bon fonctionnement au moment de l'invocation. Par conséquent, un service qui est souvent indisponible peut engendrer des effets négatifs sur la fiabilité de son fournisseur.

4.5.3 La qualité (Eval)

Contrairement aux attributs de QoS précédemment énoncés, l'attribut de qualité résulte d'une perception subjective fournie par l'utilisateur du service. En effet, une fois qu'un service s_{jl} est exécuté avec succès par le demandeur de services a_r , celui-ci attribue comme retour d'expérience une note $\nu \in [0, 1]$ à ce service. Chaque note représente un avis personnel reflétant son degré d'appréciation/regret concernant la qualité d'exécution du service. L'intérêt d'un tel attribut réside dans son caractère informel : en effet, de nombreux utilisateurs ne sont pas familiarisés avec les différents attributs de QoS et ne connaissent pas leur signification. Ils préfèrent cet attribut de qualité car il est explicite et facile à interpréter, ce qui les aide dans leur prise de décision lors de la sélection.

Habituellement, la qualité d'un service est estimée en fonction des notes communiquées par plusieurs sources [Bill 07, Huyn 04, Nard 08, Saba 02], c'est à dire provenant de tous les agents qui ont utilisé ce service.

Cependant, dans ce travail nous considérons la qualité d'un service d'un seul point de vue qui est celui du demandeur du service. Ce choix est justifié par la volonté de faire face aux manipulations et aux attaques (par exemple, les attaques Sybil²). Partant de l'hypothèse qu'un individu possède des traits de personnalité relativement stables dans une même situation, la note qu'il attribue alors à un service à la fin d'une transaction représente effectivement son retour d'expérience [Rott 80]. Si le même utilisateur fait appel aux mêmes services plusieurs fois, la valeur stockée par le système donnera une image plus précise de la fiabilité des fournisseurs. Il a donc intérêt à être honnête dans sa notation et ne pas surévaluer ou sous-évaluer le service utilisé car toutes ces notes seront exploitées dans le futur impactant ainsi sa prise de décision.

Dans ce travail, nous définissons la qualité d'un service s_{jl} à l'aide d'une mesure calculée auprès d'une seule source d'information. La valeur de la qualité, notée $Eval_n(a_r, s_{jl})$, est évaluée à partir des anciennes expériences individuelles de l'agent demandeur a_r en

2. L'attaque Sybil a tout d'abord été introduite et décrite par Douceur [Douc 02] dans le contexte des réseaux pair-à-pair. Dans cette attaque, l'agent malveillant manipule le système de réputation par la création d'un grand nombre d'identités pseudonymes qui fournissent des notes positives afin d'augmenter son niveau de confiance.

utilisant le service s_{jl} . Nous proposons deux solutions alternatives pour l'évaluation de la qualité globale d'un service :

- On peut considérer que les notes attribuées par le demandeur de services a_r ne dépendent pas du temps, autrement dit, les notes prises à des instants différents ont la même importance dans l'évaluation globale.

Définition 4.13 Soit $\{\nu_1, \nu_2, \dots, \nu_n\}$ l'ensemble des notes attribuées par a_r à s_{jl} après chaque utilisation réussie. Une façon courante d'estimer la valeur de qualité d'un service est de calculer la moyenne arithmétique de l'ensemble de ses notes comme suit :

$$Eval_n(a_r, s_{jl}) = \frac{1}{n} \sum_{i=1}^n \nu_i \quad (4.12)$$

- Comme les notes peuvent rapidement devenir obsolètes avec le temps, nous pouvons considérer la *récence* comme un paramètre temporel dans la formule de l'évaluation de la qualité. Dans ce cas, la valeur de la qualité est calculée comme une moyenne pondérée donnant plus d'importance aux notes récentes. Notre intuition est que les nouvelles notes sont censées mieux refléter la performance actuelle de l'agent que les anciennes notes.

Définition 4.14 Soit t_i l'instant de la prise de note ν_i . Soit t l'instant courant (i.e., l'instant de l'évaluation de la qualité). Soit $f(t_i, t)$ une fonction de pondération. La valeur de qualité d'un service s_{jl} selon a_r est définie par :

$$Eval_n(a_r, s_{jl}) = \frac{\sum_{i=1}^n \nu_i \cdot f(t_i, t)}{\sum_{i=1}^n f(t_i, t)} \quad (4.13)$$

La fonction de pondération $f(t_i, t)$ attribue un poids plus élevé aux notes récentes, étant donné qu'elles sont plus actualisées et probablement plus précises, que les anciennes notes. Un simple exemple de cette fonction de pondération est celui proposé par Sabater et Sierra dans [Saba 02] :

$$f(t_i, t) = \frac{t_i}{t} \quad (4.14)$$

Cependant, cette fonction, comme l'ont prouvé Huynh et Jennings dans [Huyn 06], présente quelques lacunes au niveau de l'influence des notes sur la prise de décision de l'agent. En fait, en développant la fonction 4.13 en utilisant la fonction 4.14, la fonction d'évaluation devient :

$$Eval_n(a_k, s_{jl}) = \frac{\sum_{i=1}^n \nu_i \cdot t_i}{\sum_{i=1}^n t_i} \quad (4.15)$$

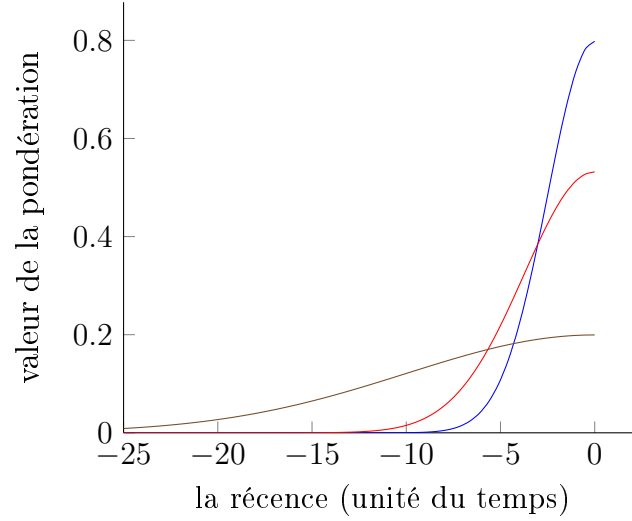


FIGURE 4.6 – Plusieurs fonctions de pondération pour différentes valeurs de σ [Leti 08]

Par conséquent, la qualité de service (Eval) dépend de l'ensemble de notes d'évaluation prises à des instants donnés et non pas de leur récence par rapport à l'instant courant t . Pour différents ensemble de notes, la fonction de pondération peut produire les mêmes poids indépendamment de l'instant courant t alors que les valeurs récentes doivent avoir plus d'influence sur la prise de décision de l'agent que les anciennes valeurs.

Pour soulever cette question, d'autres formulations ont été proposées dans la littérature [Huyn 06, Leti 08]. Dans [Leti 08], la fonction de pondération est une distribution gaussienne centrée (i.e., espérance nulle : $\mu = 0$) :

$$f(t_i) = \frac{1}{\sigma\sqrt{(2\pi)}} \cdot \exp\left(-\frac{t_i^2}{2\sigma^2}\right) \quad (4.16)$$

L'instant courant correspond à $t_i = 0$. Si l'on veut tenir compte des notes plus anciennes on augmente la valeur de l'écart type σ comme le montre la figure 4.6. Cependant, cette fonction présente quelques lacunes en matière de contrôle de la granularité de temps. Nous devons ajuster la fonction par rapport à l'unité de temps de l'application.

La fonction de pondération proposée dans [Huyn 06] est une fonction qui se base sur la différence temporelle entre l'instant courant et l'instant de la prise de note. Afin de rendre la fonction de pondération réglable par rapport à la granularité temporelle de l'application, les auteurs ont introduit dans la formulation un pa-

ramètre λ , appelé facteur de mise à l'échelle de la récence.

$$f(t_i, t) = \exp\left(-\frac{\Delta(t, t_i)}{\lambda}\right) \quad (4.17)$$

Parmi les fonctions évoquées, nous avons choisi la fonction 4.17 de [Huyn 06] pour évaluer l'importance de chaque note. Le choix de cette fonction est motivé par deux raisons : son indépendance par rapport à l'unité de temps et sa capacité à donner plus d'influence aux notes récentes qu'aux notes plus anciennes.

Une autre question importante que nous devons également prendre en considération dans l'évaluation de la qualité d'un service est *le démarrage à froid*. Pour ce faire, nous allons introduire dans le système une phase d'amorçage. L'intérêt de cette phase est d'améliorer la robustesse du modèle en initialisant les attributs de QoS des nouveaux services par des valeurs non nulles. En s'inspirant de [Huan 14], nous proposons deux stratégies d'amorçage :

- Amorçage par défaut : Cette stratégie consiste à attribuer à un service une valeur de qualité par défaut. Cette valeur varie entre 0 et 1. Si la valeur initiale attribuée est faible, cette stratégie s'est révélée être pénalisante ou négative. Cependant, si la valeur initiale attribuée est élevée, cette stratégie d'amorçage s'est révélée être positive.
- Amorçage adaptatif : Cette stratégie calcule la valeur de qualité initiale d'un service en se basant sur la moyenne des valeurs de qualité des autres services offerts par le même agent.

4.5.4 Calcul de la confiance en l'expertise

La confiance en l'expertise, notée $EC(a_r, a_j, s_{jl})$, désigne le niveau de confiance que le demandeur de services a_r accorde à un service s_{jl} offert par un fournisseur a_j agréant les trois attributs de QoS mentionnés précédemment :

$$EC(a_r, a_j, s_{jl}) = Sp(s_{jl}) \times Di(s_{jl}) \times Eval(a_r, s_{jl}) \quad (4.18)$$

Notons que $EC \in [0, 1]$ puisque $Sp \in [0, 1]$, $Di \in [0, 1]$ et $Eval \in [0, 1]$.

4.6 Confiance en la recommandation

Les techniques classiques de recommandation fondées sur la confiance évaluent la capacité de recommandation des individus et non pas la qualité de leurs recommandations en fonction du contexte de l'interaction. Cependant, selon Golbeck [Golb 06a], deux types de confiance sont nécessaires pour établir une interaction contextuelle entre deux individus : la confiance en la capacité de recommandation de l'individu et la confiance en la qualité de ses recommandations. En outre, selon l'étude réalisée par [Tang 12] dans les réseaux sociaux, il est rare qu'un individu accorde une confiance totale à la capacité de recommandation d'un autre et ce, dans tous les domaines car la qualité de recommandation de ce dernier peut varier d'un domaine à un autre [Adle 01, Wang 07a].

Par ailleurs, dans les environnements orientés services distribués, les agents sont dotés d'une perception et d'une connaissance réduites. Pour cette raison, ils peuvent demander de l'aide de leurs accointances. Ces dernières peuvent fournir comme réponse des recommandations associées à des services ou à des références pertinentes [Sing 01, Yu 03]. Cependant, aucune évaluation n'est effectuée dans ces travaux concernant la qualité des recommandations présentées. Un agent socialement digne de confiance n'est pas nécessairement un bon référent et réciproquement.

C'est pour cela que nous estimons qu'il est judicieux de vérifier l'aptitude du recommandeur à fournir de bonnes recommandations dans un contexte défini avant de les recevoir. Pour ce faire, nous intégrons une autre composante de confiance dans notre modèle appelée la confiance en la recommandation (RC) qui évalue la qualité des recommandations faites par un agent a_j à un agent a_k en fonction du contexte d'interaction (c'est à dire, la requête du demandeur de services).

La capacité d'un agent a_j à générer de bonnes recommandations repose sur son comportement antérieur. Dans les systèmes de recommandation conventionnels basés sur la confiance [Golb 05, Hess 08, Mass 07], un agent a_k évalue la confiance en la recommandation d'un agent a_j en agrégeant des avis et des opinions concernant la qualité de recommandation de ce dernier, reçus de la part de plusieurs agents de son réseau de confiance. Nous optons pour une approche d'évaluation plus *personnalisée* dans laquelle nous prenons en compte uniquement les avis de l'agent a_k concernant la qualité des recommandations de a_j . L'intérêt de cette restriction est de réduire les risques de subjectivité des autres agents. Nous décomposons la valeur de la confiance en la recommandation, notée $RC(a_k, a_j, s_{pl})$, que l'agent a_k accorde à l'agent a_j concernant sa qualité de recommandation d'un service s_{pl} proposé par un fournisseur a_p en deux parties :

- **Une partie objective** ($[r_{kj}|s_{pl}] \in [0, 1]$) qui indique la performance de a_j en

termes du nombre de recommandations sélectionnées par a_k concernant s_{pl} .

- **Une partie subjective** ($[q_{kj}|s_{pl}] \in [0, 1]$) qui reflète la satisfaction de l'agent a_k pour les recommandations de s_{pl} fournies par a_j .

Les informations nécessaires pour le calcul de ces deux parties sont stockées dans la base de croyance de l'agent a_k et plus précisément dans la table HT_k (voir Sect. 5.2.1). Les hypothèses de base de RC sont :

- 1) $RC \in [0, 1]$.
- 2) $RC = 1$ si $[r_{kj}|s_{pl}] = [q_{kj}|s_{pl}] = 0$ ou si $[r_{kj}|s_{pl}] = [q_{kj}|s_{pl}] = 1$.
- 3) $RC = [r_{kj}|s_{pl}]$ si $[q_{kj}|s_{pl}] = 1$.

D'autres propriétés sont également assurées. RC augmente plus avec l'augmentation de $[r_{kj}|s_{pl}]$ qu'avec celle de $[q_{kj}|s_{pl}]$. C'est une caractéristique désirable puisque l'importance de l'évaluation doit porter sur la satisfaction de a_k plutôt que sur le nombre des recommandations sélectionnées. Dans certains cas, a_k peut rejeter la recommandation donnée par a_j non parce que sa qualité est mauvaise mais parce qu'il a sélectionné un autre service fonctionnellement équivalent. Par conséquent, il est raisonnable d'accorder plus d'importance au critère de satisfaction qu'à celui de sélection. En se basant sur la formule proposée par Chen et Singh dans [Chen 01] et en respectant les hypothèses énoncées ci-dessus, nous définissons RC comme suit :

Definition 4.15

$$RC(a_k, a_j, s_{pl}) = \begin{cases} 1 & \text{si } [r_{kj}|s_{pl}] = 0 \text{ ou } [q_{kj}|s_{pl}] = 0 \\ ([r_{kj}|s_{pl}] + 1)^{[q_{kj}|s_{pl}]} - 1 & \text{sinon} \end{cases} \quad (4.19)$$

Comme mentionné ci-dessus, $[r_{kj}|s_{pl}]$ représente le taux de la sélection effective faite par a_k pour un service s_{pl} proposé par le fournisseur a_p parmi le nombre total de recommandations de a_j .

Definition 4.16 Soit $Nbrec_{jk|s_{pl}}$ le nombre de fois où a_j a recommandé le service s_{pl} à a_k et soit $Nbsel_{kj|s_{pl}}$ le nombre de fois où a_k a sélectionné le service s_{pl} proposée par le fournisseur a_p et recommandé par a_j . Inspiré de [Maam 10], $[r_{kj}|s_{pl}]$ est définie comme suit :

$$[r_{kj}|s_{pl}] = \begin{cases} 1 & \text{si } Nbrec_{jk|s_{pl}} = 0 \\ \frac{Nbsel_{kj|s_{pl}}}{Nbrec_{jk|s_{pl}}} & \text{sinon} \end{cases} \quad (4.20)$$

Cependant, $[q_{kj}|s_{pl}]$ représente le degré de satisfaction de l'agent a_k pour les recommandations de a_j pour le service s_{pl} fourni par a_p .

Definition 4.17 Soit $Eval(a_k, s_{pl})$ la note attribuée par a_k à un service s_{pl} proposé par un fournisseur a_p et soit $Nbsel_{kj|s_{pl}}$ le nombre de fois où a_k a sélectionné le service s_{pl} proposée par le fournisseur a_p et recommandé par a_j . $[q_{kj}|s_{pl}]$ est défini comme la moyenne des différentes notes attribuées par a_k à s_{pl} après les utilisations réussies :

$$[q_{kj}|s_{pl}] = \begin{cases} 1 & \text{si } Nbsel_{kj|s_{pl}} = 0 \\ \frac{\sum_{i=1}^{Nbsel_{kj|s_{pl}}} Eval_i(a_k, s_{pl})}{Nbsel_{kj|s_{pl}}} & \text{sinon} \end{cases} \quad (4.21)$$

4.7 Conclusion

Nous avons défini dans la première partie de ce chapitre la notion de confiance en se fondant sur les définitions énoncées dans différentes disciplines en sciences sociales comme la sociologie, la psychologie et l'économie. Ceci dans le but de mieux comprendre ce phénomène social complexe et les différents facteurs qui peuvent influencer son évaluation. Cette analyse pluridisciplinaire nous a également permis d'identifier les bases sociales utiles pour la construction de notre modèle de confiance compte tenu de notre contexte social d'application.

Dans la deuxième partie de ce chapitre, nous avons passé en revue les principaux travaux du domaine afin de définir la structure de la confiance. À travers cette étude bibliographique, nous avons effectué une catégorisation des sources d'information que nous avons utilisées dans l'évaluation de la confiance entre un "trustor" et un "trustee" à savoir : l'expérience individuelle, l'expérience communiquée, les informations sociales et les informations certifiées fournies par le "trustee". Puis, nous avons présenté la topologie de notre modèle et ses différentes composantes.

Dans la troisième et dernière partie, nous avons décrit formellement notre modèle de confiance en mettant en évidence dans chaque mesure notre apport par rapport aux travaux existants. Plus précisément, la confiance du "trustor" à l'égard du "trustee" est évaluée sur la base de trois composantes à savoir, la confiance en la crédibilité sociale, la confiance en l'expertise et la confiance en la recommandation. La confiance en la crédibilité sociale est une agrégation de trois mesures qui sont la position sociale, la proximité sociale et la similarité sociale. Le calcul de ces mesures se base sur les information issues de l'analyse du réseau social. Ces informations sont utiles pour juger si le "trustee" mérite d'être contacté avant d'utiliser ses services. La confiance

en l'expertise est aussi une agrégation de trois mesures qui sont la spécialisation, la disponibilité et la qualité. Cette composante quantifie la capacité du "trustee" à offrir le service demandé avec la qualité attendue en se basant sur son historique d'utilisation. La confiance en la recommandation évalue l'aptitude du "trustee" à fournir des recommandations pertinentes en se basant sur les expériences antérieures.

Chapitre 5

Approche multi-agents pour la découverte et la sélection de services

Sommaire

5.1	Méthodologie de conception	88
5.1.1	Choix de la méthodologie de conception et les hypothèses retenues	89
5.1.2	Définition des rôles des agents	90
5.1.3	Modèle d'organisation sociale : systèmes de références	92
5.1.4	Définition des interactions entre agents	94
5.2	Architecture d'un agent cognitif basée sur la confiance	98
5.2.1	Description de l'architecture	98
5.2.2	Protocole de propagation	103
5.2.3	Comportement de l'agent	105
5.3	Algorithme distribué pour la découverte de services Web dans les réseaux sociaux	106
5.3.1	Algorithme distribué basé sur la confiance pour la découverte de services dans les réseaux sociaux	108
5.3.2	Algorithme de propagation de confiance et modèle probabiliste et contextuel de calcul	114
	Algorithme de propagation de confiance	114
	Modèle probabiliste et contextuel pour le calcul de la confiance	116
5.3.3	Sélection de services	119

5.4 Conclusion 119

Précédemment, nous avons décrit notre modèle de confiance et ses différentes composantes. Dans le présent chapitre, nous montrons comment cette confiance guide les interactions entre les agents et intervient dans leurs prises de décision durant le processus de découverte et de sélection de services. Avant d'aborder la description de l'approche, nous présentons la méthodologie de conception et le modèle d'organisation sociale sur lesquels elle s'appuie. Pour ce faire, nous présentons succinctement dans la première section les méthodologies proposées dans la littérature pour l'utilisation des systèmes multi-agents dans la conception des systèmes complexes et distribués puis, nous mettons en relation le choix effectué dans notre approche en spécifiant les hypothèses retenues, les rôles des agents impliqués et les interactions entre agents. Nous décrivons dans la section suivante la structure interne de notre architecture combinant les technologies des agents cognitifs et celle des services web. Ces agents cognitifs constituent les blocs de construction de base du système multi-agents sur lequel se fonde notre approche. Dans la troisième section, nous présentons nos algorithmes distribués de découverte de services et de propagation des valeurs de confiance dans les réseaux sociaux. Par ailleurs, nous introduisons dans cette section notre modèle probabiliste et contextuel de calcul de confiance entre les agents non adjacents.

5.1 Méthodologie de conception

L'un des défis les plus importants auxquels fait face la communauté des services Web est d'aller au delà des annuaires centralisés UDDI (Universal Description, Discovery, and Integration) dans la découverte de services en utilisant d'autres mécanismes et outils de recherche.

Outre la question de centralité, d'autres questions doivent être abordées, comme la grande quantité d'informations, la perception et l'échange de connaissances, les requêtes complexes et le contrôle distribué.

L'approche multi-agents [Char 13, Hend 01, Huhn 02, Huhn 03, Jian 12, Lomu 10, Maxi 05a, Sial 11, Sing 05, Syca 01] apparaît comme étant l'approche la plus adéquate pour appréhender ces questions et effectuer une découverte, une sélection et une composition de services distribuées. Par définition, les agents, qu'ils soient coopératifs ou compétitifs, sont capables de réaliser des tâches complexes et distribuées, ils supportent différentes formes d'interactions y compris la négociation et la coordination, ils disposent de capacités pour raisonner formellement sur les données sémantiques, et proposent des solutions flexibles qui satisfont les besoins et les préférences des utilisateurs.

C'est dans cette perspective que nous avons décidé d'utiliser la technologie agent dans notre approche de découverte et de sélection de services dans les réseaux sociaux.

5.1.1 Choix de la méthodologie de conception et les hypothèses retenues

Plusieurs méthodologies ont été proposées pour guider les concepteurs dans leur tâche d'analyse et de conception [Braz 97, Kend 95, Sing 98]. Selon Koning [Koni 01], on distingue deux grandes familles de méthodologie de conception ; *la méthodologie de conception orientée agents* où le concepteur choisit l'architecture des agents avant la définition de son module d'interaction et *la méthodologie de conception orientée interactions* dans laquelle le concepteur spécifie les protocoles d'interaction, le langage de communication puis définit les rôles et l'architecture des agents.

Notre méthodologie de conception s'inscrit dans la lignée de la première famille car elle traite à la fois du niveau social ou macro (l'aspect organisationnel) et du niveau individuel ou micro (l'architecture agent). Dans cette perspective, plusieurs méthodologies de conception orientée agents ont été proposées, parmi elles celles de Ferber et Gutknecht [Ferb 98] et de Wooldridge et al. [Wool 00].

Avant d'entamer la conception dans un environnement spécifique, nous devons répondre à plusieurs questions :

- Quels acteurs doivent être modélisés comme des agents logiciels ?
- Quels modèles d'agents et modèles organisationnels doivent être utilisés dans la conception des agents logiciels ?
- Quelles fonctions mentales doivent être incluses dans ces agents ?
- Quelles hypothèses doivent être prises en compte pour définir le cadre et le périmètre d'application ?

Dans cette thèse, nous modélisons chaque utilisateur, inscrit dans le réseau social du demandeur de services, par un agent *social cognitif*. Doté d'une vue partielle du réseau social, l'agent agit de manière autonome au nom de son utilisateur pour satisfaire ses besoins. Chaque agent est par ailleurs caractérisé par un ensemble de services (c'est à dire, il peut offrir un ou plusieurs services) et maintient localement un ensemble de connaissances concernant ses accointances (c'est à dire, son voisinage dans le réseau social, voir définition 2.2). Ces connaissances lui permettent d'une part de savoir s'il

peut participer à la résolution de la requête et d'autre part, d'évaluer la confiance des autres agents avant d'interagir avec eux.

Notons que certaines architectures agents peuvent varier d'une application à l'autre selon les hypothèses retenues aussi qu'en fonction des caractéristiques de l'environnement (complexe, distribué, hétérogène). Dans cette thèse, nous nous sommes fixés une série d'hypothèses sur les agents ainsi que sur leurs comportements.

Hypothèse 1. Les agents sont coopératifs et partagent leurs expériences avec les autres.

Hypothèse 2. Les agents répondent aux messages et respectent le protocole d'interaction et tout message émis par un agent est forcément reçu par son destinataire.

Hypothèse 3. Les agents n'agissent pas de façon malicieuse, ils sont honnêtes dans l'échange d'informations et ils ont la bonne volonté pour satisfaire les besoins du demandeur de services.

Hypothèse 4. Les agents ont une vue limitée dans le réseau social, ils ne connaissent que les agents qui appartiennent à leur ensemble d'acointances sociales.

Hypothèse 5. Les agents appliquent un mécanisme décisionnel décentralisé lorsqu'ils veulent interagir avec d'autres agents.

La plupart de ces hypothèses présentées sont des hypothèses restrictives et fortes. Cela nous permet dans un premier temps d'évaluer l'efficacité de notre processus dans de telles conditions optimistes. Ceci dit, il est toujours possible et intéressant de relâcher certaines hypothèses au cours de l'étude empirique et ensuite analyser les résultats obtenus.

5.1.2 Définition des rôles des agents

Durant le processus de découverte et de sélection de services, certains agents peuvent être de bons fournisseurs (c'est à dire, possèdent un service adéquat) ou non. Bien qu'ils ne fournissent pas un bon services, les agents peuvent être bien connectés en recommandant des fournisseurs pertinents menant à de bons services. Afin de distinguer les responsabilités et les activités que fournit un agent au sein d'une organisation, nous utilisons la notion de *rôle*.

Dans son usage général, la notion de rôle a été mise en place pour tenir compte des relations et des comportements d'un agent dans un contexte social particulier [Stein 00].

Un rôle peut donc prendre des sens très variés selon le modèle et le contexte dans lequel il est utilisé [Odel 02].

Plusieurs approches exploitent le rôle au niveau de *l'organisation* [Colm 07, Ferb 98, Wool 00], d'autres au niveau de *l'implémentation* [Cabr 03a, Cabr 03b, Odel 04]. Si le rôle est étudié du point de vue de l'organisation (les agents sont liés aux rôles), il est perçu comme une contrainte sous laquelle l'agent prend part à certaines interactions et évolue d'une certaine façon dans l'organisation. De ce point de vue, le rôle est indépendant des agents qui vont le jouer étant donné qu'ils possèdent les capacités nécessaires pour exécuter les fonctions qui leur sont rattachées. S'il est étudié du point de vue de l'implémentation de l'agent considéré, il est vu comme une entité encapsulant certains attributs et comportements auxquels les agents sont rattachés a priori. Certains travaux (voir par exemple, [Colm 07]) essaient de trouver une association entre ces différentes appréhensions du concept de rôle et de son utilisation dans les organisations SMA. D'une manière générale, le rôle est une étiquette qui sert à catégoriser les agents à un niveau organisationnel afin d'encadrer leurs interactions et faciliter leurs localisations. Dans notre travail, un agent peut s'attribuer, au niveau de l'organisation, un seul rôle parmi trois possibles :

- *demandeur* : un demandeur est l'agent qui reçoit une requête de son utilisateur associé contenant les services requis et les préférences exprimées. Il est l'initiateur du processus de recherche. Il détermine à partir de son SA_k les accointances socialement dignes de confiance et leur envoie la requête.
- *fournisseur* : un fournisseur est un agent qui offre un ou plusieurs services demandés. Quand un agent reçoit une requête, il analyse la requête et détermine comment il participe à la satisfaction des besoins du demandeur en fonction de ses capacités. S'il y a une correspondance entre les fonctionnalités souhaitées et les services offerts, il se proclame fournisseur et participe à la recherche d'autres fournisseurs potentiels en soumettant à son tour la requête à ses accointances sociales dignes de confiance.
- *recommandeur* : un recommandeur est un agent qui n'a pas de services utiles, mais peut participer implicitement à la résolution de la requête en la propageant à des références. Autrement dit, il joue le rôle d'un « éclaireur » qui élargit la vue locale de l'agent demandeur et l'enrichit par des références pertinentes qui peuvent fournir les services souhaités. Contrairement au fournisseur, un agent recommandeur interroge seulement des connaissances dignes de confiance socialement et suffisamment expertes c'est à dire offrant les services requis.

- *stoppeur* : un stoppeur est un agent qui n'a pas pu participer à la résolution de la requête ; il ne peut ni offrir un service ni recommander une référence. Dans le processus de découverte, ce rôle est vu comme un verrou qui bloque la propagation de la requête dans le réseau social. Cela nous permet de limiter l'espace de recherche et de réduire ainsi le coût de calcul.

5.1.3 Modèle d'organisation sociale : systèmes de références

Dans notre contexte, chaque agent possède une vue partielle du système dans laquelle il maintient ses connaissances sociales. Pour satisfaire leurs besoins, les agents peuvent s'entraider en échangeant leurs expériences passées et leurs valeurs de confiance. La question qui se pose est quel mécanisme appliquer pour propager la requête du demandeur de services dans le réseau social ? et quel modèle d'organisation adéquat utiliser pour rendre cette coopération effective ?

Dans les environnements structurés en un réseau connecté, les systèmes d'organisation les plus connus sont : le système de noms de domaine (DNS) et l'architecture pair à pair (P2P) (voir Sect. 3.4).

DNS est un système de nommage hiérarchique distribué pour les ordinateurs, services ou toute ressource connecté à Internet. Il fournit un mécanisme de résolution par lequel chaque ressource traduit les noms en adresses et vice versa.

L'architecture P2P est un modèle organisationnel distribué définissant les liens de communication entre les pairs ainsi que le processus de localisation des ressources partagées.

Bien que ces deux modèles organisent et gèrent les interactions entre les ressources du système, ils n'intègrent aucun aspect décisionnel dans le processus de recherche, chose qui est prise en compte dans *les systèmes de références* [Fone 96, Kaut 97, Yu 99]. Le système de références est une organisation qui assure une relation étroite, basée sur la confiance, entre les agents en les reliant sous forme d'une chaîne de références. Il favorise la coopération entre les agents pour localiser des fournisseurs de services pertinents en se basant sur des connaissances distribuées et une prise de décision décentralisée. D'après [Al S 12], un système de références assure les fonctions suivantes :

- L'évaluation des services et des références pour aider les utilisateurs à trouver des experts.
- Le maintien des croyances concernant d'autres agents pour évaluer leur confiance.
- L'intégration de l'interaction entre agents.

Au vu de ces avantages, nous avons décidé d'utiliser le système de références dans notre processus de découverte et de sélection de services car cette approche sociale respecte non seulement la topologie du notre environnement (c'est à dire, le réseau social distribué et hétérogène), mais utilise aussi la confiance entre les agents dans la prise de décision.

Les systèmes de références ont été largement étudiés dans la littérature et plus précisément par Singh et *al.* [Sing 01, Yolu 03, Yolu 05, Yu 03, Yu 99]. Cependant, comme nous l'avons montré dans la section 3.5, le processus de référence de Singh et *al.* présente deux principales lacunes :

- La prise de décision associée à la propagation de la recherche est centralisée ; elle est toujours effectuée par le demandeur de services.
- La confiance dépend uniquement de la sociabilité et de l'expertise de l'agent et ne prend pas en considération la qualité de ses recommandations.

Afin d'illustrer ces propos, nous présentons dans la figure 5.1 un scénario décrivant notre processus de référence et celui de Singh et *al.*.

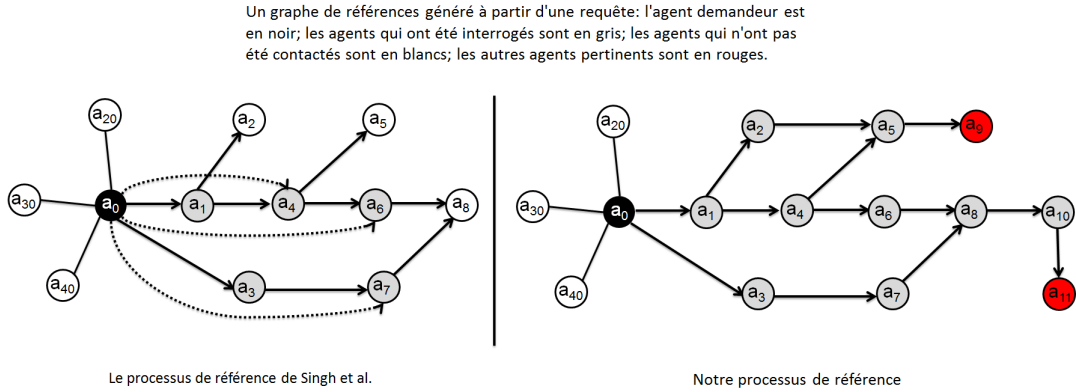


FIGURE 5.1 – Comparaison de deux systèmes de références

Selon Singh et *al.*, le processus de référence est défini comme suit ; l'agent demandeur a_0 interroge uniquement les agents dignes de confiance (a_1 et a_2) parmi ses voisins. Un agent interrogé peut soit fournir les services requis soit des références à d'autres agents. L'agent demandeur peut accepter une offre de service et peut suivre les références, le cas échéant. En d'autres termes, il recueille toutes les références possibles et décide de continuer la recherche en communiquant avec certaines des références proposées. Dans cet exemple, il interroge a_4 , a_6 et a_7 alors que a_2 , a_5 et a_8 sont ignorés. En raison de la nature distribuée des réseaux sociaux, un seul agent, qui est l'agent demandeur de

service, ne peut pas collecter toutes les informations et faire par la suite une évaluation de la confiance.

Nous proposons *une décentralisation de la prise de décision associée à la propagation de la recherche*. Comme dans l'approche de Singh et al., l'agent demandeur lance le processus de découverte en communiquant uniquement avec les agents dignes de confiance parmi ses accointances sociales. Cependant, chaque agent interrogé agit de façon autonome et décide localement de continuer ou non la recherche en propageant la requête sans revenir à l'agent demandeur. Cette propagation progressive de la recherche conduit à d'autres références pertinentes (a_9 et a_{11}) ce qui permet d'éviter les occasions manquées pour la découverte de services.

Pour résumer, un système de références est à la fois un modèle d'organisation social et un mécanisme de propagation de recherche. Son intérêt réside dans le fait qu'il rend les utilisateurs conscients de leurs communautés existantes. Habituellement, un utilisateur n'est conscient que d'une partie du réseau social auquel il appartient. En accédant à une communauté plus large, il peut découvrir de nouveaux liens avec des agents ou informations qui, autrement, sont dissimulés et inaccessibles.

5.1.4 Définition des interactions entre agents

Les communications, dans les systèmes multi-agents (SMA), sont à la base des interactions et de l'organisation sociale. Sans communication, l'agent n'est qu'un individu isolé, sourd et non coopératif avec les autres agents. Les agents cognitifs utilisent des ACL (Agent Communication Language) pour la communication inter-agent. Les principaux ACL utilisés, i.e. KQML [Fini 94] et FIPA-ACL¹, sont fondés sur *la théorie des actes de langage* et ont été conçus pour l'échange d'information, de connaissances ou encore de services entre agents.

La théorie des actes de langage a été initiée par Austin dans [Aust 62]. Il définit l'acte illocutoire qu'on accomplit en disant quelque chose et qui indique l'intention du locuteur. Searle poursuit ces travaux dans [Sear 69] où il décrit un acte illocutoire comme composé d'une force illocutoire en terme de type de message ou *performatif* et d'un contenu.

FIPA a défini différents types de performatifs² ainsi que leur sémantique opérationnelle : pré-conditions, post-conditions et effets attendus lors de l'envoi du message. Dans cette thèse, nous utilisons trois performatifs : REQUEST, INFORM et PROPAGATE.

1. <http://www.fipa.org/specs/fipa00025/>

2. <http://www.fipa.org/specs/fipa00037/SC00037J.pdf>

Ces performatifs couvrent les différents types de messages traités par le protocole de propagation (voir Sect. 5.2.2) associé au processus de découverte et de sélection de services. En s'appuyant sur le modèle de description de Labrou et Finin [Labr 94], nous décrivons sémantiquement la signification, les conditions d'application (i.e., les pré-conditions) du point de vue de l'émetteur, ainsi que les conditions de satisfaction (i.e., les post-conditions) et de complétion des performatifs utilisés comme suit :

- 1) Une description en langage naturel du sens intuitif du performatif.
- 2) Une expression logique qui décrit l'acte illocutoire. Cela est une représentation formelle de la description en langage naturel.
- 3) Les pré-conditions (Pre) qui indiquent l'état nécessaire d'un agent afin d'envoyer un performatif et celui du récepteur pour l'accepter et le traiter. Elles portent sur les intentions et les croyances de l'émetteur et sur les compétences des deux interlocuteurs.
- 4) Les post-conditions (Post) qui décrivent les états des agents après l'énoncé d'un performatif (pour l'émetteur) et après la réception d'un message (pour le récepteur).
- 5) Les conditions de complétion (Comp) qui indiquent l'état final de l'émetteur lorsque l'action déclenchée par le performatif a été remplie.

