

Université d'Artois

Approche multi-agents d'un système d'aide à la décision en environnement dynamique et incertain Application à l'entomologie médico-légale

THÈSE

présentée et soutenue publiquement le 25 novembre 2009

en vue de l'obtention du

Doctorat de l'Université d'Artois
(Spécialité : Génie Informatique et Automatique)

par

Gildas Morvan

Composition du jury

<i>Rapporteurs :</i>	Jean-Louis Deneubourg Jean-Pierre Müller	Professeur, Université Libre de Bruxelles Directeur de Recherche, CIRAD - LIRMM, Montpellier
<i>Examineurs :</i>	Daniel Dupont Didier Gosset David Hill Philippe Kubiak Eric Lefèvre	Professeur, HEI, Lille Professeur, Université de Lille 2 Professeur, Université Blaise Pascal, Clermont-Ferrand Maître de Conférences, École Centrale de Lille Maître de Conférences, Université d'Artois
<i>Directeur de thèse :</i>	Daniel Jolly	Professeur, Université d'Artois

Mis en page avec la classe thloria.

Remerciements

Je tiens à remercier chaleureusement le sympathique et compétent trio qui m’a encadré et soutenu durant ces quatre dernières années (il a également du me supporter durant mon stage de master recherche) : Daniel Jolly, professeur à l’Université d’Artois et directeur du LGI2A, Daniel Dupont, enseignant chercheur à HEI et responsable du pôle Ingénierie et Sciences du Vivant, et Philippe Kubiak, maître de conférences au LAGIS (École Centrale de Lille). Ils ont, chacun à leur manière, permis à ce projet d’aboutir.

Mes respects les plus profonds vont aux membres du jury qui ont accepté de prendre de leur temps pour juger ce travail.

Je souhaite adresser mes remerciements les plus amicaux aux doctorants avec qui j’ai collaboré et partagé des moments si particuliers dans des lieux inhospitaliers et incongrus¹ : Damien Charabidze, aujourd’hui chercheur post-doctoral à l’Institut de Médecine Légale de Lille et Alexandre Veremme, doctorant au LGI2A.

Je remercie de plus tous ceux qui, à un moment ou un autre, m’ont aidé d’une manière ou d’une autre : Gilles Goncalves, Éric Lefèvre, David Mercier, Guillem Lefait et Saïd El Hmam Mohamed du LGI2A, Benoît Bourel et Cédric Devigne du Laboratoire Environnement et Santé de la Faculté Libre des Sciences et Technologies de Lille, Jean-Louis Deneubourg de l’Unité d’Écologie Sociale de l’Université Libre de Bruxelles, Cédric Houssaye de l’Institut de Médecine Légale de Lille ainsi que tous les employés de la Verminière de Saint Pourçain-sur-Sioule.

Je remercie également les membres du département informatique de l’IUT de Lens grâce à qui j’ai pu découvrir les joies de l’enseignement : Gilles Audemard, Frédéric Bousse-mard, Valérie Capelle, Sylvie Coste-Marquis, Giovanna Giarrusso, Souhila Kaci, Christophe Lecoutre, Olivier Roussel et Vincent Vidal.

Je remercie enfin ma famille et mes amis qui m’ont soutenu et encouragé : mon père, ma mère, mes frères et mes sœurs, les « nantais » : Florian, Florence, Thomas, Nicolas, Matthieu, Aurélie et Guillaume et les « lillois » : Lucie, Bec, Frédéric, Virginie, Guillaume, François, Kristen, Régis, Florent, Céline, Maxime, Simon, La Buse, Faustine et Mariama.

¹Pour plus de précisions sur la nature de ces « moments » et de ces « lieux », je renvoie le lecteur au chapitre 6

Je dédie cette thèse à Luna

Table des matières

Table des figures	xi
Liste des tableaux	xiii
Résumé	xv
Abstract	xvii
Introduction	xix

Partie I État des connaissances I

Chapitre 1 Modélisation et simulation numérique de systèmes complexes	3
1.1 Définitions fondamentales	4
1.1.1 Système	4
1.1.1.1 Une première définition	4
1.1.1.2 Le cas des systèmes complexes	5
1.1.2 Modèle	6
1.1.2.1 Qu'est-ce qu'un modèle?	6
1.1.2.2 Pourquoi modéliser?	6
1.2 Modélisation et simulation numérique	7
1.2.1 Clés de classification d'un modèle	7
1.2.1.1 Méta-modèle	7
1.2.1.2 Echelle de modélisation	8
1.2.1.3 Déterminisme	8
1.2.1.4 Représentation du temps	9
1.2.2 Méthodologies de simulation numérique	9

1.2.2.1	Une première définition	9
1.2.2.2	Une définition plus complète	10
1.3	Simulation Orientée Agent	11
1.3.1	Notions fondamentales	13
1.3.1.1	L'agent	13
1.3.1.2	L'environnement	14
1.3.1.3	Communication entre agents	15
1.3.1.4	Le système multi-agent	16
1.3.2	Une méthodologie pour la SOA	16
1.3.3	Plates-formes de simulation orientée agent	17
1.3.3.1	MadKit	19
1.3.3.2	TurtleKit	19
1.3.4	Choix de la simulation orientée agent	20
1.4	Simulation par automates cellulaires	20
1.4.1	Définition	21
1.4.2	Méthodologie	21
1.4.3	Application	21
1.5	Résumé du chapitre	22
Chapitre 2 Problème inverse et raisonnement abductif		23
2.1	La notion de problème inverse	24
2.1.1	Définition formelle	24
2.1.2	Résolution	25
2.2	Le raisonnement abductif	25
2.2.1	Les différents types de raisonnement	26
2.2.1.1	Les trois modes de raisonnement	26
2.2.1.2	Le raisonnement temporel	27
2.2.1.3	Les différents objectifs du raisonnement	27
2.2.2	Cas d'étude : l'abduction	28
2.2.2.1	Deux points de vue sur l'abduction	29
	L'abduction comme produit	29
	L'abduction comme processus	30
2.2.3	L'abduction en Philosophie	31
2.2.3.1	La notion de modèle causal	31

2.2.3.2	Une approche classique : l'abduction comme « déduction inverse »	31
2.2.3.3	L'inférence contrastive	32
2.2.3.4	De l'importance des stratégies	32
2.2.4	L'abduction en Intelligence Artificielle	33
2.2.4.1	L'abduction comme un problème de « recouvrement d'ensembles »	33
2.2.4.2	L'abduction comme inférence logique	34
2.3	Relations entre abduction et problème inverse	34
2.4	Résumé du chapitre	35

Partie II Une architecture de système d'aide à la décision basé sur les notions de problème inverse et de simulation orientée agent 37

Chapitre 3	Architecture générale — Problématiques	39
3.1	Introduction	40
3.2	Validation de simulations	41
3.2.1	Mesure de la validité d'une simulation	42
3.2.2	Mesure de la validité d'un ensemble de paramètres	42
3.3	Observation de simulations orientées agent	42
3.3.1	Problématique générale	42
3.3.2	Description du cas d'étude	44
3.3.3	Filtrage des informations	44
3.3.3.1	Auto-observation des agents	46
3.3.3.2	Échantillonnage de la population	46
3.3.3.3	Bilan	48
3.4	Résumé du chapitre	50

Chapitre 4	Modélisation et résolution de problèmes inverses basées sur des critères d'abduction	51
4.1	Description du modèle	52
4.1.1	Définitions	52
4.1.2	Objectifs de la résolution	53
4.1.3	Algorithme de résolution	53

4.2	Heuristiques	55
4.2.1	Critères de choix	55
4.2.2	Définition des heuristiques	57
4.3	Résultats expérimentaux	57
4.4	Résumé du chapitre	61

Partie III Application à l'entomologie médico-légale 63

Chapitre 5 Introduction à l'entomologie médico-légale 65

5.1	Principes généraux	66
5.1.1	Objectifs des expertises entomologiques	66
5.1.2	Colonisation d'un cadavre par les diptères nécrophages	66
5.2	Modélisation du développement des diptères nécrophages	68
5.2.1	La température, paramètre fondamental	68
5.2.2	Quelques modèles de développement	69
5.2.2.1	Le modèle des degrés jours (heures) accumulés	69
5.2.2.2	Le modèle de Stinner	70
5.2.3	Variabilité de la vitesse de développement au sein d'une population	70
5.2.3.1	Les Time Distributed Delays	70
5.2.3.2	La méthode de Régnière	71
5.3	Calcul d'un intervalle post-mortem à l'aide de modèles de développement	72
5.3.1	Approche formelle	72
5.3.2	Problématique principale : l'estimation de la température	72
5.3.2.1	L'existence de micro-climats	72
5.3.2.2	L'inertie thermique du corps	73
5.3.2.3	L'effet des masses de larves	73
5.4	Résumé du chapitre	74

Chapitre 6 Modélisation et simulation orientées agent du développement des Diptères nécrophages 75

6.1	Introduction	76
6.1.1	Description du système étudié	76
6.1.1.1	L'environnement	76
6.1.1.2	Les diptères femelles gravides	76

6.1.1.3	Les larves de diptères	77
6.1.2	Objectifs de la modélisation	78
	Objectifs généraux	78
	Modélisation de l’environnement	80
	Modélisation des larves de diptères	80
6.1.3	Choix du méta-modèle	80
	La complexité du système étudié	80
	La complexité des comportements collectifs	80
	L’hétérogénéité des modèles existants	80
	L’échec de l’approche macroscopique	81
6.2	Description du modèle conceptuel	81
6.2.1	L’environnement	81
6.2.1.1	Caractéristiques générales	81
6.2.1.2	Règles Structurelles	81
6.2.1.3	Règles Fonctionnelles	83
6.2.2	Populations et comportements de ponte des espèces	84
6.2.3	Larves de diptères	85
6.2.3.1	Modèles d’évolution	85
	Stade et développement	85
	Vitesse de déplacement	85
	Longueur	86
	Mortalité	86
	Émission de chaleur	86
6.2.3.2	Modèle comportemental	87
6.3	Implémentation	89
6.3.1	Couplage Modèles/Agents	89
6.3.2	Agents informatiques	89
6.3.2.1	L’agent <i>Layer</i>	89
6.3.2.2	L’agent <i>Larva</i>	90
6.4	Validation du modèle et résultats de simulation	91
6.4.1	Validation des modèles de développement et de variabilité	91
6.4.2	Validation du modèle thermique du corps	91
6.4.3	Calibrage et validation du modèle comportemental	93
6.4.3.1	Calibrage qualitatif	94

6.4.3.2	calibrage quantitatif	95
6.5	Résumé du chapitre	96
Chapitre 7 Entomologie médico-légale et aide à la décision		99
7.1	Architecture du système d'aide à la décision	100
7.1.1	Stockage des données et construction des simulations	100
7.1.2	Résolution de problèmes inverses	101
7.1.2.1	Implémentation objet de l'algorithme de résolution . . .	101
7.1.2.2	Implémentation agent de l'algorithme de résolution . . .	101
7.2	Règles de décision	104
7.2.1	Taux de développement accumulé	106
7.2.2	Stade de développement	107
7.2.3	Espèces retrouvées sur le corps	108
7.3	Résultats	108
7.3.1	Premier exemple	108
7.3.2	Second exemple	109
7.4	Résumé du chapitre	109
Conclusion et perspectives		113
Bibliographie		117

Table des figures

1	Plan de la thèse	xxii
1.1	Cadre méthodologique de simulations numériques (Fichwick, 1997)	10
1.2	Cadre méthodologique de simulations numériques	12
1.3	Cadre méthodologique de conception de simulation orientée agent (Drogoul <i>et al.</i> , 2003)	18
1.4	Les trois concepts fondamentaux du modèle AGR (Gutknecht et Ferber, 1999)	19
3.1	Architecture générale du système d'aide à la décision.	40
3.2	Temps CPU moyen utilisé pour le calcul de différentes simulations avec et sans observation pour $E(Z) = N/2$	45
3.3	Différence entre les temps CPU moyens (s) de simulations observées à l'aide des méthodes basées sur l'auto-observation des agents et naïve	47
3.4	Taille de l'échantillon à observer, n , pour $d = 0.08$ et $E(Z) = N/2$	48
3.5	Différence entre les temps CPU moyens (s) de simulations observées à l'aide des méthodes basées sur l'échantillonnage de la population d'agents (avec $d =$ 0.08) et naïve	49
3.6	Conditions dans lesquelles les méthodes d'observations sont les plus efficaces ; A : auto-observation, B : naïve, C : échantillonnage	49
4.1	Valeur optimale de α_ϕ en fonction du coût maximal alloué pour calculer le résultat.	58
4.2	Exemples d'ensembles solutions générés pour évaluer les heuristiques.	59
4.3	Analyse de la performance des heuristiques pour l'objectif <i>id.</i>	59
4.4	Analyse de la performance des heuristiques pour l'objectif <i>el.</i>	60
4.5	Analyse de la performance des heuristiques pour l'objectif <i>prs.</i>	60
5.1	Cycle de développement holométabole des diptères Calliphoridae.	67
5.2	Diagramme isomorphe de <i>Lucilia sericata</i> , issu de Grassberger et Reiter (2001)	69
5.3	Forme générale du refroidissement d'un cadavre à température constante (20°C). 1 : plateau thermique initial / 2 : décroissance linéaire / 3 : phase « exponen- tielle »	73
6.1	Différents groupes de larves identifiés dans un agrégat (Charabidze, 2008, p. 145)	78

6.2	Oscillations de la température relevée au sein d'une masse de 200g de larves de second stade de <i>P. terraenovae</i> placée à 30°C en présence de nourriture . . .	79
6.3	Températures relevées dans des masses (500g) de larves avec et sans nourriture	79
6.4	Classes des cellules de l'automate	82
6.5	Taux de survie de larves de diptères en fonction de la densité d'individus	87
6.6	Modèle comportemental. b_1 : comportement normal, b_2 : comportement de panique	87
6.7	Modèle comportemental d'une larve en conditions normales. $b_{1,1}$: alimentation, $b_{1,2}$: optimisation de la température, $b_{1,3}$: recherche de nourriture	88
6.8	Variabilité intraspécifique de la vitesse de développement observée et prédite pour l'espèce <i>Lucilia sericata</i> à 15 et 35°C	91
6.9	Différence maximale entre les prédictions de Henssge et des SAC	92
6.10	Évolution de la température rectale (observation et prédiction) d'un porc de 30 kg sous températures variables et erreur absolue des prédictions	92
6.11	Différence entre l'IPM réel et calculé à l'aide du modèle de Henssge lorsque la température de l'environnement varie (amplitude 10°C, période 24 h)	93
6.12	Erreur quadratique moyenne (MSE) en fonction de λ et T_{max}	96
6.13	Valeur absolue de l'erreur moyenne ($ ME $) en fonction de λ et T_{max}	97
6.14	Erreur systématique associée à T_{ext} pour $\lambda = 0.0003$ et $T_{max} = 40$	97
6.15	Erreur systématique associée à r pour $\lambda = 0.0003$ et $T_{max} = 40$	98
7.1	Composants logiciels utilisés pour l'implémentation du SIAD ; le nom des composants tiers est donné entre parenthèses	100
7.2	Diagramme de classes représentant les notions d'hypothèse et d'ensemble solution	102
7.3	Diagramme de classes associé à l'implémentation objet de l'algorithme de résolution	103
7.4	Interactions entre les différents types d'agents et les ensembles manipulés durant la résolution du problème	105
7.5	Interface de visualisation de l'état d'avancement de la résolution d'un problème	106
7.6	Températures utilisées pour le premier exemple	110
7.7	Résultats de l'expertise (premier exemple) pour les différents modèles ADD utilisés	110
7.8	Résultats de l'expertise (premier exemple) pour les différents modèles de Stinner utilisés	111
7.9	Distribution des probabilités pignistiques sur l'ensemble des heures possibles du décès pour le premier exemple	111
7.10	Températures utilisées pour le second exemple	112
7.11	Résultats de l'expertise (second exemple) pour les deux modèles utilisés	112

Liste des tableaux

1.1	Les différents types d'agents (Ferber, 1995)	14
1.2	Modes de communication (Ferber, 1995)	16
1.3	Axes de définition d'un automate cellulaire	21
2.1	Les différents modes de raisonnement dans le modèle peircien	27
2.2	Types de raisonnements temporels associés aux modes de raisonnement du modèle peircien	28
4.1	Objectifs d'un problème inverse	54
4.2	Métriques de qualité associées aux objectifs d'un problème inverse	54
5.1	Les quatre premières escouades du modèle de succession selon Smith. Le modèle entier comporte huit escouades (Smith, 1986).	67
6.1	Succession des stades durant le développement des larves de diptères.	77
6.2	Variables d'environnement définies dans le modèle	81
6.3	Propriétés thermiques des tissus (Werner et Buse, 1988) : conductivité k , chaleur massique c , densité ρ	82
6.4	Probabilité de décès horaire d'une larve de diptère en fonction du stade (Byrd et Allen, 2000)	86
6.5	Caractéristiques des sources de chaleur	94
6.6	Paramètres du modèle identifiés par calibrage qualitatif	95

Résumé

Les travaux présentés dans cette thèse portent sur le thème de la modélisation et de l'aide à la décision dans le cadre des systèmes complexes. Plus précisément, nous nous intéressons au couplage de simulations orientées agent à une méthode de résolution de problèmes inverses. Une architecture générique de système d'aide à la décision est ainsi proposée. Nous identifions deux problématiques associées à la mise en œuvre de cette architecture : la validation de paramètres et l'observation de simulations orientées agents, et proposons des éléments de réponse. La notion de problème inverse est elle-même redéfinie dans une perspective d'aide à la décision et d'utilisation dans un cadre multi-modèles. Une méthode de résolution et des heuristiques, basées sur des idées issues de travaux sur le raisonnement abductif, sont ainsi proposées.