L'idée centrale est de définir formellement les états cognitifs des agents pour chaque performatif. Pour ce faire, nous utilisons des opérateurs en *logique de premier ordre*. L'utilisation de ces opérateurs, pour décrire les états mentaux des agents peut être trouvée dans Campbell [Camp 90], Singh [Sing 93] et Freber [Ferb 07]. Les opérateurs employés dans cette description sont :

- **croire(A,P)** : l'agent A croit P. Les croyances sont supposées être révisables. croire(A,P) signifie, en position prédicative, que l'agent A croit P, et en position d'action que l'agent A remet en cause ses croyances de manière à croire désormais que P est vrai, quelle que soit la valeur de sa croyance antérieure sur P. Pour l'instant, il suffit de dire que P existe dans la base de croyances (voir Sect. 5.2.1) de A.
- **intention(A,P)** : l'agent A a l'intention de faire l'action P.
- **savoir(A,P)** se réfère à l'état cognitif des agents. savoir(A,P) exprime un état de conscience de l'agent A concernant la connaissance de P.

- **vouloir(A,P)** : l'agent A veut que l'évènement ou l'action P se produise.
- **engager(A,B,P)** : l'agent A est engagé auprès de B pour faire P.
- **compétent(A,P)** : l'agent A est compétent pour faire l'action P. $\text{croire}(A, \text{compétent}(B,P))$ précise que l'agent A croit que l'agent B est compétent pour faire P.

Dans le reste de cette section nous donnons la description des trois performatifs que nous utilisons. Dans cette description A est l'agent émetteur, B est l'agent récepteur et P est le contenu.

- **request(A,B,P)**

- 1) l'agent A utilise ce performatif pour demander un renseignement P à l'agent B.
- 2) $\text{vouloir}(A, \text{savoir}(A,X))$ où X peut être une des expressions suivantes : $\text{croire}(B,P) \vee \text{croire}(B, \text{NOT}(P)) \vee \text{NOT}(\text{croire}(B,P))$
- 3) $\text{Pre}(A) : \text{vouloir}(A, \text{savoir}(A,X))$
 $\text{Pre}(B) : \emptyset$
- 4) $\text{Post}(A) : \text{intention}(A, \text{savoir}(A,X))$
 $\text{Post}(B) : \text{savoir}(B, \text{vouloir}(A, \text{savoir}(A,X)))$
- 5) $\text{Comp}(A) : \text{savoir}(A,X)$

- **inform(A,B,P)**

- 1) l'agent A utilise ce performatif pour répondre à une demande d'information émise par un l'agent B ou pour partager une information.
- 2) $\text{croire}(A,P) \wedge \text{vouloir}(A, \text{savoir}(B, \text{croire}(A,P)))$
- 3) $\text{Pre}(A) : \text{croire}(A,P), \text{savoir}(A, \text{vouloir}(B, \text{savoir}(B,X)))$. A ne ment pas et B est intéressé à savoir P. X est une des expressions mentionnées dans request.
 $\text{Pre}(B) : \text{intention}(B, \text{savoir}(B,X))$
- 4) $\text{Post}(A) : \text{savoir}(A, \text{savoir}(B, \text{croire}(A,X)))$
 $\text{Post}(B) : \text{savoir}(B, \text{croire}(A,P)) \vee \text{savoir}(B, \text{compétent}(A,P))$
- 5) $\text{Comp}(A) : \text{savoir}(B, \text{croire}(B,P))$

- **propagate(A,B,P)**

- 1) l'agent A utilise ce performatif pour envoyer la requête P à l'agent B.

- 2) $\text{vouloir}(A, \text{engager}(B, A, P))$
- 3) $\text{Pre}(A) : \text{vouloir}(A, \text{engager}(B, A, P)) \wedge \text{croire}(A, \text{compétent}(B, P))$
 $\text{Pre}(B) : \emptyset$
- 4) $\text{Post}(A) : \text{savoir}(A, \text{savoir}(B, P)) \wedge \text{savoir}(A, \text{savoir}(B, \text{croire}(A, P)))$
 $\text{Post}(B) : \text{savoir}(B, \text{croire}(A, P)) \wedge \text{intention}(B, \text{croire}(B, P))$
- 5) $\text{Comp}(A) : \text{engager}(B, A, P)$

Pour préciser comment les agents évoluent durant le processus de découverte et de sélection de services, nous formalisons les règles de conversation par un automate à états finis. Les états de l'automate (voir Fig. 5.2) définissent les états des interactions et les arêtes représentent les performatifs de communication. À chaque réception ou émission d'un message par un autre agent, l'automate tient à jour l'état courant de la communication. Nous utilisons la notation ? et ! devant un performatif pour indiquer s'il est respectivement reçu ou émis par l'agent.

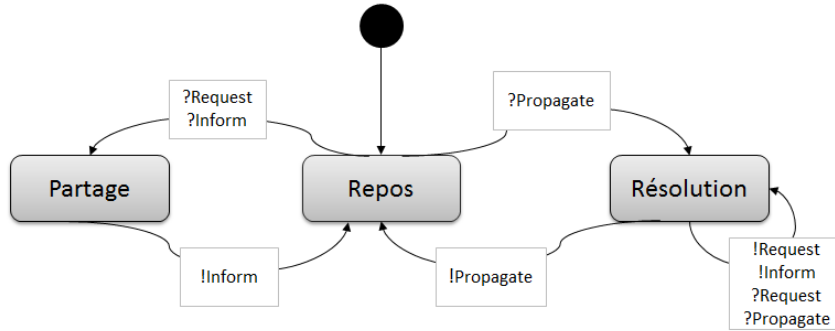


FIGURE 5.2 – Automate de contextualisation des interactions

Initialement, les agents sont dans un état **repos** en attente d'un but à résoudre. À la réception d'un message PROPAGATE, l'agent passe dans l'état **résolution**. Cet état représente la phase active de l'agent dans laquelle il met à jour sa base de croyances par l'envoi des messages REQUEST et la réception des messages INFORM correspondants. Par ailleurs, il traite localement la requête reçue afin de déterminer s'il peut offrir le service adéquat (voir Sect. 5.2.3). Notons que le message PROPAGATE peut être énoncé, soit par l'utilisateur associé à l'agent demandeur, soit par un agent qui souhaite soumettre à un groupe d'agents dignes de confiance une requête à résoudre. À la fin de la phase de résolution, l'agent passe de l'état PROPAGATE à l'état **repos** en envoyant un ou plusieurs messages PROPAGATE à des agents dignes de confiance pour continuer le processus de découverte de services. L'état **partage** est un état temporaire dans lequel l'agent se comporte comme un agent réactif qui réagit aux événements externes.

Nous distinguons deux types d'événements, soit une demande d'information exprimée par un message REQUEST soit, une demande de partage d'information exprimée par un message INFORM. Dans les deux cas, il répond par un message INFORM contenant les paramètres requis et passe à l'état **repos**. Dans ce qui suit, nous allons décrire la structure interne de notre architecture d'agent cognitif qui est basée sur la confiance.

5.2 Architecture d'un agent cognitif basée sur la confiance

La phase de conception présentée dans la section précédente définit une méthodologie de développement du système multi-agent. Les comportements, les interactions entre les agents ainsi que les ressources nécessaires à l'atteinte de leurs buts sont également spécifiés de manière conceptuelle en fonction des rôles et du modèle d'organisation. Dans cette section, nous présentons notre architecture d'agent cognitif intégrant la technologie des services Web et s'appuyant sur la confiance avec une description de ses différents composants. Nous exposons ensuite une vue dynamique du comportement de l'agent à l'aide des diagrammes AUMI d'interaction et d'activités. Ceci permet de visualiser l'échange de messages entre agents au cours du processus de découverte de services et également le flux de contrôle interne qui gère le comportement de l'agent.

5.2.1 Description de l'architecture

Comme l'ont souligné [Cast 98, Esfa 01], la confiance et la réputation sont des états mentaux de l'agent qui sont fonction de ses croyances. Pour un agent cognitif, les états mentaux guident son comportement lorsqu'il veut interagir avec un autre agent et gère les conséquences mentales de cette interaction. De ce fait, nous adoptons une architecture cognitive dans la conception de notre modèle agent. Elle est inspirée d'un certain nombre d'architectures existantes [Brys 00, Ferg 92, Geor 87, Syca 01] et composée de cinq modules : le module de raisonnement, le module de confiance, le module de contrôle, le module d'interaction et le module de planification auxquels sont ajoutées une base de croyances, une bibliothèque de plans et une pile de buts (voir Figure 5.3).

1) Base de croyances : La base de croyances contient des structures de données dans lesquelles a_k stocke les informations relatives à la fois à lui-même (le profil Pr_k de son utilisateur associé et l'ensemble $S_k = \{s_{k1}, s_{k2}, \dots, s_{km_k}\}$ des services qu'il offre avec leurs valeurs de qualité de service (QoS)) et à ses accointances sociales. Les croyances

Accountance	Profil	Accountances Sociales	Services/QoS(Sp,Di)
a_1	Pr_1	$\{a_3, a_5\}$	$\{(s_{11}, 0.4, 0.7); (s_{12}, 0.6, 0.8)\}$
a_2	Pr_2	$\{a_0, a_4, a_8\}$	$\{(s_{21}, 1, 0.9)\}$
$\dots$	$\dots$	$\dots$	$\dots$
a_j	Pr_j	$\{a_2, a_5, a_{10}\}$	$\{(s_{j1}, 0.5, 0.6); \dots\}$

TABLE 5.1 – Table d'interaction personnelle (PIT_k) d'un agent a_k

relatives aux accountances sont représentées dans deux structures de données. La première structure de données appelée la Table d'Interaction Personnelle PIT_k (voir table 5.1 pour un exemple) contient pour chaque agent $a_j \in SA_k$, la liste de ses accountances sociales SA_j , son profil Pr_j et son expertise c'est à dire, l'ensemble S_j des services offerts avec leurs valeurs de QoS. Par exemple, pour l'agent a_2 , l'ensemble d'accountances sociales est formé par trois agents a_0 , a_4 et a_8 . Il offre un seul service s_{21} dont les qualité de services, la spécialisation et la disponibilité, égalent respectivement à $Sp = 1$ et $Di = 0.9$ (voir Sect. 4.5.1 et 4.5.2). Les informations contenues dans cette table sont essentielles pour évaluer la confiance en la crédibilité sociale et en l'expertise de chaque agent $a_j \in SA_k$. Notons que la valeur de l'attribut subjectif *Eval* d'un service n'est pas affichée dans PIT_k car elle n'est pas communiquée par a_j mais plutôt calculée localement par a_k après avoir utilisé le service correspondant.

Dans la deuxième structure de donnée appelée la Table d'Histoire HT_k (voir table 5.2 pour un exemple), a_k garde une trace historique de ses interactions avec les autres agents et conserve les scores de réputation attribués aux services utilisés. L'idée principale de cette table est de capitaliser sur les informations récupérées à partir des expériences précédentes. Cette table ne contient que les agents qui ont soit proposé un service soit une recommandation, c'est à dire qui ont interagi d'une manière active avec a_k . Les informations relatives à un agent a_j contenues dans cette table sont utilisées d'une part pour évaluer la pertinence de ses recommandations générées (i.e. a_j est un recommandeur) et d'autre part pour estimer la qualité de son service fourni (i.e. a_j est un fournisseur).

L'interaction avec le module de contrôle consiste à mettre à jour les informations contenues dans PIT_k et HT_k . Initialement, les deux tables sont vides. Pour PIT_k , un nouvel enregistrement est ajouté, lorsque a_k établit un premier contact avec un agent $a_j \in SA_k$. L'initialisation et plus généralement les mises à jour sont effectuées sur réception d'un message de type REQUEST assurant l'intégrité de PIT_k compte tenu des changements possibles affectant les valeurs courantes des accountances (leurs services offerts avec leurs valeurs de QoS c'est à dire, les valeurs de Sp et de Di). Pour HT_k elle est remplie au fur et à mesure de l'interaction de a_k avec les agents du réseau

Accountance	$Nbrec_{jk}$	$Nbsel_{kj}$	Service/QoS(Eval)
a_1	7	4	$\{(s_{32}, 0.6)\}$
a_1	2	2	$\{(s_{84}, 0.8)\}$
a_2	10	6	$\{(s_{51}, 0.7)\}$
...	...	...	...
a_j	8	4	$\{(s_{j'1}, 0.8); \dots\}$

TABLE 5.2 – Table d’Historique (HT_k) d’un agent a_k

social. Par exemple, dans la table 5.2 a_k a reçu 10 recommandations de la part de a_2 concernant le service s_{51} . Sur ces 10 recommandations, a_k n’a sélectionné le service que 6 fois. La qualité de ce service, qui correspond à la moyenne arithmétique/pondérée des notes attribuées par a_k après ces utilisations (voir Sect. 4.5.3), est égale à 0.7.

2) Bibliothèque de plans : Cette bibliothèque contient des squelettes de plan abstraits (i.e. méthodes de décomposition) utilisées au cours de la planification afin de décomposer un certain but complexe. Les plans sont soit atomiques soit composites. En raffinant le but à accomplir en sous-buts en fonction des plans disponibles, un agent peut déterminer le(s) service(s) concret(s) qu’il peut fournir pour participer à la résolution de la requête.

3) Pile de buts : Un but est la description d’un comportement désiré (interne ou externe) par l’agent exprimé en termes d’exécution d’un service. Les buts sont organisés de manière hiérarchique : un but complexe est atteint lorsque les sous-buts sont satisfaits et ceci récursivement jusqu’à atteindre les buts élémentaires. Les buts élémentaires sont réalisés par l’exécution d’un service.

4) Module de calcul de confiance : Le module de calcul de confiance est responsable du calcul de la confiance qu’un agent a_k accorde à chacune de ses accountances sociales. La valeur de cette confiance influence la décision de l’agent quant au choix des agents avec lesquels il interagit au cours du processus de découverte de services. Pour évaluer la confiance, l’agent utilise les informations contenues dans sa base de croyances. Le calcul de la confiance est décomposé en une séquence de calculs élémentaires. Ce module est équipé d’un ensemble de méthodes qui sont utilisées pour effectuer les calculs élémentaires de chaque composante de confiance (voir Chap. 4).

5) Module de raisonnement : Ce module représente le composant délibératif de l'agent qui engendre les actions à entreprendre pour accomplir le but. Il joue un rôle important dans la détermination de la participation d'un agent dans la résolution de la requête aussi bien au niveau individuel qu'au niveau collectif (propagation de la requête dans le réseau social). Dans les deux cas, par le biais de ce module, l'agent vérifie si lui-même ou une de ses accointances possède localement un service adéquat. Pour ce faire, il consulte les informations contenues dans sa base de croyances. Ce module est équipé d'une méthode qui peut être utilisée pour accomplir un appariement sémantique entre les services offerts S_k par l'agent a_k et les services requis dans F (voir Définition 2.4). La méthode d'appariement sémantique entre un service $s_{kl} \in S_k$ et un service $s \in F$ peut être définie comme suit : $matching(s_{kl}, s) = True \Leftrightarrow (s_{kl}.in \subseteq s.in) \wedge (s.out \subseteq s_{kl}.out) \wedge (s_{kl}.f \equiv s.f)$.

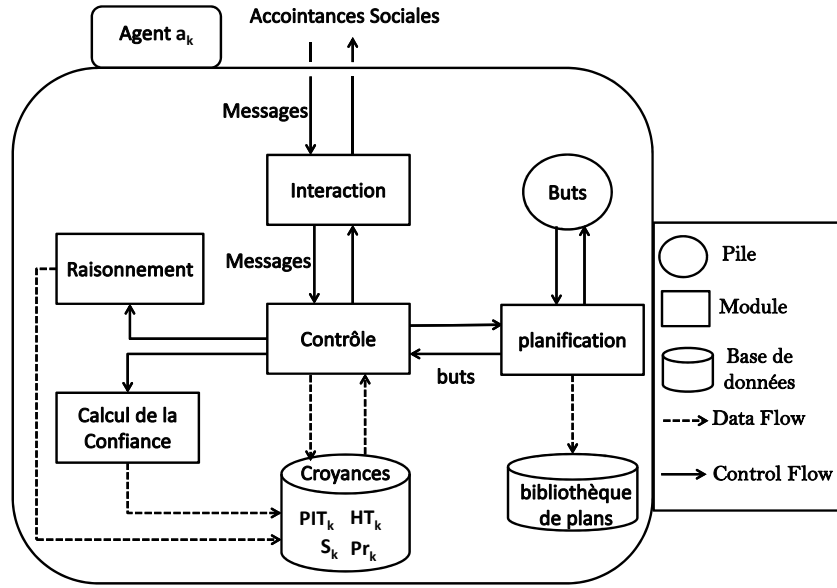


FIGURE 5.3 – Architecture d'agent cognitif orientée services et basée sur la confiance

6) Module de planification : Durant le processus de découverte de services, lorsqu'un agent reçoit une requête complexe, il essaye tout d'abord de la résoudre par le biais de son module de raisonnement, c'est à dire d'appliquer directement les méthodes d'appariement sémantique afin de déterminer le(s) service(s) qui satisfont les fonctionnalités requises. Si le but (i.e., la requête) n'est pas totalement résolu, il le raffine au niveau de son module de planification en utilisant la bibliothèque de plans.

Le composant principal de ce module est un planificateur HTN³ (hierarchical task network) [Nau 03] qui génère des sous-buts stockés dans la pile de buts. Dans ce cas, le but à atteindre n'est plus un ensemble de propriétés (input, output, ...) qui doivent être vérifiées mais une séquence d'actions primitives ou complexes à réaliser à partir de l'état initial de croyances des agents. Le processus de planification HTN procède en utilisant les méthodes de la bibliothèque de plans pour décomposer le but en sous-buts de manière récursive. Pour chaque but ou sous-but non primitif, le planificateur choisit une méthode applicable et l'instancie pour décomposer le but en sous-buts jusqu'à ce qu'on atteigne les buts primitifs qui peuvent être résolus directement en utilisant un service et s'arrête lorsque aucune décomposition n'est possible.

7) Module de contrôle : Le module central de l'architecture d'un agent a_k est le module de contrôle, qui gère son comportement et guide sa prise de décision au cours du processus de découverte et de sélection de services. Sur réception d'un message PROPAGATE, l'agent passe de l'état passif à l'état actif. À l'aide de ce module, il extrait la requête contenue dans le message reçu et la transmet au module de raisonnement pour résolution. Si la requête n'est pas totalement résolue au niveau du module de raisonnement, le module de contrôle envoie le(s) but(s) complexes qui ne peuvent pas être satisfaits par les services disponibles pour raffinement dans le module de planification. De la même manière, l'agent essaye de résoudre les sous-buts générés au niveau de son module de raisonnement. Dans tous les cas, le module de contrôle active le module de confiance pour calculer le niveau de confiance qu'accorde a_k à chacune de ses accointances. Le résultat du calcul guide la prise de décision de a_k dans le choix des accointances auxquelles il va envoyer la requête. Seules les accointances $a_j \in SA_k$ ayant $SC(a_k, a_j) > \alpha$ seront contactées par l'intermédiaire d'un message PROPAGATE.

Le module de contrôle est également le seul module qui peut lire/écrire dans la base de croyances, et donc mettre à jour son contenu compte tenu des informations courantes des accointances sociales. Pour manipuler les données, ce module est muni d'opérateurs permettant l'ajout, la suppression, le remplacement, la mises à jour et l'interrogation des données. Par exemple, le contenu d'un enregistrement de PIT_k est affiché par la méthode *get()* et mis à jour par la méthode *set()* qui remplace automatiquement toute valeur préexistante, associée à la clé spécifiée, par la nouvelle valeur.

3. HTN est une technique de planification basée sur le contrôle de la connaissance avec l'hypothèse du monde fermé (c'est à dire, tous les blocs de construction sont connus a priori). La planification HTN offre une abstraction hiérarchique avec une stratégie efficace pour faire face à la complexité et l'étendue du monde réel. Le but d'un planificateur HTN est de produire une séquence d'actions qui effectuent une activité ou une tâche.

8) Module d'interaction : Les communications inter-agents jouent un rôle déterminant dans la dynamique des systèmes multi-agents. Ces communications sont menées à travers un échange de messages géré par le module d'interaction. Selon les standards FIPA⁴, un message est constitué des champs suivants :

- L'adresse de l'émetteur
- L'adresse du récepteur
- Le performatif appelé aussi acte du message. C'est lui qui représente la force illocutoire du message (voir Sect. 5.1.4)
- Le contenu du message
- Le langage et l'ontologie utilisés pour décrire le contenu
- L'identifiant du message
- L'identifiant de la conversation

Dans cette thèse, nous n'allons considérer que les champs qui sont pertinents pour notre application SMA. Nous allons donc restreindre la structure du message à quatre champs $\langle \textit{performatif}, \textit{émetteur}, \textit{récepteur}, \textit{contenu} \rangle$.

5.2.2 Protocole de propagation

Dans cette section, nous allons présenter les interactions entre les agents durant le processus de découverte de services dans le réseau social à l'aide d'un protocole de propagation. La figure 5.4 illustre les messages échangés entre deux agents a_k et $a_j \in SA_k$ en utilisant le formalisme AUML [Caba 05]. Lorsqu'un agent a_k reçoit un message PROPAGATE, il envoie à l'ensemble de ses accointances SA_k un message REQUEST pour mettre à jour ses croyances dans PIT_k par rapport aux informations courantes de chacune de ses accointances. Sur réception d'un message de type REQUEST, un agent a_j répond avec un message INFORM contenant des informations concernant son profil Pr_j , ses accointances sociales SA_j et les services qu'il offre S_j avec leurs valeurs de QoS⁵. De même, tous les agents de SA_k répondent avec un message INFORM à a_k .

4. <http://www.fipa.org/specs/fipa00061/SC00061G.html>

5. Rappelons que nos agents ne sont pas malveillants et ont la bonne volonté pour aider et fournir des informations correctes. Sinon, nous considérons que c'est une défaillance du système qui est hors de notre champ de recherche.

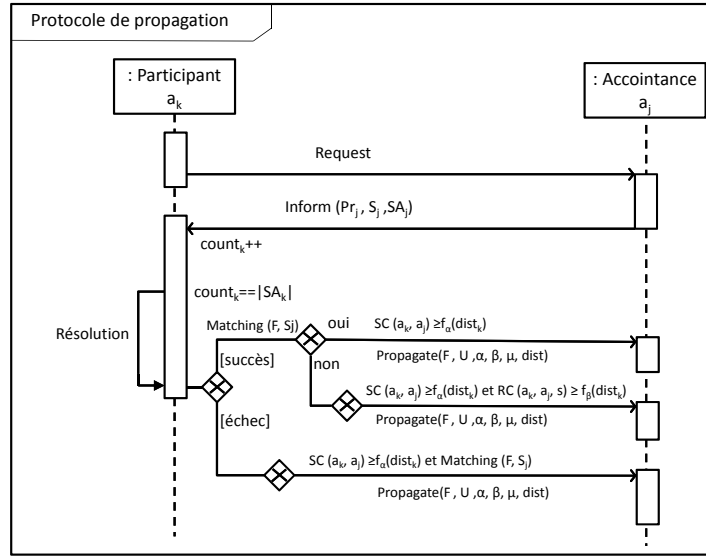


FIGURE 5.4 – Diagramme AUML d'interaction du protocole de propagation

Chaque fois qu'un message arrive, a_k incrémente son compteur de messages et met à jour sa PIT_k . Une fois que toutes les accointances ont répondu, a_k entame la phase de résolution de la requête dans laquelle il vérifie si un des services qu'il offre dans S_k correspond à un des services requis dans F . En cas de succès, a_k envoie un message PROPAGATE à a_j dans deux cas :

- Le premier est que a_j détient un service qui satisfait les besoins de l'utilisateur et qu'il est en plus socialement crédible.
- Le deuxième est qu'il ne propose pas un service adéquat mais qu'il est socialement crédible et a une bonne capacité de recommandation.

En cas d'échec, a_k envoie un message PROPAGATE à a_j si ce dernier est à la fois socialement crédible et possède un service adéquat. Cette diffusion progressive de la requête est le mécanisme original des systèmes de références dans lesquels la propagation de la recherche est assurée via une navigation dans le graphe. Si aucune de ces conditions n'est vérifiée, le processus de découverte de services est arrêté au niveau de a_k qui acquiert le rôle de stoppeur. Le critère d'arrêt implique non seulement l'incapacité d'un agent à offrir un service adéquat mais aussi son inaptitude à recommander une bonne référence.

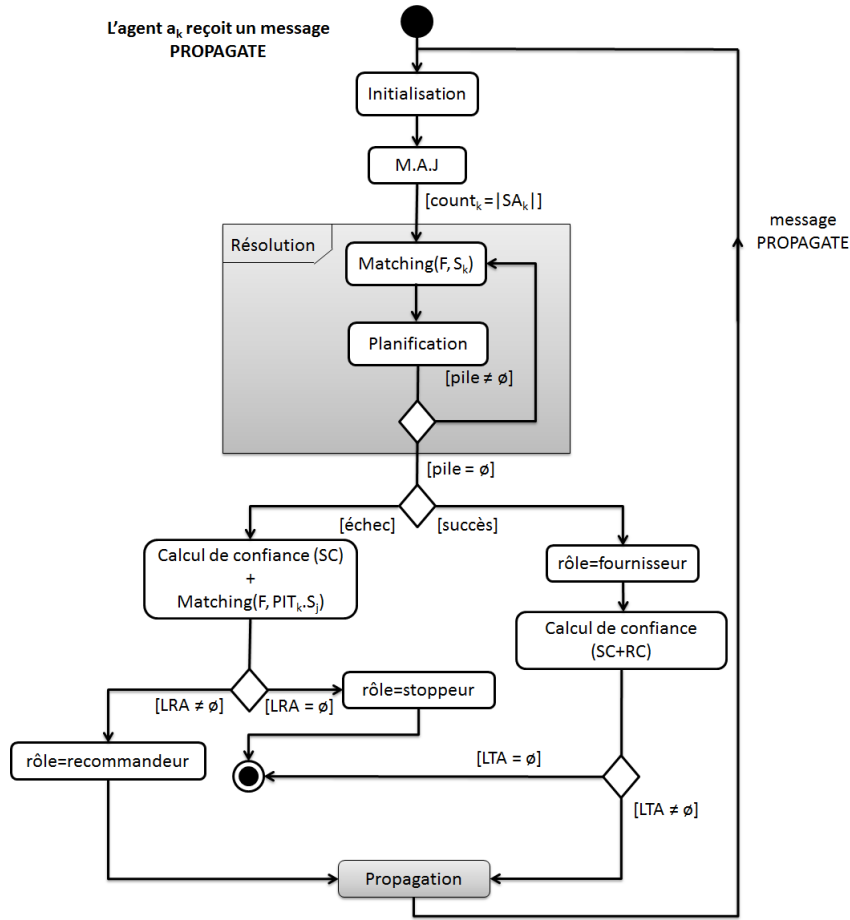


FIGURE 5.5 – Diagramme d'activité d'un agent

5.2.3 Comportement de l'agent

Le diagramme d'activité décrit le comportement interne d'un agent au cours du processus de découverte de services. La figure 5.5 représente les différentes activités de l'agent a_k ainsi que le flux de contrôle entre elles. Notons qu'il y a quatre nœuds de décision où l'agent agit d'une manière autonome et décide localement de l'action à entreprendre.

La première activité, dite activité d'« Initialisation », est déclenchée sur réception d'un message PROPAGATE. Dans cette phase, l'agent a_k met à jour $dist_k$ qui représente la distance qui le sépare de l'agent demandeur a_r dans la chaîne fournisseur-recommandeur. Une fois que l'activité « d'Initialisation » est effectuée, a_k exécute l'activité « M.A.J », destinée à la mise à jour de sa base de croyances.

L'agent entame par la suite la phase de résolution de la requête qui consiste à vérifier sa capacité à offrir des services appropriés. La résolution est composée de deux

activités qui sont l'appariement assuré par la fonction de « Matching(F, S_k) » suivi de la « Planification ». Ces deux activités s'exécutent en boucle tant que la pile de buts n'est pas vide et des raffinements des sous-buts sont possibles. À l'issue de cette phase, a_k identifie le(s) service(s) qui répond(ent) aux besoins du demandeur de services. En fonction du résultat de la condition de test, le comportement de l'agent évolue.

En cas d'appariement réussi, c'est à dire a_k a au moins un service $s_{kl} \in S_k$ qui offre la même fonctionnalité qu'un service demandé $s \in F$, il prend le rôle d'un fournisseur (rôle_k=fournisseur). Ensuite, il lance l'activité « Calcul de SC » en déterminant la liste LTA_k des agents socialement crédibles. Si $LTA_k \neq \emptyset$ alors, a_k exécute l'activité de « Propagation » qui consiste à envoyer un message PROPAGATE à toutes les accointances dans LTA_k . Sinon, a_k arrête le processus de découverte.

En revanche, si a_k ne fournit pas un service approprié, il identifie l'ensemble LRA_k qui contient des agents qui sont non seulement crédibles socialement mais qui offrent aussi des services adéquats. Si $LRA_k = \emptyset$ alors, a_k n'a pas de références pertinentes à recommander. Dans ce cas, a_k arrête la propagation de la requête dans le réseau social et prend le rôle de stoppeur (rôle_k=stoppeur). Sinon, a_k prend le rôle de recommandeur (rôle_k=recommandeur) et exécute l'activité de « Propagation » qui consiste à envoyer un message PROPAGATE à toutes les accointances dans LRA_k .

5.3 Algorithme distribué pour la découverte de services Web dans les réseaux sociaux

Dans cette section, nous présentons les algorithmes distribués qui décrivent les différentes phases de notre approche multi-agents de découverte et de sélection de services basée sur la confiance dans les réseaux sociaux. Comme illustré dans la figure 5.6 ci-dessous, il y a trois phases.

La première phase propose un algorithme distribué (voir Algo. 1) basé sur la confiance pour la découverte de services dans les réseaux sociaux. Cet algorithme décrit comment les agents raisonnent sur leurs propres actions, de raffiner une tâche en fonction de leurs compétences, et de se coordonner avec d'autres agents pour propager la recherche en se basant sur la confiance qu'ils accordent à leurs accointances dans le réseau social.

Dans la deuxième phase, nous décrivons la méthode de calcul de la confiance entre l'agent demandeur et un fournisseur non adjacent. Pour ce faire, nous présentons tout d'abord, l'algorithme de propagation qui décrit le mécanisme de transmission des valeurs de confiance entre les agents de la chaîne fournisseur-recommandeur. Puis, nous proposons un nouveau modèle probabiliste et contextuel pour calculer la confiance dans les réseaux sociaux. En plus de la confiance en la crédibilité sociale, nous intégrons dans

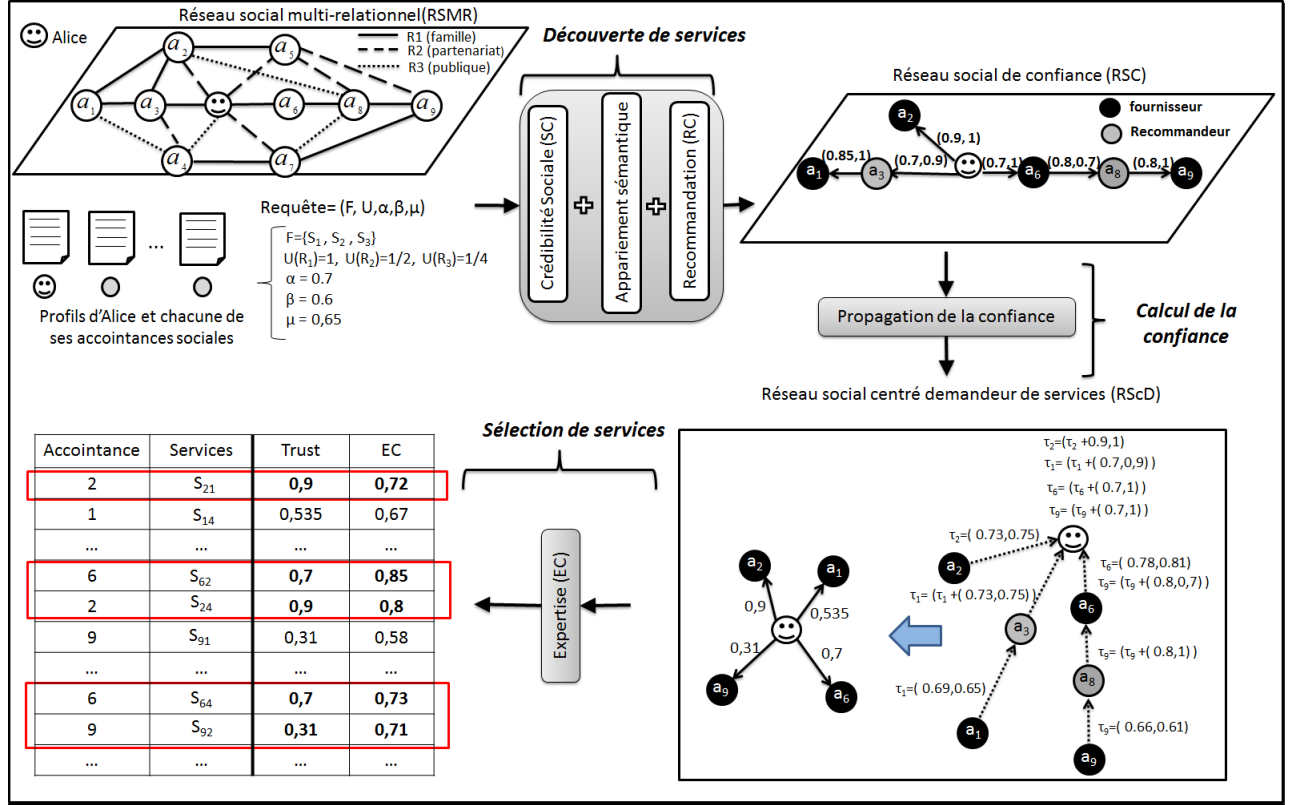


FIGURE 5.6 – Approche multi-agent de découverte et de sélection de services basée sur la confiance

le modèle probabiliste de calcul la confiance en la recommandation en fonction du rôle joué par les agents dans la chaîne.

La troisième phase porte sur la sélection des services par l'agent demandeur afin de répondre aux besoins et préférences de son utilisateur associé.

Avant de procéder à la description de l'algorithme de recherche, il convient de préciser les différents attributs et structures de données utilisés. Rappelons que chaque agent a_k propose un ensemble de services $S_k = \{s_{k1}, \dots, s_{km}\}$ et qu'il est caractérisé par un profil Pr_k ainsi que les deux tables de données dans lesquelles il maintient les informations relatives à ces accointances (voir Sect. 5.2.1). En outre, chaque agent a_k contient les variables suivantes instanciées durant le processus de découverte :

- Une variable d'état, notée $role_k$, qui représente le rôle joué par l'agent au cours du processus de recherche. Elle prend sa valeur dans l'ensemble $\{\perp, Req, Prov, Rec, Stop\}$ avec *Req* pour *demandeur*, *Prov* pour *fournisseur*, *Rec* pour *recommandeur*, et *Stop* pour *stoppeur*. Initialement, l'état de l'ensemble des agents est égal à $\perp$ à l'exception de l'agent demandeur a_r dont le statut est égal à *Req*.

- Une variable distance, notée $dist_k$, qui représente la distance qui sépare un agent a_k de l'agent demandeur a_r dans le graphe. Initialement, cette variable est égale à 0 pour l'agent demandeur a_r et $+\infty$ pour le reste des agents.
- Une variable père, notée $father_k$ dans laquelle l'agent a_k garde l'identité de son père dans la chaîne fournisseur-recommandeur. Initialement, cette variable est nulle pour tous les agents.
- Une variable compteur, notée $count_k$, qui sert à compter le nombre de messages reçus de ses accointances. Initialement, cette variable est égale à 0 pour tous les agents.
- Un ensemble d'agents, noté LTA_k , contenant la liste des agents dignes de confiance appartenant à SA_k . Initialement, LTA_k est un ensemble vide pour tous les agents.
- Un ensemble d'agents, noté LRA_k , contenant la liste des références potentielles appartenant à SA_k . Initialement, LRA_k est un ensemble vide pour tous les agents.
- Une variable pile, notée $pile_k$, qui contient l'ensemble des sous-buts générés par le module de planification. Initialement, cette variable est vide pour tous les agents.
- Un ensemble de services, noté $offer_k$ tel que $offer_k \in S_k$, qui contient l'ensemble des services offerts par a_k satisfaisant les besoins du demandeur de services. Initialement, $Offer_k$ est un ensemble vide pour tous les agents.
- Une pile, notée τ_k dans laquelle l'agent a_k maintient une séquence ordonnée des agents de la chaîne fournisseur-recommandeur relative à un service découvert. Initialement, cette pile est vide pour tous les agents.
- Une structure de donnée, noté $PSet_k$ instanciée uniquement pour l'agent demandeur dans laquelle il collecte tous les τ_p relatifs à chaque service découvert. Initialement, $PSet_k$ est vide.

5.3.1 Algorithme distribué basé sur la confiance pour la découverte de services dans les réseaux sociaux

L'objectif de cette phase est de découvrir des fournisseurs dignes de confiance au moyen d'un algorithme de recherche distribué basé sur la confiance (voir Fig. 5.6). Cet algorithme décrit la diffusion de la requête dans le réseau social par l'intermédiaire des systèmes de références. Les entrées de l'algorithme de découverte (voir Algo. 1) d'un

agent a_k ⁶ sont sa perception locale (i.e., SA_k) et une requête $Q = (F, U, \alpha, \beta, \mu)$. La requête est transmise par un utilisateur à son agent associé a_k . Prenant le rôle d'un demandeur de services ($role_k = Req$), a_k initialise sa distance à lui-même dans la chaîne fournisseur-recommandeur à 0 et l'ensemble $PSet_k$ à $\emptyset$ (voir Algo. 1 ligne 2 – 4) puis, lance le processus de découverte de services dans le réseau social.