Ces travaux sont appliqués à la conception d'un système d'aide à la décision dédié à l'entomologie médico-légale. Cette discipline à mi-chemin entre la criminalistique et l'entomologie vise à concevoir des méthodes de datation de la mort basées sur des indices entomologiques, *i.e.*, les insectes (principalement des diptères nécrophages) ou restes d'insectes retrouvés sur ou à proximité du corps. Ainsi, dans le cadre de ce travail réalisé en collaboration avec le laboratoire d'entomologie médico-légale de l'Institut de Médecine Légale de Lille, nous présentons :

- un modèle à base d'agents permettant de simuler le développement des diptères nécrophages dans un environnement complexe (*e.g.*, un corps situé dans un écosystème donné), en prenant en compte plus de paramètres que les méthodes classiques,
- une implémentation de l'architecture de système d'aide à la décision évoquée plus haut, basée sur ce modèle.

Abstract

The work presented in this thesis deals with modelling and decision support issues in the context of complex systems. Specifically, we investigate the coupling of agent-based simulations and inverse problem resolution methods. Thus, a generic architecture of decision support system is proposed. We identify two problems related to the implementation of this architecture : the validation of simulation parameters and the observation of agent-based simulations. We offer some answers to solve them. The notion of inverse problem is redefined in the context of decision support in a multi-model framework. Therefore, a resolution method and heuristics, based on ideas from works on abductive reasoning, are proposed.

This work is then applied to the conception of a decision support system dedicated to forensic entomology. This research area, midway between forensic sciences and entomology, aims to develop post-mortem estimation methods based on entomological indices, *i.e.*, insects (mainly necrophagous diptera) or insect remains, sampled on or around the cadaver. In the context of this work, done in collaboration with forensic entomology laboratory of the Legal Medicine Institute of Lille, we present :

- an agent-based model of necrophagous diptera in a complex environment such as a dead body in a specific ecosystem, taking into account more parameters than conventional methods,
- an implementation of the decision support system architecture mentioned above, based on this model.

Introduction

La complexité toujours croissante des systèmes artificiels et l'importance cruciale de mieux comprendre et maîtriser la complexité des systèmes naturels, imposent le développement de nouvelles méthodes de modélisation et de résolution de problèmes (Lempert, 2002). Les crises écologiques et économiques que nous traversons ne font que renforcer ce constat. Les travaux réalisés durant ces trois années de préparation au Doctorat portent ainsi sur le thème de l'aide à la décision dans le cadre des systèmes complexes. Dans ce champ d'étude vaste et ouvert, nous nous sommes intéressés à la conception de systèmes d'aide à la décision dans lesquels les connaissances de l'expert sont utilisées pour construire et analyser des simulations orientées agent (SOA) et la question à laquelle il souhaite répondre peut être modélisée sous la forme d'un problème inverse². La mise en œuvre d'un tel système pose un certain nombre de problèmes de conception, liés notamment à l'efficacité du processus de calcul de la solution, auxquels nous apporterons des éléments de solution tout au long de cette thèse. Mais tout d'abord, revenons sur certaines notions évoquées plus haut afin de comprendre les enjeux de ce travail.

Multiplicité des points de vue

De nombreux systèmes complexes (écosystèmes, sociosystèmes, etc.) sont caractérisés par des dynamiques non linéaires, chaotiques, stochastiques, en un mot compliquées, dans le sens où leur compréhension nécessite du temps et du talent. Les outils mathématiques classiques (équations différentielles, modèles probabilistes, etc.) peuvent ainsi se révéler inadaptés pour modéliser de tels systèmes dans lesquels l'interaction joue un rôle fondamental. Pour s'en convaincre, considérons le cas des modèles économétriques statistiques utilisés par les financiers du monde entier pour mesurer le risque associé à un placement, un prêt, etc. Ceux-ci, pourtant basés sur des théories dont le développement a valu l'attribution de prix Nobel, n'ont pas permis de prédire la crise financière amorcée en 2008 notamment parce qu'ils négligeaient les comportements des différents agents économiques (emprunteurs, banques, marchés) et les relations entre ceux-ci. Les causes de cette crise ne sont sans doute pas seulement liées à la « mutilation de la réalité » économique et sociale par des modèles mathématiques, mais la confiance aveugle dans ces derniers y est certainement pour beaucoup (Armatte, 2009).

Pourtant depuis quelques décennies, de nouvelles approches émergent, apportant un éclairage différent sur la modélisation et la simulation des systèmes complexes. En effet, il ne s'agit

²Il s'agit de déterminer l'ensemble des paramètres d'un modèle permettant d'obtenir un résultat donné. Cette notion est définie plus formellement dans le chapitre 2.

plus (seulement) de caractériser la dynamique du système sous forme mathématique, mais plutôt de la générer à partir de modèles des composants du système et de leurs interactions. Sans aller jusqu'à dire que ces approches permettront un jour d'anticiper et de résoudre les crises financières, il est intéressant de constater qu'elles se sont développées sous diverses formes dans de très nombreux domaines³, e.g., la méthode des éléments discrets en physique des milieux granulaires. Dans cette thèse, nous nous intéresserons essentiellement à la SOA qui semble la plus générale de ces méthodes de modélisation que nous qualifierons de microscopiques (Treuil *et al.*, 2008) et à la simulation par automates cellulaires (SAC).

Faut-il donc oublier les méthodes de modélisation mathématiques classiques? La réponse est bien évidemment non. Il existe de nombreuses applications où elles demeurent pertinentes. De plus, l'association de ces différentes approches semble être une voie de recherche prometteuse et pourrait permettre de résoudre certains problèmes de modélisation (Morvan *et al.*, 2009b).

Le raisonnement face à la complexité

Si modéliser des systèmes complexes est en soit un art noble et difficile (nous en avons fait l'expérience durant ces trois dernières années), résoudre des problèmes à l'aide de tels modèles, tout du moins efficacement, l'est tout autant. Les SOA sont en effet constituées de nombreux éléments (les agents) autonomes (au niveau conceptuel tout du moins) interagissant dans un environnement souvent dynamique. L'analyse d'un tel système artificiel pose un certain nombre de problèmes dont :

- l'agrégation des états des agents pour calculer efficacement l'état du système dans sa globalité,
- la détermination de la validité d'une simulation ou plus généralement d'un ensemble de paramètres d'une simulation à partir d'informations fournies par un expert.

Par ailleurs, l'exécution d'une SOA composée de nombreux agents nécessite un temps de calcul important. Cela peut être un problème en soit mais cela le devient d'avantage lorsque les simulations sont utilisées pour résoudre des problèmes, *i.e.*, s'il est nécessaire d'exécuter un grand nombre de simulations. Les méthodes de résolution de problèmes, et en particulier de problèmes inverses, considèrent généralement que le temps associé à la validation d'une hypothèse, *i.e.*, une solution possible du problème, est négligeable par rapport au temps disponible pour l'exploration de l'espace de recherche, *i.e.*, l'ensemble des solutions possibles du problème. Dans les cas des SOA, cette contrainte n'est pas toujours vérifiée. De plus, la résolution de tels problèmes est assimilée généralement à un processus d'optimisation (recherche de la meilleure solution). Dans un contexte d'aide à la décision il est intéressant d'étendre cette définition : on peut par exemple rechercher l'ensemble des solutions respectant certains critères ou encore éliminer les solutions fausses.

Dans ce contexte, nous pensons que la notion de problème inverse peut être rapprochée d'une forme de raisonnement particulière : l'abduction. Ce raisonnement a été particulièrement étudié en philosophie et en intelligence artificielle. Dans ce dernier domaine d'étude,

³Il serait aujourd'hui sans doute plus facile d'identifier les domaines qui n'ont pas recours à ces techniques que l'inverse.

l'abduction est abordée sous l'angle de la logique ou de probabilités. En revanche, la philosophie propose un ensemble de solutions, certes souvent moins formalisées, mais bien plus large (Lipton, 2004). Nous nous inspirons ainsi d'un modèle de l'abduction comme processus heuristique de collecte d'information et de validation d'hypothèses pour développer une méthode de résolution de problèmes inverses.

Application au vivant

Ces travaux ont été menés principalement dans un cadre applicatif lié à la biologie des organismes et à la criminalistique : l'entomologie médico-légale. Après un décès, dans des conditions climatiques favorables et si le cadavre est accessible, celui-ci va être colonisé par des insectes dont, en premier lieu, des diptères nécrophages. L'entomologie médico-légale cherche à déterminer la date de la mort de la victime en corrélant des informations relatives à la présence et au développement de ces diptères à des données climatiques. Les méthodes utilisées aujourd'hui, basées sur des modèles du développement des diptères nécrophages, ne considèrent pas la capacité des larves de diptères à modifier leur environnement, en régulant la température locale, pour optimiser leur développement. En d'autres termes, les caractéristiques comportementales des composants du système et les interactions entre ceux-ci sont négligées bien que de nombreuses publications, *e.g.*, Ireland et Turner (2006) et Kaneshrajah et Turner (2004), aient souligné leur importance dans l'estimation d'intervalles post-mortem. Nous avons ainsi conçu, en collaboration avec l'Institut de Médecine Légale de Lille, un modèle à base d'agents de la colonisation et du développement des diptères nécrophages dans un environnement complexe comme un cadavre humain. Ce modèle est utilisé comme base d'un outil d'aide à la décision dont l'objectif est de pouvoir déterminer plus précisément, et sur des bases formelles, des intervalles post-mortem.

Le développement d'un tel modèle a posé (et continue de poser) un certain nombre de problèmes à la fois pratiques et techniques. Ainsi, le modèle a considérablement évolué depuis le début de ce travail⁴ et, nous l'espérons, continuera d'évoluer à la lumière des nouvelles connaissances apportées au domaine et des résultats de simulation.

Plan de la thèse

Cette thèse est divisée en trois parties : la première présente un état des connaissances, tout d'abord sur la simulation numérique de systèmes complexes (chapitre 1), puis sur les notions de problème inverse et d'abduction (chapitre 2). Dans la seconde partie, nous proposons une architecture de système d'aide à la décision ainsi que des éléments de solution au problème de l'efficacité du processus de calcul (chapitre 3) ; dans ce but, une méthode de résolution de problèmes inverses est introduite (chapitre 4). Enfin, la troisième partie est dédiée à l'application à l'entomologie médico-légale. Les bases théoriques de ce domaine sont introduites dans le chapitre 5. Nous présentons le modèle et le système d'aide à la décision respectivement dans les chapitres 6 et 7. Ce plan est résumé dans la figure 1.

⁴La lecture comparative des différentes publications que nous avons écrites sur le sujet : Charabidze *et al.* (2007, 2008c,b); Morvan (2006); Morvan *et al.* (2007, 2008b), le montre tout particulièrement.

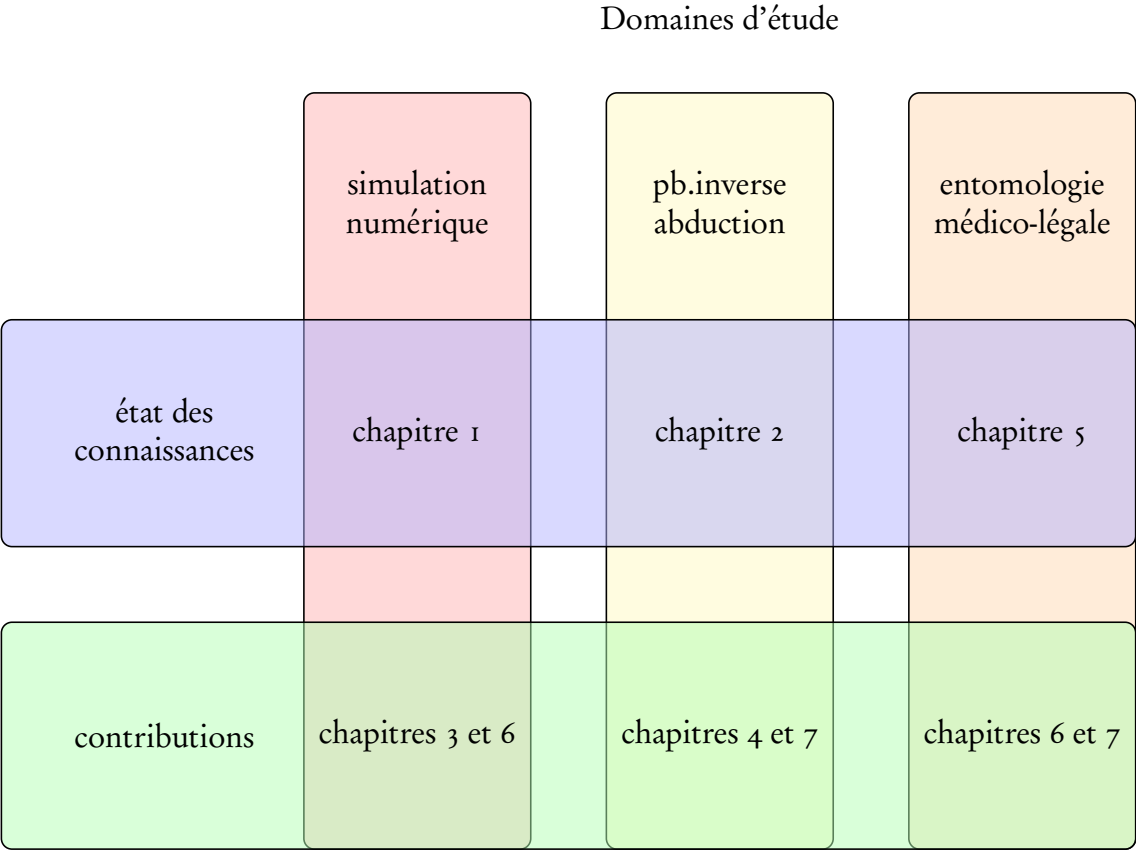


Fig. 1 – Plan de la thèse

Première partie

État des connaissances

I

Modélisation et simulation numérique de systèmes complexes

Sommaire

I.1	Définitions fondamentales	4
I.1.1	Système	4
I.1.2	Modèle	6
I.2	Modélisation et simulation numérique	7
I.2.1	Clés de classification d'un modèle	7
I.2.2	Méthodologies de simulation numérique	9
I.3	Simulation Orientée Agent	11
I.3.1	Notions fondamentales	13
I.3.2	Une méthodologie pour la SOA	16
I.3.3	Plates-formes de simulation orientée agent	17
I.3.4	Choix de la simulation orientée agent	20
I.4	Simulation par automates cellulaires	20
I.4.1	Définition	21
I.4.2	Méthodologie	21
I.4.3	Application	21
I.5	Résumé du chapitre	22

Dans ce chapitre, nous développons un certain nombre de notions relatives à la simulation numérique et plus spécifiquement à des méthodes de simulation microscopique : la simulation orientée agent (SOA) et la simulation par automates cellulaires (SAC). Mais dans un premier temps, un certain nombre de concepts, certains généraux (modèle, système) et particuliers (agent, système multi-agent) sont introduits.

1.1 Définitions fondamentales

1.1.1 Système

1.1.1.1 Une première définition

Un système peut être défini de manière minimale comme une « unité globale organisée d'interrelations entre éléments, actions, ou individus » (Morin, 1977, p. 102). La notion de système est ainsi utilisée pour désigner un vaste ensemble d'objets réels, *e.g.*, le système solaire, ou virtuels, *e.g.*, un système d'informations.

Il est possible de distinguer différents « types » de systèmes selon les relations que ces derniers entretiennent avec leur environnement. On parle ainsi de *système fermé* lorsque l'on considère que le système n'échange rien avec son environnement. *A contrario*, on parle de *système ouvert* pour désigner un système qui échange librement information, matière et énergie avec son environnement. La notion d'écosystème telle qu'elle est définie en écologie⁵ est un bon exemple de ce type de systèmes. Enfin, on parle de *système semi-ouvert* pour désigner les systèmes pouvant échanger information, matière et énergie avec son environnement, ces échanges étant contraints par la nature de la *membrane* isolant partiellement le système de son environnement. En d'autres termes, la membrane filtre les échanges en limitant les flux entrant et sortant du système. La cellule eucaryote est un exemple typique de tels systèmes. Notons que dans le cas des systèmes ouverts et semi-ouverts, les échanges entre le système et son environnement sont souvent nécessaires à l'existence même du système : il est ainsi problématique de définir de tels systèmes indépendamment de leur environnement (Morin, 1977, 1980; Varela, 1989). Nous emploierons le terme de contexte écosystémique pour désigner les variables d'état du système englobant le système étudié, influant sur l'état interne de ce dernier.