Avant de propager la requête dans son voisinage, un agent a_k calcule la crédibilité sociale des agents qui vont être interrogés. Pour effectuer le calcul de confiance, une étape de mise à jour de la table PIT_k est nécessaire pour garantir la consistance de ses croyances par rapport aux informations courantes de chaque $a_j \in SA_k$ (i.e., le profil Pr_j , les accointances sociales SA_j et les services qu'il offre avec leurs valeurs de QoS) (voir Algo. 1 lignes 5 – 6 et 20 – 24). Ensuite, a_k calcule pour chaque a_j sa crédibilité sociale $SC(a_k, a_j)$ à partir de toutes les valeurs des mesures sociales (voir Sect. 4.4 et Algo. 1 lignes 7 – 9).

Être socialement digne de confiance n'est pas suffisant pour être interrogé. C'est la raison pour laquelle a_k doit vérifier avant de propager la requête si a_j pouvait participer à la résolution de la requête ou non, soit en offrant un service approprié, soit en recommandant un fournisseur (voir Algo. 1 lignes 10 – 16). Nous utilisons aussi le seuil de la confiance en la recommandation β pour contrôler la propagation de la requête dans le réseau social (voir Algo. 1 lignes 15 – 16).

L'application de ce seuil nous permet d'une part, de réduire l'espace de recherche (c'est à dire, les nœuds visités dans le graphe) diminuant ainsi le coût de calcul et d'autre part, de garantir la réception de fournisseurs de bonne qualité recommandés par des agents dignes de confiance.

La réception d'un message déclenche l'exécution d'une procédure en fonction du type du message reçu. Un agent a_k qui reçoit un message PROPAGATE, compare la valeur de la distance reçue à celle qu'il possède (initialement égale à $+\infty$). Si la valeur reçue est inférieure à la valeur de la distance actuelle, alors a_k met à jour $dist_k$ avec cette nouvelle valeur. Celle-ci représente sa plus courte distance depuis l'agent demandeur dans la chaîne fournisseur-recommandeur.

La chaîne fournisseur-recommandeur est une séquence ordonnée d'agents dont la première extrémité est le demandeur de services et l'autre extrémité est un fournisseur de services digne de confiance. Les agents intermédiaires sont soit des fournisseurs soit des recommandeurs. Cette chaîne nous permet de définir récursivement l'ancêtre et le descendant de chaque agent participant en se basant sur la relation père-fils. La condition de test dans l'algorithme garantit la construction de l'arbre de plus court chemin qui empêche les cycles dans la propagation de la requête. L'agent a_k initialise

6. L'indice k identifie l'agent exécutant le code.

Algorithm 1: Algorithme distribué basé sur la confiance pour la découverte de services

```

1  Procedure(Propagation( $F, U, \alpha, \beta, \mu$ ))
2    if ( $role_k == Req$ ) then
3       $dist_k \leftarrow 0$ ;
4       $PSet_k \leftarrow \emptyset$ ;
5      for all  $a_j \in SA_k$  do
6        Request( $a_k, a_j, \emptyset$ );
7      wait ( $count_k == |SA_k|$ );
8       $count_k \leftarrow 0$ ;
9       $LTA_k \leftarrow \{a_j \in SA_k \mid SC(a_k, a_j) \geq f_\alpha(dist_k)\}$ ;
10   if ( $role_k \in \{Req, Pro\}$ ) then
11     for all  $a_j \in LTA_k$  do
12       if ( $\exists s \in F$  and  $s_{jl} \in PIT_k.S_j \mid matching(s, s_{jl}) == True$ ) then
13         Propagate( $a_k, a_j, (F, U, \alpha, \beta, \mu, dist_k)$ );
14       else
15         if ( $\exists s \in F \mid RC(a_k, a_j, s) \geq f_\beta(dist_k)$ ) then
16           Propagate( $a_k, a_j, (F, U, \alpha, \beta, \mu, dist_k)$ );
17   if ( $role_k == Rec$ ) then
18     for all  $a_j \in LRA_k$  do
19       Propagate( $a_k, a_j, (F, U, \alpha, \beta, \mu, dist_k)$ );

20 Procedure(Receive_Request( $a_j, a_k, \emptyset$ ))
21   Inform( $a_k, a_j, (S_k, SA_k, Pr_k)$ );

22 Procedure(Receive_Inform( $a_j, a_k, (S, SA, Pr)$ ))
23    $count_k \leftarrow count_k + 1$ ;
24    $PIT_k.set(a_j, S, SA, Pr)$ ;

25 Procedure(Receive_Propagate( $a_j, a_k, (F, U, \alpha, \beta, \mu, dist)$ ))
26   if ( $dist_k > dist + 1$ ) then
27      $dist_k \leftarrow dist + 1$ ;
28      $father_k \leftarrow a_j$ ;
29     for all  $a_j \in SA_k$  do
30       Request( $a_k, a_j, \emptyset$ );
31   wait ( $count_k == |SA_k|$ );
32    $count_k \leftarrow 0$ ;
33   resolveQuery( $F$ );
34   if ( $offer_k \neq \emptyset$ ) then
35      $role_k \leftarrow Pro$ ;
36      $\tau_k.add(a_k, s_{kl}, Sp(s_{kl}), Di(s_{kl}))$ ;
37     Inform( $a_k, father_k, \tau_k$ );
38      $LTA_k \leftarrow \{a_j \in SA_k \mid SC(a_k, a_j) \geq f_\alpha(dist_k)\}$ ;
39     Propagation( $F, U, \alpha, \beta, \mu$ );
40   else
41      $LRA_k \leftarrow \{a_j \in SA_k \mid SC(a_k, a_j) \geq f_\alpha(dist_k) \text{ and } \exists s \in F \text{ and } s_{jl} \in$ 
42        $PIT_k.S_j \mid matching(s, s_{jl}) == True\}$ ;
43     if ( $LRA_k \neq \emptyset$ ) then
44        $role_k \leftarrow Rec$ ;
45       Propagation( $F, U, \alpha, \beta, \mu$ );
46     else
47        $role_k \leftarrow Stop$ ;

```

la valeur de la variable père $father_k$ par l'identité de l'expéditeur du message et met à jour sa base de croyances (voir Algo. 1 lignes 25 – 32). Puis, a_k engage la phase de résolution de la requête.

Cette résolution repose sur une fonction itérative (voir Fonc. 1) dans laquelle l'agent raisonne et planifie conjointement afin de déterminer les services utiles correspondant aux fonctionnalités requises (voir Algo. 1 ligne 33). À la fin de cette opération de résolution, deux résultats sont possibles :

Fonction 1: $resolveQuery(F)$

```

1   $offer_k \leftarrow \emptyset$ ;
2   $pile_k \leftarrow \emptyset$ ;
3  repeat
4    for all  $s_{kj} \in S_k$  do
5      for all  $s_i \in F$  do
6        if ( $matching(s_i, s_{kj})$ ) then
7           $offer_k = offer_k \cup \{s_{kj}\}$ ;
8           $Update(F)$ ;
9     $pile_k \leftarrow planning(F)$ ;
10 until ( $pile_k = \emptyset$ );

```

- 1) L'agent a_k dispose d'un ou plusieurs services qui présentent les mêmes fonctionnalités que les services requis par le demandeur, il obtient ainsi le rôle de *fournisseur* ($role_k = Pro$) et initialise pour chaque service proposé un τ_k comme suit : $\tau_k = \{a_k, s_{kl}, Sp(s_{kl}), Di(s_{kl})\}$ où a_k est son identifiant, s_{kl} le service fourni, et $(Sp(s_{kl}), Di(s_{kl}))$ sont les valeurs de qualité de service de s_{kl} . Ensuite, a_k envoie des messages INFORM contenant chacun un τ_k différent correspondant à un service offert à son père $father_k$, i.e., l'agent auprès duquel a_k a reçu le message PROPAGATE. Puis, il détermine l'ensemble des agents LTA_k auxquels il va envoyer la requête.

Pour faire partie de cet ensemble, un agent $a_j \in SA_k$ doit avoir une crédibilité sociale $SC(a_k, a_j)$ supérieure à $f_\alpha(dist_k)$ (voir Algo. 1 lignes 35–38). Nous faisons l'hypothèse que les valeurs des seuils α et β sont proportionnelles à la longueur d de la chaîne fournisseur-recommandeur. Cela reflète la tendance naturelle qu'ont les individus à préférer les services et les recommandations fournis par leurs amis plutôt que par les amis de leurs amis, à moins que ces derniers soient plus dignes de confiance. Pour mettre en évidence cette relation, nous considérons une fonction qui ajuste les valeurs de α et β en fonction de la longueur d de la chaîne.

Cela nous permet de garantir *la terminaison* et *la convergence* de l'algorithme de découverte de services dans le réseau social. Étant donné que les valeurs initiales des seuils sont fixées par le demandeur de services ($d = 0$) dans la requête (voir définition 2.4), elles seront incrémentées localement par a_k ($d > 0$). Cela nous permet de limiter la recherche dans le réseau tout en assurant une crédibilité sociale et une capacité de recommandation élevées pour les agents contactés. Soit $seuil \in \{\beta, \alpha\}$ et soit D le diamètre du graphe représentant l'excentricité maximale des agents, c'est-à-dire la plus grande distance possible qui existe entre deux agents dans le réseau. La fonction d'ajustement du seuil est définie par :

$$f_{seuil}(dist_k) = (1 - seuil) \times (1 - e^{\frac{-dist_k}{D}}) + seuil$$

La courbe représentant l'évolution du seuil en fonction de la distance (voir Fig. 5.7) est similaire à la courbe de charge asymptotique d'un condensateur. Cela signifie que plus la longueur de la chaîne fournisseur-recommandeur est grande, plus le demandeur de services est exigeant.

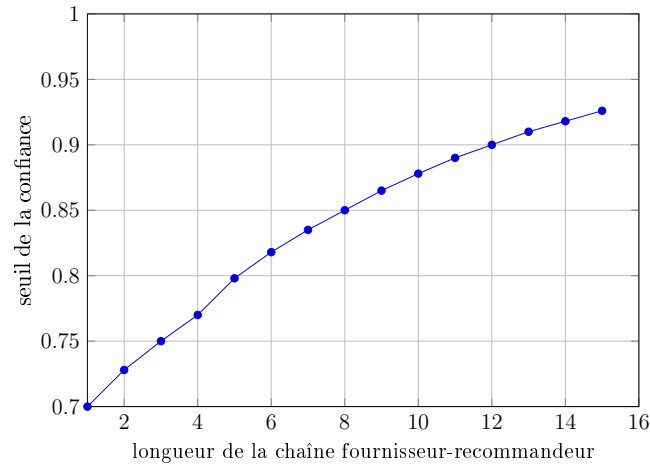


FIGURE 5.7 – L'évolution du seuil de la confiance en fonction de la distance

L'agent a_k exécute la procédure PROPAGATION dans laquelle il envoie un message PROPAGATE à tous les agents dans LTA_k (voir Algo. 1 lignes 39 et 10 – 16).

- 2) L'agent a_k ne possède aucun des services requis, dans ce cas il détermine l'ensemble LRA_k constitué d'agents qui sont non seulement dignes de confiance mais qui offrent aussi au moins un des services requis (voir Algo. 1 ligne 41). Si LRA_k est vide c'est à dire, a_k n'a aucune référence pertinente à recommander (i.e., soit ils ne sont pas suffisamment dignes de confiance, soit ils ne proposent pas de bons services), il arrête le processus de propagation de la recherche et prend le rôle *stoppeur* ($role_k = Stop$) (voir Algo. 1 lignes 45 – 46). Sinon, a_k devient un

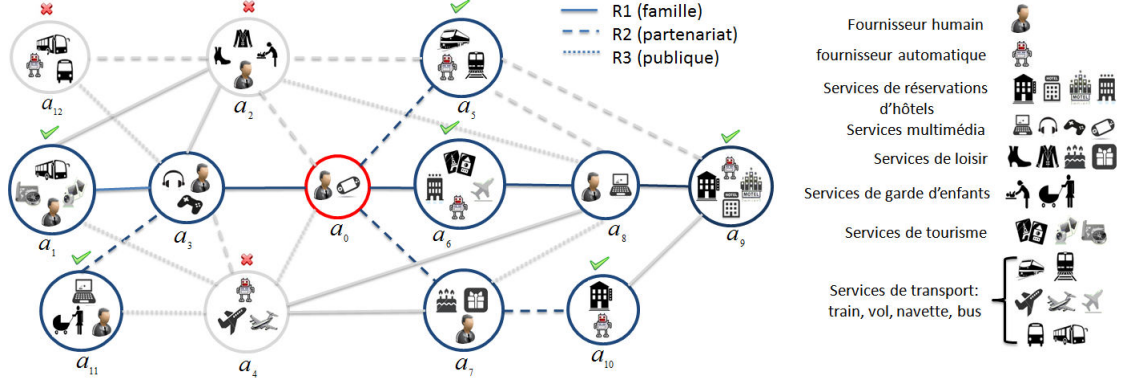


FIGURE 5.8 – Réseau social de confiance

recommandeur ($role_k = Rec$) et exécute à son tour la procédure PROPAGATION en envoyant un message PROPAGATE à tous les agents contenus dans LRA_k (voir Algo. 1 lignes 42 – 44 et 17 – 19). Dans ce cas, bien que a_k ne fournisse pas le service requis, il participe au processus de découverte en conduisant à des références pertinentes.

Maintenant, nous allons analyser la complexité de notre algorithme de découverte de services. Soit s le nombre d'agents dans le réseau social. La recherche est propagée, dans le pire cas, dans un graphe complet. Lors de la phase de mise à jour, un agent a_k contacte toutes ses accointances pour leur demander les informations utiles au calcul de la confiance. On compte alors $s - 1$ messages envoyés. Les accointances répondent chacune par un message contenant les informations demandées. On compte aussi $s - 1$ messages envoyés. Une fois la confiance calculée, dans le pire cas toutes les accointances de a_k sont dignes de confiance. Ainsi, a_k envoie de nouveau $s - 1$ messages propageant la requête. Les accointances commencent ensuite à leur tour de faire la phase de mise à jour. Dans le pire cas, toutes leurs accointances sont aussi dignes de confiance. On a alors $(s - 1)(s - 2)$ messages au plus ($s - 1$ agents dont chacun envoie un message à $s - 2$ accointances). En conséquence, la complexité de communication de notre protocole, i.e. le nombre total de messages émis au pire cas, est de l'ordre de $\mathcal{O}(s^2)$.

À la fin de cette première phase de découverte de services appliquée à l'exemple de motivation (voir Sect. 1), nous obtenons comme le montre la figure 5.8, un graphe de références appelé aussi *réseau social de confiance RSC*. Il s'agit d'un réseau social mono-relationnel, orienté et annoté dans lequel les fournisseurs dignes de confiance sont découverts. Ce réseau social de confiance est un réseau formé par les différentes chaînes fournisseur-recommandeur reliant le demandeur de services à chacun des fournisseurs découverts dans le réseau social initial. Les relations entre les agents intermédiaires sont

pondérées par leurs valeurs de confiance respectives. À ce stade, les fournisseurs et les recommandeurs sont découverts respectivement en fonction de leur degré de crédibilité sociale et de leurs aptitudes à fournir des bonnes recommandations. Cependant, l'aspect non fonctionnel (i.e., les valeurs des attributs de QoS) exprimé dans la confiance en l'expertise n'est pas encore pris en considération.

5.3.2 Algorithme de propagation de confiance et modèle probabiliste et contextuel de calcul

Une relation de confiance entre deux agents peut être représentée par une arête reliant ces deux agents dans le graphe dont le poids correspond à la valeur de cette relation, comme c'est le cas du réseau social de confiance RSC. Cependant, si les deux agents ne sont pas directement reliés, c'est à dire qu'ils n'ont pas interagi dans le passé, une inférence pourrait être appliquée pour estimer leur relation de confiance. Le calcul de la confiance, qui s'établit entre deux agents distants dans le graphe, est une opération d'agrégation des valeurs de confiance entre les agents intermédiaires tout le long des différents chemins possibles reliant ces deux agents dans le graphe. Dans cette phase, nous décrivons d'abord l'algorithme de propagation des valeurs de confiance dans les chaînes fournisseur-recommandeur. Puis, nous présentons la méthode de calcul relative au calcul de la confiance donnant naissance à un nouveau Réseau Social centré Demandeur de services appelé le RScD.

Algorithme 2: Algorithme de propagation des valeurs de confiance

```

1 Procedure(RECEIVE_INFORM( $a_j, a_k, \tau$ ))
2    $\tau_k \leftarrow \text{upgrade}(\tau)$ ;
3   if ( $\text{role}_k == \text{Req}$ ) then
4      $PSet_k \leftarrow PSet_k \cup \{\tau\}$ ;
5   else
6     Inform( $a_k, \text{father}_k, \tau_k$ );

```

Algorithme de propagation de confiance

Sur réception d'un message INFORM de la part d'un agent a_j contenant τ , a_k applique la fonction « upgrade » (voir Algo. 2 ligne 2). Selon le rôle de a_j , a_k insère un nouveau

élément dans la pile. Si a_j est un recommandeur (voir Fonc. 2 lignes 2 – 3), alors a_k empile dans τ l'élément $\{role_k, SC(a_k, a_j), RC(a_k, a_j, s)\}$ où $role_k$ est son rôle, $SC(a_k, a_j)$ et $RC(a_k, a_j, s)$ sont respectivement les valeurs de confiance en la crédibilité sociale et en la recommandation de a_j . Sinon si a_j est un fournisseur alors, a_k empile dans τ l'élément $\{role_k, SC(a_k, a_j), 1\}$. Dans ce cas, a_j se comporte comme un nœud relai dans la chaîne en transmettant des valeurs de confiance. Étant fournisseur, il ne génère pas de recommandations et ainsi $RC(a_k, a_j, s) = 1$ (voir Fonc. 2 lignes 4 – 5). Une fois l'opération d'empilement terminée, le rôle de l'agent a_k est testé. Si a_k est le demandeur de services ($role_k == Req$), alors il ajoute cette pile à son ensemble $PSet_k = PSet_k \cup \{\tau\}$ (voir Algo. 2 lignes 3 – 4). Sinon si a_k est un agent intermédiaire dans la chaîne, il applique à son tour le même mécanisme en transmettant le τ mis à jour à son père jusqu'à ce qu'il atteigne l'agent demandeur (voir Algo. 2 lignes 5 – 6). Comme exemple, prenons la chaîne de la figure 5.9 dans laquelle l'agent a_0 reçoit la pile τ issue du fournisseur a_9 contenant les rôles, les valeurs SC et RC des différents agents intermédiaires de la chaîne liant le fournisseur a_9 à l'agent demandeur a_0 : $\tau = \{a_9, s_{9l}, Sp(s_{9l}), Di(s_{9l}), \{role_8, SC(a_8, a_9), 1\}, \{role_6, SC(a_6, a_8), RC(a_6, a_8, s_{9l})\}\}$. a_0 , qui est l'agent demandeur, à son tour insère $\{role_0, SC(a_0, a_6), 1\}$, puis ajoute cet enregistrement à son $PSet_0$ pour calculer la confiance de a_9 . Ce moyen de propagation progressive des valeurs de confiance entre les agents est le mécanisme original de notre système de références distribué dans lequel la diffusion de la recherche est assurée via la navigation dans le graphe et avec une prise de décision décentralisée.

Fonction 2: upgrade(τ)

```

1  $role \leftarrow top(\tau).getRole()$  ;
2 if ( $role == Rec$ ) then
3    $\tau.push(\{role_k, SC(a_k, a_j), RC(a_k, a_j, s)\})$  ;
4 else
5    $\tau.push(\{role_k, SC(a_k, a_j), 1\})$  ;
```

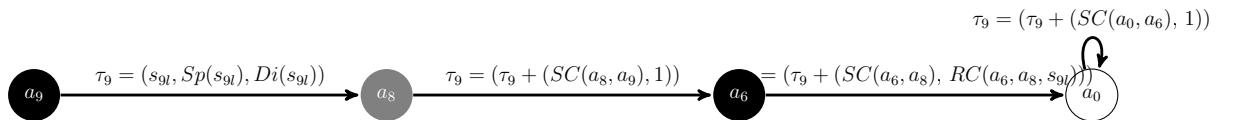


FIGURE 5.9 – Exemple de chaîne, dans laquelle chaque agent est caractérisé par son identifiant et son rôle.

À la fin de l'algorithme de propagation, le demandeur de services, noté a_r , maintient dans $PSet_r$ des enregistrements relatifs à chaque fournisseur découvert a_p . Ces enregistrements contiennent les différentes valeurs de confiance de chaque agent intermédiaire

dans la chaîne $(a_r, \dots, a_p)$ conduisant à a_p (cf. Fig. 5.9).

Les différentes chaînes de fournisseur-recommandeur forment un arbre dont la racine est l'agent demandeur. Notre algorithme de propagation évite les références redondantes (c'est à dire, un agent recommandé par deux recommandeur différents) et les chaînes cycliques car chaque agent a un seul prédécesseur (i.e., père) et ne participe qu'une seule fois au processus de propagation.

Modèle probabiliste et contextuel pour le calcul de la confiance

La propagation des valeurs de confiance pour calculer la confiance d'un agent cible en fonction de la confiance des autres agents, est un problème important dans les réseaux sociaux. Dans la littérature, les méthodes de propagation de la confiance [Golb 06b, Gray 03, Hang 09, Kute 07, Liu 11, Wang 06c, Wang 13, Zhen 14] s'appuient sur le mécanisme de référencement pas à pas pour la transmission des valeurs de confiances entre les agents.

La plupart de ces méthodes [Hang 09, Liu 11, Wang 06c] proposent des opérateurs d'agrégation pour calculer la confiance entre des agents non adjacents provenant de chemins multiples. Dans [Hang 09], les auteurs ont proposé une approche algébrique pour agréger la confiance dans les réseaux sociaux. Pour ce faire, ils ont défini trois opérateurs ; un opérateur de concaténation pour les structures séquentielles le long d'un seul chemin, un opérateur d'agrégation pour les structures parallèles, et un opérateur de sélection qui sélectionne le chemin qui offre la plus grande valeur de confiance. Gray et al. [Gray 03] ont étudié la propagation de la confiance dans les réseaux ad hoc mobiles. Ils calculent la concaténation comme étant la moyenne de la confiance le long du chemin de références en tenant compte de la profondeur. Les valeurs de confiance issues de différents chemins sont agrégées en sélectionnant ceux les plus dignes de confiance.

D'autres travaux ont adopté des stratégies de calcul simples comme la moyenne arithmétique. Par exemple, Golbeck et Hendler ont présenté dans [Golb 06b] des algorithmes de propagation multiplicatifs le long d'un chemin (par exemple, la méthode de la moyenne pondérée). Cela reflète l'intuition naturelle que la valeur de la confiance est dégressive sur un chemin (notre approche plus nuancée capte cette intuition aussi).

D'autres approches plus compliquées de propagation de la confiance ont été proposées. Dans [Guha 04], Guha et al. ont élaboré une plateforme pour traiter non seulement la propagation de la confiance, mais aussi la propagation de la méfiance. Dans la même perspective, Victor et al. ont proposé dans [Vict 09] un modèle de propagation de la confiance qui prend en compte la confiance floue, la méfiance floue, l'absence d'informations de confiance et aussi les informations de confiance contradictoires. Des études antérieures ont adopté une approche probabiliste [Kute 07, Wang 13, Zhen 14] pour

calculer la confiance. Dans [Kute 07], Kuter et Golbeck ont proposé un algorithme de calcul de confiance appelé *SUNNY* qui utilise une technique d'échantillonnage probabiliste pour estimer la fiabilité de la valeur de confiance d'un agent à partir de plusieurs sources désignées dans le réseau social. Wang et al. ont proposé [Wang 13] une autre approche probabiliste appelée *SocialTrust* qui est sensible au contexte social pour calculer la confiance dans les réseaux sociaux. Dans ce travail, les auteurs ont modélisé le réseau d'agents comme un réseau probabiliste. En appliquant la loi de la probabilité totale, ils estiment la confiance entre deux agents distants selon trois types de structures atomiques de propagation (séquentielle, parallèle et mixte) dans les réseaux sociaux. Cependant, ce processus de calcul est centralisé car il suppose que le "trustor" dispose d'une connaissance globale de la structure du réseau ce qui n'est pas notre cas.

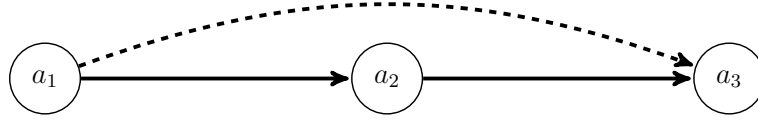


FIGURE 5.10 – Structure séquentielle de propagation de confiance.

Ces approches ont permis d'améliorer la précision de la prédiction de la confiance. Cependant, d'autres facteurs qui influent la prédiction de la confiance doivent être pris en compte, tels que la fiabilité des recommandations des agents intermédiaires. Pour ce faire, nous intégrons la confiance en la recommandation dans le modèle probabiliste de calcul proposé dans [Wang 13] et cela en fonction du rôle joué par chacun des agents intermédiaires. De plus, étant donné que chaque agent a un seul père dans le réseau social de confiance, la confiance entre les agents non adjacents est calculée uniquement pour une seule structure de propagation qui est la chaîne fournisseur-recommandeur (par exemple, Fig. 5.10). Décrivons tout d'abord le mécanisme de calcul de la confiance dans une structure de chaîne selon le modèle probabiliste de [Wang 13]. Soit $P(a_2)$ la probabilité qui désigne la confiance en a_2 et $P(\neg a_2)$ la probabilité que a_2 ne soit pas digne de confiance. Soit $P(a_2|a_1)$ la probabilité conditionnelle qui désigne la confiance de a_1 en a_2 et $P(a_2|\neg a_1)$ la probabilité conditionnelle qui désigne la confiance des autres agents connectés en a_2 sauf a_1 . En utilisant les informations contenues dans $PSet_1$ et en appliquant la loi de la probabilité totale, nous avons :

$$\begin{aligned} P(a_2) &= P(a_2|a_1)P(a_1) + P(a_2|\neg a_1)P(\neg a_1) \\ &= P(a_2|a_1)P(a_1) \end{aligned}$$

Dans notre structure de chaîne, a_1 est le seul agent en interaction avec a_2 ainsi, $P(a_2|\neg a_1) = 0$. De la même manière nous avons :

$$\begin{aligned} P(a_3) &= P(a_3|a_2)P(a_2) + P(a_3|\neg a_2)P(\neg a_2) \\ &= P(a_3|a_2)P(a_2) \end{aligned}$$

Au final, la confiance calculée entre l'agent demandeur a_1 et l'agent d'extrémité a_3 dans cette chaîne est définie comme suit :

$$P(a_3) = P(a_3|a_2)P(a_2|a_1)P(a_1)$$

Plus généralement, la confiance calculée entre l'agent demandeur a_r et un fournisseur a_j pour le service s_{pl} notée $Trust(a_r, a_j, s_{pl})$ est donnée par cette expression :

$$Trust(a_r, a_j, s_{pl}) = P(a_r) \times P(a_{r+1}|a_r) \times P(a_{r+2}|a_{r+1}) \times \dots \times P(a_j|a_{j-1})$$

où $P(a_r)$ représente la confiance de l'agent demandeur a_r .

Comme ce dernier n'a pas de prédécesseur dans la chaîne, nous supposons que $P(a_r) = 1$. Si nous ignorons les rôles pris par les agents intermédiaires tout le long de la chaîne, notre problème de calcul de la confiance est similaire à celui de Wang et al. [Wang 13]. Cependant, dans notre approche chaque agent participant à la résolution de la requête possède un rôle particulier, il peut être soit un fournisseur soit un recommandeur. Nous proposons de pondérer la crédibilité sociale d'un agent en fonction de sa capacité de recommandation exprimée par la confiance en la recommandation.

En effet, l'évaluation de la qualité d'un service ne dépend pas uniquement des valeurs de ses attributs de QoS et de la sociabilité de son fournisseur mais aussi de la qualité des recommandations faites par les agents intermédiaires qui ont conduit à sa découverte. Nous supposons que cette pondération impacte la confiance des agents recommandeurs de la chaîne car ces derniers se comportent comme des relais dans la chaîne en transmettant des informations et pas en générant de recommandations. La probabilité conditionnelle $P(a_{k+1}|a_k)$ qui désigne la confiance de a_{k+1} selon le point de vue de a_k correspond à une agrégation de SC et RC tel que :

$$P(a_{k+1}|a_k) = \begin{cases} SC(a_k, a_{k+1}) & \text{si } role_{k+1} = Pro \\ SC(a_k, a_{k+1}) \times RC(a_k, a_{k+1}, s) & \text{si } role_{k+1} = Rec \end{cases}$$

En se fondant sur les résultats de calcul de la confiance calculée pour chaque fournisseur a_p , nous construisons un nouveau réseau social appelé le RScD qui est centré

demandeur de services et modélisé par un graphe direct pondéré $G' = (V', E')$, où V' représente l'ensemble des fournisseurs découverts plus le demandeur de services et E' est l'ensemble des arêtes (voir Fig. 5.6). Une arête $(a_r, a_p) \in E'$ implique l'existence d'une relation de confiance sociale entre a_r et a_p dont le poids correspond à la valeur de la confiance calculée. Les résultats détaillés sont stockés dans une structure de données $PSet_r^*$ calculée comme suit :

$$\begin{aligned} \mathcal{F}_1 : PSet_r &\rightarrow PSet_r^* \\ \tau_p &\mapsto (\tau_p, Eval(a_r, s_{pl}), Trust(a_r, a_p, s_{pl})) \end{aligned}$$

où $\mathcal{F}_1$ est une fonction qui prend en entrée un enregistrement $\tau_p \in PSet_r$ relatif à un fournisseur digne de confiance découvert a_p auquel elle ajoute $Eval(a_r, s_{pl})$, l'évaluation de la qualité d'exécution attribuée par a_r au service s_{pl} , et la confiance calculée $Trust(a_r, a_p, s_{pl})$.

5.3.3 Sélection de services

Cette phase consiste à présélectionner les meilleurs fournisseurs répondant aux besoins et préférences de l'utilisateur au moyen d'un classement établi sur la base des services découverts dans la phase précédente. Ce classement permet de déterminer les candidats potentiels qui vont participer à la phase de composition de services. L'agent a_r calcule localement la confiance en l'expertise de chaque fournisseur découvert puis construit un ensemble ordonné noté $PSel_r$ en utilisant la fonction suivante :

$$\begin{aligned} \mathcal{F}_2 : PSet_r^* &\rightarrow PSel_r \\ (\tau_p, Eval(a_r, s_{pl}), Trust(a_r, a_p, s_{pl})) &\mapsto \\ (\tau_p, Eval(a_r, s_{pl}), Trust(a_r, a_p, s_{pl}), EC(a_r, a_p, s_{pl})) & \end{aligned}$$

où $\mathcal{F}_2$ est une fonction qui prend en entrée un enregistrement $\tau_p \in PSet_r^*$ auquel elle ajoute $EC(a_r, a_p, s_{pl})$ la confiance en l'expertise de l'agent a_r pour a_p . Ensuite, a_r sélectionne pour chaque service requis les fournisseurs dont la confiance en l'expertise est supérieure à μ comme le montre la figure 5.6.

5.4 Conclusion

Dans ce chapitre, nous avons donné une description globale de notre approche distribuée de découverte et de sélection de services dans les réseaux sociaux avec ses

différentes étapes. Nous avons précisé notamment la méthodologie de conception que nous avons adoptée, le modèle d'organisation sociale basé sur les systèmes de références distribués, les rôles des agents impliqués et la définition des interactions entre eux. Ce modèle d'organisation sociale nous a permis d'élaborer un algorithme distribué basé sur la confiance pour la découverte de services assurant la propagation de la recherche dans le réseau social.

Nous avons développé également une nouvelle méthode s'appuyant sur un modèle probabiliste pour inférer la confiance entre les agents non adjacents tout en tenant compte des rôles des agents intermédiaires.

Finalement, nous avons décrit dans ce chapitre une architecture combinant les technologies des agents cognitifs et celles des service Web sur laquelle notre approche s'appuie. Cette architecture permet à l'agent de raisonner sur la fiabilité et l'honnêteté des autres agents guidant son comportement lorsqu'il veut interagir avec eux. Par ailleurs, nous avons décrit la structure interne de l'architecture et formalisé le protocole de propagation et le diagramme d'activité.

Dans le chapitre suivant, nous introduisons notre approche multi-agents pour la composition de services fondée sur la confiance et la formation de coalitions.

Chapitre 6

Composition de services Web fondée sur la confiance et la formation de coalitions

Sommaire

6.1	Approches multi-agents pour la composition de services	122
6.2	Modèle multi-agents	127
6.2.1	Définition des concepts	127
6.2.2	Modèle multi-agents basé sur un broker	129
6.3	Description du processus de formation de coalitions	130
6.3.1	Phase de génération de coalitions initiales	131
6.3.2	Phase de sélection des membres	133
6.3.3	Phase de sélection de la meilleure coalition	139
6.4	Conclusion	139

Précédemment, nous avons décrit notre approche multi-agents fondée sur la confiance pour la découverte de services dans les réseaux sociaux. Dans le présent chapitre, nous nous intéressons à la résolution du problème de la composition de services à l'aide d'une formation de coalitions. Notre processus de formation de coalitions est un processus incrémental, dynamique et recouvrant engageant des agents égoïstes et équipés d'un ensemble de services avec leurs valeurs de qualité de service (QoS). Ces agents coopèrent pour satisfaire la requête de l'utilisateur en se basant sur une prise de décision décentralisée guidée par la confiance. À partir d'une étude bibliographique, nous

montrons dans la première section l'intérêt de l'utilisation du paradigme agent dans la composition de services en mettant l'accent sur les approches qui se fondent sur une formation de coalitions. Nous précisons les limites de chacune de ces approches pour mettre en valeur notre contribution. Dans la deuxième section, nous définissons les concepts utilisés dans ce chapitre et nous donnons un aperçu général de notre modèle multi-agents basé sur un borker en précisant les différents rôles considérés. La troisième section est dédiée à la description des trois phases de notre nouvelle approche multi-agents de composition de services. Nous introduisons une nouvelle composante de confiance appelée la confiance en la coopération permettant aux membres d'une coalition d'évaluer la fiabilité d'un candidat avant d'accepter son adhésion. Nous spécifions également les algorithmes et le protocole d'interaction utilisé pour la coordination des agents.

6.1 Approches multi-agents pour la composition de services

La notion de composition de services Web vient de la nécessité de satisfaire un besoin complexe qui ne peut pas être satisfait par un seul service. Elle se réfère à des mécanismes qui favorisent la collaboration et la combinaison des services Web disponibles afin de créer un service Web composite à valeur ajoutée. Cela permet la définition de nouvelles applications de plus en plus complexes, qui ont le potentiel de réduire le temps et l'effort de développement [Ponn 02]. La composition de services Web a fait l'objet d'une attention considérable aussi bien dans le milieu industriel que académique. Plusieurs initiatives ont été menées avec l'intention de fournir des plateformes comme eFlow [Casa 01] de HP, METEOR-S [Agga 04] développée à l'université de Géorgie et SELF-SERV [Bena 03] développée conjointement par l'université de New South Wales et l'université de Queensland of Technology et des langages tels que WSCI¹, BPML², BPEL4WS³, BPSS⁴ et XPDL⁵ pour automatiser et faciliter le processus de composition de services Web.

Malgré ces efforts, les résultats obtenus ne parviennent pas à satisfaire les préférences non fonctionnelles des demandeurs de services. Une des raisons provient du fait que

-
1. <http://www.w3.org/TR/ws-ci/>
 2. http://www.ebpml.org/bpml_1_0_june_02.htm
 3. <http://www.ibm.com/developerworks/webservices/library/ws-spec.html>
 4. http://www.service-architecture.com/articles/web-services-business_process_specification_schema_bpss.html
 5. <http://www.xpdl.org/>

les techniques proposés n'intègrent pas les dimensions sociales entre le demandeur et les fournisseurs comme la confiance mutuelle et les informations résultant de leurs expériences antérieures. De plus, ils ne prennent pas en considération l'autonomie des fournisseurs au niveau de la prise de décision lorsqu'ils veulent choisir leur partenaires dans une composition de services.

Pour faire face à ces lacunes, de nombreuses recherches [[Amor 03](#), [Hend 01](#), [Huhn 02](#), [Huhn 03](#), [Nami 04](#), [Syca 01](#), [Sree 04](#), [Maxi 05b](#), [Maam 05](#)] ont étudié l'intérêt de combiner les technologies des services web et des agents dans une seule entité. L'agentification des services Web vise à exploiter les capacités des agents comme l'autonomie, la coopération, la sociabilité, l'auto-organisation et le raisonnement. Ceci dans l'objectif d'améliorer le comportement des services Web sur les points suivants :

- Les services Web sont des composants logiciels isolés, auto-descriptifs et inconscients de leur environnement. En revanche, les agents sont sociables, auto-organisés et disposent de capacités bien développées pour exprimer et raisonner sur la confiance.
- Les services Web sont des composants passifs jusqu'à ce qu'ils soient invoqués. Les agents sont par contre des entités autonomes et pro-actives qui peuvent trouver d'autres agents pour interagir avec eux.
- Les standards actuels de la technologie service Web proposent des interactions simples qui sont principalement structurées sous la forme de questions-réponses. Les standards d'agents proposent des protocoles plus complexes pour modéliser et spécifier d'autres formes d'interactions comme la conversation, la négociation et la coordination.