Bien que la majorité des systèmes soient ouverts, l'être humain en général et le scientifique en particulier ne peuvent analyser ces derniers qu'en isolant, tout du moins partiellement, le problème de son environnement⁶. Ainsi, pour faciliter le travail d'analyse, un système ouvert sera souvent considéré, dans un laps de temps donné, comme totalement isolé (fermé) ou, dans le meilleur des cas, partiellement isolé (semi-ouvert) : les échanges entre le système et son environnement seront limités à ceux qui semblent pertinents pour répondre au problème posé. Cependant, définir les frontières ou limites d'un système peut être difficile lorsqu'il n'existe pas de barrière ou membrane (comme dans le cas d'une cellule par exemple) clairement identifiable. Cette définition peut être arbitraire, *i.e.*, dépendante de l'observateur et de la finalité de son observation.

L'observateur ou le sujet est donc d'une certaine manière au cœur de la notion de système en tant que construction intellectuelle : « Il y a donc toujours, dans l'extraction, l'isolement, la définition d'un système, quelque chose d'incertain ou d'arbitraire : il y a toujours décision et choix, ce qui introduit dans le concept de système la catégorie du *sujet*. [...] Ainsi, le système requiert un sujet, qui l'isole dans le grouillement polysystémique, le découpe, le qualifie, le

⁵Selon l'encyclopédie participative wikipedia, « un écosystème désigne l'ensemble formé par une association ou communauté d'êtres vivants (ou biocénose) et son environnement géologique, pédologique et atmosphérique (le biotope). »

⁶Le lecteur intéressé par une analyse des conséquences épistémologiques d'une telle « mutilation de la réalité » peut se référer à Morin (1977).

hiérarchise. Il renvoie, non seulement à la réalité physique dans ce qu'elle a d'irréductible à l'esprit humain, mais aussi aux structures de cet esprit humain, aux intérêts sélectifs de l'observateur/sujet, et au contexte culturel et social de la connaissance scientifique. » (Morin, 1977, p. 140)

Dans cette perspective, Cellier (1991) définit un système comme un « environnement virtuel ou réel perçu comme une source de données ou de comportements observables. Ces données sont vues ou acquises au moyen d'un protocole expérimental intéressant le modélisateur⁷. »

1.1.1.2 Le cas des systèmes complexes

On parle de *système complexe* lorsque une « unité globale » est composée de nombreux éléments interagissant localement et simultanément. Les systèmes complexes peuvent être le siège de processus de rétroaction, d'auto-organisation ou d'autopoïèse (auto-production), générant des émergences au niveau macroscopique.

Le terme « émergence » est fréquemment rencontré dans la littérature relative à de nombreux domaines comme la biologie, la philosophie (Maturana et Varela, 1980), la sociologie (Schelling, 1971), l'économie (Derveeuw, 2006), la conception de systèmes de production (Ueda *et al.*, 2001) et d'une manière générale, l'étude des systèmes complexes. Cependant, la définition même de la notion d'émergence pose un grand nombre de problématiques non résolues. Bien que dans la suite de cette thèse une définition volontairement « naïve » et générale, *i.e.*, admise par l'ensemble des chercheurs s'intéressant à ce concept, soit adoptée, il semble important de revenir sur les points importants permettant de spécifier cette notion.

Toute définition du concept d'émergence débute généralement par la citation de la maxime suivante, attribuée à Aristote : « la totalité est plus que la somme des parties » (Aristote, 2008). Cette maxime pose la non-équivalence entre les propriétés de « parties » ou composants pris indépendamment et les propriétés résultant de l'interaction de ces mêmes composants, formant un « tout » ou système. L'émergence est donc le processus par lequel un système acquière de nouvelles propriétés ne pouvant être expliquées ou déduites par la seule connaissance de ses parties. Cette idée est formalisée par Baas (1992) de la manière (ici simplifiée) suivante : soit un ensemble de structures⁸ S_1 , une fonction obs_1 permettant d'observer certaines propriétés de S_1 . De plus, on suppose que les éléments de S_1 peuvent interagir. On pose $S_2 = R_1(S_1)$ où R_1 est le résultat du processus d'interactions des structures de niveau 1 et S_2 est un ensemble de structures de niveau 2. Soit obs_2 , une fonction d'observation permettant d'observer certaines propriétés de S_2 . P est une *propriété émergente* de S_2 si et seulement si $P \in obs_2(S_2)$ mais $P \notin obs_1(S_1)$. Cette définition est généralisable à tout ensemble de structures de rang N .

Bien que cette définition pose le problème de l'existence de nouvelles propriétés « triviales⁹ » et qui ne devraient donc pas être considérées comme émergentes au sens où l'en-

⁷Traduction : Coquillard et Hill (1997)

⁸Le terme « structure » est une abstraction pour désigner, par exemple, les notions de système, d'organisme, etc.

⁹S'il est aisé de déterminer des exemples d'« émergence triviale », *e.g.*, le fait qu'il soit possible de s'asseoir et de se maintenir à une certaine distance du sol sur une chaise mais que cela soit en revanche impossible sur les atomes, pris indépendamment, qui constituent cette même chaise, le problème de la définition de la notion de

tendent par exemple McGregor et Fernando (2005), elle est intéressante car elle pose le problème des différents niveaux d'observation d'un système, *i.e.*, d'une hiérarchie de structures, et souligne l'importance de l'interaction dans le processus d'émergence.

1.1.2 Modèle

1.1.2.1 Qu'est-ce qu'un modèle ?

La notion de modèle peut être définie selon différents points de vue. Ainsi, Minsky (1965) adopte une définition utilitaire du modèle : « *To an observer B, an object A* is a model of an object A to the extent that B can use A* to answer questions that interest him about A* ». Le modèle est vu ici comme un outil permettant de répondre à des questions à propos d'un système donné.

Néanmoins, cette définition ne permet pas de comprendre comment l'on construit un modèle. Un modèle est essentiellement la description abstraite ou concrète (on parle alors de « modèle réduit » ou de « maquette ») d'un système. La caractéristique la plus fondamentale d'un modèle par rapport au système qu'il décrit (système source) est sans doute que le modèle doit être plus facilement manipulable que le système source.

La construction d'un modèle procède par sélection, hiérarchisation et simplification des données pertinentes ; le modèle ne peut donc pas être la réplique en miniature du système source. Il peut, et est conçu pour, reproduire certains aspects du système source, et ce dans une certaine mesure, mais pas tous.

1.1.2.2 Pourquoi modéliser ?

Une réponse au titre de cette section peut être trouvée dans une définition donnée dans la section précédente : un modèle sert à « répondre à certaines questions ». Nous allons voir ici à quels types de questions un modèle permet de répondre. Les services que peuvent rendre un modèle peuvent être classés en deux catégories, selon qu'ils résultent de l'exploitation directe du modèle ou qu'ils fassent appel à des techniques automatisées de traitement des résultats de simulations.

Un modèle peut simplement **décrire** certains aspects de la structure d'un système : on parle alors de **modèle statique**. Ainsi, une carte routière est un modèle du réseau routier d'un espace géographique donné, un diagramme de classes UML est un modèle d'un composant logiciel. Ces modèles, purement descriptifs, peuvent être toutefois utilisés pour aider un humain ou une machine à réaliser des tâches de plus haut niveau comme le routage d'un véhicule ou la détection d'erreurs de conception d'un logiciel.

La notion de modèle est généralement associée, dans la communauté scientifique, à celle de **prédiction**. Ainsi, un **modèle dynamique** peut permettre de prédire l'évolution d'un système. Ces modèles peuvent être eux aussi utilisés pour des tâches de plus haut niveau, relatives à la gestion ou au contrôle de systèmes, comme la prévention de dysfonctionnements dans un système artificiel, *e.g.*, un système de production, la résolution de problèmes inverses (cf. partie II) ou encore l'identification des contraintes à appliquer sur un système pour obtenir le résultat souhaité.

trivialité est tout sauf trivial. Le lecteur intéressé par ce problème peut se référer à Gross et McMullin (2001).

Enfin, un modèle peut permettre d'**expliquer** certaines observations en apportant des connaissances nouvelles sur la structure du système étudié. Notons qu'un modèle explicatif n'est pas nécessairement prédictif. Par exemple, le modèle de ségrégation décrit par Schelling (1971) permet d'expliquer les mécanismes de ségrégation raciale dans un environnement urbain à l'échelle macroscopique à partir des préférences individuelles des agents mais ne permet pas de prédire l'état d'un système de ce type.

Le triptyque description/explication/prédiction n'est pas exhaustif. Ainsi, Epstein (2008) distingue, outre l'explication et la prédiction définies précédemment, quinze buts potentiels de la modélisation d'un système dynamique :

- guider la collecte de données,
- mettre en avant des dynamiques fondamentales,
- suggérer des analogies,
- découvrir de nouvelles questions,
- promouvoir une méthode scientifique,
- restreindre (ou borner) l'ensemble des solutions possibles,
- mettre en avant des incertitudes fondamentales,
- offrir des solutions à un problème en temps réel,
- démontrer l'avantage apporté par une méthode,
- démontrer la résistance à des perturbations,
- montrer l'incompatibilité de connaissances avec les données disponibles,
- former ou entraîner des professionnels,
- aider à la résolution de conflits,
- éduquer le grand public,
- montrer que ce qui semble simple (resp. complexe) est en réalité complexe (simple).

1.2 Modélisation et simulation numérique

Nous avons précédemment défini la notion de modèle de manière très générale. Dans cette section, nous précisons cette définition en la situant dans le champ de la simulation numérique, *i.e.*, lorsque le modèle est conçu pour être implémenté puis exécuté (simulé) sur un ordinateur.

1.2.1 Clés de classification d'un modèle

Dans cette section, nous présentons un certain nombre de clés nous semblant pertinentes, permettant de distinguer différents modèles. Notons que ces clés ne sont pas totalement indépendantes : *e.g.*, un méta-modèle donné est généralement associé à une certaine représentation du temps.

1.2.1.1 Méta-modèle

Le méta-modèle désigne l'ensemble des concepts, ainsi que leurs relations, sur lequel repose la définition du modèle ou d'un composant du modèle (un modèle peut être basé sur plusieurs méta-modèles). Voici par exemple, quelques méta-modèles qui ont été utilisés dans le cadre de cette thèse :

- les équations différentielles,
- les systèmes multi-agents,
- les automates cellulaires,
- les automates à états finis.

1.2.1.2 Echelle de modélisation

Pour modéliser un système, il est nécessaire de disposer de données, même lacunaires, sur celui-ci. Bien souvent, le modélisateur ne dispose pas d'informations exhaustives sur le système à un niveau d'observation donné. Il est généralement confronté à un ensemble de données éparses, ne concernant que quelques aspects du système. On distingue généralement les **micro-connaissances**, relatives au niveau microscopique (*i.e.*, aux parties du système), et les **macro-connaissances**, relatives au niveau macroscopique (*i.e.*, au système lui-même) (Drogoul *et al.*, 2002). Il peut être, dans certains cas, intéressant de considérer les connaissances issues d'un niveau mésoscopique, *i.e.*, de groupes de parties du système, identifiés selon des critères propres à la problématique.

Après étude des données dont il dispose, deux possibilités s'offrent au modélisateur : considérer le système comme un tout cohérent, *i.e.*, d'un point de vue macroscopique, dont certaines propriétés présentent des régularités quantifiables ou qualifiables, ou comme un ensemble de parties en interrelations faisant émerger ces mêmes régularités. Le choix de l'une ou l'autre alternative repose généralement sur des raisons pratiques et/ou liées à l'objectif de modélisation : nature du système, résultats attendus, données disponibles, difficultés liées à l'implémentation, familiarité avec certains méta-modèles, intuition, etc.

Ainsi, un **modèle macroscopique** permet de représenter la dynamique d'un système sans considérer les comportements éventuellement complexes de ses constituants et leur relations alors qu'un **modèle microscopique** permet de représenter comment la dynamique d'un système émerge à partir des interactions entre ses constituants dont le modélisateur a caractérisé le comportement.

Les approches « multi-niveaux » proposent d'associer ces deux représentations, *i.e.*, d'intégrer dans un même modèle des connaissances issues de différents niveaux d'observation (David et Courdier, 2008, 2009; Müller *et al.*, 2005; Ratzé *et al.*, 2007; Servat *et al.*, 1998). Dans Morvan *et al.* (2008c), nous proposons de modéliser, du point de vue microscopique, la façon dont une organisation émerge à partir des interactions entre composants du système et, du point de vue macroscopique, les contraintes imposées aux composants du système par l'organisation ainsi que les propriétés émergentes. Une implémentation de ce concept, basée sur les notions d'agent pulsionnel et de stigmergie¹⁰ est décrite dans Morvan *et al.* (2009b).

1.2.1.3 Déterminisme

Un modèle est **déterministe** si aucune variable aléatoire n'intervient dans la définition du modèle. A l'inverse, on parlera de modèle **stochastique**. L'introduction de variables aléatoires dans un modèle permet de prendre en compte la variabilité intrinsèque (liée par exemple à une variabilité interindividuelle dans une population), ou extrinsèque (liée à des facteurs extérieurs au système) de données observées dans le système source.

¹⁰Ces termes sont définis respectivement dans les sections 1.3.1.1 et 1.3.1.3.

Ainsi, les résultats attendus de la simulation de modèles déterministes sont de simples valeurs ; dans le cas de modèles stochastiques, on cherchera à obtenir des distributions de valeurs.

1.2.1.4 Représentation du temps

Lorsque l'on modélise un système, on tente de caractériser la variation de certaines propriétés du système (variables d'état) en fonctions d'autres. L'une de ces « autres propriétés » est bien souvent le temps. Nous allons voir dans cette section différentes représentations possibles de la variable temps dans un modèle.

Les modèles à *temps continu* considèrent les variations des variables d'état du système de manière continue. Le formalisme généralement associé à ce type de représentation du temps est celui des systèmes d'équations différentielles. Dans son expression la plus simple, la variation infinitésimale d'une variable d'état x dépend d'une fonction du temps f et de la variation infinitésimale du temps t :

$$dx = f(t) \cdot dt. \quad (1.1)$$

Dans les modèles à *temps discret*, le temps est représenté comme un ensemble de valeurs discrètes : $t = \{t_0, t_1, \dots, t_n\}$ tel que pour tous t_i et t_{i+1} de t , $t_{i+1} - t_i = \Delta t$. Δt est une constante appelée pas de temps. On considère donc qu'une variable d'état du système change d'état instantanément à chaque pas de temps.

Dans les modèles à *événements discrets*, le temps est représenté à travers une série d'événements, correspondant aux instants auxquelles les variables du système changent d'état. Le formalisme le plus simple associé à ce type de représentation est DEVS (*Discrete Event System Specification*). Nous ne reviendrons pas sur ce formalisme dans cette thèse. Le lecteur intéressé peut se référer à Zeigler *et al.* (2000) pour une présentation détaillée de DEVS ainsi que certaines de ses évolutions.

1.2.2 Méthodologies de simulation numérique

La conception de simulations numériques est une activité compliquée, nécessitant la mise en œuvre d'une méthodologie adaptée. Les cadres méthodologiques présentés dans cette section ne doivent pas être vus comme des instructions strictes à appliquer systématiquement, mais plutôt comme des guides souples devant être adaptés aux différentes situations rencontrées.

1.2.2.1 Une première définition

La simulation numérique peut être décrite comme un processus itératif composé de trois activités fondamentales (Fichwick, 1997) :

1. conception d'un modèle,
2. exécution du modèle,
3. analyse des résultats.

Ce processus est itératif car l'analyse des résultats de la simulation peut amener à reconsidérer certaines hypothèses retenues lors de la conception du modèle ou la méthode retenue pour exécuter le modèle. Ainsi, ces trois étapes seront répétées jusqu'à l'obtention de résultats satisfaisants (figure 1.1).

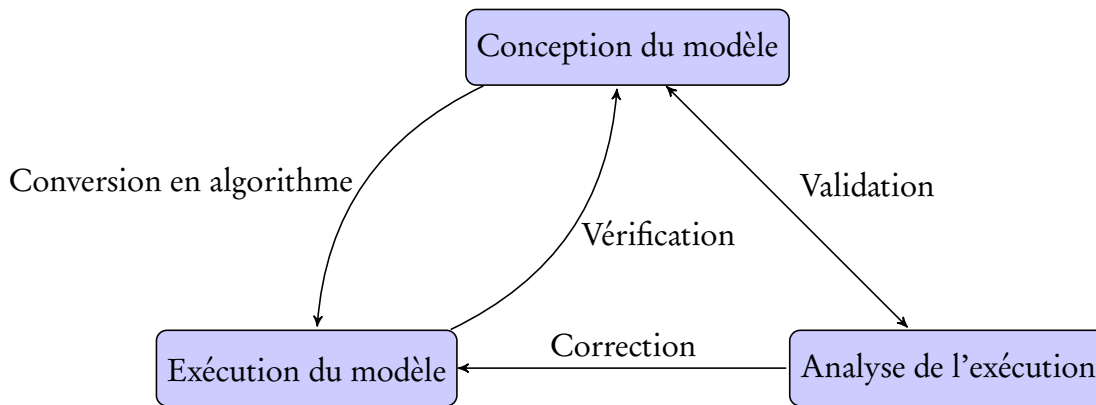


Fig. 1.1 – Cadre méthodologique de simulations numériques (Fichwick, 1997)

1.2.2.2 Une définition plus complète

Bien que séduisante, la définition précédente est sans doute trop simple et il peut être intéressant d'entrer dans les détails. Ainsi, Shannon (1998) propose de décomposer la conception d'une simulation numérique en douze étapes. Nous retiendrons ici dix étapes qui nous semblent essentielles.