Dans le contexte de la composition de services, l'utilisation d'une approche multi-agents à été traitée dans la littérature avec différentes formulations qui ne posent pas nécessairement les mêmes contraintes, les mêmes hypothèses et n'utilisent pas les mêmes techniques et méthodes de résolution. Par exemple, de nombreux efforts de recherche ont été menés pour apporter des solutions via la planification à la composition de services Web [[Paik 06](#), [Ponn 02](#), [Siri 04](#), [Tong 11](#), [Xu 11](#)]. L'idée est de traiter le problème de composition de services comme un problème de recherche dans un graphe en incluant les relations de dépendance entre les services.

Ce sont des approches automatiques qui considèrent respectivement les données d'entrée et les données de sortie désirées comme l'état initial et l'état final du problème, et les services Web disponibles comme l'ensemble des actions et opérateurs que le planificateur peut appliquer pour passer d'un état à un autre. Cependant, ces approches

sont centralisées autour d'un moteur de composition.

La centralisation de ces approches découle du principe même d'orchestration qu'elles implémentent. De plus, la sélection des services dans ces approches est effectuée à partir des annuaires centralisés de services Web (e.g. UDDI, ebXML). Elle est soit statique (les services sont pré-sélectionnés), soit basée sur un simple mécanisme d'appariement sémantique entre les besoins de l'utilisateur et les descriptions fonctionnelles et non-fonctionnelles des services publiés. Cependant, elle ne garantit pas que les services identifiés répondent aux contraintes et préférences exprimées par l'utilisateur. La réalité est que le nombre croissant de services disponibles limite la portée des annuaires de services causant un manque d'évolutivité et introduisant un point de défaillance unique dans le système.

D'autres approches multi-agents pour la composition de services se sont intéressées à la capacité de coordination des agents. [Char 13] ont proposé une approche dynamique basée sur la chorégraphie pour la composition de services. Les agents impliqués sont capables de raisonner sur leurs propres actions (ou sur les services qu'ils offrent), de décomposer une tâche en fonction de leurs compétences et de se coordonner dynamiquement avec d'autres agents pour satisfaire les besoins de l'utilisateur. Bien que le processus de composition de services soit réalisé d'une manière dynamique et distribué, la sélection de services porte uniquement sur les préférences de l'utilisateur exprimées par les contraintes et ne considère pas celles des agents participants.

Dans ce cas, les agents manquent d'autonomie au niveau du choix des partenaires avec qui ils préfèrent coopérer dans une composition de services. [Sial 11] ont présenté une approche multi-agents pour la sélection des services Web composites en se basant sur une méthode de raisonnement par cas dirigée par les attributs de QoS. La coordination entre les agents se fait par négociation à l'aide du protocole contract-net. Ce protocole précise la manière avec laquelle les agents interagissent et échangent les informations entre eux pour réaliser le service composite. Le processus de sélection de services est un processus centré-utilisateur qui ne considère que les préférences de l'utilisateur en termes de valeurs de QoS et ignore la contribution de la dimension sociale.

Une sélection efficace de services ne doit pas être uniquement axée sur les préférences de l'utilisateur mais elle doit inclure aussi celles des fournisseurs de services. [Maam 05] ont présenté une approche orientée contexte pour la composition de services Web basée sur l'utilisation d'agents. Le terme contexte reflète le point de vue des services plutôt que des utilisateurs. Lors de l'exécution, Maamar et al. ont considéré trois types d'agents : un agent composite, un agent maître et un agent service. Ces agents sont engagés dans des conversations pour sélectionner les services Web qui vont participer au processus de composition en fonction du contexte de l'exécution et du modèle global de la composition. Suite à la réception d'une nouvelle demande pour rejoindre le

service composite, les auteurs ont décrit l'évolution de la conversation en fonction de la décision de l'agent maître. Cependant, ils n'ont pas présenté le mécanisme d'évaluation des demandes qui guide les prises de décision dans un contexte d'organisation de tâches. [Wang 06a] ont proposé un modèle de workflow à base d'agents pour améliorer la collaboration inter-entreprises. Ce modèle est géré par un système multi-agents comprenant quatre types d'agents : les agents brokers, les agents planificateurs de workflow, les agents d'ontologie et les agents de service. Les agents brokers sont en charge de la coordination de l'exécution des tâches entre les agents de service. Les agents planificateur de workflow indiquent aux agents brokers l'ordre correct des tâches à exécuter. Les agents d'ontologie font correspondre les besoins des utilisateurs avec les services disponibles puis les remettent aux agents brokers pour satisfaire les tâches sur cet ensemble de services. Les agents de service exécutent les workflows dans un environnement Web. Bien que ce modèle fournisse un cadre général de coordination couvrant les processus internes et externes d'une entreprise, il souffre d'un manque de flexibilité et d'évolutivité pour satisfaire les processus complexes.

Dans les approches mentionnées ci-dessus, l'utilisation des agents se limite à leurs capacités d'interaction, de raisonnement et de gestion de connaissances. Ils ne coopèrent pas au sein d'une organisation pour réaliser directement la composition de services mais agissent comme des intermédiaires qui coordonnent la construction du service composite. Cette manière de procéder ne met pas en valeur la contribution de la dimension sociale dans le processus de composition de services car celle-ci provient principalement de la capacité de coopération et des informations qui en résultent.

Dans la littérature, la coopération entre les agents apparaît sous différents modèles organisationnels (par exemple, les hiérarchies, les coalitions, les teamworks, les sociétés [Horl 05]). La sélection du modèle organisationnel approprié dépend du type de l'agent, de la particularité du domaine d'application et des caractéristiques de l'environnement. Un travail récent [Asl 13] a montré l'intérêt d'utilisation des coalitions pour former des communautés optimales et stables maximisant l'efficacité individuelle et celle du groupe. Nous pensons que la formation de coalitions [Sand 97, Sheh 98, Cail 02, Akni 00, Akni 04] est bien adaptée à la modélisation de la coopération entre des agents égoïstes ayant un objectif commun qui est la composition de services dans un environnement distribué comme le réseau social. Dans ce contexte, trois défis se présentent :

- 1) Intégrer *la dimension sociale* dans le processus de composition : généralement, les utilisateurs préfèrent les fournisseurs qui proposent non seulement les services requis mais qui sont aussi dignes de confiance. La plupart des travaux antérieurs sur la formation de coalitions pour la composition de services [Ermo 03, Mull 06, Hong 09] ne prennent pas en compte la contribution de la dimension sociale au cours du processus

de formation de coalitions. Les agents peuvent décider de participer ou non à une coalition et d'accepter ou de refuser l'adhésion d'un partenaire sans aucune considération liée à la confiance.

2) Accorder *l'autonomie aux fournisseurs* pour décider avec qui ils coopèrent : aussi bien les demandeurs que les fournisseurs sont des agents égoïstes qui doivent avoir la possibilité de décider localement le choix de leurs partenaires dans le service composite. Griffiths et Luck ont présenté dans [Grif 03] un modèle de formation de coalitions qui combine la confiance et la motivation pour générer des coopérations réussies entre des agents égoïstes. Bien que ce modèle prenne en compte la dimension sociale, son processus de prise de décision associé à la sélection d'un candidat manque d'une vision globale. Les auteurs accordent de l'autonomie aux candidats pour décider de participer ou non à la coalition en se basant sur leur confiance envers l'initiateur de la coalition. Cependant, les membres de la coalition ne peuvent pas décider d'accepter ou de refuser l'adhésion d'un candidat.

3) Incorporer *le dynamisme* dans le processus de formation de coalitions : une coalition dynamique signifie qu'elle peut évoluer au cours du temps en fonction de l'état de ses membres. Bourdon et al. ont proposé dans [Bour 09] un modèle multi-agents basé sur un broker capable de trouver des coalitions dignes de confiance centrées fournisseurs pour accomplir une composition de services Web. Bien que leur modèle intègre la dimension sociale en considérant la confiance entre les agents et accorde l'autonomie aux fournisseurs dans le choix de leurs partenaires, il ne permet pas aux agents de rejoindre ou de quitter les coalitions dynamiquement. Un problème se pose quand un membre est insatisfait de la présence d'un agent spécifique et ne peut pas quitter sa coalition courante pour en rejoindre une autre.

Dans ce chapitre, nous proposons un nouveau modèle multi-agents basé sur un broker pour une composition dynamique de services. Le processus de composition est réalisé par des agents égoïstes équipés d'un ensemble de services avec les valeurs de leurs attributs de QoS. Ils coopèrent dans une formation de coalitions pour fournir collectivement plusieurs services composites répondant à la requête complexe de l'utilisateur. Pour relever les défis susmentionnés, nous intégrons :

- Premièrement la dimension sociale en utilisant d'une part un mécanisme de confiance permettant à chaque agent d'évaluer la fiabilité des autres agents impliqués avant de coopérer avec eux et d'autre part, les informations résultant des anciennes coalitions.
- Deuxièmement, nous accordons l'autonomie aux fournisseurs pour décider avec qui ils coopèrent dans le service composite en se basant sur la confiance en la

coopération. L'idée est de donner aux fournisseurs la capacité de participer activement au PFC. La satisfaction de tous les membres donne lieu à une coalition stable et efficace. Si certains membres ne sont pas satisfaits, nous devons disposer d'un mécanisme qui permet de retrouver la stabilité de la coalition.

- Troisièmement, nous intégrons le dynamisme dans notre PFC pour permettre aux membres insatisfaits de quitter la coalition.

6.2 Modèle multi-agents

Cette section est divisée en deux parties. Nous définissons dans un premier temps les concepts que nous allons utiliser pour la description du processus de formation de coalitions puis nous donnons un aperçu général de notre modèle multi-agents basé sur un broker en précisant les différents rôles considérés.

6.2.1 Définition des concepts

Le réseau social utilisé dans ce chapitre est le réseau de confiance résultant de la phase de découverte (voir Fig. 5.6). Il est modélisé par un graphe orienté où chaque utilisateur est représenté par un agent égoïste et autonome, et chaque arête entre deux utilisateurs correspond à une relation sociale de confiance entre eux.

Definition 6.1 (Réseau de confiance) Soit $A = \{a_1, a_2, \dots, a_s\}$ un ensemble d'agents et $E \subseteq A \times A$ un ensemble d'arêtes, un réseau social $G = \langle A, E \rangle$ est un graphe orienté et connecté où chaque arête $(a_k, a_j) \in E$ représente une relation sociale de confiance entre a_k et a_j .

Ce réseau social mono-relationnel est considéré comme un système multi-agents dans lequel les agents sont équipés d'un ensemble de services avec les valeurs de QoS. Ces agents agissent de manière autonome afin de satisfaire les besoins de leurs utilisateurs associés. Dans ce chapitre, nous simplifions l'architecture agent définie dans la section 5.2.1.

Definition 6.2 (Agent) Un agent $a_k \in A$ est une entité autonome définie par un tuple $a_k = \langle S_k, Trust, CC, EC, \lambda Inf_k, \lambda Sup_k, \gamma_k, Blist_k \rangle$ où :

- S_k est l'ensemble des m_k services offerts avec les valeurs de QoS,
- $Trust(a_k, a_j)$ est la confiance de l'agent a_k en un autre agent a_j . Elle est définie comme étant l'agrégation de la confiance en la crédibilité sociale et la confiance en la recommandation (pour plus de détail voir 5),
- $CC(a_k, a_j)$ représente la confiance en la coopération de l'agent a_k en un autre agent a_j ,
- $EC(a_k, a_j, s)$ désigne la confiance en expertise que l'agent a_k accorde à un service s offert par un fournisseur a_j qui est définie comme l'agrégation des trois attributs de qualité de services à savoir : la spécialisation, la disponibilité et la qualité (pour plus de détail voir Sect. 4.5),
- λInf_k et $\lambda Sup_k \in [0, 1]$ sont respectivement, les seuils inférieur et supérieur de la confiance en la coopération,
- $\gamma_k \in [0, 1]$ est le seuil de la confiance en une coalition.
- $Blist_k$ est une liste noire formée par un ensemble particulier d'agents avec lesquels a_k s'interdit de coopérer car ils ne respectent pas les termes d'utilisation des services qu'il offre.

Rappelons qu'une requête utilisateur contient un ensemble fini $F = \{f_1, f_2, \dots, f_n\}$ de fonctionnalités requises représentant le service composite (voir définition 2.4). Cet ensemble ne fait pas référence à des services spécifiques, mais plutôt à un ensemble d'interfaces sémantiques décrivant la capacité des services adéquats pour les satisfaire. Le but de la composition est de trouver des services quiinstancient cet ensemble.

Definition 6.3 (Service) *Un service s est un n -uplet $\{in, out, f, q^1, \dots, q^d\}$ où in est un ensemble d'entrées requises pour utiliser le service, out est un ensemble de sorties prévues à la fin de l'exécution du service, f est une interface sémantique décrivant la fonctionnalité offerte, et $q^1, \dots, q^d$ sont les valeurs des d attributs de QoS.*

Nous désignons par $A_p \subseteq A$ l'ensemble des fournisseurs dignes de confiance, par $A_i = \{a_k \mid a_k \in A_p \text{ and } a_k.s.f \equiv f_i\}$ l'ensemble des fournisseurs offrant un service avec la fonctionnalité f_i et par $A_F = \bigcup_{i=1}^n A_i$ l'ensemble de fournisseurs offrant des services pour toutes les fonctionnalités requises dans F .

Comme plusieurs agents peuvent offrir différents services avec différentes valeurs de QoS pour les fonctionnalités requises, la formation de coalitions entre eux peut résoudre le problème de la composition.

Definition 6.4 (Coalition) Soit F un ensemble de fonctionnalités requises. Une coalition $c = \{x_1, x_2, \dots, x_n \mid 1 \leq i \leq n, x_i = a_k \text{ and } a_k \in A_i\}$ est un ensemble d'agents qui instancient F .

Durant le processus de formation de coalitions (PFC), les agents s'organisent dans des coalitions où chacun d'entre eux est en mesure de satisfaire une ou plusieurs fonctionnalités requises. Une coalition qui ne contient pas un ensemble complet d'agents qui instancient toutes les fonctionnalités requises est appelée *coalition intermédiaire*. Une coalition intermédiaire, notée c_z , est une instantiation partielle de F telle que $c_z = \{x_1, x_2, \dots, x_n \mid x_i \in \{a_k, f_i\} \text{ et } a_k \in A_i\}$. Durant le PFC, le contenu d'une coalition intermédiaire évolue. La transition d'une coalition intermédiaire c_z à une autre c_{z+1} est effectuée à l'aide d'un *proposal*.

Definition 6.5 (Proposal) Un *proposal* $\phi = \{y_1, y_2, \dots, y_n \mid y_i \in \{a_k, \emptyset\} \text{ et } a_k \in A_i\}$ représente soit une demande d'adhésion soit une offre d'adhésion. Dans le cas d'une demande d'adhésion, $\phi = \{y_1, y_2, \dots, y_n \mid \exists! y_i \text{ tel que } y_i = a_k \in A_i \text{ et } \forall j \neq i, y_j = \emptyset\}$. Dans le cas d'une offre d'adhésion, $\phi = \{y_1, y_2, \dots, y_n \mid y_i = x_i \text{ si } x_i \equiv a_k, y_i = \emptyset \text{ sinon}\}$.

6.2.2 Modèle multi-agents basé sur un broker

Basé sur les travaux de Klusch et Sycara [Klus 01], notre modèle multi-agents est un modèle basé sur un broker comme représenté sur la Figure 6.1. Il englobe trois rôles différents :

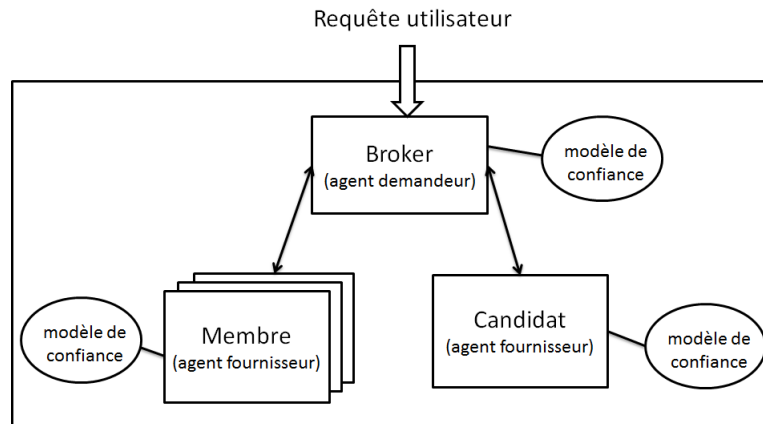


FIGURE 6.1 – Notre modèle multi-agents basé sur un broker

- *Candidat* : un candidat est un fournisseur de services susceptible de rejoindre une coalition.
- *Membre* : un membre est un fournisseur de services qui est attribué à une ou à plusieurs coalitions.
- *Broker* : un broker est un demandeur de services qui se charge de la mise en relation des candidats avec les membres durant le PFC et orchestre les interactions entre eux. En cas de plusieurs coalitions, il choisit la coalition qui a la meilleure valeur d'expertise.

Les comportements associés à chaque rôle vont être décrits en détail dans la section suivante à l'aide des protocoles d'interactions.

6.3 Description du processus de formation de coalitions

Habituellement, la composition de services est effectuée à la suite d'une phase préalable de découverte de services. La phase de découverte de services décrite dans le chapitre 5 permet à l'agent demandeur a_r d'identifier un ensemble A_p de fournisseurs dignes de confiance sur lequel il va lancer le processus de formation de coalitions (PFC). Cet ensemble de fournisseurs est situé dans un arbre (voir Fig. 6.2(b)), appelé *réseau social de confiance (RSC)*, résultant du réseau social initial (voir Fig. 6.2(a)) où le poids d'une arête représente la valeur de confiance entre deux agents. Si nous considérons l'agent demandeur a_r comme la racine de l'arbre, chaque fournisseur $a_k \in A_p$ est localisé par une chaîne d'agents dignes de confiance appelée *chaîne fournisseur-recommandeur*. Une chaîne fournisseur-recommandeur est une séquence père-fils d'agents commençant à l'agent demandeur a_r et conduisant à un agent fournisseur a_k dans laquelle les agents intermédiaires sont soit des fournisseurs soit des recommandeurs. La figure 6.3 illustre cette structure arborescente en utilisant une représentation à plusieurs couches où l'agent demandeur a_r est la racine et chaque couche représente un niveau dans l'arbre.

Dans le RSC, les agents coopèrent dans une formation de coalitions pour fournir des services composites qui répondent à la requête complexe de l'utilisateur. Notre PFC est original car il est :

- *Incrémental* : Le PFC se fait d'une manière itérative où à chaque itération un seul agent peut rejoindre la coalition en favorisant en premier lieu les fournisseurs

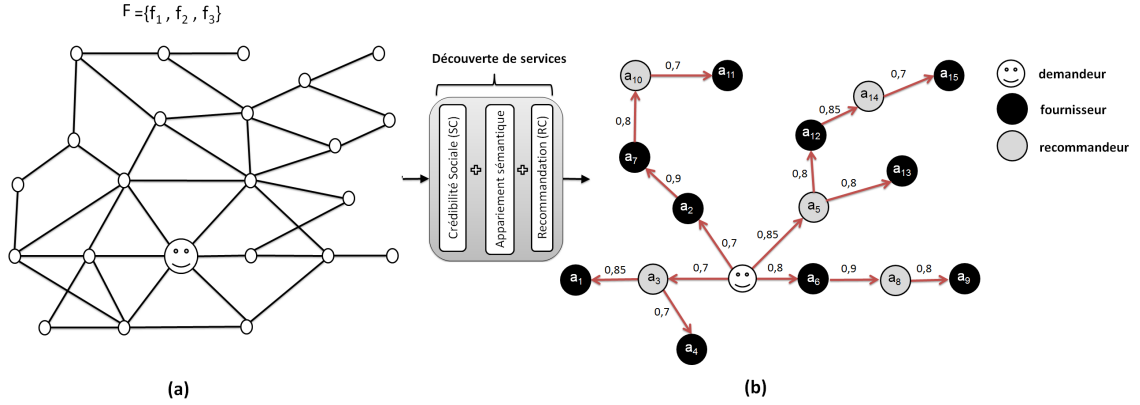


FIGURE 6.2 – (a) Réseau social et (b) Réseau social de confiance : représentation plane

découverts dans la première couche du RSC. Le processus se poursuit en cherchant des fournisseurs dans la seconde couche et ainsi de suite.

- *Dynamique* : Les agents peuvent rejoindre et quitter leurs coalitions de façon autonome à tout moment en fonction des messages reçus et de leurs décisions locales basées sur la confiance.
- *Overlapping* : Étant donné qu'un fournisseur peut offrir plus d'un service, il peut appartenir à plusieurs coalitions au même temps.

Le PFC se compose de trois phases organisées en ordre séquentiel : la phase de génération des coalitions initiales, la phase de sélection des membres et la phase de sélection de la meilleure coalition. Dans chacune de ces phases, différentes actions sont exécutées. Chaque phase produit un résultat qui est utilisé comme entrée de la phase suivante.

6.3.1 Phase de génération de coalitions initiales

Le but de cette phase est de générer un ensemble de coalitions initiales $\mathcal{C}$ dans le RSC. Les entrées de l'algorithme 1 sont l'ensemble des fonctionnalités requises F , l'ensemble des fournisseurs A_p et le réseau social de confiance RSC. Initialisant $\mathcal{C}$ à $\emptyset$ et son rôle à *brok*, a_r commence par l'identification des fournisseurs dans la première couche RSC(1) qui est à une profondeur 1 de lui. Ensuite, il attribue chaque fournisseur a_k à une nouvelle coalition initiale c_z (par exemple, dans la Fig. 6.3, $\mathcal{C} = \{c_1 = \{a_2\}, c_2 = \{a_6\}\}$) et envoie à a_k un message INFORM contenant l'indice de la coalition dans laquelle il est un membre (voir Algo. 1 lignes 4 – 5). Nous faisons l'hypothèse que les agents ont

la bonne volonté pour rejoindre la coalition initiale vide. Cependant, dans la suite ils seront complètement autonomes dans leur prises de décisions de rejoindre ou de quitter une coalition intermédiaire ou d'accepter une demande d'adhésion. Sur réception d'un message INFORM contenant l'indice d'une coalition (voir Algo. 3 ligne 12), un agent a_k prend le rôle de membre (voir Algo. 3 ligne 13). Chaque nouvelle coalition initiale est ajoutée à $\mathcal{C}$ (voir Algo. 1 ligne 6).

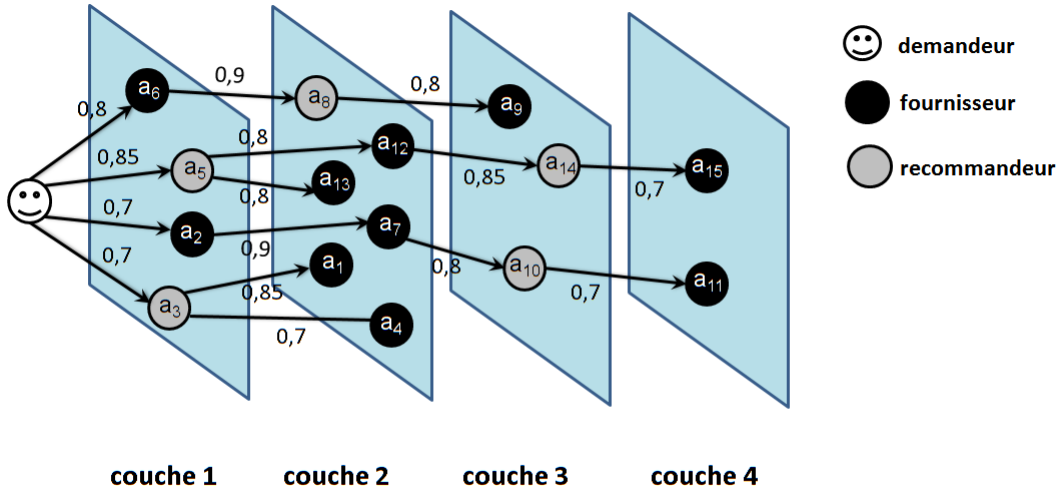


FIGURE 6.3 – Réseau social de confiance : représentation multi-couche.

À la fin de cette phase, le broker détermine un ensemble $\mathcal{C}$ de coalitions initiales sur lesquelles il lancera la deuxième phase du PFC qui est la phase de sélection des membres.

Algorithme 1: Algorithme de génération de coalitions initiales

Input: F est l'ensemble des fonctionnalités requises, A_p est l'ensemble des fournisseurs, RSC est le réseau social de confiance où RSC(1) est la première couche.

Variables: $\mathcal{C} = \{c_z\}$ est l'ensemble des coalitions, $role_r$ est un tableau de $|\mathcal{C}|$ valeurs $\in \{brok, mem, cand\}$ initialement fixées à *brok*.

```

1  $\mathcal{C} \leftarrow \emptyset$ ;
2  $z \leftarrow 1$ ;
3 for all ( $a_k \in RSC(1) \cap A_p$ ) do
4    $c_z = \text{new initial\_coalition}(a_k, F)$ ;
5   Inform( $a_r, a_k, (z, mem)$ );
6    $\mathcal{C}.add(c_z)$ ;
7    $z \leftarrow z + 1$ ;
8 return  $\mathcal{C}$ ;

```

/* voir Algo. 3 ligne 12 */

6.3.2 Phase de sélection des membres

L'objectif de cette phase est de compléter séquentiellement chacune des coalitions initiales $c_z \in \mathcal{C}$ par des membres fournissant des services nécessaires à la réalisation du service composite F . Pour plus de simplicité, nous limitons la description de cette phase à une seule coalition c_z .

La phase de sélection des membres est une phase itérative gérée par le broker a_r où, à la fin de chaque itération, un seul membre peut rejoindre c_z . Elle est basée sur une stratégie de recherche qui consiste à donner la priorité d'adhésion aux fournisseurs appartenant aux couches supérieures de l'arbre i.e., plus proches de la racine a_r , car ils sont plus dignes de confiance. Rappelons que notre PFC est dynamique c'est à dire, les agents peuvent rejoindre et quitter les coalitions intermédiaires lors de l'exécution. Par conséquent, il est essentiel d'appliquer un mécanisme de timeout $timer_r[z]$ associé à c_z qui permet au broker d'arrêter le processus s'il dure trop longtemps (voir Algo. 2 ligne 4).

Pour chaque couche $RSC(l)$, a_r identifie un ensemble de fournisseurs susceptibles de rejoindre c_z . Ceci se fait en déterminant d'abord l'ensemble des fonctionnalités non satisfaites $func_r[z]$ dans c_z et ensuite en identifiant les fournisseurs $candP_r[z]$ qui offrent des services avec ces fonctionnalités non satisfaites (voir Algo. 2 lignes 6 – 7). Dans le cas où il n'y a que des recommandeurs dans cette couche ou/et tous les fournisseurs identifiés ne proposent pas de services avec les fonctionnalités requises dans $func_r[z]$ ($candP_r[z] = \emptyset$), a_r étend l'identification des fournisseurs à la couche suivante $RSC(l + 1)$. Le passage d'une couche l à une autre couche $l + 1$ se fait si et seulement si toutes les fonctionnalités requises dans $func_r[z]$ sont satisfaites ou l'ensemble des fournisseurs identifiés $candP_r[z]$ est vide (voir Algo. 2 lignes 8 et 24). Dans le cas où a_r dispose d'un ensemble d'agents susceptibles de rejoindre c_z ($candP_r[z] \neq \emptyset$), il sélectionne un candidat parmi cet ensemble en se basant sur sa valeur de confiance et transmet sa demande d'adhésion aux membres de c_z .

La confiance est reconnue comme un mécanisme efficace pour la modélisation et le raisonnement sur l'honnêteté et la fiabilité des autres avant de coopérer avec eux [Grif 03, Adle 01, Al S 12, Cast 98, Mars 94]. Plus précisément, a_r ordonne les fournisseurs identifiés dans $candP_r[z]$ en se basant sur leurs valeurs de confiance (calculée dans la phase de découverte voir Sec. 5.3.1) et choisit le plus digne de confiance pour être candidat (voir Algo. 2 ligne 9). Puis, il déclenche une conversation de sélection d'un membre en engageant les membres de c_z et le candidat choisi. La description de cette conversation nécessite la définition de protocoles d'interactions afin de structurer les messages échangés entre les agents. La figure 6.4 représente le protocole de sélection d'un membre à l'aide du formalisme AUML.

Algorithme 2: Algorithme de sélection des membres

Input: A_p est l'ensemble des fournisseurs, $\mathcal{C} = \{c_z\}$ est l'ensemble des coalitions résultat de l'Algo 1, RSC est le réseau social de confiance, l_{max} est la couche maximale dans RSC.

Variables: $timer_r$ est un tableau de $|\mathcal{C}|$ entier, $candP_r$ est un tableau de $|\mathcal{C}|$ ensembles de fournisseurs candidats initialement $\emptyset$, $func_r$ est un tableau de $|\mathcal{C}|$ ensembles de fonctionnalités initialement $\emptyset$, $count_r$ est un tableau de $|\mathcal{C}|$ entier initialement fixés à 0, $reply_r$ est un tableau de $|\mathcal{C}|$ boolean initialement fixés à False.

```

1   $z \leftarrow 1$ ;
2  for all ( $c_z \in \mathcal{C}$ ) do
3       $l \leftarrow 1$ ;
4      Activate( $timer_r[z]$ );
5      while ( $(timer_r[z])$  and  $(\exists x \in c_z \mid x \equiv f)$  and  $(l \leq l_{max})$ ) do
6           $func_r[z] \leftarrow$  functionalities_identification( $c_z$ );
7           $candP_r[z] \leftarrow$  candidates_identification( $func_r[z]$ ,  $RSC(l) \cap A_p$ );
8          while ( $(candP_r[z] \neq \emptyset)$  and  $(\exists f_i \in func_r[z])$ ) do
9               $a_j \leftarrow \text{Argmax}_{a_t \in candP_r[z] \cap A_i} Trust(a_r, a_t)$ ;
10             for all ( $a_k \in c_z$ ) do
11                 Propose( $a_r, a_k$ , MembershipRequest( $z, \phi(a_j)$ )));          /* voir Algo. 3 ligne 1 */
12             wait ( $count_r[z] == |c_z|$ );
13              $count_r[z] \leftarrow 0$ ;
14             if ( $(OK) > \frac{|c_z|}{2}$ ) then
15                 Propose( $a_r, a_j$ , MembershipOffer( $z, \phi(c_z)$ )));          /* voir Algo. 4 ligne 1 */
16             else
17                  $candP_r[z] \leftarrow candP_r[z] \setminus \{a_j\}$ ;
18                  $reply_r[z] \leftarrow True$ ;
19                  $count_r[z] \leftarrow |c_z|$ ;
20             wait ( $reply_r[z] == True$ );
21              $reply_r[z] \leftarrow False$ ;
22             wait ( $count_r[z] == |c_z|$ );
23              $count_r[z] \leftarrow 0$ ;
24          $l \leftarrow l + 1$ ;
25      $z \leftarrow z + 1$ ;
26 return  $\mathcal{C}$ ;

```

Après avoir choisi un candidat a_j , le broker a_r envoie à chaque membre a_k de la coalition intermédiaire c_z un message PROPOSE contenant une demande d'adhésion de a_j (voir Algo. 2 lignes 10–11). Sur réception d'un message PROPOSE (voir Algo. 3 ligne 1), un membre a_k décide localement d'accepter ou de rejeter la demande d'adhésion de a_j .

Comme mentionné ci-dessus, le processus de prise de décision d'un agent est basé sur la confiance. Il pourrait également se baser sur le résultat de la vérification de la liste noire $Blist_k$ (voir Algo. 3 ligne 2) : un membre a_k peut refuser de coopérer avec un agent candidat a_j si ce dernier est présent dans sa liste noire. Ainsi, une demande d'adhésion d'un candidat inclus dans cette liste noire est directement rejeté suivi de l'envoi d'un message REJECT_PROPOSAL au broker a_r (voir Algo. 3 lignes 5–6). Sinon, a_k évalue la confiance en la coopération qu'il a en a_j en se basant sur l'historique des coalitions antérieures dans lesquelles ils étaient tous les deux impliqués. La confiance

Algorithme 3: Algorithme de demande d'adhésion

Variables: $role_k$ est un tableau de $|\mathcal{C}|$ valeurs $\in \{brok, mem, cand\}$ initialement fixées à $cand$, $Blist_k$ est une liste noire d'agents, $NbSoll_k[[A]]$ est un tableau d'entier.

```

1 Procedure (RECEIVE_PROPOSE( $a_r, a_k, MembershipRequest(z, \phi(a_j))$ ))
2   if ( $CC(a_k, a_j) > \lambda Sup_k$  and  $a_j \notin Blist_k$ ) then
3      $Accept\_Proposal(a_k, a_r, a_j)$ ;
4      $NbSoll_k[j] \leftarrow NbSoll_k[j] + 1$ ;
5   else
6      $Reject\_Proposal(a_k, a_r, a_j)$ ;

7 Procedure (RECEIVE_ACCEPT_PROPOSAL( $a_k, a_r, a$ ))
8    $count_r[z] \leftarrow count_r[z] + 1$ ;
9    $|OK| \leftarrow |OK| + 1$ ;

10 Procedure (RECEIVE_REJECT_PROPOSAL( $a_k, a_r, a$ ))
11    $count_r[z] \leftarrow count_r[z] + 1$ ;

12 Procedure (RECEIVE_INFORM( $a_r, a_k, (z, mem)$ ))
13    $role_k[z] \leftarrow mem$ ;

```

en la coopération $CC(a_k, a_j)$ d'un agent a_j selon le point de vue d'un autre agent a_k est définie par la formule 6.1.

Definition 6.6 (Confiance en la coopération (CC)) Soit $NbSoll_k[j]$ le nombre total de sollicitations de a_k auprès de a_j pour qu'il rejoigne sa coalition intermédiaire et soit $NbMem_k[j]$ le nombre total d'adhésions de a_j à la coalition intermédiaire dans laquelle a_k est un membre.

$$CC(a_k, a_j) = \begin{cases} 1 & \text{si } NbSoll_k[j] = 0 \\ \frac{NbMem_k[j]}{NbSoll_k[j]} & \text{sinon} \end{cases} \quad (6.1)$$

La décision d'un membre a_k d'accepter ou de rejeter la demande d'adhésion d'un candidat a_j repose sur sa valeur de confiance en la coopération : si elle dépasse λSup_k alors a_k accepte la demande d'adhésion de a_j en envoyant au broker a_r un message ACCEPT_PROPOSAL indiquant le résultat de sa décision. Il incrémente aussi la valeur de $NbSoll_k[j]$ (voir Algo. 3 lignes 2–4) sinon, il rejette la demande d'adhésion et envoie un message REJECT_PROPOSAL au broker (voir Algo. 3 lignes 5–6).

Sur réception d'un message ACCEPT_PROPOSAL indiquant une réponse positive à la demande d'adhésion, a_r incrémente son compteur de messages $counter_r[z]$ et le nombre de réponses positives (voir Algo. 3 lignes 7–9).

Sur réception d'un message REJECT_PROPOSAL indiquant une réponse négative à la demande d'adhésion, a_r incrémente uniquement son compteur de messages $counter_r[z]$ (voir Algo. 3 lignes 10–11).

Le broker attend un message de chaque membre de c_z (voir Algo. 2 ligne 12). Une fois

Algorithme 4: Algorithme d'offre d'adhésion

Variables: $evalC_k$ est un tableau de $|\mathcal{C}|$ évaluation de la confiance des coalitions, $NbMem_k[|A|]$ est un tableau d'entier.

```

1  Procedure (RECEIVE_PROPOSE( $a_r, a_k, MembershipOffer(z, \phi(c))$ ))
2  |   if ( $\exists a_t \in c$  and  $a_t \in Blist_k$ ) then
3  |   |   Reject( $a_k, a_r, (c, z)$ );                                     /* voir ligne 28 */
4  |   else
5  |   |    $evalC_k[z] \leftarrow evalC(a_k, c)$ ;
6  |   |   if ( $evalC_k[z] > \beta_k$ ) then
7  |   |   |   Accept_Proposal( $a_k, a_r, (c, z)$ );                       /* voir ligne 19 */
8  |   |   |    $role_k[z] \leftarrow mem$ ;
9  |   |   else
10 |   |   |   Reject_Proposal( $a_k, a_r, (c, z)$ );                       /* voir ligne 28 */
11 Procedure (RECEIVE_FAILURE( $a_k, a_r, (a_k \notin c)$ ))
12 |    $c_z.remove(a_k)$ ;
13 |   for all ( $t \mid a_k \in A_t$ ) do
14 |   |    $func_r[z] \leftarrow func_r[z] \cup \{f_t\}$ ;
15 |   |    $v \leftarrow 1$ ;
16 |   |   for all ( $v \leq l$ ) do
17 |   |   |    $candP_r[z] \leftarrow candP_r[z] \cup \{RSC(v) \cap A_t \setminus \{a_k\}\}$ ;
18 |    $count_r[z] \leftarrow count_r[z] + 1$ ;
19 Procedure (RECEIVE_ACCEPT_PROPOSAL( $a_k, a_r, (z, c)$ ))
20 |   for all ( $a_t \in c_z$ ) do
21 |   |   Inform( $a_r, a_t, (z, a_k \in c_z)$ );                           /* voir ligne 32 */
22 |    $c_z.add(a_k)$ ;
23 |    $func_r[z] \leftarrow func_r[z] \setminus \{f_i\}$ ;
24 |    $candP_r[z] \leftarrow candP_r[z] \setminus \{a_k\}$ ;
25 |   for all ( $a_t \in candP_r[z] \cap A_i$ ) do
26 |   |    $candP_r[z] \leftarrow candP_r[z] \setminus \{a_t\}$ ;
27 |    $reply_r[z] \leftarrow True$ ;
28 Procedure (RECEIVE_REJECT_PROPOSAL( $a_k, a_r, (z, c)$ ))
29 |    $candP_r[z] \leftarrow candP_r[z] \setminus \{a_k\}$ ;
30 |    $count_r[z] \leftarrow |c_z|$ ;
31 |    $reply_r[z] \leftarrow True$ ;
32 Procedure (RECEIVE_INFORM( $a_r, a_k, (z, a_j \in c)$ ))
33 |   if ( $(answer_k == KO)$  and  $(CT(a_k, a_j) < \lambda Inf_k)$ ) then
34 |   |    $role_k[z] \leftarrow cand$ ;
35 |   |    $Blist_k.add(a_j)$ ;
36 |   |   Failure( $a_k, a_r, a_k \notin c_z$ );                               /* voir ligne 11 */
37 |   else
38 |   |    $NbMem_k[l] \leftarrow NbMem_k[l] + 1$ ;
39 |   |   Confirm( $a_k, a_r, Yes$ );                                         /* voir ligne 40 */
40 Procedure (RECEIVE_CONFIRM( $a_k, a_r, Yes$ ))
41 |    $count_r[z] \leftarrow count_r[z] + 1$ ;

```

qu'il a reçu tous les messages (un message par membre), il met $counter_r[z]$ à 0 et établit une décision finale reposant sur la règle de la majorité (voir Algo. 2 lignes 13 – 14). Dans le cas où la majorité a accepté la demande d'adhésion du candidat a_j , a_r initie une

conversation avec l'agent candidat a_j en lui envoyant un message PROPOSE contenant une offre d'adhésion (voir Algo. 2 ligne 15). Sinon, a_r retire a_j de $candP_r[z]$ et met les paramètres de synchronisation $counter_r[z]$ à 0 et $reply_r[z]$ à *True* pour débloquer son état d'attente (voir Algo. 2 lignes 16 – 19).