1. **Définition des objectifs.** Tout d'abord, il s'agit de définir les objectifs de la simulation, *i.e.*, la question à laquelle on souhaite répondre à propos du système étudié (cf. section 1.1.2.2) et éventuellement la précision des résultats attendus.
2. **Définition du système.** Il s'agit de caractériser le système en le situant dans son environnement (notamment en fixant ses limites), en définissant les échanges qu'il entretient avec lui (cf. section 1.1.1.1) et enfin en choisissant les aspects (composants, propriétés, etc.) du système que l'on va considérer.
3. **Choix ou définition d'un méta-modèle.** Dans cette étape, le modélisateur doit choisir comment modéliser le système d'un point de vue conceptuel. Cela implique un certain nombre de choix (discrétisation du temps, prise en compte de l'incertitude des données, etc.) importants pour la suite.
4. **Écriture du modèle conceptuel.** Il s'agit de traduire les relations entre les aspects du système que l'on considère dans le paradigme choisi. Selon le paradigme, le résultat de cette étape peut être très formalisé (équations, graphes, etc.), défini de manière plus intuitive (schémas, dessins, etc.) ou encore comme un mélange des deux. Parfois, le modèle peut se suffire à lui-même : le modèle est le résultat. C'est par exemple le cas de certains diagrammes UML. Cependant, si l'on considère un modèle dynamique, il est nécessaire de l'exécuter pour obtenir des résultats.

5. **Choix ou écriture d'un simulateur.** Si le modèle a besoin d'être exécuté, il s'agit de concevoir, ou de choisir parmi une offre existante, un logiciel permettant d'exécuter le modèle. Un tel simulateur peut être écrit spécialement pour un modèle précis, ou au contraire être très généraliste, *i.e.*, conçu pour exécuter un modèle conçu selon un ou plusieurs méta-modèles.
6. **Vérification du simulateur.** Il s'agit de s'assurer que le simulateur est exempt d'erreurs de conceptions. Cette étape est particulièrement importante lorsque le simulateur a été écrit. Il s'agit d'une étape transversale aux étapes d'écriture du simulateur et d'exécution du modèle.
7. **Traduction du modèle.** Il s'agit de traduire le modèle dans un langage tel qu'il puisse être exécuté par le simulateur.
8. **Exécution du modèle.** Il s'agit d'exécuter le modèle traduit sur le simulateur et d'obtenir des résultats quantitatifs ou qualitatifs.
9. **Validation du modèle.** Il s'agit de vérifier que le modèle et/ou les résultats de son exécution, permettent d'atteindre les objectifs définis à l'étape 1. Cette étape peut mettre en évidence des erreurs lors des étapes de définition du système, d'écriture du modèle, de sa traduction, ou encore dans le choix du paradigme de modélisation. Il sera alors nécessaire de corriger les erreurs faites précédemment.
10. **Exploitation du modèle.** Le modèle peut enfin être exploité. À partir de cette exploitation, de nouveaux objectifs peuvent être définis, *e.g.*, préciser les résultats obtenus ou décrire le système selon un point de vue différent.

La figure 1.2 illustre les relations entre ces différentes étapes.

1.3 Simulation Orientée Agent

Nous nous intéressons dans cette section aux simulations orientées agent (SOA), c'est-à-dire à des simulations basées sur un méta-modèle particulier : le système multi-agent (SMA). La modélisation à base d'agents constitue très certainement aujourd'hui l'une des techniques de modélisation microscopique les plus populaires. Treuil *et al.* (2008) définissent un modèle à base d'agent ainsi :

Définition 1 (Modèle à base d'agent) « *Tout modèle à base d'agent est un système composé d'entités multiples ou **agents** qui évoluent dans un **environnement**, conçu comme une entité particulière, dans lequel ils sont localisés. Ces agents sont dotés d'attributs, de **comportements**, et de capacité de **perception** et de **communication**. L'ensemble des valeurs des attributs d'une entité à un instant donné constitue l'état de cette entité, et la réunion de l'ensemble des états des entités forme l'état microscopique ou — dit plus simplement — l'état du système. Les capacités de perception des entités leur permettent de consulter un sous-ensemble de cet état microscopique, habituellement de façon localisée dans l'environnement. Les comportements sont des règles contrôlant à chaque instant l'évolution de cet état, en intervenant sur les états des entités qui les portent ou sur leur existence même (création et destruction), ainsi que sur les états et existences des autres entités intervenant dans les éventuelles actions, communications ou interactions décrites dans les comportements.* » (pp. 13 – 14).

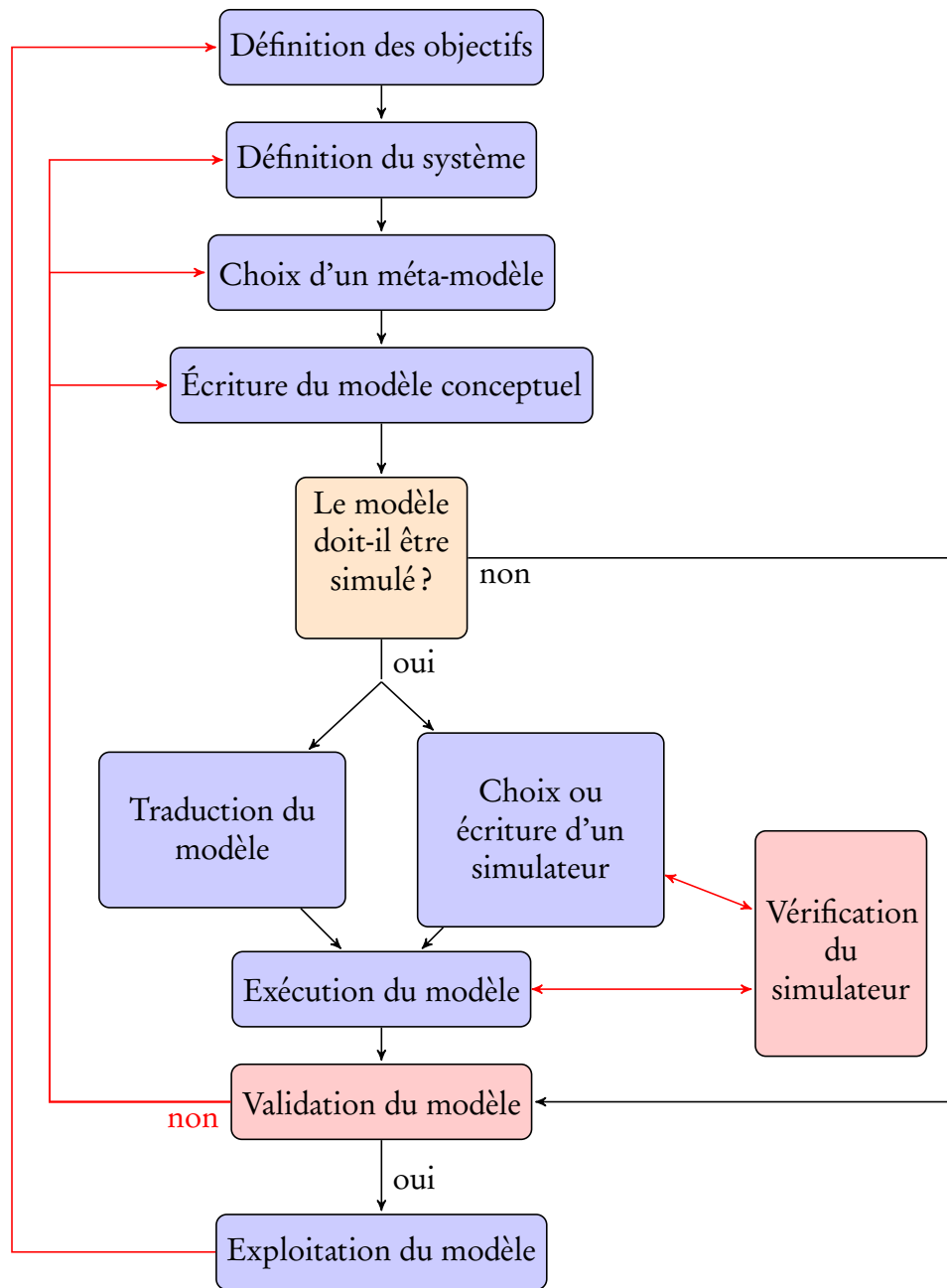


Fig. 1.2 – Cadre méthodologique de simulations numériques

Cette définition fait intervenir un certain nombre de concepts (agent, environnement, etc.) définis succinctement. Dans la section suivante, nous revenons plus en détails sur ces notions.

1.3.1 Notions fondamentales

1.3.1.1 L'agent

Au sens premier (philosophique), un agent peut être défini comme une « entité agissante ». Dans les sciences de l'information, on parle d'agent computationnel (ou tout simplement agent) pour désigner un système informatique, logiciel et éventuellement matériel, **autonome** capable :

- d'**agir** dans son **environnement**, ses actions étant dirigées vers un but, sans en être nécessairement conscient,
- de **percevoir** et éventuellement de se **représenter** partiellement son environnement,
- d'**interagir** avec d'autres agents.

On parle de *comportement* pour désigner l'ensemble des actions et réactions d'un agent observées dans une situation donnée. En revanche, l'*architecture* est l'ensemble des mécanismes internes à l'agent permettant de sélectionner et de réaliser un comportement. L'architecture interne d'un agent est généralement décomposée en modules : *i.e.*, en composants logiciels assurant une fonction précise. L'agent est ainsi conçu comme une chaîne de modules exécutés séquentiellement ou simultanément. On distingue au minimum trois modules :

- un module de perception, permettant d'acquérir des informations en provenance de l'environnement,
- un module d'action, permettant de modifier localement l'environnement,
- un module de prise de décision, permettant de choisir les actions à effectuer et éventuellement de les planifier.

Ce dernier est bien évidemment le plus complexe et différentes architectures ont été proposées pour le modéliser. D'une manière très générale, il est classique d'opposer deux types d'architectures, selon la manière dont l'agent se représente son environnement. On parle ainsi d'agent cognitif si celui-ci dispose d'une représentation symbolique de son environnement et d'agent réactif dans le cas contraire. Ces termes sont utilisés moins formellement pour désigner respectivement des agents individuellement intelligents, possédant des croyances à propos de leur environnement, capables de planifier des actions pour atteindre des objectifs, etc., et des agents dont le module de prise de décision peut se réduire à un système de réaction à des perceptions (stimuli) et pouvant être représenté sous la forme d'un ensemble de règles de type : *si perception alors action*.

Cependant, Ferber (1995) montre que la seule distinction cognitif/réactif est trop réductrice et qu'il convient de dégager une autre dimension d'analyse relative aux objectifs ou buts des agents. Si les buts des agents sont définis dans les agents eux-même, on parlera de conduite téléonomique, si au contraire les buts des agents sont présents dans l'environnement on parlera de conduite réflexe. Cette opposition selon deux dimensions (représentation du monde – conduite) permet de différencier quatre types d'agents présentés dans le tableau 1.1.

Détaillons, à titre d'exemple, le cas des agents réactifs. On parle d'agent tropique lorsque l'agent ne réagit qu'à des stimuli externes, *i.e.*, provenant de l'environnement, et d'agent pul-

sionnel lorsque celui-ci réagit à des stimuli internes, *i.e.*, relatifs à l'état de l'agent, et externes. L'état de l'agent peut donc être vu comme une source de motivation pour celui-ci. Par exemple, les agents modélisant des larves de diptères décrits dans le chapitre 6 réagissent à des stimuli environnementaux (température, signal d'agrégation) et à un stimulus interne représentant le besoin de se nourrir.

Enfin, Ferber (1995) ajoute que ces distinctions ne sont que conceptuelles : il est possible de concevoir des agents hybrides, cognitifs par certains aspects et réactifs par d'autres et plus ou moins téléonomiques.

Relation au monde Conduites	Agents cognitifs	Agents réactifs
Téléonomiques	Agents intentionnels	Agents pulsionnels
Réflexes	Agents « modules »	Agents tropiques

Tab. 1.1 – Les différents types d'agents (Ferber, 1995)

1.3.1.2 L'environnement

La notion d'environnement est fondamentale : dans un système multi-agents, la conception de l'environnement est généralement préalable à la conception des agents. En effet, l'architecture des agents, notamment les effecteurs et les percepteurs, est particulièrement dépendante de la nature de l'environnement. Russel et Norvig (1995) ont proposé des clés de classification d'environnements :

- **Accessibilité** : dans un environnement accessible, un agent peut obtenir l'ensemble des informations relatives à l'état de l'environnement.
- **Déterminisme** : dans un environnement déterministe, une action exécutée dans un contexte donné ne peut avoir qu'un seul et même effet.
- **Dynamisme** : un environnement statique ne change pas sans l'action des agents. En revanche, un environnement dynamique peut se modifier indépendamment des agents.
- **Continuité** : un environnement est discret si l'ensemble des actions et des perceptions possibles est fini, continu sinon.

La classe d'environnements la plus générale (et la plus complexe) comprend les environnements inaccessibles, non-détermistes, dynamiques et continus. Ces environnements sont qualifiés d'*ouverts* (Wooldridge, 2002). L'architecture d'un agent sera d'autant plus complexe que l'environnement dans lequel il agit est lui-même complexe¹¹.

Dans le cadre des simulations multi-agents, l'environnement est souvent défini comme un espace métrique, *i.e.*, un espace topologique¹² dans lequel il est possible de définir une notion de distance, ou métrique, entre tous les éléments de cet ensemble. De plus, l'environnement est souvent discret : on appelle alors cellule une portion de l'environnement considé-

¹¹Il ne s'agit pas là du seul facteur déterminant la complexité de l'agent : la nature des interactions avec l'environnement et du système de prise de décision influent également sur la complexité de l'architecture de l'agent

¹²Un espace topologique est un ensemble muni d'une structure permettant de définir la notion de voisinage d'un élément de cet ensemble.

rée comme homogène. L'exemple le plus typique est sans doute l'environnement représenté comme une grille à deux dimensions, découpée en cellules carrées, chaque cellule disposant généralement de 4 (voisinage de von Neumann) ou 8 (voisinage de Moore) voisins.

1.3.1.3 Communication entre agents

La capacité à communiquer est une caractéristique importante des agents : la communication est, dans les SMA, la forme la plus courante d'interaction. Un acte de communication, intentionnel ou non, est la transmission d'un message d'un émetteur à un ou plusieurs destinataires. Ces derniers doivent donc disposer d'un médium et d'un langage commun afin que le message arrive à destination et puisse être compris par le ou les destinataires. On parlera de *communication point à point* si le message est destiné à un destinataire unique et de *diffusion* sinon. De plus, un acte de communication est rarement isolé : il fait parti d'une conversation. Un message ne peut être alors interprété qu'au regard des messages échangés précédemment. Les acteurs de la conversation doivent donc disposer d'un protocole de communication. Un protocole peut être vu comme un ensemble de couples (séquence de messages, sémantique). De plus, un acte de communication peut être *intentionnel* ou *incident*, i.e., décidé ou subis par l'émetteur. Ainsi, en étant assis à mon bureau, je communique malgré moi : je « dis » à tous ceux qui peuvent me voir que je travaille sur ma thèse. Dans cette section, nous aborderons diverses méthodes de communication utilisées dans le cadre des systèmes multi-agents. Nous insisterons particulièrement sur la communication stigmergique, souvent utilisée dans le cadre de la simulation orientée agent d'insectes sociaux ou présociaux.

La communication directe est la méthode de communication la plus simple : le message est envoyé directement au destinataire, lequel peut le stocker dans une structure de type *boîte à lettres*. Ferber (1995) emploie le terme de message symbolique pour qualifier un tel message. Ce mode de communication autorise la communication point à point et la diffusion. Cependant, l'émetteur doit pouvoir identifier le ou les agents à qui il souhaite envoyer un message. Les plates-formes multi-agents proposent généralement de telles fonctionnalités.

Un tableau noir ou tableau d'affichage est une structure de données accessible par l'ensemble des agents. Les agents peuvent y déposer et consulter des messages et éventuellement les modifier et les effacer. Ce mode de communication permet de diffuser des messages sous la forme d'annonces.

La stigmergie est une méthode de communication indirecte utilisant l'environnement comme médium (Bonabeau, 1999). Les individus communiquent en modifiant localement, dans l'espace et le temps, leur environnement. D'abord étudiée en entomologie (les fourmis et les termites notamment, utilisent ce type de communication), la notion de stigmergie a été utilisée dans des algorithmes d'optimisation, basés sur le comportement de fourragement des fourmis (Dorigo *et al.*, 1996). Concrètement, et dans le cadre des SMA, les agents communiquent en diffusant des *signaux* et en déposant des *marques* dans l'environnement. Un signal est une propriété environnementale se propageant selon une loi qui lui est propre. Une marque est un signal ne se propageant pas et au temps de vie éventuellement limité. Par exemple, dans un environnement discret, il est possible d'associer à chaque cellule un ensemble de *variables d'environnement*. Le comportement d'une variable d'environnement s'inspire de celui d'une phéromone¹³ et peut être défini à l'aide de deux paramètres : un co-

¹³Une phéromone peut être définie comme une substance émise dans l'environnement par un individu afin

efficient de diffusion et un coefficient d'évaporation. Le coefficient de diffusion représente la proportion de la valeur de la variable d'environnement qui sera répartie équitablement entre les voisins de la cellule au pas d'exécution suivant. Le coefficient d'évaporation représente la proportion de la valeur de la variable d'environnement (après diffusion) qui sera perdue par la cellule au pas d'exécution suivant. Les agents doivent bien sûr disposer d'effecteurs leur permettant de manipuler des variables d'environnement. Il est à noter que la définition d'une variable d'environnement peut être modifiée en intégrant des notions comme la saturation.

Contrairement à la stigmergie, les méthodes de communication directe et par tableau d'affichage permettent de structurer le message de manière complexe et sont donc bien adaptées à la conduite de conversations. Ainsi, des standards de communication (ou langages) comme KQML ou ACL, fondés sur la théorie des actes de langage, ont été développés. Le tableau 1.2, tiré de Ferber (1995), récapitule les caractéristiques des différents modes de communication.

Type de message	Mode de communication	Acheminement	Intentionnalité
Message point à point	Point à point	Direct	Généralement intentionnel
Message diffusé	Diffusion	Direct	Généralement intentionnel
Annonce	Diffusion / point à point	Tableau d'affichage	Généralement intentionnel
Signal	Diffusion	Propagation	Incident

Tab. 1.2 – Modes de communication (Ferber, 1995)

1.3.1.4 Le système multi-agent

Nous pouvons maintenant définir ce qu'est un SMA. Ainsi, un SMA est caractérisé par :

- un ensemble B d'entités plongées dans un environnement E ,
- un ensemble A d'agents avec $A \subseteq B$,
- un système d'action (percepteurs, effecteurs) permettant aux agents d'agir dans l'environnement,
- un système d'interaction (e.g., de communication) entre agents,
- éventuellement, une organisation structurant l'ensemble des agents.

1.3.2 Une méthodologie pour la SOA

En partant du constat que les méthodologies classiques de conception de simulations sont mal adaptées aux simulations orientées agent, Drogoul *et al.* (2002) proposent un cadre méthodologique original reposant sur l'idée que la seule décomposition en tâches ou étapes (cf. section 1.2.2) est simplificatrice. En effet, l'élaboration d'un modèle fait intervenir différents acteurs manipulant des notions différentes, cela étant particulièrement vrai dans le cas des

de communiquer d'autres individus de la même espèce.

simulations orientées agent. Il est ainsi possible de distinguer trois rôles dans le processus de conception :

- le **thématicien** est le spécialiste du domaine d'application considéré, il manipule des données relatives à des **agents réels**, *i.e.*, des composants du système considéré. L'ensemble des connaissances que manipule le thématique est appelé **modèle du domaine**.
- le **modélisateur** doit transcrire les agents réels en **agents conceptuels**, *i.e.*, des agents tels que nous les avons définis dans la section 1.3.1.1. Le rôle du modélisateur est de formaliser les connaissances du thématique afin de construire un **modèle conceptuel**.
- l'**informaticien** doit traduire le modèle conceptuel en **modèle opérationnel** (composé d'**agents informatiques**), *i.e.*, exécutable sur un ordinateur et fournir un environnement de simulation.

De plus, des fonctions sont associées aux connaissances issues des différents niveaux d'observation :

- les **micro-connaissances** permettent de construire et de paramétrer le modèle,
- les **macro-connaissances** permettent de valider le modèle.

La figure 1.3 illustre le cadre méthodologique complet.

1.3.3 Plates-formes de simulation orientée agent

Une plate-forme est un ensemble de composants logiciels permettant de faciliter l'implémentation d'une SOA. Elle est généralement composée :

- d'implémentations génériques et réutilisables d'un ou plusieurs méta-modèles d'environnements et d'agents,
- d'un environnement d'exécution d'agents (gestion des communications, du cycle de vie et de l'ordonnancement des agents),
- d'outils permettant à l'utilisateur d'interagir avec le simulateur (visualisation, sauvegarde des résultats, etc.).

Il existe de nombreuses plates-formes dédiées à la SOA. Selon Treuil *et al.* (2008), en 2007, une recherche sur Internet permettait d'identifier environ 400 plates-formes. Ils notent cependant qu'une très large majorité d'entre elles n'étaient soit plus maintenues (ou n'offraient pas de garantie suffisante qu'elles le soient ultérieurement) soit dédiées à une application précise. Ainsi, en dépit de nombreux articles (Railsback *et al.*, 2006), rapports techniques (Leszczyna, 2004), chapitres de livres (Ricordel et Demazeau, 2000; Treuil *et al.*, 2008) et sites web ¹⁴, relatifs au choix d'une plate-forme multi-agents, il est souvent difficile de se décider. En effet, ce choix est basé sur des critères pouvant évoluer très rapidement (*e.g.*, l'activité des communautés de développeurs et d'utilisateurs de la plate-forme) ou difficiles à évaluer sans implémenter réellement le système (efficacité, adéquation du système avec le métamodèle de la plate-forme, etc.).

Pour notre travail de thèse, nous souhaitons pouvoir profiter des possibilités d'une plate-forme générique et de la facilité de mise en œuvre d'une plate-forme dédiée à la simulation pour implémenter respectivement le système d'aide à la décision et le simulateur. Notre choix s'est ainsi porté sur MadKit/TurtleKit.

¹⁴*e.g.*, <http://www.abmsystemsbiology.info>

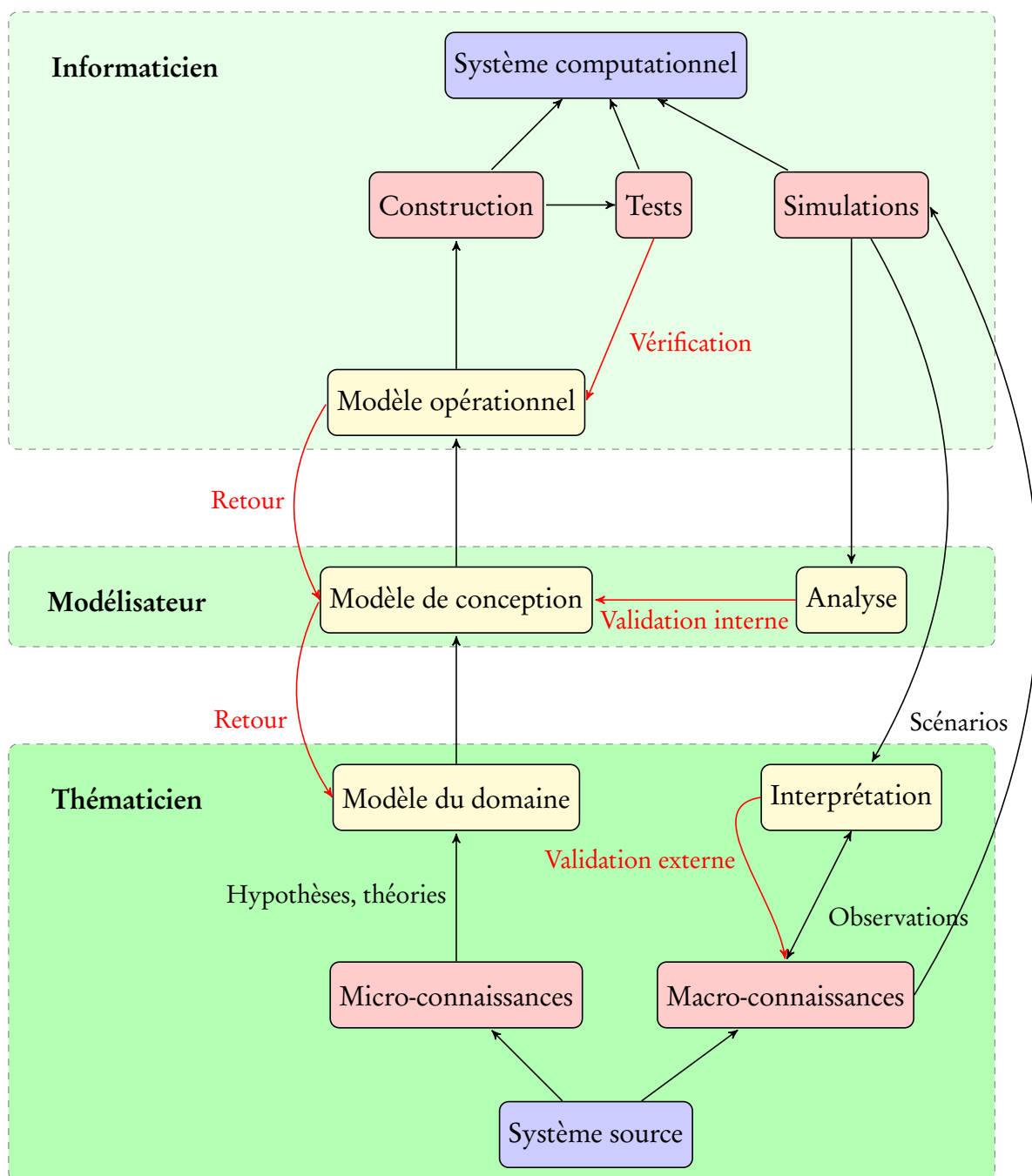


Fig. 1.3 – Cadre méthodologique de conception de simulation orientée agent (Drogoul *et al.*, 2003)

1.3.3.1 MadKit

Madkit¹⁵ (Multi-Agents Developpement Kit) est une plate-forme de développement et d'exécution de systèmes multi-agents générique, écrite en JAVATM et basée sur le métamodèle organisationnel : AGR (Agent – Groupe – Rôle) (Ferber et Gutknecht, 1998; Ferber *et al.*, 2004, 2005; Gutknecht et Ferber, 1997). Ce métamodèle est basé sur trois concepts essentiels :

- un **agent** est une entité autonome communicante qui joue des rôles au sein de différents groupes,
- un **groupe** représente un regroupement d'agents,
- un **rôle** représente une fonction qu'occupe un agent dans un groupe.

Un agent peut appartenir à plusieurs groupes et prendre plusieurs rôles dans un groupe (figure 1.4).

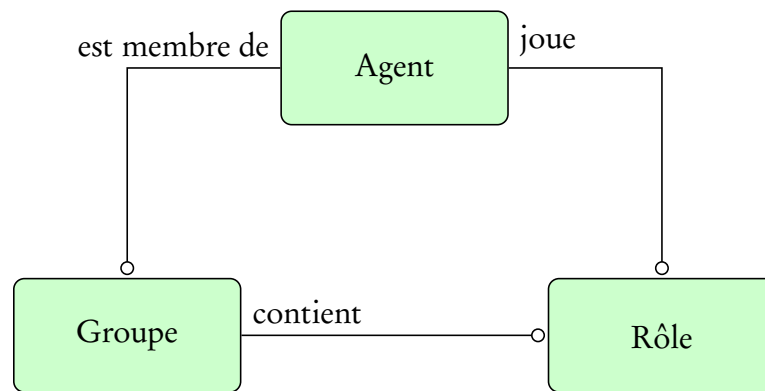


Fig. 1.4 – Les trois concepts fondamentaux du modèle AGR (Gutknecht et Ferber, 1999)

L'environnement d'exécution des agents¹⁶ prend en charge les services suivants : gestion des groupes et rôles locaux, gestion du cycle de vie des agents, passage de messages locaux. Tout autre service est assuré par un agent exécuté par le micro-noyau. Cette architecture modulaire est particulièrement efficace. Un autre exemple de cette modularité est le découplage entre l'agent lui-même et son éventuelle interface graphique : chaque agent la gère en toute indépendance.

MadKit définit nativement deux méthodes d'exécution d'agent. La première associe un processus à chaque agent : chaque agent gère de façon autonome son cycle de vie. Dans la seconde, l'exécution des agents est contrôlée par un ordonnanceur : les agents ordonnancés sont encapsulés dans un unique thread. Cette dernière méthode permet l'exécution de milliers d'agents simultanément sur un même microprocesseur. De plus, l'évolution des propriétés des agents peut être observée facilement.

1.3.3.2 TurtleKit

TurtleKit est un environnement de simulations de type Logo, conçu comme un plugin MadKit (Michel, 2002; Michel *et al.*, 2005). Il se caractérise par sa simplicité et sa facilité de

¹⁵<http://www.madkit.org>

¹⁶Cet environnement est appelé « micro-noyau » par analogie avec les systèmes d'exploitation.

mise en œuvre. Le temps et l'espace sont discrétisés et l'environnement est défini en deux dimensions. Les agents de simulations de TurtleKit (appelées tortues) sont ordonnancés. Leur comportement est défini comme un automate à état fini, chaque état représentant un comportement atomique. La transition d'un état à l'autre est effectuée à chaque pas de simulation.

L'environnement d'exécution de TurtleKit a été quelque peu modifié. Le principal changement, outre quelques modifications cosmétiques, concerne le mode d'ordonnancement des tortues. Afin d'éviter tout artefact de simulation lié au déterminisme dans l'ordre d'exécution, l'ordre d'exécution des tortues change aléatoirement à chaque pas de simulation.

TurtleKit intègre un ensemble d'outils de visualisation permettant d'observer en ligne différentes propriétés associées à l'état des agents ou de l'environnement. Ces outils étant relativement pauvres, certaines interfaces ont été entièrement redéveloppées en utilisant la librairie JFreeChart¹⁷. Ces dernières peuvent être activées et désactivées durant la simulation. L'implémentation de tels outils peut paraître anecdotique. Pourtant, fournir à l'utilisateur ou à des systèmes automatiques de calibrage ou de validation, des indicateurs pertinents sur les simulations est primordial.

1.3.4 Choix de la simulation orientée agent

Enfin, il nous paraît important de soulever une question fondamentale : pourquoi s'orienter vers une modélisation orientée agent plutôt que vers une modélisation basée sur un autre méta-modèle ? Plusieurs réponses sont possibles. Tout d'abord, le paradigme agent est très riche. Il permet ainsi de représenter facilement des processus qualitatifs et quantitatifs, de faire interagir des entités hétérogènes aux architectures variées. Cependant, la motivation première est souvent liée à l'objectif de modélisation : comprendre la relation entre comportements individuels et comportement collectif (Treuil *et al.*, 2008).

Enfin une dernière motivation, peut être moins souvent évoquée, est la robustesse des modèles microscopiques par rapport aux modèles macroscopiques, notamment dans des cas où le contexte écosystémique est très variable et perturbe fortement le comportement du système étudié (Morvan *et al.*, 2009b). Ce dernier aspect, s'il est peu cité dans les travaux relatifs à la modélisation et à la simulation orientée agent est en revanche au cœur des travaux traitant de la conception de systèmes, qu'il s'agisse de systèmes informatiques, de régulation du trafic routier, ou encore de production (voir par exemple Ueda *et al.* (2006)).

1.4 Simulation par automates cellulaires

La simulation par automates cellulaires (SAC), moins populaire (sans doute car plus spécifique) que la SOA, est une technique de simulation s'appuyant sur un méta-modèle des plus simple : l'automate cellulaire (AC). Dans cette section, nous définissons formellement la notion d'automate cellulaire et discutons de son applicabilité pour la modélisation.

¹⁷<http://www.jfree.org/jfreechart/>

1.4.1 Définition

Un AC peut être décrit comme une grille G de dimension d composée de cellules. Une cellule c est caractérisée par :

- l'ensemble des états S_c que peut prendre c ,
- un voisinage de c , $V_c \subseteq G$ pouvant inclure c ,
- une fonction de transition

$$F_c : \{S_{c'}\} \rightarrow S_c, \forall c' \in V_c, \text{ permettant de faire évoluer l'état de } c. \quad (1.2)$$

Un modèle basé sur la notion d'automate cellulaire est donc fondé sur une discrétisation de l'espace et du temps. On considère généralement que l'ensemble des cellules évoluent en parallèle. En d'autres termes, l'état d'une cellule à t_{i+1} ne dépend que de l'état de son voisinage à t_i . De plus, de nombreux AC, étudiés notamment d'un point de vue théorique, sont homogènes : toutes les cellules de l'automate possèdent le même ensemble d'état, la même fonction de transition et une définition commune du voisinage. Cependant, dans un contexte de modélisation de système réel, les modélisateurs s'affranchissent généralement de cette contrainte.

1.4.2 Méthodologie

Un AC est souvent décrit, dans un contexte de modélisation, selon deux axes : le niveau d'observation (distinction micro/macro ou cellule/automate) et la fonction/structure. Le tableau 1.3 synthétise cela.