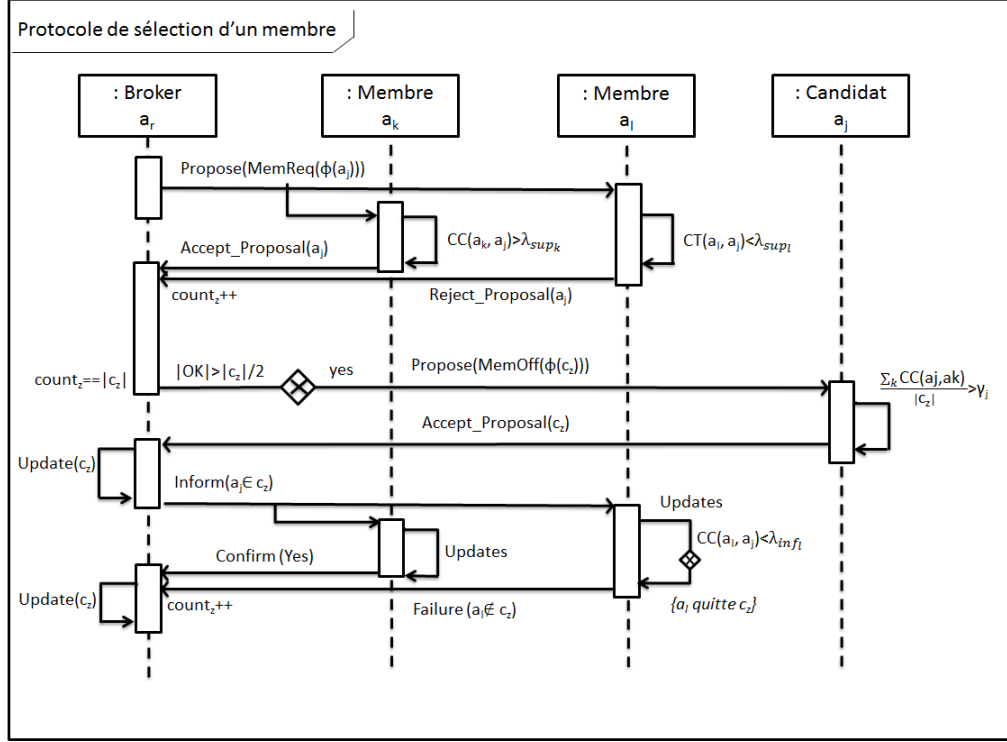


FIGURE 6.4 – Protocole AUMML de sélection d'un membre

Sur réception d'un message PROPOSE contenant une offre d'adhésion à une coalition intermédiaire c_z , un agent a_k vérifie si c_z contient un membre présent dans sa liste noire. Si c'est le cas, il rejette l'offre d'adhésion (voir Algo. 4 lignes 2 – 3). Sinon, il évalue la fiabilité de c_z en utilisant la mesure définie dans la formule 6.2 (voir Algo. 4 ligne 5).

Definition 6.7 (Confiance en la coalition (evalC)) Soit c_z une coalition intermédiaire et soit a_k un candidat. La confiance en la coalition est définie comme étant la moyenne arithmétique de la confiance en la coopération de ses membres.

$$evalC(a_k, c_z) = \frac{\sum_{a_t \in c_z} CC(a_k, a_t)}{|c_z|} \quad (6.2)$$

Une fois l'évaluation terminée, l'agent a_k accepte l'offre d'adhésion si sa confiance en la coalition c_z est supérieure à γ_k . Par conséquent, il prend le rôle d'un membre

($role_k[z] = mem$) et répond au broker avec un message `ACCEPT_PROPOSAL` indiquant le résultat de la décision (voir Algo. 4 lignes 6 – 8). Sinon, il répond au broker avec un message `REJECT_PROPOSAL` (voir Algo. 4 lignes 9 – 10).

En cas de rejet (voir Algo. 4 ligne 28), le broker a_r retire a_k de $candP_r[z]$ et met $counter_r[z]$ à $|c_z|$ (voir Algo. 4 lignes 29 – 30) pour débloquent son état d'attente (voir Algo. 2 ligne 22). En cas d'acceptation (voir Algo. 4 ligne 19), le broker a_r envoie à tous les membres de c_z un message `INFORM` annonçant le succès de la sélection de a_k (voir Algo. 4 lignes 20 – 21). Ensuite, il ajoute a_k à c_z , supprime la fonctionnalité satisfaite f_i de $func_r[z]$ et met à jour $candP_r[z]$ en supprimant tous les fournisseurs qui offrent un service avec la fonctionnalité f_i (voir Algo. 4 lignes 22 – 26).

Dans les deux cas, a_r met $reply_r[z]$ à *True* (voir Algo. 4 lignes 27 and 31) pour débloquent son état d'attente (voir Algo. 2 ligne 20).

Sur réception d'un message `INFORM` contenant une sélection réussie (voir Algo. 4 ligne 32), un membre $a_k \in c_z$ soit a déjà accepté la demande d'adhésion, dans ce cas il incrémente la valeur de $NbMem_k[j]$ et envoie un message `CONFIRM` à a_r pour exprimer son engagement (voir Algo. 4 lignes 37 – 39) soit n'a pas accepté la demande d'adhésion alors, il peut décider de quitter c_z . Cela arrive en cas de mécontentement de a_k concernant la sélection du nouveau membre i.e., sa confiance en la coopération est inférieure à λInf_k .

Le mécontentement d'un membre peut conduire à un groupe malsain et instable. Par conséquent, a_k quitte c_z en prenant le rôle de candidat ($role_k[z] = cand$), ajoute le nouveau membre à sa liste noire et envoie au broker un message `FAILURE` indiquant le résultat de sa décision (voir Algo. 4 lignes 33 – 36). La mise à jour dynamique de la liste noire des agents évite notre PFC l'effet *ping-pong* dans lequel la coalition reste instable i.e. les agents quittent et rejoignent la coalition indéfiniment. Sur réception d'un message `CONFIRM` contenant un engagement (voir Algo. 4 ligne 40), le broker a_r incrémente $counter_r[z]$ (voir Algo. 4 ligne 41).

Sur réception d'un message `FAILURE` d'un agent a_k pour quitter la coalition intermédiaire c_z (voir Algo. 4 ligne 11), a_r retire a_k de c_z puis, ajoute les fonctionnalités de ses services éliminés à l'ensemble des fonctionnalités non satisfaites dans c_z (voir Algo. 4 lignes 12 – 14).

Enfin, il enrichit l'ensemble des fournisseurs candidats $candP_r$ avec les fournisseurs offrant les fonctionnalités ajoutées de toutes les couches précédentes et incrémente $counter_r[z]$ (voir Algo. 4 lignes 15 – 19). Une fois que a_r a reçu tous les messages, (voir Algo. 2 lignes 21 – 22), il débloquent son état d'attente et met $counter_r[z]$ à 0. Le processus de sélection des membres d'une coalition c_z se répète et continue jusqu'à ce qu'une des trois conditions de terminaison est remplie :

- La coalition est complète i.e. toutes les fonctionnalités requises sont instanciés,
- Le timeout $timer_r[z]$ du PFC a expiré,
- La couche maximale de l'arbre est atteinte.

Le résultat de cette phase est un ensemble $\mathcal{C}$ de coalitions qui seront utilisées comme entrée de la phase suivante.

6.3.3 Phase de sélection de la meilleure coalition

Notre approche permet de trouver plusieurs coalitions et de choisir la plus appropriée pour l'utilisateur. Pour ce faire, le broker élimine en premier lieu les coalitions non complètes (voir Algo. 5 lignes 1 – 3) et établit un classement en fonction de leur niveau de confiance en l'expertise. La confiance en l'expertise d'une coalition donnée est définie comme étant la moyenne de l'expertise de ses membres (voir Algo. 5 lignes 4 – 5). Rappelons que la confiance en l'expertise d'un membre est un score calculé à partir des valeurs de QoS de(s) service(s) qu'il offre dans la coalition sous-jacente. La coalition qui a la meilleure confiance en l'expertise consitue le service composite qui sera transmis au demandeur.

Algorithme 5: Best Coalition Choice Algorithm

Input: $\mathcal{C} = \{c_z\}$ is the set of coalitions.

Variables: Exp_r est un tableau d'évaluation de confiance en l'expertise, c est la meilleure coalition.

```

1 for all ( $c_z \in \mathcal{C}$ ) do
2   if ( $\exists x \in c_z \mid x \equiv f$ ) then
3      $\mathcal{C}.remove(c_z)$ ;
4 for all ( $c_z \in \mathcal{C}$ ) do
5    $Exp_r[z] \leftarrow \frac{\sum_{a_t \in c_z} EC(a_t)}{|c_z|}$ ;
6  $c \leftarrow Argmax_{1 \leq z \leq |\mathcal{C}|} Exp[z]$ ;
7 return  $c$ ;
```

6.4 Conclusion

Dans ce chapitre, nous avons proposé un processus original de formation de coalitions fondé sur la confiance pour une composition de services dans les réseaux sociaux. Le processus de formation de coalitions est incrémental, dynamique et recouvrant engageant des agents égoïstes et équipés d'un ensemble de services avec leurs valeurs de

QoS. Les agents coopèrent en se basant sur un processus de prise de décision décentralisé guidé par la confiance en la coopération. Celle-ci permet aux membres (resp. aux candidats) d'une coalition d'évaluer la fiabilité d'un candidat (resp. d'une coalition) et de décider d'accepter ou pas sa demande d'adhésion (resp. l'offre d'adhésion).

L'approche proposée permet la génération de coalitions multiples qui répondent à la requête du demandeur de services. Elle accorde l'autonomie aux fournisseurs pour décider avec qui coopérer dans une composition de services. Ainsi, ils peuvent quitter leur coalition s'ils sont insatisfaits de l'adhésion d'un nouveau membre dans cette coalition.

Dans le chapitre suivant, nous présenterons la mise en œuvre et la validation de notre approche multi-agents de découverte, de sélection et de composition de services dans les réseaux sociaux.

Chapitre 7

Mise en œuvre et validation expérimentale

Sommaire

7.1	Choix techniques et implémentation	142
7.1.1	La plateforme JADE et les agents développés	143
	Présentation de la plateforme JADE	143
	Structure des agents développés	143
7.1.2	Choix du langage de stockage	144
7.2	Description des données	146
7.2.1	Description des jeux de données simulées	147
7.2.2	Description des jeux de données réelles	147
7.3	Évaluations empiriques et résultats	150
7.3.1	Implémentation	152
	Évaluation du temps de calcul	153
	Évaluation du coût de communication	153
7.3.2	Évaluation de la performance de notre approche de découverte de services	154
	Efficacité	155
	Efficience	157
	Qualité des résultats	159
7.3.3	Évaluation de la performance de notre approche de sélection de services	161

Méthodologie d'expérimentation	162
Utilité et satisfaction des demandeurs	162
7.3.4 Évaluation de la performance de notre approche de composition de services	166
7.4 Conclusion	169

Nous avons spécifié dans les précédents chapitres les algorithmes et les structures de données permettant de concevoir une approche multi-agent fondée sur la confiance pour la découverte, la sélection et la composition de services dans les réseaux sociaux. Dans le présent chapitre, nous présentons la mise en œuvre et la validation expérimentale de l'approche proposée. Dans une première partie, nous justifions les choix techniques, à savoir la plateforme de développement et le langage de stockage de données, adoptés pour l'implémentation d'un prototype. Dans une seconde partie, nous décrivons les données simulées et réelles utilisées pour notre validation expérimentale. Dans la dernière partie, nous présentons les expérimentations menées sur les données simulées et réelles pour évaluer successivement les performances de l'approche de découverte, de sélection et de composition de services. Nous présentons à chaque fois notre méthodologie d'expérimentation et les différentes politiques adoptées, et expliquons comment chacune d'elles influence la prise de décision des agents.

7.1 Choix techniques et implémentation

Pour implémenter notre approche SMA, nous avons utilisé la plateforme JADE¹. Nous avons conçu sur cette plateforme des agents interactifs capables de communiquer par envoi de messages en fonction des rôles et actes de langage que nous avons définis dans le chapitre précédent. Dans cette section, nous présentons les choix techniques effectués notamment le choix du langage de stockage de données.

1. Telecom Italia Lab. JADE 4.3 <http://jade.tilab.com/>.

7.1.1 La plateforme JADE et les agents développés

Il existe actuellement plusieurs plateformes pour le développement des systèmes multi-agents, par exemple Madkit², Zeus³, FIPA-OS⁴, JACK⁵, JADE, etc. La plupart de ces plateformes ne proposent pas de solutions qui combinent les technologies agent et services Web et facilitent l'utilisation des protocoles d'interaction, à l'exception de la plate-forme JADE. C'est pour cette raison que nous avons décidé d'utiliser cette plateforme dans le développement de notre application.

Présentation de la plateforme JADE

JADE (Java Agent DEvelopment Framework) est une plateforme logicielle écrite en langage JAVA. Elle simplifie la mise en œuvre de systèmes multi-agents à travers une middleware conforme aux spécifications FIPA et à travers un ensemble d'outils graphiques qui prennent en charge les phases de débogage et de déploiement. Un SMA développé avec JADE peut être réparti sur plusieurs machines et la configuration peut être contrôlée via une interface graphique à distance.

Outre l'abstraction du comportement de l'agent, JADE fournit :

- une exécution simple mais puissante selon les besoins et les capacités des agents,
- une communication pair à pair basée sur le paradigme d'échange asynchrone de messages,
- un service pages jaunes supportant les mécanismes de publication, de souscription, de découverte de services,
- de nombreuses autres fonctionnalités avancées qui facilitent le développement d'un système distribué.

Structure des agents développés

Chaque agent correspond à un *thread* JAVA. D'un point de vue architecture logicielle, un agent peut être vu comme une liste de comportements (classe Behaviour). Chaque

2. [http : // www.madkit.org/](http://www.madkit.org/).

3. [http : // sourceforge.net/projects/zeusagent/](http://sourceforge.net/projects/zeusagent/).

4. [http : // fipa-os.sourceforge.net/index.htm](http://fipa-os.sourceforge.net/index.htm)

5. [http : // aosgrp.com/products/jack/](http://aosgrp.com/products/jack/)

comportement est une routine JAVA exécutable. L'ensemble des comportements forme une file d'attente et ceux-ci sont exécutés successivement par un ordonnanceur. Un comportement est composé de deux méthodes, la méthode *action()*, qui est exécutée en premier et qui correspond au code du comportement défini pour l'agent, et la méthode *done()* qui est exécutée après la méthode *action()*, et vérifie si le comportement doit être remis dans la file d'attente de l'ordonnanceur ou non. Un comportement peut donc être exécuté plusieurs fois. Un agent n'exécute qu'un seul comportement à la fois.

Dans les expérimentations, nous avons initialisé la base de croyance de chaque agent, notamment son profil, sa liste de services offerts et l'ensemble de ses accointances dans le réseau social. Lorsqu'un agent reçoit une requête, un comportement dédié lui permet de raisonner sur la résolution de la requête. Les agents sont dotés aussi d'un comportement permettant l'envoi de messages à un ou plusieurs autres agents de la plateforme. Les messages contiennent les informations relatives à l'évaluation de la confiance, ainsi que les primitives des protocoles d'interactions. JADE offre par ailleurs la possibilité de réceptionner de manière automatique les messages dans une liste d'attente.

Nous avons développé un comportement de lecture de la file d'attente, qui se répète de manière cyclique pendant toute la simulation. Les agents récupèrent les messages par ordre de réception.

7.1.2 Choix du langage de stockage

Les données sont structurées dans un format universel qui est non seulement compréhensible par la majorité des applications participantes, mais dont la structure est aussi le dénominateur commun de toutes les autres structures et peut être étendue pour répondre à un format particulier.

Dans le cas de notre application, en plus des critères de portabilité et de flexibilité, nous souhaitons que les fichiers représentant les graphes soient d'une part facilement interprétables par un humain, (cela exclut donc les formats binaires et nous restreint aux formats de fichiers basés sur du texte « brut ») et d'autre part, rapidement accessibles en termes de temps de chargement puisqu'on manipule des graphes de grande taille.

Plus précisément, les fonctionnalités nécessaires pour un format de fichier commun sont :

- Indépendance de la plateforme et facile à mettre en œuvre.
- Capacité de représenter des structures de données arbitraires.

- Souplesse et indépendance de l'ordre des déclarations.

Dans ce qui suit, nous allons présenter les formats les plus utilisés.

DOT⁶ (**graph description language**) modélise aussi bien les graphes orientés que les graphes hiérarchiques. Il peut fonctionner comme un programme de ligne de commande, un service de visualisation Web, ou comme une interface graphique compatible . Il dispose aussi d'un panel d'algorithmes de layout pour bien positionner les nœuds et les arêtes, attribuer des étiquettes, des formes pour le dessin des structures de données personnalisées et d'un langage de fichier sous-jacent pour les outils de graphe orienté flux.

GML (Graph Modelling Language) est un format de graphes défini par Himsolt⁷, créé dans un but de portabilité (proposer un standard pour la description de graphes) et de souplesse. En effet, beaucoup de programmes et logiciels travaillent avec des graphes, mais la plupart d'entre eux utilisent un format de fichier propriétaire. En conséquence, l'échange entre les différents programmes est presque impossible. Des tâches simples comme l'échange de données, les résultats reproductibles à l'extérieur ou une suite de tests commune sont beaucoup plus difficiles, d'où la naissance du nouveau format *GML* qui permet d'attribuer des données arbitraires à des graphes, des nœuds et des arêtes. Ainsi, il est capable d'émuler presque tous les autres formats.

GraphML⁸ (**Graph Markup Language**) est un format de fichier complet et facile à utiliser pour les graphes. Il se compose et d'un noyau de langage pour décrire les propriétés structurelles d'un graphe et d'un mécanisme d'extension flexible pour ajouter des données spécifiques aux applications. En plus de sa richesse au niveau de la représentation (les graphes orientés, non orientés et mixtes, les hypergraphes, les graphes hiérarchiques et autres représentations de graphes notamment les graphes imbriqués ...), *GraphML* fournit d'autres fonctionnalités et avantages :

- Les références à des données externes,
- La personnalisation d'attributs par rapport aux données,
- Les analyseurs sont faciles à mettre en place.

6. <http://www.tonyballantyne.com/graphs.html>

7. <https://www.fm.uni-passau.de/fileadmin/files/lehrstuhl/brandenburg/projekte/gml/gml-technical-report.pdf>

8. <http://graphml.graphdrawing.org/>

Contrairement à de nombreux autres formats de représentation de graphes, *GraphML* n'utilise pas une syntaxe personnalisée mais il est basé sur XML. Il est donc parfaitement adapté à toutes sortes d'opérations de génération, d'archivage, ou de traitement de graphes. Le seul bémol c'est qu'il exige plus d'espace de stockage pour la représentation des réseaux sociaux. Cela peut éventuellement poser des problèmes au niveau de la lecture des grands graphes.

Choix effectué : GML. Les critères qui ont guidé notre choix sont :

1) Extensibilité et flexibilité : La possibilité de définir nos propres attributs, ce qui exclut DOT, langage fait pour le dessin des graphes et dont les seuls attributs sont définis pour cette utilisation.

2) Structure simple et portabilité : Un fichier *GML* est un fichier ASCII 7-bit. Il est donc simple de le produire par des routines standard. Par ailleurs, les analyseurs sont faciles à mettre en œuvre manuellement. En outre, les fichiers sont des fichiers texte, ils peuvent être échangés entre les plateformes sans convertisseurs spéciaux.

Suite à ces critères, le standard *GML*, nous est apparu comme le langage le plus approprié à utiliser pour le stockage de graphes dans l'implémentation de notre modèle.

7.2 Description des données

Les jeux de données doivent décrire les agents du réseau social et les relations qui les relient, ainsi que les requêtes qui seront lancées sur le réseau. Rappelons que l'environnement expérimental de notre approche multi-agents est un réseau social dans lequel nous distinguons deux familles d'agents ; ceux qui fournissent des services (appelés fournisseurs) et ceux qui utilisent ces services (appelés demandeurs).

Pour l'évaluation, nous utilisons aussi bien des données simulées que des données réelles. Les données simulées sont des graphes générés dont la structure est aléatoire avec un nombre d'agents et une densité précisés a priori. Les données réelles ont été récupérées sur Internet à partir de sources de données connues et accessibles au public. Contrairement aux données simulées, la structure des graphes réels possèdent certaines caractéristiques (e.g., « small-world », « scale-free ») des réseaux sociaux.

Notons que pour tous les jeux de données (simulées ou réelles), les types de relations, les profils ainsi que les services ont été distribués aléatoirement parmi les agents, ainsi chaque agent peut être un fournisseur potentiel en fonction de la requête d'un demandeur.

Dans cette partie, nous décrivons les données utilisées au cours de l'évaluation empirique.

7.2.1 Description des jeux de données simulées

La génération des réseaux simulés se fait selon le modèle de génération aléatoire de *Erdős-Rényi* [Erdo 60], noté $G(n, p)$. La construction du réseau se fait d'une manière progressive. Initialement, le graphe contient n nœuds et pas d'arêtes. Ensuite, pour chaque arête possible entre deux nœuds distincts dans le graphe, on ajoute une arête avec une probabilité p . Le réseau résultant est homogène suivant une distribution de Poisson. La plupart des nœuds ont approximativement le même degré (voir Fig. 7.1(A)). Par exemple, le réseau routier correspond à ce type de modèle de réseau. Les données simulées sont utilisées principalement pour mesurer la performance (efficacité, efficience et qualité des résultats) de notre approche de découverte et aussi pour étudier la robustesse du système dans des configurations différentes. De plus, les données simulées nous permettent d'analyser les comportements des agents en faisant varier divers paramètres.

7.2.2 Description des jeux de données réelles

Les expérimentations réalisées sur des données réelles permettent de mettre en évidence l'intérêt de notre approche par rapport aux approches existantes. Nous considérons deux familles de réseaux fréquemment utilisés : les réseaux « small-world » et « scale-free ». Les réseaux « small-world » présentent trois propriétés qui les distinguent des autres familles de graphes (i.e., les graphes aléatoires, les graphes complets et les graphes réguliers) : une connectivité faible, un coefficient de clustering élevé et un petit diamètre.

De nombreux réseaux du monde réel⁹ exhibent la propriété « small-world » (voir Fig. 7.1(C)) comme le World Wide Web, les réseaux électriques, le réseau des citations scientifiques et les réseaux sociaux. Ces réseaux, par définition, relient la plupart des paires de nœuds par des chemins courts. Notre recherche basée sur la confiance peut être particulièrement bien adaptée pour trouver ces chemins car la structure d'un réseau « small-world » peut résulter de l'homophilie [Klei 00]. L'homophilie est l'une des régularités empiriques les plus notables et robustes de la vie sociale. Elle se traduit

9. <http://toreopsahl.com/datasets/>

par le fait que les agents voisins dans le réseau ont tendance à avoir des valeurs d'attributs similaires e.g., ils vivent dans la même zone géographique ou ils ont la même profession [Sims 05].

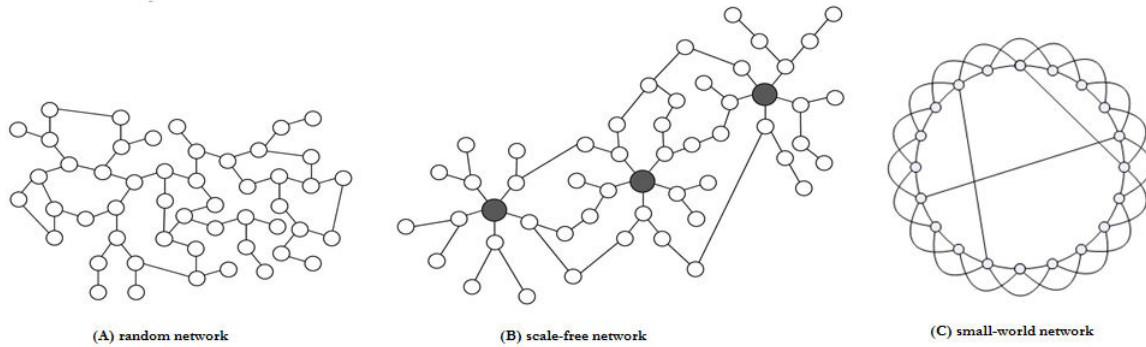


FIGURE 7.1 – Les réseaux « random », « scale-free » et « small-world » [Xiao 10a].

Les réseaux « scale-free » sont des réseaux qui suivent une loi de puissance pour la répartition des degrés des sommets du graphe (voir Fig. 7.1(B)). En statistique, une loi de puissance est une relation entre deux quantités où une quantité varie en puissance de l'autre. Cela signifie que la probabilité d'un degré k donné est proportionnelle à $k^{-\beta}$, où β est un paramètre représentant l'exposant ou l'indice de la loi. La plupart des nœuds de ces réseaux ont un degré faible et quelques uns ont un degré beaucoup plus élevé. Les quelques nœuds fortement connectés sont appelés les « hubs ».

L'existence de « hubs » rend le réseau fragile en proposant des cibles raisonnables à attaquer. Cependant, les événements naturels qui causent des interruptions de réseau (comme les mutations) sont classés beaucoup plus comme des pannes du système plutôt que comme des attaques, puisque ceux-ci sont des événements non-spécifiques. En l'absence d'attaques spécifiques, les « hubs » sont relativement exempts de dommages et rendent la structure du réseau plus stable et robuste [Jeon 01] contre les attaques aléatoires. Beaucoup de réseaux « small-world » sont aussi « scale-free » [Bara 99]. Les réseaux réels sélectionnés pour réaliser nos expérimentations sont décrits ci-dessous.

Le réseau d'Enron Email Le jeu de données de Enron email¹⁰ est un jeu de données très connu qui a été utilisé dans de nombreuses études de réseaux sociaux (voir par exemple, [Shet 04, Wang 13, Liu 11]). De plus, il a été prouvé dans [Mats 08] qu'il possède les deux caractéristiques distinctives des réseaux sociaux, « small-world » et « scale-free ».

Pour nos expérimentations, nous considérons la version fournie par Golbeck¹¹. Pour

10. <http://www.cs.cmu.edu/enron/>

11. <https://www.cs.umd.edu/golbeck/LBSC708L/hw5-enron.shtml>

construire ce réseau social, nous n'avons gardé que la plus grande composante connexe. De plus, nous n'avons considéré que les liens bidirectionnels, cela signifie qu'une arête existe entre deux nœuds si et seulement si ces derniers ont échangé des emails. Cela garantit qu'il y a eu un échange d'informations entre les deux employés d'Enron, et donc qu'ils sont impliqués dans une sorte de conversation. Comme le montre la figure 7.2, le réseau social de Enron email est un graphe contenant 7087 nœuds et 14786 arêtes. La densité du réseau est de 0,001, le diamètre est 8 et le plus court chemin est de 3,755.

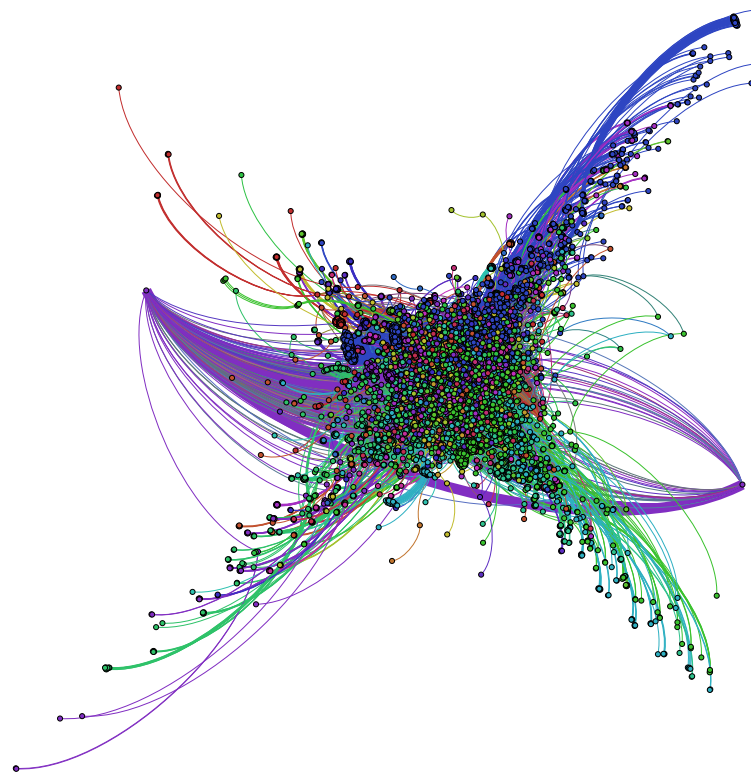


FIGURE 7.2 – Le réseau Enron Email (Visualisation à l'aide de Gephi).

Le réseau social Facebook. Nous avons récupéré sur internet un réseau social type

Facebook¹² contenant 4039 nœuds et 88234 arêtes (voir Fig. 7.3).

[Klei 00] a prouvé que ce réseau social possède les caractéristiques og « small-world » et « scale-free ». Les données ont été anonymisées en remplaçant les identifiants Facebook de chaque utilisateur par une nouvelle valeur. De plus, étant donné que le contenu sémantique de ces données a été fourni, l'interprétation a été obscurcie. Par exemple, pour un attribut reflétant la sensibilité politique "politique=Parti démocrate", les nouvelles données peuvent tout simplement contenir "politique= valeur anonyme 1". Ainsi, en utilisant des données anonymisées, il est possible de déterminer si deux utilisateurs ont les mêmes affiliations politiques, mais pas ce que leurs affiliations politiques individuelles représentent.

Afin de rendre les réseaux sociaux (réels ou simulés) des réseaux sociaux multi-relationnel (RSMR), nous introduisons différents types de relations $\{R_i \mid i \in \mathbb{N}^*\}$. L'utilité d'une relation est donnée par l'expression $U(R_i) = \frac{1}{2^i - 1}$ avec $i \in \mathbb{N}^*$ est le rang de la relation selon l'ordre de préférence du demandeur de services. Dans ce cas, nous supposons que le demandeur de services préfère deux fois plus la première relation que la deuxième. Cela permet de favoriser les chemins composés d'un nombre minimal de types de relations différentes, compte tenu des préférences du demandeur de services.

7.3 Évaluations empiriques et résultats

Après avoir détaillé les jeux de données réels et simulés, nous présentons maintenant le processus de validation. Nous explorons dans la première série d'expériences le temps de calcul et le nombre moyen des messages échangés en fonction des paramètres du système à savoir :

- Le nombre d'agents dans le réseau social,
- Le seuil de confiance en la crédibilité sociale,
- Le nombre des types de relations,
- Le longueur de la chaîne fournisseur-recommandeur.

Dans la deuxième série d'expériences, nous examinons la performance de notre approche de découverte de services en évaluant l'efficacité, l'efficacité et la qualité des résultats.

12. <http://snap.stanford.edu/data/egonets-Facebook.html>

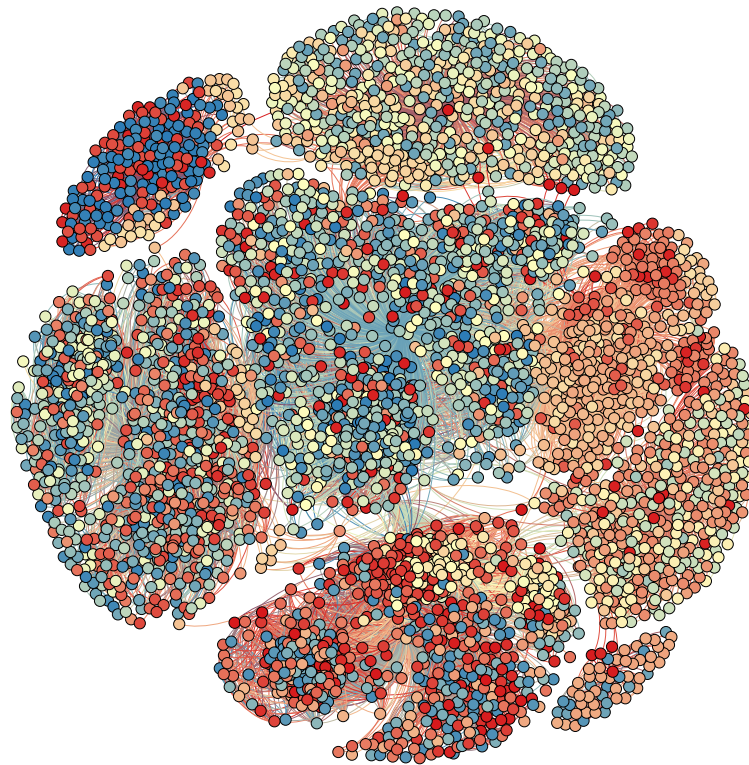


FIGURE 7.3 – Le réseau Facebook (Visualisation à l’aide de Gephi).

L’efficacité est menée selon trois politiques de référencement au vu desquelles nous examinons la qualité d’expertise des fournisseurs découverts pour différentes valeurs du seuil de la confiance en la crédibilité sociale.

L’efficacité consiste à étudier la dispersion du nombre des fournisseurs découverts pour différentes longueurs de la chaîne fournisseur-recommandeur afin de contrôler la propagation de la recherche dans le réseau social.

La qualité des résultats sera évaluée à l’aide des mesures de la précision et du rappel.

Nous nous concentrons dans la troisième série d’expériences sur l’évaluation de la performance de notre approche de sélection de services. L’examen de la performance

est effectué à travers une étude comparative avec d'autres approches de sélection basées sur la confiance selon deux critères : l'utilité et la satisfaction des demandeurs. Le premier critère relève du caractère objectif de la performance en évaluant l'utilité des demandeurs de services. Le deuxième critère porte sur l'aspect subjectif de la performance en analysant la satisfaction des demandeurs.

Finalement, nous présentons dans la quatrième série d'expériences les résultats des premières évaluations afin de souligner la validité de notre approche de composition. Les résultats obtenus à partir des simulations sont utilisés pour analyser d'une part le comportement des agents dans différents jeux de données et d'autre part, pour déterminer les configuration à adopter pour améliorer la qualité de la découverte, de la sélection et de la composition de services.

7.3.1 Implémentation

Dans cette section, nous présentons les résultats du temps de calcul et le nombre moyen des messages échangés durant la découverte de services pour différents graphes du réseau social multi-relationnel (RSMR). Le temps de calcul mesuré correspond au temps de l'évaluation de la crédibilité sociale effectué par les agents participants afin de propager la requête dans le réseau social. Le nombre moyen des messages échangés représente le coût de communication de la propagation de la recherche dans le réseau social.

Toutes les expériences ont été réalisées avec une machine tournant sous Windows 7 de capacités 3.1GHz Core(TM) i5-2400 et 8Go de RAM. Nous déroulons plusieurs expérimentations afin d'étudier les compromis entre le nombre d'agents et le nombre de relations contenues dans le réseau et leur impact sur le temps de calcul et sur le coût de communication. Ces expérimentations ont été réalisées sur 10 jeux de graphes simulés et respectant la propriété de la connectivité des réseaux sociaux. Le nombre d'agents varie de 1000 à 10000 et le nombre d'arêtes varie de 6000 à 60000 respectivement. Dans chaque graphe nous varions le nombre de types de relations de 2 à 5 donnant lieu à un RSMR. Dans le premier RSMR, le nombre d'agents est 1000 tandis que le nombre de types de relations est 2. Dans le dernier RSMR, le nombre d'agents est 10000 et le nombre de types de relations est 5. Dans tous les RSMR, les préférences du demandeur pour ces relations sont $U(R_1) = 1$, $U(R_2) = \frac{1}{2}$, $U(R_3) = \frac{1}{4}$, $U(R_4) = \frac{1}{8}$, et $U(R_5) = \frac{1}{16}$.

Évaluation du temps de calcul

La figure 7.4 montre le temps de calcul (en secondes) relatif à la mesure de la crédibilité sociale en fonction du nombre d'agents dans le RSMR et en variant le nombre de types de relations. Pour chaque RSMR, l'exécution a été réalisée 10 fois et le temps de calcul moyen a été enregistré.

Comme nous l'avions prévu, le temps de calcul augmente avec l'augmentation du nombre d'agents dans le RSMR. Par ailleurs, nous notons que le temps de calcul est sensible au nombre de types de relations.

En effet, pour des réseaux contenant moins de 4000 agents, le nombre de types de relations n'influe pas trop sur les coûts de calcul. Cependant, pour les grands réseaux ce nombre impacte considérablement le temps de calcul. Par exemple, l'écart du temps de calcul entre un réseau de 4000 agents avec 2 relations et le même avec 5 relations est de 7 secondes (de 11 à 18). Cet écart devient plus significatif valant 34 secondes (de 65 à 99) entre un réseau de 10000 agents avec 2 relations et le même avec 5 relations. En terme de pourcentage, l'écart du temps de calcul entre les différentes courbes est quasiment linéaire. Par exemple, si nous considérons la même paire de courbes (2 relations et 5 relations), le pourcentage de l'écart est égal à 62%. En général, plus la taille (en terme de nombre d'agents) ou la multiplicité (en terme de nombre de types de relations) du RSMR augmente, plus le calcul est coûteux.

Évaluation du coût de communication

Dans cette série de simulations, nous étudions l'effet de la taille du réseau social (en terme de nombre d'agents) sur le coût de la communication pour différentes valeurs du seuil de confiance. Rappelons que la valeur du seuil de confiance détermine la possibilité d'un agent d'envoyer un message à une accointance si la confiance qu'il a en elle est supérieure à ce seuil. Dans la figure 7.5, nous représentons le nombre moyen des messages échangés en fonction de la taille du réseau social tout en faisant varier la valeur du le seuil de confiance en la crédibilité sociale. Comme prévu, plus la valeur du seuil α est grande, plus le nombre moyen de messages échangés est petit. Cela peut être expliqué par le fait que les interactions entre les agents ne peuvent avoir lieu que si et seulement si la valeur de la crédibilité sociale entre eux est supérieure à α . Pour des valeurs élevées de α , les agents explorent moins d'espace de recherche dans le RSMR ce qui réduit considérablement le coût de la communication. Nous pouvons remarquer aussi que plus la taille du RSMR est grande, plus le nombre de messages échangés diminue. Ce résultat est dû au fait que dans les instances de grande taille, la densité est

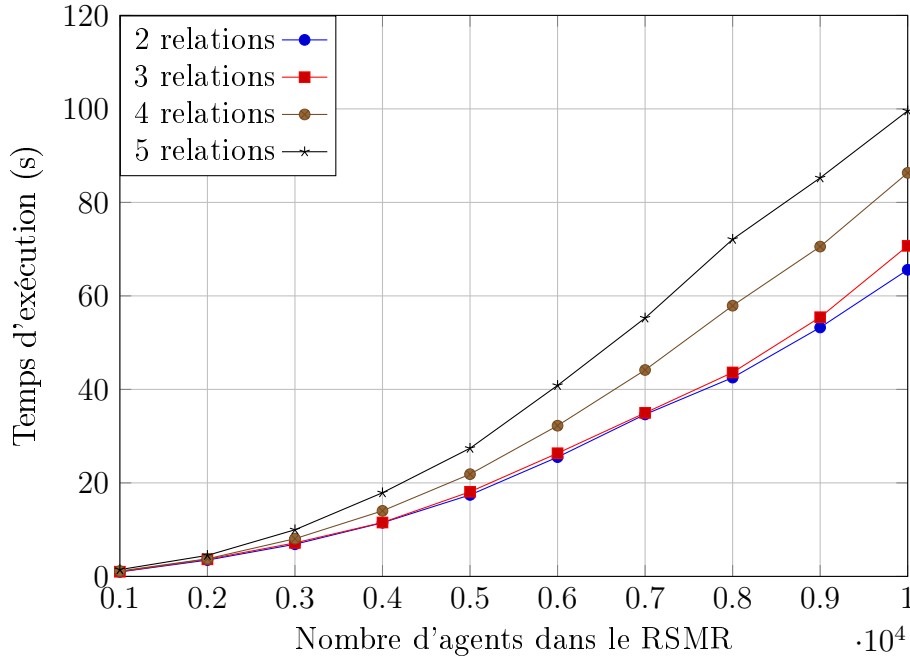


FIGURE 7.4 – Temps de calcul de la confiance en la crédibilité sociale en fonction de la taille du réseau

plus faible ¹³. Par conséquent, le nombre de voisins communs entre deux agents diminue ce qui influence négativement la valeur de la similarité sociale qui est une composante de la crédibilité sociale. Ceci engendre une diminution du nombre d'interactions (i.e. messages envoyés).

7.3.2 Évaluation de la performance de notre approche de découverte de services

Dans cette section, nous évaluons la performance de notre approche de découverte, en termes d'*efficacité*, d'*efficience* et de *qualité des résultats* via des simulations. Nous générons 5 instances de graphes où le nombre d'agents est égal à 1000 et le nombre d'arêtes est égal à 6000. Nous considérons trois catégories de fonctionnalités : transport, hébergement et divertissement dans chacune d'elles nous définissons 5 services différents. Nous équipons chaque agent avec trois services différents, un de chaque catégorie, choi-

13. Rappelons que le nombre d'agents dans les jeux de données varie de 1000 à 10000 et le nombre d'arêtes varie de 6000 à 60000 respectivement. Ainsi, la densité du premier jeu de données est 0.012. Par contre, celle du dernier est 0.001.

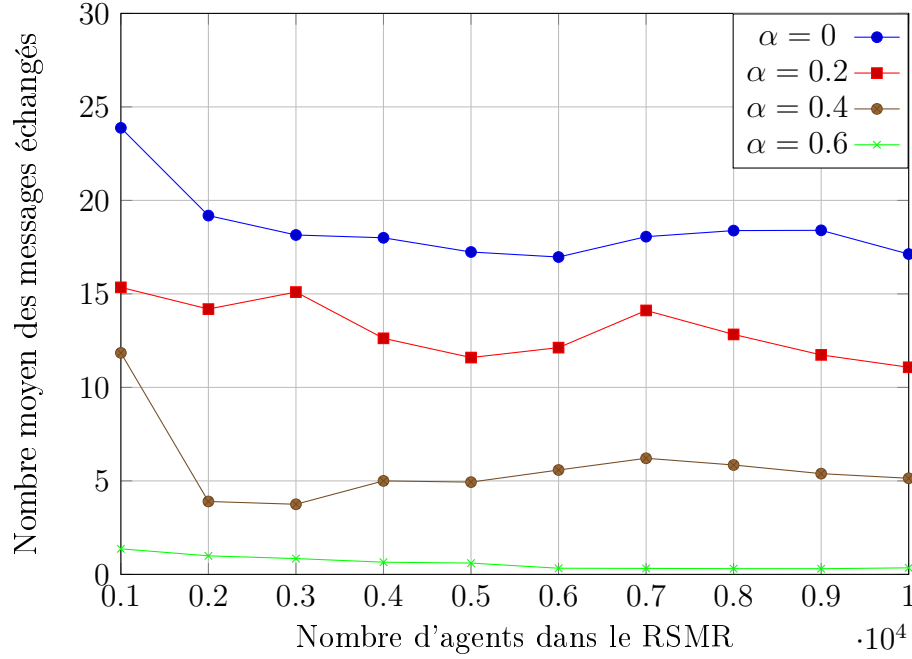


FIGURE 7.5 – Effet de la taille du réseau social sur le coût de la communication.

sis au hasard. Pour les valeurs de QoS, nous considérons le jeu de données WSDream¹⁴ de [Zhan 10]. Rappelons que la requête du demandeur est définie par $(F, U, \alpha, \beta, \mu)$. Par souci de simplicité, nous supposons que les demandeurs ont besoin d'une seule fonctionnalité $F = \{f\}$. Nous considérons trois types de relations $\{R_1, R_2, R_3\}$. Les préférences du demandeur pour ces relations sont égales à $U(R_i) = \frac{1}{2^{i-1}}$ avec $i \in \llbracket 1, 3 \rrbracket$. Nous varions la valeur du seuil de confiance α 's dans $[0, 1]$, alors que β et μ sont fixés à 0,4.

Efficacité

Cette expérience évalue l'efficacité de notre approche de découverte basée sur les systèmes de références. L'efficacité consiste à examiner la qualité moyenne de l'expertise des fournisseurs découverts. L'expertise d'un fournisseur correspond à la valeur de sa confiance en l'expertise établie à partir des valeurs des attributs de QoS des services qu'il offre (voir Sect. 4.5). Cela nous permet d'estimer à l'avance la qualité des services découverts pour différents intervalles de confiance.

Comme méthodologie d'expérimentation, nous considérons trois politiques de référencement :

14. <http://www.wsdream.net/dataset.html>

- **referAll** : un agent envoie la requête à tous les agents de l'ensemble de ses accointances. C'est un cas particulier de référencement dans lequel la confiance n'est pas prise en considération. Cela ressemble au processus de recherche de *GNUtella* [Kan 01] où les requêtes sont propagées par inondation c'est à dire, chaque agent transmet la requête entrante à tous ses voisins s'il la reçoit pour la première fois.
- **referTrust** : uniquement les agents dont la crédibilité sociale est au-dessus d'un seuil de confiance prédéfini sont référés. Plus la valeur du seuil augmente, plus le nombre d'agents contactés diminue.
- **referBest** : un agent réfère un seul agent, son meilleur voisin, dont le score de crédibilité sociale est le plus élevé. Cela est similaire au routage de messages de Freenet [Lang 01] dans lequel chaque client transmet la demande à l'agent le plus susceptible d'avoir les informations demandées.

La figure 7.6 illustre l'évolution de la qualité moyenne de l'expertise en fonction de la variation du seuil de confiance α pour les différentes politiques de référencement. Durant la simulation, chaque agent présente 5 requêtes générant au total en 5000 requêtes pour lesquelles nous calculons la valeur moyenne de la confiance en expertise des fournisseurs découverts. Nous limitons la longueur des chaînes de références à cinq comme la valeur "time-to-live" de *GNUtella*.

Dans la figure 7.6, la politique referAll correspond au cas où le seuil de confiance $\alpha = 0$. Par contre, la politique referBest référant le meilleur voisin est indépendante du seuil de confiance. Elle demeure constante pour toutes les valeurs de α . La première observation à noter est que parmi les trois politiques mises en place, la politique referAll est la moins performante car sa qualité d'expertise est égale à 0.27. Ce résultat est justifié par le fait qu'étant donné que la confiance n'est pas prise en considération, tous les fournisseurs sont découverts ce qui diminue la qualité moyenne de l'expertise. Par conséquent, échanger plus de références ne garantit pas que la qualité d'expertise soit plus élevée d'où l'intérêt d'intégrer la confiance dans le processus de découverte.

La deuxième observation à noter concerne l'évolution de la qualité moyenne de l'expertise de referTrust contre la variation de la valeur de α . Nous relevons trois importants points.

Tout d'abord, la meilleure performance d'expertise enregistrée dans la figure 7.6 correspond à $\alpha = 0.6$. Selon ces résultats, nous pouvons considérer cette valeur comme une valeur de référence qui pourrait être recommandée au demandeur avant de lancer le processus de découverte.

Deuxièmement, lorsque $\alpha = 1$, les courbes de referBest et referTrust se croisent. Cela

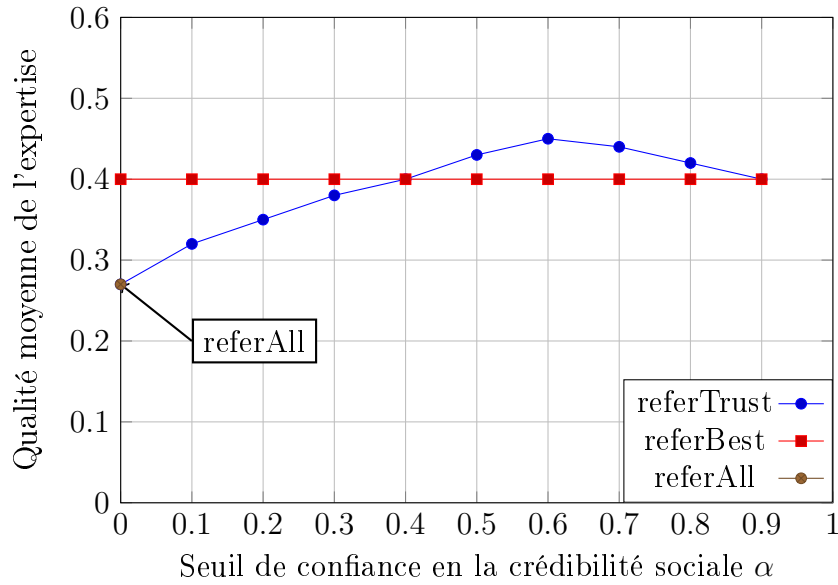


FIGURE 7.6 – Performance de l'approche de découverte : Efficacité.

est équivalent à garder pour les deux politiques un seul agent, qui est l'agent le plus digne de confiance.

Troisièmement, dans la plage des valeurs $\alpha \in [0, 0.6]$, la courbe de la politique `referTrust` a une pente positive (i.e., les fournisseurs avec une faible crédibilité sociale ont également une faible expertise). Cela indique l'existence d'une corrélation entre la valeur de la crédibilité sociale et la qualité d'expertise d'un agent. Avec l'augmentation de la valeur de α , les fournisseurs faiblement dignes de confiance sont filtrés et la valeur moyenne de la qualité d'expertise augmente.

Cependant, cette corrélation n'est plus valable (c'est à dire, l'expertise d'un fournisseur s'améliore avec l'augmentation de sa crédibilité sociale) pour des grandes valeurs de α . Les agents deviennent trop sélectifs dans le référencement et peu d'agents dignes de confiance sont contactés pour trouver des services utiles. Cela signifie que pour des grandes valeurs de confiance, certains fournisseurs avec une bonne expertise ne sont plus découverts car ils sont socialement déconnectés de l'agent demandeur c'est à dire, qu'il n'y a pas de chaîne de références dignes de confiance menant à eux. Ce qui n'est pas le cas de `referBest`, car il y a toujours un agent à contacter.

Efficience

Nous évaluons dans cette section l'efficience de notre approche de découverte en examinant la dispersion du pourcentage des fournisseurs découverts en fonction de la

longueur de la chaîne fournisseur-recommandeur. Durant la simulation, chaque agent présente 5 requêtes résultant en 5000 requêtes en total pour lesquelles nous calculons la moyenne en pourcentage des fournisseurs découverts.

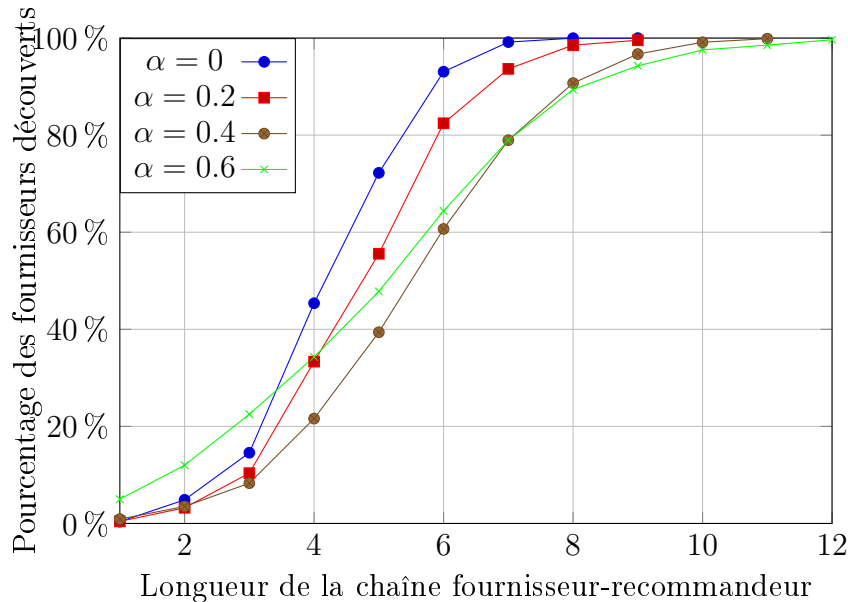


FIGURE 7.7 – Performance de l'approche de découverte : Efficience.

Les études en sciences sociales [Dunn 84, Misz 96, Rowl 79, Mckn 96] ont indiqué que la confiance d'un individu A en un individu B est une relation sociale établie à partir de leurs expériences directes ou à partir des expériences indirectes communiquées par d'autres individus dignes de confiance qui ont interagi avec B. Autrement dit, les individus désirent recevoir des services de leur réseaux égocentriques ou au plus de leur réseaux égocentriques *2-degrés* (voir Sect. 2.2.2). En sciences informatiques, les approches qui évaluent la confiance dans les réseaux sociaux proposent des modèles de calcul qui opèrent sur des relations sociales inférées établies à partir des chemins plus longs (par exemple, [Golb 06b, Gray 03, Hang 09, Kute 07, Liu 11, Wang 06c, Wang 13, Zhen 14]). Pour illustrer ce point, nous représentons dans la figure 7.7 la répartition moyenne en pourcentage cumulé des fournisseurs découverts en fonction de la longueur de la chaîne fournisseur-recommandeur. Pour différentes valeurs de seuil de confiance, nous notons dans la figure 7.7 que le nombre des fournisseurs découverts dans l'ensemble d'acointances des demandeurs de services (i.e., leur réseaux égocentriques) ne représente qu'une petite fraction (environ 1%) de l'ensemble total des fournisseurs découverts. En réalité, certains agents peuvent ne pas être de bons experts, mais peuvent être bien connectés en recommandant de bonnes références. D'où l'intérêt des systèmes de références dans la recherche des fournisseurs.

Dans notre approche, la crédibilité sociale attribue la capacité de fournir de bonnes

références. La propagation de la recherche est effectuée via navigation dans le réseau social ce qui explique l'augmentation du nombre de fournisseurs découverts en fonction de la longueur de la chaîne.

Il est aussi intéressant de noter qu'il y a un compromis entre la longueur de la chaîne et le nombre de fournisseurs découverts : nous avons six fois plus de chances de trouver un fournisseur de services digne de confiance pour une longueur de chaîne égale à 7 qu'une chaîne de longueur 3 (de 14% à 80%). De plus, nous remarquons que pour des valeurs de seuil supérieures à 0,4, les fournisseurs dignes de confiance sont découverts relativement loin du demandeur de services. Cela indique que certains fournisseurs sont faiblement reliés aux demandeurs de services et requièrent une longue chaîne pour être découverts. De plus, le nombre moyen des fournisseurs découverts tend à se stabiliser une fois la longueur atteint une certaine valeur (ici, elle est égale à 8) indépendamment de la confiance. Concrètement, lorsque la longueur de la chaîne va au-delà d'une certaine limite, les fournisseurs restants sont socialement déconnectés du demandeur de services.

Selon ces résultats, nous pouvons considérer la chaîne fournisseur-recommandeur comme un paramètre d'entrée qui pourrait être ajusté avant de lancer le processus de découverte de services. Au cours de la recherche, il est inutile de propager la requête au delà d'une chaîne de longueur 7 étant donné que la plupart des fournisseurs (environ 80%) sont découverts dans cette zone. Fixant la valeur 7 comme étant la longueur maximale autorisée nous permet d'une part, d'être en accord avec le phénomène du « small-world » et d'autre part, de contrôler l'espace de recherche limitant ainsi les coûts de calcul.

Qualité des résultats

Nous projetons la validation de notre approche de découverte dans le domaine de l'extraction d'information. Pour évaluer la qualité des résultats, nous utilisons les métriques de *précision* et de *rappel*, introduites par Van Rijsbergen [Rijs 79]. Ces deux métriques sont définies par deux ensembles, l'ensemble des services trouvés et l'ensemble des services pertinents. Un service est dit pertinent s'il répond au besoin du demandeur avec une valeur d'expertise supérieure à un seuil prédéfini. Rappelons que la valeur d'expertise correspond au produit des trois attributs de QoS, la spécialisation (SP), la disponibilité (Di) et la qualité (Eval) définis dans la section 4.5. La précision évalue le nombre des services pertinents parmi les services découverts par notre approche, tandis que le rappel évalue le nombre de services pertinents découverts par notre approche parmi les services pertinents existant [Xiao 10b]. Par conséquent, l'approche de découverte est efficace quand le rappel et la précision sont proches de 1. Ces

deux métriques sont définies comme suit :

$$\text{Précision} = \frac{|\{\text{services pertinents}\} \cap \{\text{services découverts}\}|}{|\{\text{services découverts}\}|}, \text{ Rappel} = \frac{|\{\text{services pertinents}\} \cap \{\text{services découverts}\}|}{|\{\text{services pertinent}\}|}$$

Les simulations ont été réalisées sur 10 jeux de graphes générés en respectant la propriété de la connectivité des réseaux sociaux. Le nombre d'agents varie entre 1000 à 10000 et le nombre d'arêtes varie de 6000 à 60000 respectivement. Pour chaque jeu, nous générons trois instances différentes. Chaque instance est peuplée d'un pourcentage défini d'experts (i.e., agents qui offrent le service requis) : 10%, 15% et 25%. Dans cette expérience, nous fixons les seuils de confiance comme suit : $\alpha = 0.6$, $\beta = 0.4$ et $\mu = 0.4$. Ainsi, un service découvert est pertinent si la valeur de son expertise est supérieure ou égale à 0.4. Pour chaque exécution, nous comptabilisons le nombre de services pertinents retournés par notre approche de découverte et nous notons à chaque fois la précision et le rappel.

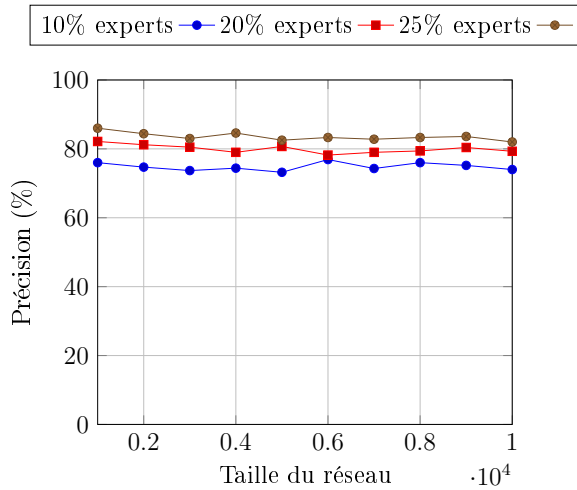


FIGURE 7.8 – Valeur de la précision en fonction de la taille du réseau.

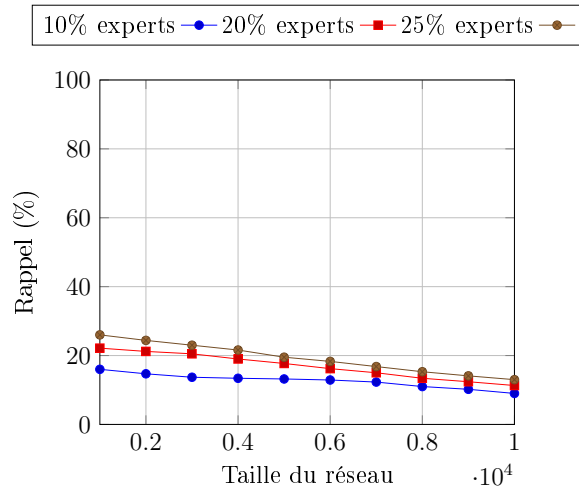


FIGURE 7.9 – Valeur du rappel en fonction de la taille du réseau.

Les figures 7.8 et 7.9 présentent la précision et le rappel en fonction de la taille du réseau. Les résultats affichés dans la figure 7.8 indiquent qu'une découverte de services réalisée dans un réseau avec 25% d'experts affiche un niveau de précision supérieur à celui des autres instances. La précision moyenne de la découverte pour cette instance est proche de 84%, alors que celle pour une instance avec 10% d'experts est proche de 75%. Ces observations indiquent que plus le pourcentage d'experts est élevé, plus la possibilité de trouver des services pertinents pour le demandeur est grande. Cela peut être expliqué par la nature de notre mécanisme de référencement.

Contrairement à Singh et al. [Sing 01, Yu 99, Yolu 03], nos chaînes de références, appelées chaîne fournisseur-recommandeur, sont particulières étant donné qu'elles n'au-

torisent pas la succession des recommandeurs c'est à dire, un recommandeur ne peut pas recommander un autre recommandeur (voir Sect. 5.3). Par conséquent, dans le cas où un agent contacté n'a pas un fournisseur dans l'ensemble de ces accointances la recherche est arrêtée. Cela indique que les experts restants sont socialement déconnectés du demandeur de services et ne peuvent pas être découverts.

Les résultats obtenus pour la précision sont accompagnés par d'intéressants résultats concernant le rappel, comme l'illustre la figure 7.9. Nous remarquons que plus le nombre d'experts augmente dans l'instance, plus ceux-ci sont découverts. Le taux de rappel moyen pour une instance avec 25% d'experts avoisine les 19%, alors que celui pour une instance avec 10% d'experts est proche de 12.6%. Cependant, contrairement à ce qui est observé pour la précision, la figure 7.9 montre que le rappel est sensible à la taille du réseau. Les valeurs du rappel diminuent progressivement avec l'augmentation du nombre d'agents dans le réseau. Cette décroissance est prévisible car la propagation de la recherche dépend de la longueur de la chaîne. Rappelons que les seuils de la crédibilité sociale et de la recommandation sont ajustés en fonction de la longueur de la chaîne, ce qui limite l'espace de recherche (voir Sect. 5.3.1). Parmi les experts non découverts, certains ne sont pas suffisamment socialement crédibles et d'autres sont inaccessibles car ils nécessitent une longue chaîne pour être identifiés.

L'analyse conjointe des métriques précision et rappel démontre que notre approche de découverte basée sur la confiance ne permet pas de trouver tous les services pertinents disponibles dans le graphe. Nous pensons que le mécanisme de découverte proposé est bien adapté aux réseaux sociaux réels où les demandeurs préfèrent les services et les recommandations fournies par des individus dignes de confiance se trouvant dans un voisinage proche.

7.3.3 Évaluation de la performance de notre approche de sélection de services

Dans cette section, nous évaluons la performance de notre approche de sélection de services appelée *Trusty* en terme d'*utilité* et de *satisfaction* des demandeurs via une étude comparative avec d'autres approches de sélection. Les approches que nous avons mises en place sont les approches de Bansal [Bans 10], Maaradji [Maar 10] et une approche de sélection purement fonctionnelle sans confiance. Les expériences sont réalisées sur les données réelles de Facebook¹⁵ décrites précédemment dans la section 7.2.2. Nous utilisons le jeu de données WSDream¹⁶ de [Zhan 10] qui contient 5825

15. <http://snap.stanford.edu/data/egonets-Facebook.html>

16. <http://www.wsdream.net/dataset.html>

services Web avec les valeurs de QoS. Chaque agent est équipé de trois services différents. De même, pour la requête nous supposons que les demandeurs ont besoin d'une seule fonctionnalité $F = \{f\}$. Nous fixons les seuils de confiance comme suit : $\alpha = 0.6$, $\beta = 0.4$ et $\mu = 0.4$.

Méthodologie d'expérimentation

Comme méthodologie d'expérimentation, nous avons mis en place quatre stratégies de sélection de services :

- Stratégie **NoTrust** : cette stratégie permet aux demandeurs de sélectionner aléatoirement un fournisseur parmi l'ensemble des candidats découverts (i.e., offrant un service avec la fonctionnalité requise dans F). Ainsi, la sélection d'un service est purement fonctionnelle excluant toute considération liée à la confiance.
- Stratégies **Bansal/Maaradji** : pour ces deux stratégies, la prise de décision d'un demandeur de services lors de la sélection d'un service est guidée par l'un des deux modèles de confiance de Bansal et Maaradji. Cela veut dire qu'il est amené à évaluer la confiance des fournisseurs découverts avant de sélectionner le meilleur.
- Stratégie **Trusty** : cette stratégie met en œuvre notre approche de sélection *Trusty* telle qu'elle est décrite dans le chapitre 5. Similaire aux deux stratégies précédentes, la prise de décision relative à la sélection de services est basée sur la confiance. Le modèle de confiance de Trusty est plus riche et compositionnel comme il intervient aussi bien au niveau de la sélection que la découverte de services.

Dans le réseau social, nous définissons quatre ensembles d'agents où chaque ensemble d'agents est équipé d'une stratégie différente. Étant donné que la seule différence entre les demandeurs est la stratégie de sélection qu'ils utilisent, l'utilité et la satisfaction exprimée par chaque agent reflétera la performance de son modèle de confiance.

Utilité et satisfaction des demandeurs

La première chose à vérifier est de savoir si Trusty aide les demandeurs de services à sélectionner de bons fournisseurs tout en acquérant une meilleure utilité en comparaison

aux autres stratégies de sélection.

L'utilité est un critère objectif défini par les valeurs des attributs de QoS objectifs (i.e., la disponibilité et la spécialisation). Pour un attribut de QoS objectif, l'utilité correspond à la moyenne de ses valeurs relevées pour les services sélectionnés. Chaque agent génère 10 requêtes durant l'exécution de la simulation résultant en 40390 requêtes au total sur lesquelles nous calculons la valeur moyenne de la spécialisation et de la fiabilité des services sélectionnés.

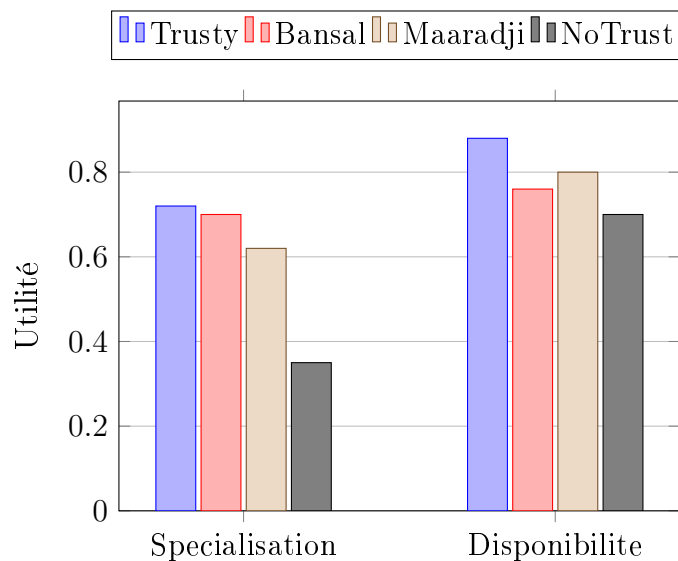


FIGURE 7.10 – Performance de l'approche de sélection : Utilité

Utilité. Comme nous l'avions prévu, la figure 7.10 montre que le groupe d'agents NoTrust (i.e., les agents qui sélectionnent au hasard parmi les services découverts sans aucune considération liée à la confiance) possède la plus faible performance pour tous les attributs de QoS. D'un autre côté, Trusty, Bansal et Maaradji s'avèrent bénéfiques pour les demandeurs en les aidant à obtenir une utilité significativement plus élevée c'est à dire, des services avec de meilleures QoS. Cela montre qu'une approche de sélection basée sur la confiance est plus efficace qu'une autre n'utilisant pas la confiance car elle permet aux agents d'interagir avec des fournisseurs plus dignes de confiance donc plus pertinents. Nous remarquons que notre approche se comporte mieux que les approches de Bansal et Maaradji sur les deux attributs. Cela est concrétisé par le fait que la mesure de Trusty est plus expressive et plus riche étant donné qu'elle évalue non seulement la crédibilité sociale et l'expertise d'un agent mais aussi la qualité de recommandation des agents intermédiaires conduisant à cet agent. Cela augmente la qualité des services sélectionnés et par la suite l'utilité mesurée. Bansal par exemple se base uniquement sur la crédibilité sociale évaluée à l'aide de la mesure de degré. La différence de performance entre Trusty et Maaradji s'explique par le fait que Trusty

sépare la sociabilité de l'expertise dans l'évaluation de la confiance au niveau du choix d'un agent avec qui interagir.

Cette séparation nous permet d'éviter les occasions manquées dans la découverte de services. En effet, un agent socialement digne de confiance qui a une expertise faible ne sera pas contacté dans la stratégie de Maaradji car l'agrégation des deux composantes donne une valeur de confiance inférieure au seuil. Par contre, dans la stratégie Trusty cet agent sera contacté étant donné qu'il est digne de confiance et même s'il ne fournit pas un service avec la fonctionnalité requise, il peut être utile en recommandant un bon fournisseur. Bien que le modèle de confiance de Maaradji soit plus riche que celui de Bansal, étant donné qu'il considère aussi bien la sociabilité (i.e., la proximité sociale) que l'expertise dans l'évaluation de la confiance d'un agent, nous notons qu'il est moins performant sur l'attribut spécialisation. Cela peut être expliqué par le fait que la spécialisation d'un fournisseur est naturellement corrélée à son degré dans le réseau qui est la mesure utilisée par Bansal. De ce fait, plus un agent est spécialiste dans un service, plus il est sollicité dans le réseau ce qui augmente le nombre de ses interactions et par la suite son degré. Cette observation justifie le fait que l'utilité de Bansal est plus performante que celle de Maaradji pour cet attribut.

Satisfaction des demandeurs. Dans la deuxième expérience, nous réalisons la même étude comparative mais d'un point de vue subjectif. Plus précisément, nous nous concentrons sur la détermination de l'effet des stratégies de sélection définies ci-dessus sur la satisfaction des demandeurs.

La satisfaction est un critère subjectif qui reflète l'opinion d'un demandeur concernant le comportement d'un service après utilisation. Elle correspond à la moyenne des valeurs de l'attribut *Eval* de tous les services sélectionnés. Pour les valeurs initiales d'*Eval*, nous appliquons un amorçage négatif (voir Sect. 4.5.3) pour mieux observer l'influence des stratégies de sélection sur la satisfaction des demandeurs.

Comme le montre la figure 7.11, l'objectif de cette expérience est d'examiner l'évolution de la satisfaction des demandeurs au cours du temps pour chacune des stratégies de sélection mises en place : Trusty, Bansal, Maaradji et NoTrust. Pour ce faire, chaque agent génère 100 requêtes (résultant en 403900 requêtes au total) qu'il soumet séquentiellement dans le réseau social. Afin de surveiller la progression de la satisfaction des demandeurs, nous relevons après chaque 10 requêtes la moyenne des notes attribuées aux services après utilisation. Au démarrage de la simulation, nous notons que la satisfaction des demandeurs se comporte de façon similaire pour les différentes stratégies de sélection. Cela est dû au fait que pour un petit nombre de requêtes, ces valeurs sont modifiées lentement. Cependant, au bout de 10 requêtes nous remarquons une différence notable dans les valeurs de satisfaction. Les résultats des premières sélections de services réalisées par la stratégie Bansal montrent qu'elle est plus performante

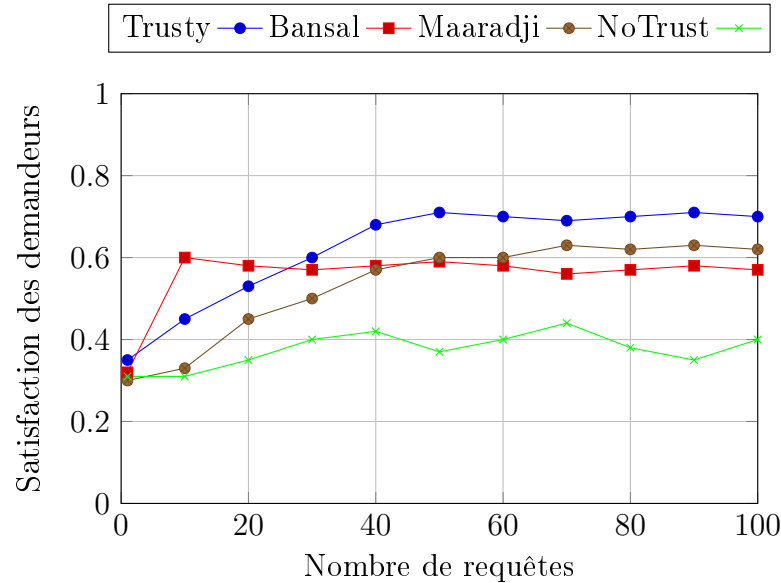


FIGURE 7.11 – Performance de l'approche de sélection : Satisfaction des demandeurs

que celles des autres stratégies. Cela peut s'expliquer par le fait qu'au démarrage de la simulation, l'attribut qualité de l'ensemble des services est initialisé avec une valeur faible (amorçage négatif). Par conséquent, pour des stratégies de sélection comme Trusty et Maaradj qui utilisent cet attribut dans l'évaluation de l'expertise, les agents échouent à trouver les bons fournisseurs. Par contre, ces valeurs faibles de la qualité n'influencent pas l'évaluation de la stratégie Bansal car elle se base uniquement sur la mesure de degré. Ainsi, les fournisseurs sélectionnés par Bansal dont le degré est élevé ont de fortes chances d'offrir de bons services, ce qui est le cas. Cela confirme notre conviction sur la grande influence de la crédibilité sociale dans l'évaluation de la confiance.