	Niveau microscopique		Niveau macroscopique
Structure	$\forall c \in G$	S_c et V_c	Topologie de G
Fonction		F_c	Parallélisme

Tab. 1.3 – Axes de définition d'un automate cellulaire

La méthodologie décrite dans la section 1.3.2 semble adaptée à la conception de SAC (moyennant une modification du vocabulaire). Une SAC fait généralement intervenir différents acteurs, manipulant micro et macro connaissances.

1.4.3 Application

La SAC est une technique de modélisation s'appliquant en particulier à des systèmes pouvant être décrits comme des continuums de zones homogènes (cellules). Elle reste donc plus spécifique que la SOA. Elle est cependant très intéressante car elle permet de comprendre comment, à partir de règles très simples au niveau local, il est possible d'obtenir des configurations globales très complexes. De nombreuses SAC ont ainsi été développées dans les domaines de la physique (Chopard et Droz, 2004), de la biologie (Deutsch et Dormann, 2004) ou encore des sciences humaines (Langlois, 1997).

Ferber (1995) note qu'un AC peut être un bon modèle d'environnement dans la conception de SMA. Cette idée a été conservée pour le développement du modèle présenté au chapitre 6.

1.5 Résumé du chapitre

Dans ce chapitre, nous avons tout d'abord définis les notions de système et de modèle, afin d'introduire le lecteur au domaine de la simulation numérique. Nous avons ensuite axé notre propos sur des méthodes de modélisation et de simulation microscopiques, *i.e.*, dans lesquelles le système source est modélisé comme un ensemble de composants en interactions : la simulation orientée agent (SOA) et la simulation par automates cellulaires (SAC).

Problème inverse et raisonnement abductif

Sommaire

2.1	La notion de problème inverse	24
2.1.1	Définition formelle	24
2.1.2	Résolution	25
2.2	Le raisonnement abductif	25
2.2.1	Les différents types de raisonnement	26
2.2.2	Cas d'étude : l'abduction	28
2.2.3	L'abduction en Philosophie	31
2.2.4	L'abduction en Intelligence Artificielle	33
2.3	Relations entre abduction et problème inverse	34
2.4	Résumé du chapitre	35

La notion de problème inverse est très importante dans le domaine de la modélisation et de la simulation numérique. Nous avons vu au chapitre précédent que l'un des objectifs d'une simulation pouvait être de prédire l'état d'un système à partir d'un modèle dont les paramètres sont connus ou supposés connus. Dans le cas du problème inverse, il s'agit à l'inverse de déterminer le paramétrage d'un modèle permettant de produire des résultats les plus proches possibles de données expérimentales.

Un problème inverse est généralement mal posé ou sous-déterminé. En d'autres termes, le problème n'est pas suffisamment (ou à l'inverse est trop) contraint par les données expérimentales fournies et l'unicité de la solution n'est pas garantie. Un tel problème accepte donc souvent une multiplicité de solutions (ou à l'inverse ne possède pas de solution). Bien qu'étudiées dans des domaines scientifiques différents, les notions de problème inverse et de raisonnement abductif sont très semblables. Elles peuvent d'ailleurs être définies informellement de la même manière : « déterminer les causes d'une observation surprenante ». Mais s'agit-il réellement de la même chose ?

Dans ce chapitre, nous revenons sur ces deux notions afin de déterminer s'il est possible d'établir des liens entre les méthodes et les idées associées à l'une et à l'autre. Les formalismes

associés au problème inverse et au raisonnement abductif sont très différents. Afin de ne pas troubler le lecteur et pour privilégier l'intuitivité de l'exposé, nous avons essayé, dans la mesure du possible, de n'utiliser qu'un minimum de formalisme et d'harmoniser les notations.

2.1 La notion de problème inverse

2.1.1 Définition formelle

Soit m , un modèle acceptant des paramètres. L'espace des paramètres de m de dimension k est noté

$$\Phi = \Phi_1 \times \Phi_2 \times \dots \times \Phi_i \times \dots \times \Phi_k, \quad (2.1)$$

où $\Phi_i (1 \leq i \leq k)$ représente l'ensemble des valeurs que peut prendre le paramètre i . m peut être représenté sous une forme fonctionnelle :

$$m : \Phi \rightarrow 2^\Omega, \quad (2.2)$$

où

$$\Omega = \Omega_1 \times \Omega_2 \times \dots \times \Omega_j \times \dots \times \Omega_l \quad (2.3)$$

représente l'espace des données de m de dimension l , *i.e.*, l'ensemble des données qui peuvent être produites par m . $\Omega_j (1 \leq j \leq l)$ représente ainsi l'ensemble des valeurs que peut prendre une donnée produite par m . Un modèle associe donc un paramétrage particulier à un ensemble d'observations. Notons qu'il est possible, et cela est souvent fait dans le cadre de la résolution de problèmes inverses portant sur un type de modèle particulier, de contraindre la définition de Φ et Ω , par exemple avec une notion de volume, afin d'exploiter certaines propriétés du modèle (Tarantola, 2008).

Soit une fonction c , permettant de « mesurer » la cohérence entre un élément ϕ de Φ , *i.e.*, un paramétrage particulier de m et un sous-ensemble Ω^* de Ω , représentant un ensemble de données observées avec une possible imprécision et pouvant être produit par m :

$$c : \Phi \times 2^\Omega \rightarrow \mathcal{E}, \quad (2.4)$$

où $\mathcal{E}$ représente l'ensemble dans lequel est définie la mesure de cohérence. On peut ainsi définir $\mathcal{E}$ à l'aide des ensembles $\{\top, \perp\}$, $[0, 1]$ ou encore $\mathbb{R}$. La sémantique associée à c dépend des caractéristiques de $\mathcal{E}$ et des contraintes imposées sur c . Prenons par exemple, le cas où $\mathcal{E} = [0, 1]$. On peut associer à c la sémantique suivante : pour tout $\phi \in \Phi$ et pour tout $\Omega^* \subseteq \Omega$

- si $c(\phi, \Omega^*) = 1$, alors ϕ et Ω sont totalement compatibles,
- si $0 < c(\phi, \Omega^*) < 1$, alors ϕ et Ω sont partiellement compatibles,
- si $c(\phi, \Omega^*) = 0$, alors ϕ et Ω sont totalement incompatibles.

c est alors appelée *mesure de cohérence*.

Mais on peut donner d'autres sens à c . Si par exemple on impose la condition suivante (Φ étant discret) :

$$\sum_{\forall \phi \in \Phi} c(\phi, \Omega^*) = 1, \quad (2.5)$$

alors c représente une loi discrète de probabilité sur Φ .

Définition 2 (Problème inverse) *L'objectif d'un problème inverse est, étant donné un ensemble d'observations Ω^* réalisées sur le système source, de déterminer un sous-ensemble Φ^* de Φ tel que*

$$\forall \phi^* \in \Phi^*, c(\phi^*, \Omega^*) = \sup(c). \quad (2.6)$$

Cependant, nous verrons dans le chapitre 4 que dans un contexte d'aide à la décision, la définition du problème inverse peut être étendue.

2.1.2 Résolution

Les méthodes de résolutions peuvent être classées en deux catégories : analytique et numérique. L'approche **analytique** (appelée aussi **directe**) consiste à déterminer

$$m^{-1} : 2^\Omega \rightarrow 2^\Phi, \quad (2.7)$$

telle que pour un modèle m donné,

$$\forall \Omega^* \subseteq \Omega, \forall \phi \in m^{-1}(\Omega^*), m(\phi) = \Omega^*. \quad (2.8)$$

Cependant, la mise en œuvre d'une telle méthode est inenvisageable si m est complexe ou si il n'est pas exprimé sous forme mathématique (modèle à base d'agents par exemple).

Les approches **numériques** (appelées aussi **indirectes**) consistent à modifier itérativement un paramétrage initial tant qu'un critère d'arrêt n'est pas atteint, afin de maximiser une fonction objectif, *e.g.*, c . Il s'agit donc de résoudre un problème d'optimisation. Le principal problème associé à cette approche est le nombre important de paramétrages possibles¹⁸. Il n'est donc pas possible d'explorer systématiquement l'espace des paramètres. De nombreuses méthodes intelligentes de parcours d'espaces de recherches (heuristiques ou métaheuristiques), peuvent être alors mises en œuvre (recherche tabou, recuit simulé, algorithmes génétiques, etc.) (Pardalos, 2009).

2.2 Le raisonnement abductif

Le raisonnement est un processus cognitif aux objectifs et formes multiples. La modélisation du raisonnement demeure une problématique centrale en Épistémologie et en Intelligence Artificielle (IA). Dans cette section, nous distinguerons les différents types de raisonnement identifiés par les philosophes et analyserons les finalités possibles d'un raisonnement. Enfin, nous dresserons un panorama des outils développés par les chercheurs en Philosophie et en IA pour modéliser le raisonnement abductif.

¹⁸Un autre problème est le coût associé à l'évaluation de la fonction objectif. Ce problème est généralement éludé dans les méthodes d'optimisation classiques : ce coût est généralement considéré comme négligeable. La méthode proposée dans le chapitre 4 considère la prise en compte de ce problème comme fondamentale.

2.2.1 Les différents types de raisonnement

2.2.1.1 Les trois modes de raisonnement

Un syllogisme permet de représenter (relativement) facilement certains raisonnements. Ainsi, C.S. Peirce¹⁹ utilise cette représentation pour distinguer différents types de raisonnement (Peirce, 1931, vol.2, p. 623). Un syllogisme S peut être défini²⁰ comme un triplet

$$S = \langle c, r, C \rangle, \quad (2.9)$$

où c et C sont respectivement un *cas* et une *conclusion*, i.e., des propositions particulières, vraie ou fausse, et r est une règle permettant de lier le *cas* à la *conclusion*, notamment par l'utilisation de l'implication logique. Le raisonnement est ici défini comme le processus d'inférence permettant d'obtenir l'un des éléments de S à partir des deux autres. Il est ainsi possible d'identifier trois types, ou modes de raisonnement présentés dans le tableau 2.1.

Si un agent²¹ croit qu'un cas c et une règle r , e.g., $r = c \supset C$ ²², sont vrais, alors l'agent est capable d'inférer C ; c'est une *déduction*²³, représentée dans le tableau 2.1 par l'opérateur $\vdash$. La déduction est le seul mode de raisonnement *non révisable*, ou *logiquement valide*, dans le système de croyances considéré : si les croyances de l'agent ne sont pas erronées, alors une déduction ne peut être contredite, même si de nouvelles croyances sont introduites.

Si un agent croit que si c est vrai, alors C est vrai, il peut inférer qu'il existe une relation de cause à effet entre c et C , en d'autres termes, qu'il existe une règle $r = c \supset C$; c'est une induction, représentée dans le tableau 2.1 par l'opérateur $\vdash_i$. Contrairement à la déduction, l'induction est un mode de raisonnement hypothétique ou révisable. En effet, l'induction est essentiellement un mode de raisonnement permettant d'inférer des règles générales à partir de cas particuliers : de nouvelles observations (contradictaires) peuvent donc infirmer un résultat précédemment établi. Cependant, l'observation répétée de la même corrélation entre deux faits, permet de valider *statistiquement* une induction.

Enfin, si un agent croit qu'un fait surprenant C est vrai, il peut, à l'aide d'une règle r qu'il croie également vraie, inférer une cause probable c de l'observation de ce fait ; c'est une abduction, représentée dans le tableau 2.1 par l'opérateur $\vdash_a$. Tout comme l'induction, l'abduction est un mode de raisonnement hypothétique mais pour des raisons différentes. Soient par exemple deux règles $r_1 = c_1 \supset C$ et $r_2 = c_2 \supset C$ considérées vraies par un agent. Ce même agent observe C . Que peut-il inférer ? Logiquement seulement que c_1 ou c_2 sont vraies (ou éventuellement une autre proposition si la base de croyances de l'agent contient d'autres règles ou si elle n'est pas close déductivement). Cette inférence n'est pas une abduction au sens où nous l'avons définie au début de ce chapitre : pour ce faire il doit choisir entre c_1 , c_2 ou la conjonction de c_1 et c_2 . Il doit donc, selon le contexte dans lequel l'inférence est réalisée,

¹⁹C.S. Peirce (1839 — 1914), philosophe pragmatiste à l'œuvre abondante, a été le pionnier des travaux modernes sur le raisonnement abductif.

²⁰Il s'agit là d'une définition particulièrement simple mais cependant suffisante dans le cas qui nous intéresse. Cette définition est notamment utilisée dans Peirce (1931, 1958); Aliseda-Llera (1998) et, d'une manière générale, dans les publications relatives aux travaux de C.S. Peirce.

²¹Dans ce chapitre, nous utilisons la notion d'agent dans son sens le plus général : *ce ou celui qui exerce une action*.

²² $\supset$ représente l'implication logique.

²³Il s'agit dans ce cas de la règle du *modus ponens*

privilégier l'une ou l'autre règle. L'abduction est donc un mode de raisonnement particulièrement complexe dont la validité ne peut être que *contextuelle*. L'abduction est décrite plus longuement dans la section 2.2.2.

Forme du raisonnement	Mode	Validité
$c, r \vdash C$	Déduction	Logique
$c, C \vdash_i r$	Induction	Statistique
$r, C \vdash_a c$	Abduction	Contextuelle

Tab. 2.1 – Les différents modes de raisonnement dans le modèle peircien

2.2.1.2 Le raisonnement temporel

Dans la section précédente, nous n'avons considéré que des propositions pouvant être vraies ou fausses dans un contexte général. Cependant, nombre de nos raisonnements font intervenir une ou plusieurs modalités qui conditionnent la vérité d'une proposition. Ici nous nous intéresserons à une modalité particulière²⁴ : le temps. Ainsi, la proposition « Bob Marley est vivant » dépend du moment auquel elle est énoncée. On parle de raisonnement temporel lorsque *l'objet de l'inférence est distant dans le temps du moment auquel l'inférence est produite*. Il est donc possible de distinguer deux types de raisonnement temporel selon que le moment sur lequel porte l'objet de l'inférence est situé dans le passé ou dans le futur, relativement au moment de production de l'inférence. On parle ainsi de *rétrodiction* (ou d'*explication*), et de *prédiction* pour qualifier respectivement l'un et l'autre.

Il semble intéressant de lier ces deux types de raisonnement temporel aux modes de raisonnement du modèle peircien. La solution la plus évidente est d'associer la prédiction à la déduction et la rétrodiction à l'abduction (Shanahan, 1989). En effet, les modèles permettant d'associer certaines propositions à d'autres dans le temps sont généralement prédictifs. En d'autres termes, ces modèles permettent de déterminer l'état de certaines propriétés d'un système dans le futur à partir de l'état de certaines propriétés de ce même système observées antérieurement. Or, cela n'est pas nécessairement le cas ; nous verrons par la suite que des modèles temporels peuvent être rétrodictifs. Les types de raisonnement temporel et les modes de raisonnement du modèle peircien ne doivent donc être liés qu'en fonction du type de modèle temporel utilisé comme le montre le tableau 2.2.

Il est également possible de considérer des formes de raisonnement temporel de plus haut niveau comme la planification : il s'agit, partant d'une situation donnée, de planifier un ensemble d'actions dans le temps afin d'atteindre un objectif. Nous définissons ces raisonnements comme étant *de plus haut niveau* car ils mettent en œuvre déduction et/ou abduction.

2.2.1.3 Les différents objectifs du raisonnement

Tout comme l'action, le raisonnement est, parfois inconsciemment, dirigé vers un but. Cependant, la nature de ce but dépend du contexte. Ainsi, le raisonnement, notamment

²⁴Nous pourrions parler de raisonnement modal au sens général, mais cela serait, dans le cadre de cette thèse, quelque peu hors de propos.

Modèle temporel	Type de raisonnement temporel	Mode dans le modèle peircien
Prédictif	Prédiction	Déduction
Prédictif	Rétrodiction	Abduction
Rétrodictif	Rétrodiction	Déduction
Rétrodictif	Prédiction	Abduction

Tab. 2.2 – Types de raisonnements temporels associés aux modes de raisonnement du modèle peircien

scientifique, est utilisé pour démontrer, empiriquement ou formellement, la véracité ou la fausseté d'une proposition ; en d'autres termes pour produire une inférence. Le raisonnement permet ainsi à un agent d'obtenir de nouvelles connaissances ou croyances.

Le raisonnement peut également être un préalable à l'action. Le raisonnement permet alors de prédire les conséquences éventuelles d'une action envisagée, de déterminer si les objectifs de l'action peuvent être atteints, etc. D'une manière générale le raisonnement permet de prendre une décision relative à une action à mener.

2.2.2 Cas d'étude : l'abduction

Nous avons précédemment donné une première définition de l'abduction, proposée par C.S. Peirce et basée sur la notion de syllogisme. L'une des définitions de l'abduction la plus citée, bien que souvent critiquée, est également due à Peirce : « *The surprising fact, C, is observed. But if A [an explanatory hypothesis] were true, C would be a matter of course. Hence, there is reason to suspect that A is true.* » (Peirce, 1931, vol. 5, p. 189). Les définitions de ce type, i.e., très générales, soit ne définissent pas grand chose, soit omettent certains aspects du problème et sont systématiquement critiquées. Aussi, nous ne nous risquons pas à cet exercice et donnerons par la suite des définitions relatives au contexte d'étude, Philosophie ou IA, dans lequel elles ont été élaborées. Notons de plus que l'abduction se retrouve dans des domaines applicatifs très divers (enquête scientifique, diagnostic, etc.) et que les modèles abductifs peuvent être propres à l'un de ces domaines.

Mais tout d'abord, afin d'illustrer notre propos, nous reprenons un exemple de « sens commun » issu de Peirce (1931) en le complexifiant. Nous proposons ainsi au lecteur l'expérience de pensée suivante :

En vous réveillant de votre sieste, vous constatez que la pelouse de votre jardin est mouillée. Bien qu'il existe plusieurs explications possibles à cette observation surprenante, l'une d'elle s'impose a priori : il a plu durant votre sommeil. Cependant, il semble intéressant de valider cette hypothèse, ne serait-ce que pour savoir si vous devez prendre un parapluie pour sortir faire votre promenade. Ainsi, si en regardant le ciel, vous voyez que le ciel est couvert, cela confirme l'hypothèse de la pluie. Si en revanche aucun nuage n'obscurcit l'horizon, il faut rejeter cette hypothèse et en considérer une autre : par exemple le déclenchement inopiné de votre système d'arrosage automatique. Là aussi, il semble intéressant de valider cette hypothèse afin de déterminer si vous devez ou non réparer ce système : le prix de l'eau et votre conscience écologique ne vous permettent pas de le laisser dans un tel état de dysfonctionnement. En admettant que vous ayez les connaissances suffisantes, vous pouvez ainsi déterminer la validité de cette hypothèse. Si votre système fonctionne

correctement, il ne vous reste plus qu'une hypothèse à envisager : des enfants du voisinage ont joué sur votre pelouse pendant que vous dormiez. Cependant, allez-vous valider cette hypothèse ? À moins que vous détestiez les enfants en général, ou les enfants de vos voisins en particulier, et que vous souhaitiez les voir punis, le fait de savoir que cette hypothèse est vraie ou fausse n'aura pas d'influence sur vos actions futures.

2.2.2.1 Deux points de vue sur l'abduction

Le raisonnement abductif peut être analysé selon deux points de vue : celui du processus inférentiel, que nous appellerons par la suite « raisonnement abductif » et celui du résultat de ce processus, que nous appellerons simplement « abduction »²⁵. Certains auteurs ont pu opposer ces deux approches ; nous préférons les voir comme complémentaires. Nous verrons dans les sections suivantes les problématiques qui sous-tendent ces deux points de vue.

L'abduction comme produit Penser l'abduction comme le produit d'un raisonnement revient à se poser le problème de sa forme en fonction de la question posée et des autres solutions possibles. Un certain nombre de critères (*theoretical virtues*), permettant de caractériser une bonne abduction, ont été identifiés, ceux-ci pouvant par ailleurs être utilisés pour guider le raisonnement abductif, comme nous le verrons par la suite. Il s'agit de critères permettant d'ordonner des abductions *valides*, dans le cadre de la théorie utilisée.

Le premier critère est la **cohérence** : l'abduction proposée doit être la plus cohérente avec l'ensemble des connaissances dont l'agent dispose. Afin de quantifier un tel « degré de cohérence », il semble naturel de déterminer une *mesure de cohérence* entre l'abduction proposée et le problème posé. Il s'agit là d'une notion très générale pouvant être spécialisée en se plaçant dans un cadre théorique plus précis, e.g., la théorie des probabilités ou des possibilités.

Le second critère permettant de déterminer si une abduction est intéressante ou non peut sembler trivial : il s'agit de l'**explicativité**. Une abduction doit permettre d'expliquer le fait surprenant observé. Il existe cependant des degrés d'explicativité qui dépendent des contextes d'observation et d'inférence : ainsi il est courant de dire qu'une proposition est plus explicative qu'une autre.

Le troisième critère, fréquemment cité et particulièrement utilisé dans les approches logiques est celui du rasoir d'Occam, en d'autres termes, la **simplicité**. Une solution « simple » ou « minimale » devrait être préférée à une solution plus générale incluant un grand nombre d'hypothèses. Ainsi, dans le cadre de l'exemple présenté plus haut, nous n'avons pas retenu la solution expliquant l'humidité de la pelouse par la présence conjointe de la pluie, des enfants et du dysfonctionnement du système d'arrosage. Cette notion de simplicité est souvent interprétée comme la simplicité logique. Pourtant, pour un agent cognitif, la simplicité semble être bien plus complexe (Paul, 1993).

Un quatrième critère pouvant être utilisé est l'**utilité**. Par exemple dans le cas de l'exemple de la pelouse mouillée, l'explication des enfants n'est pas du tout utile dans le sens que sa vérité ou sa fausseté n'influenceront pas, ou très peu les actions de l'agent ; en revanche l'explica-

²⁵Cette terminologie nous est propre. Dans la littérature, ces deux appellations sont utilisées pour désigner l'un ou l'autre point de vue.

tion de la pluie est utile car elle conditionne l'utilisation d'un parapluie pour la promenade. Notons que les critères d'explicativité et d'utilité peuvent être contradictoires : une proposition très explicative peut se révéler absolument inutile (indépendamment de la vérité de la proposition).

Un cinquième critère est parfois cité, il s'agit de la **spécificité**. Une explication spécifique sera ainsi préférée à une explication générale. Par exemple, si l'on considère le cas du dysfonctionnement du système d'arrosage automatique, une proposition relative à la défaillance d'un élément technique très précis sera préférée à une proposition relative à la défaillance du dispositif en général.

Enfin, il est nécessaire de prendre en compte un dernier aspect particulièrement important : l'abduction ne peut être considérée comme un mode d'inférence valide que si la meilleure solution, *e.g.*, au sens des différents critères évoqués ci-dessus, est suffisamment bonne (« *if the "best" is good enough* »). Ainsi, parfois la meilleure solution est l'agnosticisme ; en d'autres termes, il peut être préférable de ne rien inférer (Lipton, 2004, p. 104).

L'abduction comme processus Nous venons de voir quelles pouvaient être les caractéristiques d'une bonne abduction. Nous n'avons cependant rien dit de la façon dont on pouvait la construire. Il paraît délicat d'établir les caractéristiques générales de ce processus tant elles dépendent du modèle de raisonnement. Il nous semble pourtant intéressant de développer quelques idées évoquées précédemment.

Tout d'abord, notons qu'un raisonnement est nécessairement déclenché par une observation. Il est généralement établi que cette observation peut être de deux natures différentes : nouvelle ou anormale. En quelques mots, une observation nouvelle est une observation qui nous paraît surprenante car nous ne disposons pas de modèle²⁶ qui aurait permis de la prédire ; une observation anormale est elle surprenante car nous disposons d'un modèle qui prédit un résultat contradictoire.

Ainsi, là où l'on peut conduire des raisonnements déductif et inductif indépendamment de l'environnement dans lequel nous sommes situés (en se basant sur des connaissances établies à partir d'observations précédentes), le raisonnement abductif est nécessairement contextuel. Ainsi, dans notre exemple nous établissons que c'est la pluie (ou des enfants ou un dysfonctionnement du système d'arrosage) qui a mouillé la pelouse *dans ce cas précis* ; cette abduction étant basée sur le résultat d'un raisonnement inductif qui nous donne que la pluie (ou des enfants ou un dysfonctionnement du système d'arrosage) est susceptible de mouiller la pelouse *d'une manière générale*.

De nombreux auteurs, *e.g.*, Paavola (2003), voient dans le raisonnement abductif un processus *heuristique*, *i.e.*, reposant sur une stratégie ou une méthode permettant de guider le processus. Cette idée sera détaillée dans la section 2.2.3.4. Il faut cependant souligner que certains modèles, *e.g.*, le modèle bayésien, sont fondamentalement incompatibles avec cette idée.

²⁶Dans un cadre logique, nous aurions utilisé le terme « théorie » ; nous nous référons ici à la notion de modèle telle qu'elle a été définie au chapitre précédent.

2.2.3 L'abduction en Philosophie

2.2.3.1 La notion de modèle causal

Un modèle causal permet d'associer un ensemble de causes à un ensemble d'effets. De nombreux travaux sur l'abduction se basent sur des modèles causaux décrits à l'aide d'une théorie particulière, essentiellement la théorie des probabilités et la logique du premier ordre, plutôt que de travailler sur la notion de modèle causal en général²⁷. Cela pose le problème des notions « cachées », au delà des différences des formes évidentes, par la théorie utilisée. L'une d'elle nous semble particulièrement problématique : il s'agit de l'*inversibilité* du modèle causal. Nous dirons qu'un modèle causal est inversible s'il est possible de manipuler sa structure interne afin d'inverser le sens du raisonnement. Par exemple un modèle d'inférence logique classique comporte la règle du *modus ponens* : si je crois qu'une proposition ϕ est vraie et que si ϕ est vraie alors une proposition ω l'est aussi (ϕ implique ω), alors je peux inférer que ω est vraie. Cette règle est formalisée de la manière suivante :

$$\frac{\phi \supset \omega, \phi}{\omega}. \quad (2.10)$$

Il est possible d'inverser le *modus ponens* pour obtenir la règle d'inférence suivante :

$$\frac{\phi \supset \omega, \omega}{\phi}. \quad (2.11)$$

Cette règle peut être traduite en langage naturel ainsi : si je crois qu'une proposition ω est vraie et que ϕ est une cause possible de ω , alors je peux supposer (par abduction) que ϕ est vraie.

Or, dans le cas général, rien n'assure que le modèle qui sous-tend les connaissances d'un agent soit inversible. Ainsi, les contributions présentées dans cette thèse considéreront la notion de modèle causal en général (bien que des méthodes d'implémentation soient par la suite développées dans le cas particulier des modèles orientés agent).

2.2.3.2 Une approche classique : l'abduction comme « déduction inverse »

L'approche la plus « classique » de l'abduction en Philosophie est logique : elle considère que le raisonnement abductif est une forme d'inférence déductive « inversée » (*backward reasoning*) basée, par exemple, sur la règle présentée dans l'équation 2.11. Cependant l'application de cette seule règle pose le problème de la multiplicité des solutions ; ainsi, le raisonnement abductif a pu être considéré comme une forme dégradée ou « faible » d'inférence, permettant d'inférer « n'importe quoi ».

Ainsi, l'approche logique du raisonnement abductif, décompose généralement le processus de détermination d'une solution en deux étapes : (1) *construction* de l'ensemble des solutions possibles ; (2) *sélection* de la meilleure solution (Aliseda-Llera, 1998, ch. 2). Le processus de sélection d'une solution est un problème complexe reposant sur la quantification puis l'application d'un ou plusieurs critères définis dans la section 2.2.2.1 ou sur un ordonnancement *a priori* de l'ensemble des solutions possibles.

²⁷Certains travaux au contraire, e.g., Lipton (2004) sont basés sur des notions plus empiriques et intuitives et donc moins formalisés, sans que cela nuise à la rigueur des démonstrations.

2.2.3.3 L'inférence contrastive

Une approche pragmatique du raisonnement abductif est de considérer qu'il existe effectivement une méthode permettant de construire un ensemble de solutions possibles ou *hypothèses* et une méthode permettant d'ordonner préférentiellement ces hypothèses de manière empirique, une telle méthode devant, si elle est complète, ne retourner qu'une seule solution : la meilleure, en fonction des connaissances obtenues. Cette méthode de test encapsule donc des processus de collecte d'informations et d'inférence.

Certains auteurs considèrent qu'il est impossible, dans une très large majorité de cas²⁸, de conduire ce processus pour une hypothèse prise indépendamment (Lipton, 2004). En revanche, les hypothèses peuvent être ordonnées préférentiellement par un processus de *comparaison*. Cette idée est particulièrement séduisante car elle semble cohérente avec l'idée, intuitive certes, que nous (humains) avons de nos propres pratiques inférentielles.

Une implémentation possible d'une telle méthode consiste par exemple à comparer les hypothèses deux à deux, itérativement jusqu'à l'obtention de l'hypothèse la plus probable. Dans le cas de l'exemple de la pelouse mouillée, imaginons que vous ne disposez pas de méthode de validation des hypothèses expliquant cette observation surprenante, *i.e.*, que vous n'avez pas accès aux données de la station météorologique la plus proche de chez vous, que vous n'avez pas les connaissances techniques suffisantes pour déterminer si votre système d'arrosage est défectueux et que vous n'avez aucun contact avec vos voisins. Si vous considérez par exemple les hypothèses de la pluie et de la défaillance du système d'arrosage automatique, le fait de regarder le ciel et d'y observer de gros nuages, puis d'aller voir votre système d'arrosage et de constater que tout à l'air normal ne prouve pas que l'hypothèse de la pluie est la bonne : cela favorise l'hypothèse de la pluie par rapport à celle du dysfonctionnement du système d'arrosage. En d'autres termes, ce processus ne permet pas formellement d'expliquer de manière absolue pourquoi l'hypothèse de la pluie semble bonne ; il permet d'expliquer pourquoi l'hypothèse de la pluie semble plus vraisemblable que l'hypothèse du dysfonctionnement du système d'arrosage.

2.2.3.4 De l'importance des stratégies

La conduite d'un raisonnement abductif selon les approches (logique et pragmatique) évoquées plus haut peut être problématique si le nombre d'hypothèses est élevé. En effet, dans un grand nombre de cas pratiques, il sera impossible pour un agent *d'ordonner* l'ensemble des solutions possibles du problème car il ne dispose pas des ressources (par exemple en temps) nécessaires pour ce faire. Il va donc devoir choisir (souvent itérativement) un sous-ensemble d'hypothèses à valider. Par ailleurs, même si l'agent dispose des ressources nécessaires pour réaliser ce processus, il souhaite généralement minimiser leur utilisation et se diriger le plus rapidement possible vers la meilleure solution.

Il semble en effet que nous disposions, à la suite d'un processus d'apprentissage, de telles facultés. Si Peirce attribua cette disposition à l'intuition, (Peirce, 1931, vol. 6, p. 526), ce qui d'un point de vue analytique ne semble pas être très satisfaisant, plus récemment des philosophes ont travaillé sur les méthodes (heuristiques) et les critères sous-tendant ces méthodes, permet-

²⁸Ces auteurs s'intéressent essentiellement au raisonnement scientifique, cependant il semble raisonnable d'étendre leurs conclusions à l'ensemble des « formes » de raisonnement.

tant de guider le raisonnement abductif (Hintikka, 2001; McGrew, 2003; Paavola, 2003). Il s'agit essentiellement d'idées trop peu formalisées, au sens où peut l'entendre un informaticien, pour être implémentées en tant que telles. Certains de ces travaux proposent d'utiliser des critères permettant d'évaluer la qualité d'une abduction pour guider la recherche.

2.2.4 L'abduction en Intelligence Artificielle

La deuxième « école de pensée » à s'être intéressée, plus tardivement, à l'abduction, est l'IA. Essentiellement inspirée par la tradition philosophique, l'IA a proposé des méthodes qui ont été elles-mêmes re-développées par des philosophes, *e.g.*, (Aliseda-Llera, 1998). Sans pour autant quitter le monde de l'intuition, il sera nécessaire d'utiliser des formalismes précis pour décrire les approches développées en IA ; afin de ne pas nuire à la lisibilité de cette thèse, ceux-ci seront introduits avec parcimonie et au fur et à mesure.

Deux approches seront décrites : l'abduction comme « recouvrement d'ensemble » (*set cover*) et comme inférence logique. Cependant, cette présentation ne sera pas exhaustive²⁹ ; elle servira essentiellement l'objectif suivant : montrer que ces approches sont intrinsèquement inadaptées à la résolution de problèmes dans lesquels des modèles de systèmes complexes sont utilisés comme « base de connaissance ».

2.2.4.1 L'abduction comme un problème de « recouvrement d'ensembles »

L'approche qui considère l'abduction comme un problème de « recouvrement d'ensemble » définit, de manière minimale, un tel « cadre abductif », $\mathcal{A}$, comme un triplet

$$\mathcal{A} = \langle \Phi, \Omega, e \rangle, \quad (2.12)$$

où Φ est un ensemble d'hypothèses, Ω est un ensemble de « manifestations », *i.e.*, l'ensemble de tous les faits surprenants qu'il est possible d'observer, et e est une application de 2^Φ vers 2^Ω (Reggia *et al.*, 1983). Les algorithmes de résolutions supposent généralement que e possède certaines propriétés comme :

$$\text{la calculabilité : } \forall \Phi' \subseteq \Phi, e(\Phi') \text{ peut être calculé} \quad (2.13)$$

$$\text{l'indépendance : } \forall \Phi_1, \Phi_2 \subseteq \Phi, e(\Phi_1 \cup \Phi_2) = e(\Phi_1) \cup e(\Phi_2) \quad (2.14)$$

$$\text{la monotonie : } \forall \Phi_1, \Phi_2 \subseteq \Phi, \Phi_1 \subseteq \Phi_2 \Rightarrow e(\Phi_1) \subseteq e(\Phi_2) \quad (2.15)$$

Le problème du calcul d'une abduction peut ainsi être décrit comme suit : soit $\Omega^* \subseteq \Omega$ un ensemble de manifestations observées ; il s'agit de déterminer un ensemble minimal $\Phi^* \subseteq \Phi$ tel que $e(\Phi^*) = \Omega^*$.