Un autre point notable qui se dégage de cette expérience est que la satisfaction finit par se stabiliser après un nombre fini de requêtes et cela pour toutes les stratégies de sélection qui utilisent la confiance (i.e., Trusty, Bansal, Maaradji). En réalité, ces groupes d'agents sont capables d'apprendre progressivement, mais avec des vitesses différentes, la qualité effective des services. Par exemple, le groupe d'agents Trusty est le groupe le plus rapide, il est capable d'apprendre au bout de 50 requêtes. Cependant, pour une approche aléatoire qui n'utilise pas de confiance, la satisfaction reste fluctuante et chute progressivement ce qui est tout à fait prévisible. Nous remarquons qu'au bout de 45 requêtes la satisfaction associée à la stratégie de Maaradji finit par surpasser celle du groupe d'agents Bansal. Aussi, nous constatons que parmi toutes les stratégies de sélection avec confiance, Trusty est le plus performant car il maintient une satisfaction élevée et stable.

Configuration \ Scénario	A	B	C	D	E
requête utilisateur	2	3	4	5	6
services	5	5	5	5	5
timeout (ms)	3000	3000	3000	3000	3000

TABLE 7.1 – Définition de scénarios de test

7.3.4 Évaluation de la performance de notre approche de composition de services

Dans cette section, nous présentons les résultats expérimentaux de l'évaluation de la performance de notre approche de composition de services. Les expériences ont été menées sur les données réelles de Facebook¹⁷ contenant 4039 agents et 88234 arêtes. Nous avons créé un bassin de services Web contenant 6 catégories de fonctionnalités, dans chacune d'elles nous avons défini 5 services différents résultant en 30 services au total. Nous avons équipé chaque agent avec trois services différents, choisis au hasard dans le bassin.

Initialement, nous fixons la taille de la blacklist $Blist_k$ de chaque agent a_k à 100 peuplée aléatoirement. Par ailleurs, nous fixons les seuils de confiance comme suit : $\lambda Sup_k = 0.7$, $\lambda Inf_k = 0.3$ et $\gamma_k = 0.7$. Finalement, nous initialisons les paramètres $NbSoll_k[A]$ et $NbMem_k[A]$ à des valeurs aléatoires avec $NbSoll_k[A] > NbMem_k[A]$.

Cinq scénarios de test ont été créés comme représenté dans le tableau 7.1 où la ligne *requête utilisateur* désigne le nombre de fonctionnalités requises dans sa requête, la deuxième ligne *service* définit le nombre de services par catégorie de fonctionnalité et la troisième ligne *timeout* spécifie la durée en milliseconde du timeout appliqué. 100 exécutions de formation de coalitions ont été effectuées par scénario de test.

Dans la première série d'expériences, nous étudions le coût de communication de notre approche pour les différents scénarios.

Dans la deuxième série d'expériences, nous discutons la complétion de notre processus de formation de coalitions pour les différents scénarios.

Dans la troisième série d'expériences nous étudions le comportement des agents durant le processus de formation de coalitions en notant la fréquence moyenne des agents qui quittent la coalition.

La figure 7.12 représente le nombre moyen de messages échangés par scénario de test. Comme prévu, plus le nombre de fonctionnalités requises est élevé, plus le nombre de

17. <http://snap.stanford.edu/data/egonets-Facebook.html>

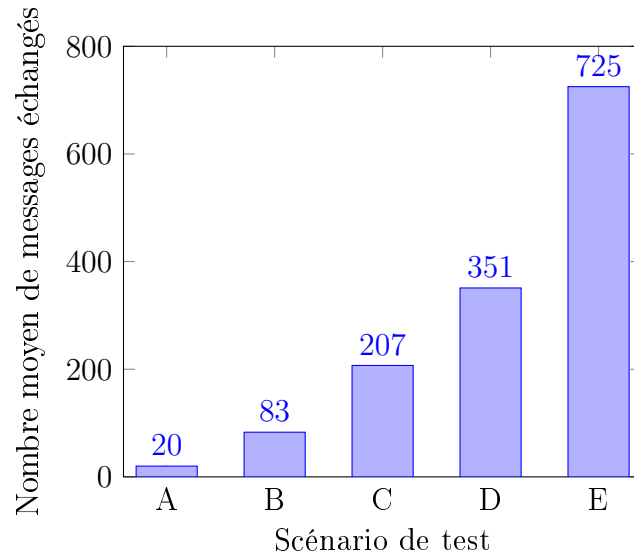


FIGURE 7.12 – Nombre moyen de messages échangés par scénario de test.

messages échangés est grand (avec une moyenne de 20 messages dans le scénario A et une moyenne de 725 messages dans le scénario E). Le coût de communication est particulièrement élevé dans le scénario E où plus d'interactions sont nécessaires pour satisfaire les fonctionnalités requises. L'écart du nombre de messages est enregistré dans la phase de sélection des membres du processus de formation de coalitions. En effet, comme nous l'avons décrit dans le chapitre 6, l'adhésion d'un nouveau membre dans une coalition doit garantir la satisfaction en termes de confiance de ce dernier et de la majorité des membres existants également. Ainsi, une grande coalition est intuitivement plus vulnérable à l'instabilité étant donné qu'il y a plus de chances qu'un membre de cette coalition refuse l'adhésion d'un nouveau membre et quitte par la suite la coalition. Cette intuition sera confirmée plus tard (voir Fig. 7.14 scénarios D et E). À cause de l'instabilité, le processus requiert plus d'interactions pour compenser la perte des anciens membres et continuer à chercher les membres manquants à la complétion de la coalition.

L'un des critères clés reflétant la performance d'un processus de composition de services est sa capacité à fournir un service composite complet qui satisfait la requête du demandeur de services. La figure 7.13 illustre la moyenne des résultats sur 100 exécutions, où l'axe des abscisses représente le scénario de test et l'axe des ordonnées le pourcentage du nombre des coalitions générées. Tout d'abord, notons que le nombre maximal de coalitions générées dans les scénarios A et B est de 3 coalitions, dans les scénarios C et D c'est 4 et enfin dans le scénario E c'est 5. Bien que le nombre de fonctionnalités requises dans le scénario A est plus petit que celui du scénario B,

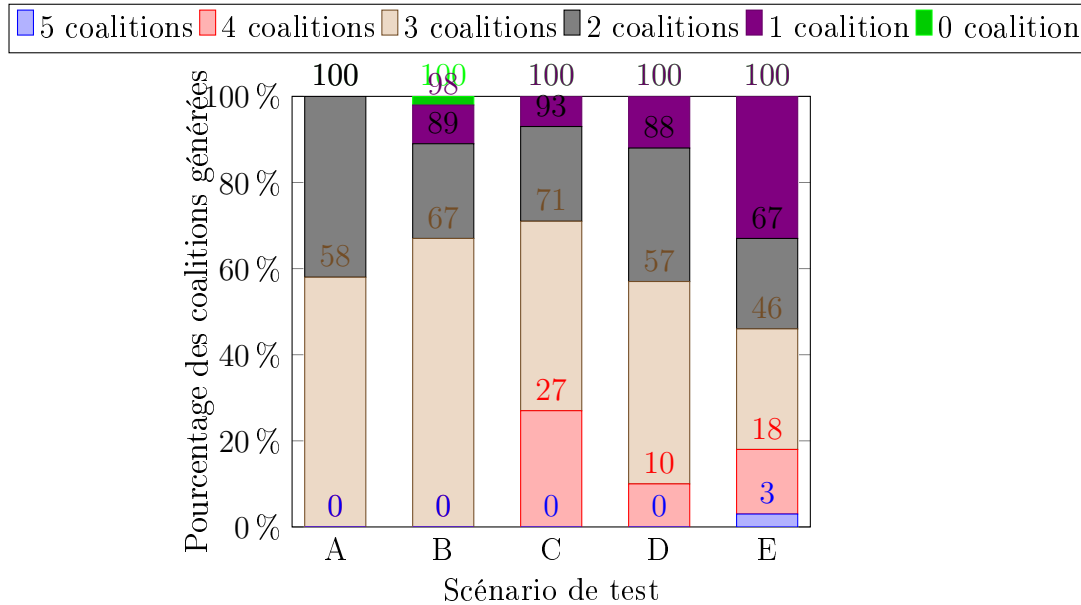


FIGURE 7.13 – Pourcentage de coalitions générées par scénario de test

les résultats obtenus montrent que le scénario A génère dans 58% des exécutions 3 coalitions alors que le scénario B génère les 3 coalitions dans 67% des exécutions. Cela peut être expliqué par le fait que l'évaluation de la confiance en la coopération se base sur des valeurs générées aléatoirement, ce qui affecte par la suite la prise de décision des agents. Nous notons également que lorsque le nombre de fonctionnalités requises augmente (comme dans les scénarios C, D et E), le pourcentage du nombre maximal de coalitions générées diminue. Par exemple, les exécutions du scénario E donnent 5 coalitions seulement dans 3% du temps. Cela explique notre intuition que plus la taille d'une coalition est grande plus le risque d'insatisfaction chez ses membres existants est élevé, ce qui réduit le nombre de coalitions réussies. Le troisième point à interpréter concerne le cas d'absence de coalitions. La figure 7.13 montre que le scénario B échoue dans 2% des exécutions à générer une coalition. Cela est dû au fait qu'aucune des 3 coalitions initiales n'a réussi à garantir la confiance requise par les membres et chacun des candidats proposés par le broker.

Les résultats obtenus prouvent que l'approche que nous avons proposée est efficace pour sa capacité à produire de multiples services composites dignes de confiance (pour une configuration de seuils de confiance $\lambda\text{Sup}_k = 0.7$, $\lambda\text{Inf}_k = 0.3$ et $\gamma_k = 0.7$) à faible coût de communication.

Afin de confirmer empiriquement l'intuition évoquée dans la première expérimentation (la corrélation potentielle entre la stabilité d'une coalition et sa taille), nous allons examiner de plus près comment ces coalitions se forment et plus précisément comment

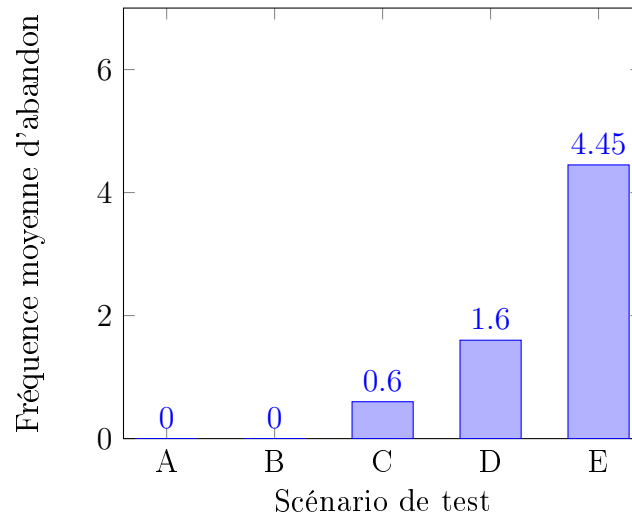


FIGURE 7.14 – La fréquence moyenne d’abandon par scénario de test

les agents se comportent durant le processus de formation de coalitions. La figure 7.14 illustre la fréquence moyenne d’abandon des agents sur 100 simulations par scénario de test.

La première observation à noter est que les scénarios A et B demeurent stables durant les 100 exécutions et qu’aucun agent ne quitte sa coalition. Pour le scénario A (i.e. 2 fonctionnalités requises) cela n’est pas envisageable car une seule sélection réussie complète la coalition intermédiaire singleton. De la même façon, dans le scénario B l’abandon d’une coalition n’est théoriquement pas possible, étant donné que dans ce cas la majorité correspond aux deux membres de la coalition. Ainsi, l’adhésion d’un nouveau membre n’entraîne pas l’instabilité du groupe.

La deuxième observation à noter concerne l’évolution de la fréquence d’abandon des agents. La figure 7.14 montre que la fréquence d’abandon des agents est proportionnelle au nombre de fonctionnalités requises. Par exemple, dans les scénario de test C, D et E, en moyenne 0.6, 1.6 et 4.45 agents quittent respectivement leurs coalitions durant le processus de formation. Cette capacité de quitter une coalition accorde aux agents l’autonomie pour décider avec qui coopérer assurant une satisfaction intra-groupe ce qui améliore la qualité du service composite sous-jacent.

7.4 Conclusion

Dans ce chapitre, nous avons exposé les résultats expérimentaux de notre approche de découverte, de sélection et de composition de services. Dans un premier temps,

nous avons justifié les choix techniques, à savoir la plateforme de développement et le langage de stockage de données, adoptés pour l'implémentation d'un prototype. Puis, nous avons décrits les données simulées et réelles utilisées pour notre validation expérimentale. La validation expérimentale est composée de quatre parties.

Dans la première partie, nous avons présenté le temps de calcul et le nombre moyen des messages échangés en fonction du nombre d'agents dans le réseau social, le seuil de confiance en la crédibilité sociale, le nombre des types de relations, et le longueur de la chaîne fournisseur-recommandeur.

Nous avons évalué dans la deuxième partie la performance de notre approche de découverte de services en termes d'efficacité, efficience et qualité des résultats. Les résultats obtenus ont montré que le mécanisme de découverte proposé est bien adapté aux réseaux sociaux réels.

Dans la troisième partie, nous avons évalué la performance de notre approche de sélection de services en terme d'utilité et de satisfaction des demandeurs via une étude comparative avec d'autres approches de sélection. Les résultats ont démontré que notre approche surpasse les autres approches similaires et donne des résultats plus efficaces.

Dans la dernière partie, nous avons évalué la performance de notre approche de composition de services basée sur la formation de coalitions. Ces résultats démontrent que le modèle offre certains avantages aux agents, mais à un coût important pour établir la stabilité. Les travaux à venir sont nécessaires pour explorer davantage les implications de notre approche sur la stabilité des coalitions ; et pour réviser et optimiser le modèle en conséquence. Nous devons apprendre davantage au sujet de plusieurs questions telles que la façon de déterminer les valeurs appropriées pour les seuils utilisés dans la prise de décision ou de rajouter des contraintes qui conditionnent l'abandon d'une coalition.

Chapitre 8

Conclusion générale

Étant donné que dans chacun des chapitres précédents nous avons déjà fourni un bref résumé du travail présenté dans le chapitre, nous exposons ici les plus importantes contributions de la thèse et nous discutons les pistes de recherches qui permettraient d'améliorer les approches proposées.

Contribution du travail

La principale contribution de cette thèse consiste en la proposition à la fois d'approches multi-agents distribuées pour la découverte, la sélection et la composition de services dans les réseaux sociaux et d'un modèle de confiance guidant les interactions entre les agents en agissant sur leurs prises de décision. Pour ce faire, nous avons proposé :

- Un modèle de confiance bâti sur les propriétés sociales de la confiance dans le monde réel et basé sur trois composantes : la crédibilité sociale, la qualité de la recommandation et l'expertise ;
- Un modèle de confiance précis, expressif et robuste intégrant une variété de sources d'information : l'expérience individuelle, l'expérience communiquée, les informations sociales et les informations certifiées fournies par l'agent évalué ;
- Une architecture basée sur la confiance combinant technologies service Web et agent cognitif ;

- Un modèle d'interaction basé sur un système de références distribué permettant aux agents à la fois de raisonner localement sur les besoins complexes du demandeur et d'échanger leurs informations et les valeurs de confiance ;
- Un algorithme distribué fondé sur la confiance pour la découverte de services assurant une propagation efficace de la recherche des fournisseurs dans le réseau social ;
- Un nouveau modèle probabiliste et contextuel pour calculer la confiance entre le demandeur de services et un fournisseur découvert non adjacent en prenant en considération les rôles des agents intermédiaires ;
- Un nouveau processus incrémental, dynamique et recouvrant de formation de coalitions pour une composition de services Web fondée sur la confiance ;
- Nous introduisons une nouvelle composante de confiance appelée la confiance en la coopération permettant aux membres d'une coalition d'évaluer la fiabilité d'un candidat afin d'accepter son adhésion.

A la lumière de ce qui précède, nous espérons que cette thèse constitue une contribution dans la modélisation de la confiance et la résolution distribuée de problèmes en utilisant une approche multi-agents et, à travers de ces domaines d'application, dans la découverte, la sélection et la composition de services Web dans les réseaux sociaux.

Perspectives

Nos travaux de recherches permettent d'ouvrir plusieurs perspectives de recherche qui requièrent une investigation plus profonde soit en enrichissant le modèle de confiance ou en cherchant à inscrire notre approche dans un champ d'interaction plus large. Nous allons présenter des perspectives envisageables à court et moyen terme, et des perspectives de recherches plus générales et envisageables à long terme.

Perspectives à court terme

Enrichir le modèle de confiance. Dans cette thèse, le modèle de confiance proposé est un modèle riche et expressif bâti sur les composantes de la crédibilité sociale, la recommandation, l'expertise et la coopération. Cependant, son mécanisme d'évaluation

est statique durant l'exécution. Au cours d'une conversation, la valeur de la confiance entre deux agents peut évoluer en fonction de leur comportement. Afin de rendre le mécanisme d'évaluation *dynamique*, nous planifions de mettre en place une méthode efficace de mise à jour de type récompense/pénalité. Les valeurs de confiance seront ajustées en se basant sur le suivi et l'analyse en temps réel des conversations entre les agents.

Par ailleurs, nous avons l'intention d'intégrer la notion de *méfiance* appelée aussi *relation de confiance négative* dans la modélisation de la confiance. Cela nous permettra de couvrir d'autres situations du monde réel où la relation entre deux individus est marquée par la méfiance et la déception. Notons que la confiance nulle et la méfiance ne sont pas équivalentes et ne peuvent par conséquent être confondues. La modélisation explicite de la méfiance a des conséquences directes sur l'évaluation de la confiance et sa concrétisation nécessite une étude approfondie. Les premiers travaux de la mise en œuvre de la méfiance ont été initiés par [Josa 03, Guha 04, Zieg 05].

Généraliser le modèle de requête. Dans la définition de requête proposée dans la thèse, nous nous sommes limités à l'expression des préférences globales d'un demandeur de services associées à tous les services requis. Ses préférences représentent son ordre de priorité sur les types de relations et les valeurs des seuils de confiance en la crédibilité sociale, en la recommandation et en l'expertise. Un premier travail serait d'affiner les préférences en fonction de la nature du service. Par exemple, pour la garde de son enfant, Alice de l'exemple de motivation (voir Chap. 1) peut favoriser une relation famille à une relation d'ami. Par contre, pour le transport elle préfère emprunter un moyen de transport public (par exemple, train ou avion) qu'un transport privé (par exemple, covoiturage). Cela permettra d'avoir plus de précision et d'efficacité dans la recherche. De plus, il serait intéressant de mettre en place un mécanisme de suggestion de services par l'application d'algorithmes d'apprentissage de profil demandeur en mémorisant ses préférences. Ainsi, lorsque plusieurs services candidats répondent aux besoins du demandeur, son profil pourrait permettre de faire une sélection adéquate selon ses préférences. Enfin, il devrait être possible pour l'utilisateur d'exprimer plusieurs contraintes sur les services qu'il requiert, par exemple "*Je souhaite réserver une chambre d'hôtel pour moins de 90€ et si possible dans l'hôtel le plus étoilé*". Il arrive alors que ces contraintes concernent le même service demandé, et que plusieurs services candidats répondent à toutes ou une partie des contraintes formulées.

Exploiter les interactions entre les services. Dans le modèle de confiance proposé, nous avons considéré les interactions de type agent/agent et de type agent/service (voir Sect. 4.3). Les interactions de type agent/agent sont les relations sociales entre les utilisateurs dans le réseau social et les interactions de type agent/service sont les évaluations attribuées par les utilisateurs aux services après une exécution réussie. Un

troisième type d'interaction que nous pouvons prendre en considération dans la modélisation de la confiance est l'interaction de type service/service dans une composition de services. L'idée est de capitaliser sur ces interactions à l'aide d'un réseau social de services (où les nœuds désignent les services et les arêtes représentent les relations de coopérations entre eux). L'utilisation d'un réseau social de services a été introduite par [Maam 10, Maam 11a, Maam 11b]. Les auteurs ont montré comment les interactions entre les services Web à savoir, la collaboration, la substitution et la concurrence peuvent être exploitées pour améliorer la découverte de services.

Dans notre cas, nous envisageons de réutiliser les compositions de services réussies en les transformant en une séquence d'interactions qui peuvent guider et améliorer le processus de sélection. L'avantage d'une telle représentation réside dans le fait que d'une part, elle constitue une alternative intéressante pour la découverte de services autre que les systèmes de références et d'autre part, elle permet aux agents d'enrichir leur accointances sociales pour les futures coopérations et d'établir une classification de services corrélés ou des partenariats.

Perspectives à moyen terme

Intégrer la négociation dans le processus de formation de coalitions. Le processus de formation de coalitions que nous avons présenté dans le chapitre 6 est un processus dynamique permettant aux agents de quitter une coalition d'une manière autonome en fonction de leurs décisions locales basées sur la confiance. Cependant, nous avons pu constater durant les évaluations que lorsque le nombre de fonctionnalités requises augmente, le nombre de coalitions complètes générées diminue. Cela est expliqué par le fait que les grandes coalitions sont plus sujettes à l'instabilité. Pour pallier à ce problème, nous envisageons d'élaborer *un mécanisme de négociation* engageant le broker et un membre voulant quitter la coalition. Doté de stratégies de persuasions, le broker négocie avec ce membre pour le convaincre de changer d'avis et rester dans la coalition. Les stratégies de persuasions peuvent dépendre de l'état de la coalition (i.e., nombre de fonctionnalités non satisfaites) et du niveau d'expertise du membre.

Étudier le facteur de malveillance. Dans cette thèse, nous avons fait l'hypothèse que les agents sont honnêtes dans l'échange et le partage d'informations et qu'ils ont la bonne volonté pour satisfaire les besoins du demandeur de services. Cependant, dans des environnements réels tels que les réseaux sociaux, les agents sont susceptibles d'être malveillants en propageant de mauvaises informations dans le réseau social. Ils peuvent aussi manipuler les valeurs annoncées de leurs qualité de services dans le but d'améliorer les valeurs de leur confiance. Il est donc recommandé de prendre en considération ce

facteur dans les simulations. Cela nous permettra d'évaluer la performance de notre approche dans des conditions de manipulation et de mesurer sa robustesse.

Examiner la stabilité de la coalition. Le processus de formation de coalitions que nous avons présenté dans le chapitre 6 permet une génération séquentielle de coalitions. Un agent quitte sa coalition en cas d'insatisfaction par rapport à la présence d'un nouveau membre. Dans ces conditions, la stabilité de la coalition dépend exclusivement de la composition de ses membres et de la confiance mutuelle entre eux. En théorie de jeux, chaque coalition est caractérisée par une valeur appelée utilité ou gain partagé. La stabilité indique qu'aucune coalition ne peut dévier de telle sorte que les agents n'ont pas intérêt à quitter la coalition pour rejoindre une autre plus profitable. Nous distinguons ainsi la stabilité d'un joueur et celle d'une coalition. La stabilité d'une coalition est définie par l'écart-type des gains à l'intérieur de cette coalition, cela varie donc en fonction de la coalition. La stabilité d'un joueur par contre est intrinsèque. C'est un trait de sa personnalité prédéfini et figé. Plusieurs méthodes ont été proposées dans la littérature pour étudier la stabilité d'une coalition comme la notion du cœur, l'équilibre de Nash, la stabilité individuelle et la stabilité contractuelle individuelle.

Comme futur travail, nous avons l'intention d'examiner la notion de stabilité en proposant un modèle qui définit l'utilité d'une composition de services. Nous envisageons également de générer les coalitions d'une manière parallèle pour permettre aux agents de quitter une coalition pour rejoindre une autre si celle-ci s'avère plus profitable.

Perspectives à long terme

Les approches actuelles pour la composition de services traitent toutes d'une composition à *la demande* où les besoins sont explicitement formulés par le demandeur. Des recherches pourraient être conduites dans la conception d'approches de composition anticipée de services. Les besoins à satisfaire dans ce type d'approches ne seraient pas explicitement exprimés mais inférés des actions du demandeur et de son réseau social. Le contexte et les actions du demandeur peuvent par exemple être un simple changement d'horaire d'un billet d'avion, la fin des provisions en nourriture, un anniversaire, prise de congés, etc. Il faut alors concevoir des services capables de percevoir *le contexte du demandeur* et son réseau social, détecter et répercuter les changements qui y ont opéré (le demandeur devant prendre un train après son vol se voit changer l'horaire de son billet de train), découvrir dynamiquement et collaborer avec les services en rapport avec le contexte du demandeur.

Bibliographie

- [Abdu 00] A. Abdul-Rahman and S. Hailes. “Supporting Trust in Virtual Communities”. In : *Proceedings of the 33rd Hawaii International Conference on System Sciences - Volume 6*, pp. 6007–6016, IEEE Computer Society, 2000.
- [Aber 01] K. Aberer and Z. Despotovic. “Managing Trust in a Peer-2-peer Information System”. In : *Proceedings of the Tenth International Conference on Information and Knowledge Management*, pp. 310–317, ACM, 2001.
- [Adle 01] P. Adler. “Market, Hierarchy, and Trust : The Knowledge Economy and the Future of Capitalism”. *Organization Science*, Vol. 12, pp. 215–234, 2001.
- [Adom 05a] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin. “Incorporating Contextual Information in Recommender Systems Using a Multidimensional Approach”. *ACM Trans. Inf. Syst.*, Vol. 23, No. 1, pp. 103–145, 2005.
- [Adom 05b] G. Adomavicius and A. Tuzhilin. “Toward the Next Generation of Recommender Systems : A Survey of the State-of-the-Art and Possible Extensions”. *IEEE Trans. on Knowl. and Data Eng.*, Vol. 17, No. 6, pp. 734–749, 2005.
- [Agga 04] R. Aggarwal, K. Verma, J. A. Miller, and W. Milnor. “Constraint Driven Web Service Composition in METEOR-S”. In : *IEEE SCC*, pp. 23–30, IEEE Computer Society, 2004.
- [Akco 11] C. Akcora, G. Carminati, and E. Ferrari. “Network and profile based measures for user similarities on social networks”. In : *IRI*, pp. 292–298, 2011.

- [Akni 00] S. Aknine and S. Pinson. “Méthodes de Coordination multi-agent par formation de coalitions”. In : *Proceedings of the 8èmes Journées Francophones d’Intelligence Artificielle et Systèmes Multi-Agents*, pp. 149–161, Hermès, 2000.
- [Akni 04] S. Aknine, S. Pinson, and M. Shakun. “A Multi-Agent Coalition Formation Method Based on Preference Models”. *Group Decision and Negotiation*, Vol. 13, No. 6, pp. 513–538, 2004.
- [Al S 09] J. Al-Sharawneh and M.-A. Williams. “ABMS : Agent-Based Modeling and Simulation in Web Service Selection”. In : *Management and Service Science*, pp. 1–6, 2009.
- [Al S 12] J. Al-Sharawneh. *Social Networks : Service Selection and Recommendation*. PhD thesis, University of Technology, Sydney, 2012.
- [Amor 03] M. Amor, L. Fuentes, and J. M. Troya. “Putting Together Web Services and Compositional Software Agents”. In : *Proceedings of the 2003 International Conference on Web Engineering*, pp. 44–53, 2003.
- [Artz 07] D. Artz and Y. Gil. “A Survey of Trust in Computer Science and the Semantic Web”. *Web Semant.*, Vol. 5, No. 2, pp. 58–71, 2007.
- [Asl 13] E. K. Asl, J. Bentahar, H. Otrok, and R. Mizouni. “Efficient Coalition Formation for Web Services.”. In : *IEEE SCC*, pp. 737–744, IEEE, 2013.
- [Aust 62] J. L. Austin. *How to do things with words*. Harvard University Press, 1962.
- [Badr 10] Y. Badr and Z. Maamar. “Can Enterprises Capitalize on their Social Networks?”. *The Cutter IT Journal*, Vol. 22, No. 10, pp. 10–14, 2010.
- [Badr 11] Y. Badr, N. Faci, Z. Maamar, and F. Biennier. “Multi-level Social Networking to Enable and Foster Collaborative Organizations.”. In : *PRO-VE*, pp. 3–10, Springer, 2011.
- [Bala 97] M. Balabanovic. “An Adaptive Web Page Recommendation Service”. In : *Proceedings of the First International Conference on Autonomous Agents*, pp. 378–385, ACM Press, 1997.
- [Bans 10] S. K. Bansal, A. Bansal, and M. B. Blake. “Trust-based Dynamic Web service Composition using Social Network Analysis”. In : *IEEE International Workshop on Business Applications for Social Network Analysis*, pp. 1–8, 2010.

- [Bara 99] A.-L. Barabasi and R. Albert. "Emergence of Scaling in Random Networks". *Science*, Vol. 286, No. 5439, pp. 509–512, 1999.
- [Barb 14] C. Barbara, F. Elena, and V. Marco. *Security and Trust in Online Social Networks Synthesis Lectures on Information Security, Privacy, and Trust*. Morgan & Claypool publishers, 2014.
- [Bedi 07] P. Bedi, H. Kaur, and S. Marwaha. "Trust Based Recommender System for the Semantic Web". In : *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pp. 2677–2682, 2007.
- [Bena 03] B. Benatallah, Q. Z. Sheng, and M. Dumas. "The Self-Serv environment for Web services composition". *Internet Computing, IEEE*, Vol. 7, No. 1, pp. 40–48, 2003.
- [Bers 98] E. Berscheid, H. T. Reis, *et al.* "Attraction and close relationships". *The handbook of social psychology*, Vol. 2, pp. 193–281, 1998.
- [Bill 07] H. Billhardt, R. Hermoso, S. Ossowski, and R. Centeno. "Trust-based Service Provider Selection in Open Environments". In : *Proceedings of the 2007 ACM Symposium on Applied Computing*, pp. 1375–1380, 2007.
- [Bill 09] E. Billionniere, D. Greiman, and K. Gosha. "A Comparison of Social Service Selection Techniques". *DASC '09*, pp. 260–265, 2009.
- [Boon 91] S. Boon and J. Holmes. *Cooperation and prosocial behaviour*. Cambridge University Press, 1991.
- [Bori 08] S. Boriah, V. Chandola, and V. Kumar. "Similarity Measures for Categorical Data : A Comparative Evaluation". *SIAM*, Vol. 30, No. 2, pp. 243–254, 2008.
- [Bour 09] J. Bourdon, L. Vercouter, and T. Ishida. "A Multiagent Model for Provider-Centered Trust in Composite Web Services.". In : *PRIMA*, pp. 216–228, Springer, 2009.
- [Bras 11] P. Brassier. *Boostez vos ventes grâce aux réseaux sociaux*. Editions Eyrolles, 2011.
- [Braz 97] F. M. T. Brazier, B. Dunin-Keplicz, N. R. Jennings, and J. Treur. "DESIRE : Modelling Multi-Agent Systems in a Compositional Formal Framework.". *Int. J. Cooperative Inf. Syst.*, Vol. 6, No. 1, pp. 67–94, 1997.

- [Bree 98] J. S. Breese, D. Heckerman, and C. Kadie. "Empirical Analysis of Predictive Algorithms for Collaborative Filtering". In : *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pp. 43–52, 1998.
- [Brys 00] K. Bryson, M. Luck, M. Joy, and D. T. Jones. "Applying Agents to Bioinformatics in GeneWeaver". In : *CIA*, pp. 60–71, 2000.
- [Burk 02] R. Burke. "Hybrid Recommender Systems : Survey and Experiments.". *User Modeling and User-Adapted Interaction*, Vol. 12, No. 4, pp. 331–370, 2002.
- [Burk 07] R. Burke. "Hybrid Web Recommender Systems". In : *The Adaptive Web : Methods and Strategies of Web Personalization*, pp. 377–408, 2007.
- [Burn 70] T. Burnaby. "On a method for character weighting a similarity coefficient, employing the concept of information". *Mathematical Geology*, Vol. 2, pp. 25–38, 1970.
- [Caba 05] L. Cabac and D. Moldt. "Formal Semantics for AUML Agent Interaction Protocol Diagrams". In : *5th International Workshop on Agent-Oriented Software Engineering*, pp. 47–61, Springer, 2005.
- [Cabr 03a] G. Cabri, L. Leonardi, and F. Zambonelli. "BRAIN : A Framework for Flexible Role-Based Interactions in Multiagent Systems.". In : *CoopIS/DOA/ODBASE*, pp. 145–161, Springer, 2003.
- [Cabr 03b] G. Cabri, L. Leonardi, and F. Zambonelli. "Implementing Role-based Interactions for Internet Agents.". In : *SAINT*, pp. 380–389, IEEE Computer Society, 2003.
- [Cail 02] P. Caillou, S. Aknine, and S. Pinson. "A Multi-agent Method for Forming and Dynamic Restructuring of Pareto Optimal Coalitions". In : *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 1074–1081, ACM, 2002.
- [Camp 90] J. A. Campbell and M. P. D’Inverno. "Knowledge Interchange Protocols". In : *Decentralized A.I. : Proc. of the First European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, pp. 63–80, 1990.
- [Cart 77] D. Cartwright and F. Harary. "A graph theoretic approach to the investigation of system-environment relationships.". *The Journal of Mathematical Sociology*, Vol. 5, pp. 87–111, 1977.

- [Casa 01] F. Casati, M. Sayal, and M.-C. Shan. "Developing E-Services for Composing E-Services.". In : *CAiSE*, pp. 171–186, Springer, 2001.
- [Cast 95] C. Castelfranchi. "Commitments : From Individual Intentions to Groups and Organizations.". In : *ICMAS*, pp. 41–48, The MIT Press, 1995.
- [Cast 98] C. Castelfranchi and R. Falcone. "Principles of Trust for MAS : Cognitive Anatomy, Social Importance, and Quantification". In : *ICMAS*, pp. 72–79, IEEE Computer Society, 1998.
- [Char 13] Y. Charif-Djebbar and N. Sabouret. "Dynamic service composition enabled by introspective agent coordination". *Autonomous Agents and Multi-Agent Systems*, Vol. 26, No. 1, pp. 54–85, 2013.
- [Chen 01] M. Chen and J. P. Singh. "Computing and using reputations for internet ratings.". In : *ACM Conference on Electronic Commerce*, pp. 154–162, ACM, 2001.
- [Chiu 09] D. K. W. Chiu, H.-F. Leung, and K. man Lam. "On the making of service recommendations : An action theory based on utility, reputation, and risk attitude.". *Expert Syst. Appl.*, Vol. 36, No. 2, pp. 3293–3301, 2009.
- [Clar 01] D. Clark. "Face-to-Face with Peer-to-Peer Networking.". *IEEE Computer*, Vol. 34, No. 1, pp. 18–21, 2001.
- [Clay 01] M. Claypool, P. Le, M. Wased, and D. Brown. "Implicit interest indicators". In : *Proceedings of the 6th international conference on Intelligent user interfaces*, pp. 33–40, ACM, 2001.
- [Cole 98] J. S. Coleman. *Foundations of Social Theory*. Harvard University Press, 1998.
- [Colm 07] A. W. Colman and J. Han. "Roles, players and adaptable organizations.". *Applied Ontology*, Vol. 2, No. 2, pp. 105–126, 2007.
- [Cont 03] R. Conte and M. Paolucci. "Social Cognitive Factors of Unfair Ratings in Reputation Reporting Systems". In : *Proceedings of the 2003 IEEE/WIC International Conference on Web Intelligence*, pp. 316–322, IEEE Computer Society, 2003.
- [Cotr 02] D. Cotroneo, M. Gargiulo, S. Russo, and G. Ventre. "Improving the Availability of Web Services". In : *22nd International Conference on Software Engineering (ICSE 2002)*, pp. 59–63, 2002.

- [Da S 12] I. Da Silva and A. Zisman. "A framework for trusted services". In : *Proceedings of the 10th international conference on Service-Oriented Computing*, pp. 328–343, Springer-Verlag, 2012.
- [Dami 02] E. Damiani, D. C. Di Vimercati, and S. Paraboschi. "A Reputation-Based Approach for Choosing Reliable Resources in Peer-to-Peer Networks". In : *Proceedings of the 9th Conference on computer and communications security*, pp. 207–216, ACM, 2002.
- [Day 04] J. Day and R. Deters. "Selecting the Best Web Service". In : *Proceedings of the 2004 Conference of the Centre for Advanced Studies on Collaborative Research*, pp. 293–307, IBM Press, 2004.
- [Dema 95] Y. Demazeau. "From Interactions To Collective Behaviour In Agent-Based Systems". In : *Proceedings of the 1st. European Conference on Cognitive Science. Saint-Malo*, pp. 117–132, 1995.
- [Douc 02] J. R. Douceur. "The Sybil Attack". In : *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, pp. 251–260, Springer-Verlag, 2002.
- [Dunn 84] J. Dunn. "The concept of trust in the politics of John Locke". *Cambridge University Press*, 1984.
- [Dust 05] S. Dustdar and W. Schreiner. "A survey on web services composition". *International journal of web and grid services*, Vol. 1, No. 1, pp. 1–30, 2005.
- [Elsa 96] A. E. Elsayed. *Reliability Engineering*. Addison Wesley, 1996.
- [Erdo 60] P. Erdos and A. Renyi. "On the evolution of random graphs". *Publ. Math. Inst. Hung. Acad. Sci.*, Vol. 7, p. 17, 1960.
- [Ermo 03] V. Ermolayev, N. Keberle, and S. Plaksin. "Towards Agent-Based Rational Service Composition - RACING Approach.". In : *ICWS-Europe*, pp. 167–182, Springer, 2003.
- [Esfa 01] B. Esfandiari and S. Chandrasekharan. "On How Agents Make Friends : Mechanisms for Trust Acquisition". In : *Proceedings of the Fourth Workshop on Deception, Fraud and Trust in Agent Societies*, pp. 27–34, 2001.
- [Falc 04] R. Falcone and C. Castelfranchi. "Trust Dynamics : How Trust Is Influenced by Direct Experiences and by Trust Itself". In : *Proceedings of the*

- Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 740–747, IEEE Computer Society, 2004.
- [Ferb 07] J. Ferber. *Les Systèmes multi-agents : Vers une intelligence collective*. Dunod, 2007.
- [Ferb 98] J. Ferber and O. Gutknecht. “A Meta-Model for the Analysis and Design of Organizations in Multi-Agent Systems.”. In : *ICMAS*, pp. 128–135, IEEE Computer Society, 1998.
- [Ferg 92] I. A. Ferguson. “Touring Machines : Autonomous Agents with Attitudes.”. *IEEE Computer*, Vol. 25, No. 5, pp. 51–55, 1992.
- [Fini 94] T. Finin, R. Fritzson, D. McKay, and R. McEntire. “KQML As an Agent Communication Language”. In : *Proceedings of the Third International Conference on Information and Knowledge Management*, pp. 456–463, ACM, 1994.
- [Fisk 13] S. T. Fiske. *Social Beings : Core Motives in Social Psychology*. John Wiley and Sons Press, 2013.
- [Fone 96] L. N. Foner. “A Multi-Agent Referral System for Matchmaking”. In : *Proceedings of the 1st International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology*, pp. 245–261, 1996.
- [Gamb 88] D. Gambetta. *Can We Trust Trust ?* Department of Sociology, University of Oxford, 1988. This was originally published in hardcopy in 1988, this is electronic copy circa 2000.
- [Gart 97] L. Garton, C. Haythornthwaite, and B. Wellman. “Studying Online Social Networks”. *Journal of Computer Mediated Communication*, Vol. 3, No. 1, 1997.
- [Geor 50] S. Georg. *The Sociology of Georg Simmel*. The free Press, 1950.
- [Geor 87] M. P. Georgeff and A. L. Lansky. “Reactive Reasoning and Planning.”. In : *AAAI*, pp. 677–682, 1987.
- [Golb 05] J. Golbeck. *Computing and Applying Trust in Web-Based Social Networks*. PhD thesis, University of Maryland at College Park, 2005.
- [Golb 06a] J. Golbeck. “Generating predictive movie recommendations from trust in social networks”. In : *Proceedings of the 4th international conference on Trust Management*, pp. 93–104, Springer-Verlag, 2006.

- [Golb 06b] J. Golbeck and J. Hendler. “Inferring binary trust relationships in Web-based social networks”. *ACM Transactions on Internet Technology*, Vol. 6, No. 4, pp. 497–529, 2006.
- [Gold 08] D. H. Goldbaum. “Follow the Leader : Simulations on a Dynamic Social Network”. Tech. Rep. 155, Finance Discipline Group, UTS Business School, University of Technology, Sydney, 2008.
- [Gome 12] S. Gómez, A. Díaz-Guilera, J. Gómez-Gardeñes, C. J. P. Vicente, Y. Moreno, and A. Arenas. “Diffusion dynamics on multiplex networks”. *CoRR*, 2012.
- [Gray 03] E. Gray, J.-M. Seigneur, Y. Chen, and C. D. Jensen. “Trust Propagation in Small Worlds.”. In : *iTrust*, pp. 239–254, Springer, 2003.
- [Grif 03] N. Griffiths and M. Luck. “Coalition formation through motivation and trust”. In : *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pp. 17–24, ACM, 2003.
- [Gros 05] R. Gross and A. Acquisti. “Information revelation and privacy in online social networks”. In : *Proceedings of the 2005 ACM workshop on Privacy in the electronic society*, pp. 71–80, ACM, 2005.
- [Guha 04] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. “Propagation of Trust and Distrust”. In : *Proceedings of the 13th International Conference on World Wide Web*, pp. 403–412, ACM, 2004.
- [Hang 09] C.-W. Hang, Y. Wang, and M. P. Singh. “Operators for propagating trust and their evaluation in social networks”. In : *AAMAS*, pp. 1025–1032, International Foundation for Autonomous Agents and Multiagent Systems, 2009.
- [Hang 10] C.-W. Hang and M. P. Singh. “Trust-based recommendation based on graph similarity”. In : *in 13th AAMAS Workshop on Trust in Agent Societies (Trust)*, 2010.
- [Hard 93] R. Hardin. “The Street-Level Epistemology of Trust”. *Politics & Society*, Vol. 21, No. 4, pp. 505–529, 1993.
- [Hend 01] H. Hendler. “Agents and the Semantic Web”. *IEEE intelligent systems*, Vol. 16, No. 2, pp. 30–37, 2001.
- [Hess 08] C. Heß. *Trust-based Recommendations in Multi-layer Networks*. IOS Press, 2008.

- [Hong 09] T. Hongxia, C. Jian, Z. Shensheng, and L. Minglu. "A Distributed Agent Coalition Algorithm for Web Service Composition". *2014 IEEE World Congress on Services*, pp. 62–69, 2009.
- [Horl 05] B. Horling and V. Lesser. "A Survey of Multi-Agent Organizational Paradigms". *The Knowledge Engineering Review*, Vol. 19, No. 4, pp. 281–316, 2005.
- [Huan 14] K. Huang, Y. Liu, S. Nepal, Y. Fan, S. Chen, and W. Tan. "A Novel Equitable Trustworthy Mechanism for Service Recommendation in the Evolving Service Ecosystem". In : *Service-Oriented Computing*, pp. 510–517, Springer Berlin Heidelberg, 2014.
- [Huhn 02] M. Huhns. "Agents as Web Services". *IEEE Internet Computing*, Vol. 6, No. 4, pp. 93–95, 2002.
- [Huhn 03] M. Huhns. "Software Agents : The Future of Web Services". In : *Agent Technologies, Infrastructures, Tools, and Applications for E-Services*, Chap. 1, pp. 1–18, Springer, 2003.
- [Huyn 04] T. D. Huynh, N. R. Jennings, and N. R. Shadbolt. "FIRE : An Integrated Trust and Reputation Model for Open Multi-Agent Systems". In : *ECAI*, pp. 18–22, 2004.
- [Huyn 06] T. D. Huynh, N. R. Jennings, and N. R. Shadbolt. "An Integrated Trust and Reputation Model for Open Multi-agent Systems". *Autonomous Agents and Multi-Agent Systems*, Vol. 13, No. 2, pp. 119–154, 2006.
- [Indr 04] R. Indrajit and C. Sudip. "A Vector Model of Trust for Developing Trustworthy Systems". In : *Computer Security*, pp. 260–275, Springer, 2004.
- [Ivan 08] M. Ivana and G. Alex. *Trust and Distrust : Sociocultural Perspectives*. Information Age Publishing, 2008.
- [Jack 96] M. Jackson and A. Wolinsky. "A Strategic Model of Social and Economic Networks". *Journal of Economic Theory*, Vol. 71, pp. 44–74, 1996.
- [Jeon 01] H. Jeong, S. Mason, A. L. Barabási, and Z. N. Oltvai. "Lethality and centrality in protein networks". *Nature*, Vol. 411, pp. 41–42, 2001.
- [Jian 12] S. Jiang, J. Zhang, and Y. Ong. "A multiagent evolutionary framework based on trust for multiobjective optimization.". In : *AAMAS*, pp. 299–306, 2012.

- [Josa 03] A. Jøsang, E. Gray, and M. Kinatader. “Analysing topologies of transitive trust.”. In : *Proceedings of the Workshop of Formal Aspects of Security and Trust*, pp. 403–412, ACM, 2003.
- [Josa 07] A. Jøsang, R. Ismail, and C. Boyd. “A Survey of Trust and Reputation Systems for Online Service Provision”. *Decis. Support Syst.*, Vol. 43, No. 2, pp. 618–644, 2007.
- [Josa 08] A. Jøsang. “Online reputation systems for the health sector”. *electronic Journal of Health Informatics*, Vol. 3, No. 1, 2008.
- [Kale 04] S. Kalepu, S. Krishnaswamy, and S. W. Loke. “Reputation = F(User Ranking, Compliance, Verity)”. In : *Proceedings of the IEEE International Conference on Web Services*, pp. 200–, IEEE Computer Society, 2004.
- [Kan 01] G. Kan. *Peer-to-Peer : Harnessing the Power of Disruptive Technologies*. O’Reilly & Associates, Inc., 2001.
- [Kart 05] K. Karta. *An investigation on personalized collaborative filtering for web service selection*. PhD thesis, University of Western Australia, 2005.
- [Kauf 05] L. Kaufman and P. Rousseeuw. *Finding Groups in Data : An Introduction to Cluster Analysis*. Wiley-Interscience, 2005.
- [Kaut 97] H. Kautz, B. Selman, and M. Shah. “ReferralWeb : Combining Social Networks and Collaborative Filtering”. *Communications of the ACM*, Vol. 40, No. 3, 1997.
- [Kend 95] E. A. Kendall, M. T. Malkoun, and C. H. Jiang. “A Methodology for Developing Agent Based Systems.”. In : *DAI*, pp. 85–99, Springer, 1995.
- [Klei 00] J. Kleinberg. “The Small-world Phenomenon : An Algorithmic Perspective”. In : *Proceedings of the Thirty-second Annual ACM Symposium on Theory of Computing*, pp. 163–170, ACM, 2000.
- [Klus 01] M. Klusch and K. P. Sycara. “Brokering and Matchmaking for Coordination of Agent Societies : A Survey.”. In : *Coordination of Internet Agents : Models, Technologies, and Applications*, pp. 197–224, Springer-Verlag, 2001.
- [Koni 01] J.-L. Koning. “Designing and Testing Negotiation Protocols for Electronic Commerce Applications.”. In : *AgentLink*, pp. 34–60, Springer, 2001.

- [Kram 99] R. M. Kramer. "Trust and distrust in organizations : Emerging perspectives, enduring questions". *Annual review of psychology*, Vol. 50, No. 1, pp. 569–598, 1999.
- [Krit 07] A. Kritikopoulos, M. Sideri, and I. Varlamis. "Success index : measuring the efficiency of search engines using implicit user feedback". In : *Proceedings of the 11th Pan-Hellenic Conference on Informatics*, Web Search and Mining Track, PCI, 2007.
- [Kute 07] U. Kuter and J. Golbeck. "SUNNY : A New Algorithm for Trust Inference in Social Networks Using Probabilistic Confidence Models.". In : *AAAI*, pp. 1377–1382, AAAI Press, 2007.
- [Labr 94] Y. Labrou and T. Finin. "A Semantics Approach for KQML a General Purpose Communication Language for Software Agents". In : *Proceedings of the Third International Conference on Information and Knowledge Management*, pp. 447–455, ACM, 1994.
- [Lala 12] F. Lalanne, A. Cavalli, and S. Maag. "Quality of Experience as a Selection Criterion for Web Services". In : *SITIS*, pp. 519–526, 2012.
- [Lang 01] A. Langlet. *Peer-to-Peer : Harnessing the Power of Disruptive Technologies*. O'Reilly & Associates, Inc., 2001.
- [Lee 03] K. Lee, J. Jeon, W. Lee, S.-H. Jeong, and S.-W. Park. "QoS for Web Services : Requirements and Possible Approaches". Tech. Rep., W3C, Web Services Architecture Working Group, 2003.
- [Leti 08] I. Letia and O. Pop. "Web service selection based on trust.". In : *International conference on Development and Application Systems*, pp. 222–227, 2008.
- [Lewi 95] R. J. Lewicki and B. B. Bunker. "Trust in relationships : A model of development and decline". In : *Conflict, cooperation, and justice : Essays inspired by the work of Morton Deutsch.*, pp. 133–173, 1995.
- [Li 03] Q. Li and B. M. Kim. "An Approach for Combining Content-based and Collaborative Filters". In : *Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages*, pp. 17–24, Association for Computational Linguistics, 2003.
- [Li 09] L. Li, Y. Wang, and E.-P. Lim. "Trust-Oriented Composite Service Selection and Discovery.". In : *ICSOC/ServiceWave*, pp. 50–67, 2009.

- [Li 12] M. Li, Z. Hua, J. Zhao, Y. Zou, and B. Xie. “ARIMA model-based web services trustworthiness evaluation and prediction”. In : *ICSOC*, pp. 648–655, 2012.
- [Li 14a] L. Li and Y. Wang. “The Roadmap of Trust and Trust Evaluation in Web Applications and Web Services.”. In : *Advanced Web Services*, pp. 75–99, Springer, 2014.
- [Li 14b] Z. Li and Y. Jiang. “Cross-layers Cascade in Multiplex Networks”. In : *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems*, pp. 269–276, 2014.
- [Liu 05] W. Liu. “Trustworthy service selection and composition—Reducing the entropy of service-oriented Web”. In : *Proceedings of the 3rd IEEE International Conference on Industrial Informatics*, pp. 104–109, IEEE Computer Society, 2005.
- [Liu 11] G. Liu, Y. Wang, and M. A. Orgun. “Trust Transitivity in Complex Social Networks.”. In : *AAAI*, pp. 1222–1229, AAAI Press, 2011.
- [Liu 14] G. Liu and Y. Wang. “Trust-Oriented Service Provider Selection in Complex Online Social Networks.”. In : *Advanced Web Services*, pp. 363–380, Springer, 2014.
- [Lomu 10] A. Lomuscio, M. Solanki, W. Penczek, and M. Szreter. “Runtime Monitoring of Contract Regulated Web Services”. In : *AAMAS*, pp. 1449–1450, 2010.
- [Loua 14] A. Louati, J. El Haddad, and S. Pinson. “A Multilevel Agent-Based Approach for Trustworthy Service Selection in Social Networks”. In : *Proceedings of the 2014 IEEE/WIC/ACM IAT - Volume 03*, pp. 214–221, IEEE Computer Society, 2014.
- [Luhm 12] N. Luhmann. *Trust and Power*. John Wiley and Sons Ltd., 2012.
- [Ma 08] H. Ma, H. Yang, M. R. Lyu, and I. King. “SoRec : Social Recommendation Using Probabilistic Matrix Factorization”. In : *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, pp. 931–940, ACM, 2008.
- [Ma 11] H. Ma, T. C. Zhou, M. R. Lyu, and I. King. “Improving Recommender Systems by Incorporating Social Contextual Information”. *ACM Trans. Inf. Syst.*, Vol. 29, No. 2, pp. 9 :1–9 :23, 2011.

- [Maam 05] Z. Maamar, S. K. Mostéfaoui, and H. Yahyaoui. "Toward an Agent-Based and Context-Oriented Approach for Web Services Composition.". *IEEE Trans. Knowl. Data Eng.*, Vol. 17, No. 5, pp. 686–697, 2005.
- [Maam 10] Z. Maamar, L. Wives, B. Youakim, E. Said, B. Khouloud, and F. Noura. "LinkedWS : A Novel Web Services Discovery Model Based on the Metaphor of Social Networks". *Simulation Modelling Practice and Theory*, Vol. 19, No. 2, pp. 121–132, 2010.
- [Maam 11a] Z. Maamar, H. Hacid, and M. Huhns. "Why Web Services Need Social Networks". *IEEE Internet Computing*, Vol. 15, No. 2, pp. 90–94, 2011.
- [Maam 11b] Z. Maamar, P. Santos, L. Wives, Y. Badr, N. Faci, and J. de Oliveira. "Using Social Networks for Web Services Discovery". *IEEE Internet Computing*, Vol. 15, pp. 48–54, 2011.
- [Maar 10] A. Maaradji, H. Hakim, J. Daigremont, and N. Crespi. "Towards a Social Network Based Approach for Services Composition". In : *IEEE International Conference on Communications*, pp. 1–5, 2010.
- [Mali 09] Z. Malik and A. Bouguettaya. "RATEWeb : Reputation Assessment for Trust Establishment among Web services". *VLDB J.*, Vol. 18, No. 4, pp. 885–911, 2009.
- [Mani 05] U. S. Manikrao and T. V. Prabhakar. "Dynamic Selection of Web Services with Recommendation System". *Next Generation Web Services Practices, International Conference on*, Vol. 0, pp. 117–121, 2005.
- [Mars 94] S. P. Marsh. *Formalising trust as a computational Concept*. PhD thesis, 1994.
- [Mass 04] P. Massa and B. Bhattacharjee. "Using Trust in Recommender Systems : An Experimental Analysis". In : *Proceeding of iTrust*, pp. 221–235, 2004.
- [Mass 05] P. Massa and P. Avesani. "Controversial Users Demand Local Trust Metrics : An Experimental Study on Epinions.com Community.". In : *AAAI*, pp. 121–126, 2005.
- [Mass 06] P. Massa. "A Survey of Trust Use and Modeling in Current Real Systems". In : *Trust in E-Services : Technologies, Practices and Challenges*, Idea Group Publishing, 2006.

- [Mass 07] P. Massa and P. Avesani. "Trust-aware Recommender Systems". In : *Proceedings of the 2007 ACM Conference on Recommender Systems*, pp. 17–24, ACM, 2007.
- [Mats 08] S. Matsuyama and T. Terano. "Analyzing the ENRON Communication Network Using Agent-Based Simulation.". *JNW*, Vol. 3, No. 7, pp. 26–33, 2008.
- [Maxi 05a] E. M. Maximilien and M. P. Singh. "Multiagent System for Dynamic Web Services Selection". In : *ISOCABE at AAMAS*, pp. 25–29, 2005.
- [Maxi 05b] E. M. Maximilien and M. P. Singh. "Toward Web Services Interaction Styles". In : *Proceedings of 2nd IEEE International Conference on Services Computing*, 2005.
- [McAu 12] J. J. McAuley and J. Leskovec. "Learning to Discover Social Circles in Ego Networks.". In : *NIPS*, pp. 548–556, 2012.
- [McCa 02] C. McCarty. "Structure in Personal Networks". *Journal of Social Structure*, Vol. 3, No. 1, 2002.
- [McDo 00] D. McDonald and M. S. Ackerman. "Expertise Recommender : A flexible recommendation system and architecture". In : *Proc. Int. Conf. on CSCW*, pp. 231–240, 2000.
- [Mcdo 03] D. W. McDonald. "Recommending collaboration with social networks : a comparative evaluation". In : *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 593–600, ACM Press, 2003.
- [McKn 01] D. H. McKnight and N. L. Chervany. "Trust and Distrust Definitions : One Bite at a Time". In : *Trust in Cyber-societies, Integrating the Human and Artificial Perspectives*, pp. 27–54, Springer-Verlag, 2001.
- [Mckn 96] D. H. Mcknight and N. L. Chervany. "The Meanings of Trust". Tech. Rep., Carlson School of Management, University of Minnesota, 1996.
- [Medj 14] B. Medjahed, Z. Malik, and S. Benbernoui. *Web Services Foundations*. Springer New York, 2014.
- [Misz 96] B. Misztal. *Trust in Modern Societies*. Polity, Oxford, 1996.
- [Mood 03] J. Moody and D. White. "Structural cohesion and embeddedness : A hierarchical concept of social groups". *American Sociological Review*, Vol. 68, No. 1, pp. 103–127, 2003.

- [More 33] J. L. Moreno. “Emotions mapped by new geography”. *New York Times*, 1933.
- [Mull 05] I. Muller and R. Kowalczyk. “Service composition through agent-based coalition formation”. In : *Proceedings of the Workshop on WWW Service Composition with Semantic Web Services*, pp. 34–43, 2005.
- [Mull 06] I. Muller, R. Kowalczyk, and P. Braun. “Towards Agent-Based Coalition Formation for Service Composition”. In : *Proceedings of the IEEE/WIC/ACM international conference on Intelligent Agent Technology*, pp. 73–80, IEEE Computer Society, 2006.
- [Nami 04] A. S. Namin, W. Shen, and H. Ghenniwa. “Web Services/Agent-Based Model for Inter-Enterprise Collaboration.”. In : *BASYS*, pp. 231–240, Springer, 2004.
- [Nard 08] L. G. Nardin, A. Brandão, J. S. a. Sichman, and L. Vercouter. “SOARI : A Service Oriented Architecture to Support Agent Reputation Models Interoperability.”. In : *AAMAS-TRUST*, pp. 292–307, Springer, 2008.
- [Nau 03] D. S. Nau, T.-C. Au, O. Ilghami, U. Kuter, J. W. Murdock, D. Wu, and F. Yaman. “SHOP2 : An HTN Planning System.”. *J. Artif. Intell. Res.*, Vol. 20, pp. 379–404, 2003.
- [Nevi 15] B. Neville, M. Fasli, and J. Pitt. “Utilising social recommendation for decision-making in distributed multi-agent systems.”. *Expert Systems with Applications*, Vol. 42, No. 6, pp. 2884–2906, 2015.
- [Nord 10] K. Nordheimer, T. Schulze, and D. Veit. “Trustworthiness in Networks : A Simulation Approach for Approximating Local Trust and Distrust Values.”. In : *IFIPTM*, pp. 157–171, Springer, 2010.
- [Odel 02] J. Odell, H. V. D. Parunak, and M. Fleischer. “The Role of Roles in Designing Effective Agent Organizations.”. In : *SELMAS*, pp. 27–38, Springer, 2002.
- [Odel 04] J. Odell, M. H. Nodine, and R. Levy. “A Metamodel for Agents, Roles, and Groups.”. In : *AOSE*, pp. 78–92, Springer, 2004.
- [Page 99] L. Page, S. Brin, R. Motwani, and T. Winograd. “The PageRank Citation Ranking : Bringing Order to the Web”. Tech. Rep., Stanford University, 1999.

- [Paik 06] I. Paik, D. Maruyama, and M. N. Huhns. "A Framework for Intelligent Web Services : Combined HTN and CSP Approach.". In : *ICWS*, pp. 959–962, IEEE Computer Society, 2006.
- [Papa 08] M. P. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann. "SERVICE-ORIENTED COMPUTING : A Research Roadmap". 2008.
- [Ping 05] X. Ping, G. Ji, and G. Hang. "Rating Reputaion : A Necessary Consideration in Reputation Mechanism". In : *Proceedings of 2005 International Conference on Machine Learning and Cybernetics*, pp. 182–187, IEEE, 2005.
- [Ponn 02] S. R. Ponnekanti and A. Fox. "SWORD : A developer toolkit for web service composition". In : *Proceedings of the 11th International WWW Conference (WWW2002)*, pp. 83–107, 2002.
- [Pujo 02] J. M. Pujol, R. Sangüesa, and J. Delgado. "Extracting Reputation in Multi Agent Systems by Means of Social Network Topology". In : *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems : Part 1*, pp. 467–474, ACM, 2002.
- [Raab 03] J. Raab and H. B. Milward. "Dark networks as problems.". *Journal of Public Administration Research and Theory*, Vol. 13, No. 4, pp. 413–439, 2003.
- [Ran 03] S. Ran. "A Model for Web Services Discovery with QoS". *SIGecom Exch.*, Vol. 4, No. 1, pp. 1–10, 2003.
- [Ratn 01] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. "A Scalable Content-addressable Network". *SIGCOMM Comput. Commun. Rev.*, Vol. 31, No. 4, pp. 161–172, 2001.
- [Rein 86] P. H. Reingen and J. B. Kernan. "Analysis of Referral Networks in Marketing : Methods and Illustration". *Journal of Marketing Research*, Vol. 23, No. 4, pp. pp. 370–378, 1986.
- [Resn 00] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman. "Reputation Systems". *Commun. ACM*, Vol. 43, No. 12, pp. 45–48, 2000.
- [Resn 02] P. Resnick and R. Zeckhauser. "Trust Among Strangers in Internet Transactions : Empirical Analysis of eBay's Reputation System". In : *The Economics of the Internet and E-Commerce*, pp. 127–157, Elsevier Science, 2002.

- [Rijs 79] C. J. V. Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, 1979.
- [Rott 80] J. B. Rotter. *Interpersonal trust, trustworthiness, and gullibility*. Vol. 35, 1980.
- [Rowl 79] M. Rowland. *Intimate Relationships*. McGraw-Hill College Press, 1979.
- [Saba 02] J. Sabater and C. Sierra. “Regret : a reputation model for gregarious societies”. In : *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pp. 475–482, ACM Press, 2002.
- [Saba 05] J. Sabater and C. Sierra. “Review on Computational Trust and Reputation Models”. *Artificial Intelligence Review*, Vol. 24, No. 1, pp. 33–60, 2005.
- [Saba 13] J. Sabater and L. Vercouter. “Trust and reputation in multiagent systems”. In : *Multiagent Systems*, pp. 381–420, MIT Press, 2013.
- [Sand 97] T. W. Sandholm and V. R. Lesser. “Coalitions Among Computationally Bounded Agents”. *Artif. Intell.*, Vol. 94, No. 1-2, pp. 99–137, 1997.
- [Sarw 00] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. “Analysis of Recommendation Algorithms for e-Commerce”. In : *Proceedings of the 2Nd ACM Conference on Electronic Commerce*, pp. 158–167, ACM, 2000.
- [Sarw 01] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. “Item-based Collaborative Filtering Recommendation Algorithms”. In : *Proceedings of the 10th International Conference on World Wide Web*, pp. 285–295, ACM, 2001.
- [Sear 69] J. R. Searle. *Speech Acts*. Cambridge University Press, 1969.
- [Seli 00] A. B. Seligman. *The Problem of Trust*. Princeton University Press, 2000.
- [Sell 09] M. Sellami, S. Tata, Z. Maamar, and B. Defude. “A Recommender System for Web Services Discovery in a Distributed Registry Environment.”. In : *ICIW*, pp. 418–423, IEEE Computer Society, 2009.
- [Sheh 98] O. Shehory and S. Kraus. “Methods for task allocation via agent coalition formation”. *Artif. Intell.*, Vol. 101, pp. 165–200, 1998.
- [Shet 04] J. Shetty and J. Adibi. “The Enron Email Dataset Database Schema and Brief Statistical Report”. 2004.

- [Sial 11] F. Siala, S. Lajmi, and K. Ghedira. “Multi-agent selection of multiple composite web services based on CBR method and driven by QoS”. In : *iiWAS*, pp. 90–97, ACM, 2011.
- [Sier 09] C. Sierra and J. Debenham. “Information-Based Reputation”. *First International Conference on Reputation : Theory and Technology*, pp. 5–19, 2009.
- [Sims 05] Ö. Simsek and D. Jensen. “Decentralized search in networks using homophily and degree disparity”. In : *Proceedings of the 19th international joint conference on Artificial intelligence*, pp. 304–310, Morgan Kaufmann Publishers Inc., 2005.
- [Sing 01] M. P. Singh, B. Yu, and M. Venkatraman. “Community-Based Service Location”. *Communications of the ACM*, Vol. 44, No. 4, pp. 49–54, 2001.
- [Sing 05] M. P. Singh and M. Huhns. *Service-oriented computing : semantics, processes, agents*. Wiley, 2005.
- [Sing 93] M. P. Singh. “A Semantics for Speech Acts.”. *Annals of Mathematics and Artificial Intelligence*, Vol. 8, pp. 47–71, 1993.
- [Sing 98] M. P. Singh. “Developing Formal Specifications to Coordinate Heterogeneous Autonomous Agents.”. In : *ICMAS*, pp. 261–268, IEEE Computer Society, 1998.
- [Sinh 01] R. R. Sinha and K. Swearingen. “Comparing Recommendations Made by Online Systems and Friends”. In : *DELOS Workshop : Personalisation and Recommender Systems in Digital Libraries*, 2001.
- [Siri 04] E. Sirin, B. Parsia, D. Wu, J. A. Hendler, and D. S. Nau. “HTN planning for Web Service composition using SHOP2.”. *J. Web Sem.*, Vol. 1, No. 4, pp. 377–396, 2004.
- [Sree 04] R. M. Sreenath and M. P. Singh. “Agent-based service selection”. *Web Semantics : Science, Services and Agents on the World Wide Web*, Vol. 1, No. 3, pp. 261 – 279, 2004.
- [Stan 94] W. Stanley and F. Katherine. *Social network analysis : Methods and applications*. Cambridge University Press, 1994.
- [Stei 00] F. Steimann. “On the representation of roles in object-oriented and conceptual modelling”. *Data Knowledge Engineering*, Vol. 35, No. 1, pp. 83–106, 2000.

- [Stoi 01] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. “Chord : A Scalable Peer-to-peer Lookup Service for Internet Applications”. In : *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pp. 149–160, ACM, 2001.
- [Sure 00] C. Surendar, E. Carla, Schlatter, and V. Amin. “Differentiated Multimedia Web Services Using Quality Aware Transcoding”. 2000.
- [Syca 01] K. P. Sycara. “Multi-agent Infrastructure, Agent Discovery, Middle Agents for Web Services and Interoperation.”. In : *EASSS*, pp. 17–49, Springer, 2001.
- [Syca 03] K. P. Sycara, M. Paolucci, A. Ankolekar, and N. Srinivasan. “Automated discovery, interaction and composition of Semantic Web Services”. *Journal of Web Semantics*, Vol. 1, pp. 27–46, 2003.
- [Szel 10] M. Szell, R. Lambiotte, and S. Thurner. “Multirelational Organization of Large-scale Social Networks in an Online World”. In : *Proceedings of the National Academy of Science*, pp. 13636–13641, 2010.
- [Tang 12] J. Tang, H. Gao, and H. Liu. “mTrust : discerning multi-faceted trust in a connected world.”. In : *WSDM*, pp. 93–102, ACM, 2012.
- [The 90] The Institute of Electrical and Eletronics Engineers. “IEEE Standard Glossary of Software Engineering Terminology”. IEEE Standard, 1990.
- [Tong 11] H. Tong, J. Cao, S. Zhang, and M. Li. “A Distributed Algorithm for Web Service Composition Based on Service Agent Model.”. *IEEE Trans. Parallel Distrib. Syst.*, Vol. 22, No. 12, pp. 2008–2021, 2011.
- [Vict 09] P. Victor, C. Cornelis, M. D. Cock, and P. P. da Silva. “Gradual trust and distrust in recommender systems.”. *Fuzzy Sets and Systems*, Vol. 160, No. 10, pp. 1367–1382, 2009.
- [Vu 05] L.-H. Vu, M. Hauswirth, and K. Aberer. “QoS-Based Service Selection and Ranking with Trust and Reputation Management”. In : *OTM*, pp. 466–483, Springer-Verlag, 2005.
- [Walt 08] F. E. Walter, S. Battiston, and F. Schweitzer. “A model of a trust-based recommendation system on a social network.”. *Autonomous Agents and Multi-Agent Systems*, Vol. 16, No. 1, pp. 57–74, 2008.

- [Wang 06a] S. Wang, W. Shen, and Q. Hao. "An agent-based Web service workflow model for inter-enterprise collaboration.". *Expert Syst. Appl.*, Vol. 31, No. 4, pp. 787–799, 2006.
- [Wang 06b] X. Wang, T. Vitvar, M. Kerrigan, and I. Toma. "A Qos-aware Selection Model for Semantic Web Services". In : *ICSOC*, pp. 390–401, Springer-Verlag, 2006.
- [Wang 06c] Y. Wang and M. P. Singh. "Trust Representation and Aggregation in a Distributed Agent System.". In : *AAAI*, pp. 1425–1430, AAAI Press, 2006.
- [Wang 07a] Y. Wang and V. Varadharajan. "Role-based Recommendation and Trust Evaluation.". In : *CEC/EEE*, pp. 278–288, IEEE Computer Society, 2007.
- [Wang 07b] Y. Wang and J. Vassileva. "Toward Trust and Reputation Based Web Service Selection : A Survey". *ITSSA Journal*, Vol. 3, pp. 118–132, 2007.
- [Wang 09] Y. Wang, K.-J. Lin, D. S. Wong, and V. Varadharajan. "Trust management towards service-oriented applications". *Service Oriented Computing and Applications*, Vol. 3, No. 2, pp. 129–146, 2009.
- [Wang 13] Y. Wang, L. Li, and G. Liu. "Social Context-Aware Trust Inference for Trust Enhancement in Social Network based Recommendations on Service Providers". *World Wide Web Journal*, pp. 1–26, 2013.
- [Wool 00] M. Wooldridge, N. R. Jennings, and D. Kinny. "The Gaia Methodology for Agent-Oriented Analysis and Design.". *Autonomous Agents and Multi-Agent Systems*, Vol. 3, pp. 285–312, 2000.
- [Xiao 10a] G. Xiao and J. Kertész. "Editorial : Beyond Small-World and Scale-Free Networks.". *Advances in Complex Systems*, Vol. 13, No. 1, pp. 1–2, 2010.
- [Xiao 10b] H. Xiao, Y. Zou, J. Ng, and L. Nigul. "An Approach for Context-Aware Service Discovery and Recommendation". In : *ICWS*, pp. 163–170, IEEE Computer Society, 2010.
- [Xie 08] X. Xie, B. Du, and Z. Zhang. "Semantic service composition based on social network". In : *Proceedings of the 17th International World Wide Web Conference (WWW'2008)*, pp. 149–160, 2008.
- [Xu 07] Z. Xu, P. Martin, W. Powley, and F. H. Zulkernine. "Reputation-Enhanced QoS-based Web Services Discovery.". In : *ICWS*, pp. 249–256, IEEE Computer Society, 2007.

- [Xu 11] J. Xu, K. Chen, and S. Reiff-Marganiec. "Using Markov Decision Process Model with Logic Scoring of Preference Model to Optimize HTN Web Services Composition.". *Int. J. Web Service Res.*, Vol. 8, No. 2, pp. 53–73, 2011.
- [Yan 08] W. Yan and L. Kwei-Jay. "Reputation-Oriented Trustworthy Computing in E-Commerce Environments". *IEEE Internet Computing*, Vol. 12, No. 4, pp. 55–59, 2008.
- [Yolu 03] P. Yolum and M. P. Singh. "Emergent properties of referral systems.". In : *AAMAS*, pp. 592–599, ACM, 2003.
- [Yolu 05] P. Yolum and M. P. Singh. "Engineering self-organizing referral networks for trustworthy service selection.". *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, Vol. 35, No. 3, pp. 396–407, 2005.
- [Yu 00] B. Yu and M. P. Singh. "A Social Mechanism of Reputation Management in Electronic Communities". In : *Proceedings of the 4th International Workshop on CIA*, pp. 154–165, Springer-Verlag, 2000.
- [Yu 03] B. Yu and M. P. Singh. "Searching social networks". In : *AAMAS*, pp. 65–72, 2003.
- [Yu 06] T. Yu and K.-J. Lin. "QCWS : An Implementation of QoS-capable Multimedia Web Services". *Multimedia Tools Appl.*, Vol. 30, No. 2, pp. 165–187, 2006.
- [Yu 99] B. Yu, M. Venkatraman, and M. P. Singh. "A Multiagent Referral System for Expertise Location". In : *In working notes of the AAAI workshop on intelligent information systems*, pp. 66–69, 1999.
- [Zach 05] G. Zacharia. *Collaborative Reputation Mechanisms for Online Communities*. PhD thesis, Massachusetts Institute of Technology, School of Architecture and Planning, 2005.
- [Zarg 09] A. Zarghami, S. Fazeli, N. Dokoochaki, and M. Matskin. "Social Trust-Aware Recommendation System : A T-Index Approach.". In : *Web Intelligence/IAT Workshops*, pp. 85–90, IEEE, 2009.
- [Zhan 05] J. Zhang and M. S. Ackerman. "Searching for expertise in social networks : a simulation of potential strategies". In : *Proc. ACM SIGGROUP Conf. on Supporting Group Work*, pp. 71–80, ACM Press, 2005.

- [Zhan 07] J. Zhang, M. S. Ackerman, and L. Adamic. "Expertise networks in online communities : structure and algorithms". In : *Proceedings of the 16th international conference on World Wide Web*, pp. 221–230, ACM Press, 2007.
- [Zhan 10] Y. Zhang, Z. Zheng, and M. R. Lyu. "WSExpress : A QoS-aware Search Engine for Web Services". In : *Proc. IEEE Int'l Conf. Web Services*, pp. 83–90, 2010.
- [Zhen 09] Z. Zheng, H. Ma, M. R. Lyu, and I. King. "WSRec : A Collaborative Filtering Based Web Service Recommender System.". In : *ICWS*, pp. 437–444, IEEE, 2009.
- [Zhen 14] X. Zheng, Y. Wang, M. Orgun, Y. Zhong, and G. Liu. "Trust Prediction with Propagation and Similarity Regularization". In : *AAAI Conference on Artificial Intelligence*, pp. 237–244, 2014.
- [Zhu 14] Y. Zhu, M. Halpern, and V. J. Reddi. "Event-based scheduling for energy-efficient QoS (eQoS) in mobile Web applications". In : *High Performance Computer Architecture (HPCA), 2015 IEEE 21st International Symposium on*, 2014.
- [Zieg 04] C. Ziegler and G. Lausen. "Analyzing Correlation between Trust and User Similarity in Online Communities". In : *Proceedings of Second International Conference on Trust Management*, pp. 251–265, Springer-Verlag, 2004.
- [Zieg 05] C.-N. Ziegler and G. Lausen. "Propagation Models for Trust and Distrust in Social Networks". *Information Systems Frontiers*, Vol. 7, No. 4-5, pp. 337–358, 2005.
- [Zieg 07] C. Ziegler and J. Golbeck. "Investigating Correlations of Trust and Interest Similarity - Do Birds of a Feather Really Flock Together?". *Decision Support Systems*, Vol. 43, pp. 460–475, 2007.