Cette approche peut être étendue à des cas dans lesquels l'application e est *floue* et/ou les manifestations ne sont observées qu'avec un certain « degré de certitude (ou d'incertitude) ». Les théories de l'incertain, comme la théorie des possibilités, offrent un cadre particulièrement adapté pour ce type de problèmes (Dubois et Prade, 1995).

L'approche basée sur la notion de recouvrement d'ensembles est particulièrement adaptée à certains problèmes de diagnostic pour lesquels les relations entre causes et effets peuvent être définies exhaustivement.

²⁹Elle ne pourrait l'être faute de place, tant les travaux relatifs à ces approches sont nombreux.

2.2.4.2 L'abduction comme inférence logique

La seconde approche considère que le raisonnement abductif peut être défini comme un mode d'inférence logique. On suppose que la connaissance de l'agent est définie par une théorie $\mathcal{T}$ définie sur un langage $\mathcal{L}$. Soit Φ , un ensemble de propositions de $\mathcal{L}$, définissant les hypothèses, *i.e.* l'ensemble des explications possibles à l'ensemble des observations surprenantes possibles. Soit une proposition ω une observation surprenante, *i.e.*, $\mathcal{T} \not\vdash \omega$ (il n'est pas possible de déduire ω de $\mathcal{T}$). ϕ est une explication de ω si et seulement si :

- $\mathcal{T} \cup \phi \vdash \omega$;
- $\mathcal{T} \cup \phi$ est cohérent ;
- $\phi \in \Phi$.

2.3 Relations entre abduction et problème inverse

D'après les définitions précédentes, il est possible de dire que la résolution d'un problème inverse se base sur une forme particulière de raisonnement abductif, déclenché par une observation anormale. Cependant, la différence principale entre raisonnement abductif et résolution de problème tient à l'objet d'étude. Dans le premier cas, il s'agit de modèle de connaissances, dans le second de modèles de systèmes réels. Si la pertinence de cette distinction peut être discutée, il demeure que les formes des modèles dans l'un et l'autre cas sont très différentes. Ainsi, nous ne nous intéresserons pas aux méthodes associées à un méta-modèle particulier. C'est le cas de l'ensemble des méthodes d'abduction développées en IA et plus généralement basées sur un langage particulier, comme la logique, et de la résolution analytique de problèmes inverses. Par ailleurs, ces méthodes peuvent être rapprochées car elles supposent que le modèle causal étudié soit inversible. Elles semblent donc inapplicables dans le cadre de systèmes complexes. Elles supposent en effet une connaissance *a priori* de l'ensemble des liens causaux entre des explications et des observations : les activités de collecte d'information et de raisonnement sont supposées s'exécuter l'une après l'autre. Pourtant, dans un contexte empirique, *e.g.*, un raisonnement scientifique, les relations de cause à effet sont généralement construites progressivement : collecte d'information et inférence sont intimement liées (Lipton, 2004). Ainsi, la dynamique d'un raisonnement, et donc sa conclusion, dépendent des « choix expérimentaux » de l'agent.

Ajoutons que dans un système complexe, les liens de causalité entre propriétés peuvent être très dépendants du contexte écosystémique et ne peuvent être définis *a priori*. Dans des cas très complexes, dont nous verrons un exemple dans la dernière partie de cette thèse, les dynamiques engendrées par ces caractéristiques semblent difficilement traduisibles, si cela n'est pas tout simplement impossible, dans les représentations classiques : logiques mathématiques, théorie des probabilités, systèmes d'équations différentielles, *etc.*

En revanche, l'approche pragmatique de l'abduction et la méthode de résolution indirecte d'un problème inverse peuvent être rapprochées. Tout d'abord, elles considèrent la notion de modèle causal en général. De plus, on peut leur associer un algorithme de résolution commun. La notion de stratégie, ou en d'autres termes d'heuristique, est fondamentale dans un cas comme dans l'autre. Ainsi, dans le chapitre 4, nous proposons une généralisation de la définition de problème inverse ainsi que des heuristiques basées sur la quantification de certains des critères utilisés pour caractériser une bonne abduction (cf. section 2.2.2.1).

2.4 Résumé du chapitre

Dans ce chapitre nous avons défini les notions de problème inverse et d'abduction. Nous avons ensuite étudié différents modèles d'abduction tout d'abord en Philosophie, puis en IA. Nous nous sommes enfin interrogé sur la pertinence d'établir des liens entre ces modèles et les méthodes de résolution de problèmes inverses, en particulier dans le cadre des systèmes complexes. Nous avons ainsi montré que si les méthodes logiques ou basées sur la notion de recouvrement d'ensemble semblent difficilement applicables à la résolution de problèmes inverses, l'approche pragmatique, issue de la Philosophie, peut être une piste de recherche intéressante.

Deuxième partie

**Une architecture de système d'aide à la
décision basé sur les notions de problème
inverse et de simulation orientée agent**

Architecture générale – Problématiques

Sommaire

3.1	Introduction	40
3.2	Validation de simulations	41
3.2.1	Mesure de la validité d'une simulation	42
3.2.2	Mesure de la validité d'un ensemble de paramètres	42
3.3	Observation de simulations orientées agent	42
3.3.1	Problématique générale	42
3.3.2	Description du cas d'étude	44
3.3.3	Filtrage des informations	44
3.4	Résumé du chapitre	50

Dans ce chapitre, nous définissons une architecture hiérarchique de système d'aide à la décision dans laquelle les connaissances de l'expert sont définies par des modèles, au sens où nous l'avons défini dans le chapitre 1, implémentés sous la forme de simulateurs capables de valider des simulations. Dans le cas de modèles à base d'agents, un système basé sur une telle architecture peut se révéler difficile à exploiter, l'exécution de nombreuses simulations étant très coûteuse en temps de calcul. Ainsi, nous présenterons des méthodes d'observation de SOA permettant d'optimiser le processus de validation de simulations et de paramètres.

3.1 Introduction

L'architecture présentée ici ambitionne de servir de base générique pour la construction de systèmes d'aide à la décision dans des contextes variés et complexes. Nous nous intéressons plus particulièrement à des problèmes pour lesquels :

- il n'existe pas de modèle unique du système source,
- aucune information n'est disponible a priori sur la forme de l'espace de recherche,
- l'unicité ou l'existence de la solution n'est pas garantie.

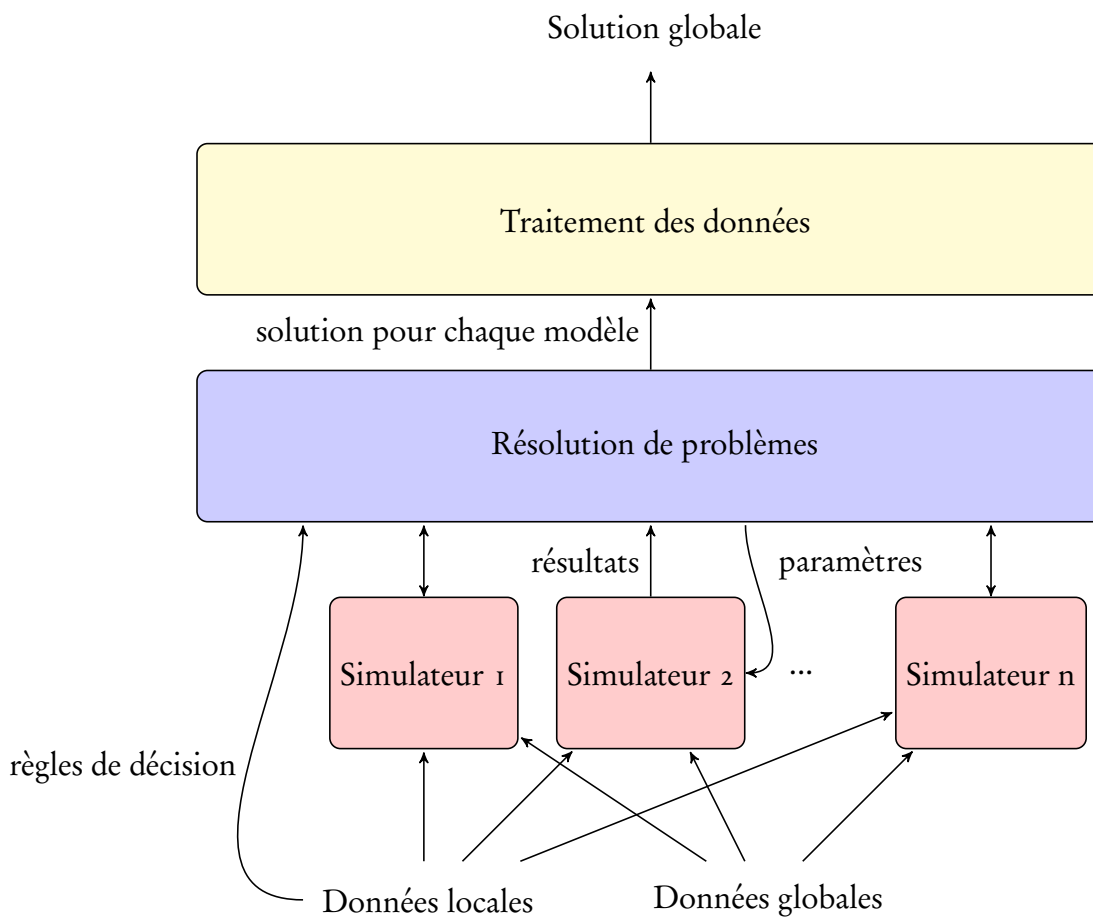


Fig. 3.1 – Architecture générale du système d'aide à la décision.

Cette architecture, représentée dans la figure 3.1, est basée sur trois « couches » fonctionnelles :

- une couche de simulation, pouvant intégrer un ou plusieurs simulateurs différents du même système cible,
- une couche dédiée à la résolution de problème générant des simulations (par paramétrage des simulateurs) et analysant les résultats,
- une couche de manipulation des connaissances permettant par exemple de fusionner les solutions obtenues à l'aide des différents simulateurs.

Ce système est « alimenté » par des données de deux types :

- des données relatives au problème considéré (ou **données locales**), qui serviront à construire des règles de décisions et à paramétrer les simulations,
- des données relatives au système étudié (ou **données globales**) qui permettront de paramétrer et de valider les modèles.

Comme le laissait sous-entendre la première partie de cette thèse, nous nous focaliserons sur des simulateurs orientés agent et la résolution de problèmes inverses. L'aspect traitement de connaissances ne sera que très brièvement abordée³⁰. Ainsi, le chapitre 4 présente un modèle de résolution de problème inverse original. Dans ce chapitre, nous nous concentrerons sur les problèmes soulevés par la mise en œuvre de cette architecture dans ce contexte.

Le premier problème est lié à la validation d'une simulation et d'un ensemble de simulations basées sur des paramètres communs. Nous formalisons ce problème et proposerons une méthode de validation adaptée au champ d'application de cette thèse.

Le second problème est lié à l'efficacité du calcul des résultats de simulations, prélude nécessaire à sa validation ; cela peut être, et cela est souvent vrai dans le cas des SOA, particulièrement coûteux en temps. Or, la résolution de problèmes, notamment inverse, repose sur l'exécution et l'analyse de très nombreuses simulations. Nous avons ainsi travaillé sur des méthodes d'observation de SOA non intrusives, *i.e.*, pour lesquelles il n'a pas été nécessaire de modifier la plate-forme de simulation utilisée, permettant de minimiser le temps de calcul.

3.2 Validation de simulations

L'un des raisons pour lesquelles on réalise une simulation est sa validation. Souvent, on cherche à déterminer si la valeur d'une propriété observée p de la simulation reste dans un domaine de validité. Ainsi, on évalue la condition suivante :

$$obs_p(\mathcal{A})(t) \in val_p(t), \quad (3.1)$$

où obs_p est la fonction d'observation de p et val_p représente le domaine de validité de p en fonction du temps. On peut distinguer deux cas limites :

- on n'évalue cette condition qu'à un horizon de temps donné, *e.g.*, à la fin de la simulation,
- on évalue cette condition durant toute la simulation (on s'intéresse à la dynamique de la simulation).

Le second cas est plus riche en information mais nécessite d'observer la simulation en ligne.

Ainsi, du point de vue de sa validation, une simulation est caractérisée par un ensemble discret et fini de tuples :

$$\mathcal{P} = \{\langle p_1, obs_{p_1}, val_{p_1}, t_{p_1} \rangle, \dots, \langle p_i, obs_{p_i}, val_{p_i}, t_{p_i} \rangle, \dots, \langle p_m, obs_{p_m}, val_{p_m}, t_{p_m} \rangle\}, \quad (3.2)$$

où pour $0 \leq i \leq m$, p_i représente l'identifiant de la propriété, obs_{p_i} sa fonction d'observation, val_{p_i} la fonction permettant de définir son domaine de validité et t_{p_i} l'ensemble des instants auxquels elle doit être observée.

³⁰Les travaux relatifs à cette couche fonctionnelle sont réalisés dans le cadre d'une thèse menée parallèlement à celle-ci.

3.2.1 Mesure de la validité d'une simulation

De nombreuses mesures de la validité d'une simulation peuvent être définies. Pour des raisons liées à l'application de notre travail, nous considérons ici une mesure particulièrement stricte. Soit val_s la fonction permettant de déterminer la validité d'une simulation :

$$val_s : \mathcal{P} \rightarrow \{0, 1\}. \quad (3.3)$$

Si val_s retourne 1, la simulation est considérée comme valide, invalide sinon. Ici val_s est définie par des règles de décision (si *condition* alors *décision*) génériques :

$$val_s(\mathcal{P}) = \begin{cases} 1 & \text{si } \forall \langle p_i, obs_{p_i}, val_{p_i}, t_{p_i} \rangle \in \mathcal{P}, \forall t \in t_{p_i}, obs_{p_i}(t) \in val_{p_i}(t) \\ 0 & \text{sinon.} \end{cases} \quad (3.4)$$

En d'autres termes, une simulation est valide si aucune des règles permettant de définir sa validité n'est violée.

3.2.2 Mesure de la validité d'un ensemble de paramètres

Dans le cas d'un modèle déterministe, la définition précédente se suffit à elle-même pour déterminer la validité d'un ensemble de paramètres. Dans le cas d'un modèle stochastique, un ensemble de simulations S devra être exécuté avec le même ensemble de paramètres pour en déterminer la validité. Soit val_p , la fonction permettant de déterminer la validité d'un ensemble de paramètres :

$$val_p : S \rightarrow [0, 1], \quad (3.5)$$

val_p est calculé en utilisant la moyenne arithmétique des mesures de validité des différentes simulations :

$$val_p(S) = \frac{\sum_{s \in S} val_s(\mathcal{P})}{|S|}. \quad (3.6)$$

3.3 Observation de simulations orientées agent

3.3.1 Problématique générale

Les avancées à la fois théoriques et pratiques dans le domaine de la SOA ont permis la modélisation de systèmes toujours plus complexes, *i.e.*, formés de composants en interaction toujours plus nombreux. L'exécution et l'analyse des systèmes artificiels résultants posent de nombreux problèmes d'ingénierie. Ainsi, un problème classique lié aux SOA et plus généralement aux SMA est leur observation. Ces systèmes étant composés de nombreuses entités indépendantes, il est souvent nécessaire d'agréger leurs états pour fournir à l'utilisateur un ou plusieurs indicateurs de l'état du système dans sa globalité. On trouve dans la littérature relative au problème de l'observation en temps réel (ou monitoring) d'agents deux familles de méthodes :

- l'observation basée sur les rapports d'agents (report-based monitoring) suppose que chaque agent du système envoie un message contenant les valeurs des propriétés à observer à un « agent superviseur » qui agrégera ces données (Wilkins *et al.*, 2003),

- l’observation indirecte consiste à écouter les conversations des agents (ou plus généralement toute conséquence facilement observable de l’activité des agents) pour inférer les valeurs des propriétés à observer (Kaminka *et al.*, 2002).

Kaminka *et al.* (2002) soulignent que, dans le cadre des SMA, l’observation indirecte est intéressante comparativement à l’observation directe pour au moins deux raisons :

1. il n’est pas nécessaire de modifier les agents pour la mettre en œuvre,
2. son impact sur le temps de calcul est faible.

En revanche, il n’est pas toujours possible de mettre en œuvre une telle méthode. En effet, rien ne garantit qu’une propriété du système puisse être observée indirectement. Notons cependant que l’observation des interactions, et en particulier des conversations entre agents d’une simulation, présente un intérêt intrinsèque, notamment pour comprendre, déboguer ou encore valider une SOA (Botía *et al.*, 2006). Des outils basés sur ce principe ont ainsi été développés (Ralambondrainy *et al.*, 2007).

Les objectifs de l’observation d’agents dans les SMA (au sens général) et les SOA peuvent être, si l’on se réfère à la littérature, très différents. Cela est dû aux caractéristiques des agents (notamment leur capacité à élaborer collectivement des plans), du médium de communication (e.g., l’accessibilité), du système (physiquement distribué ou non) et des observations elles-mêmes. Ainsi, dans le cas des SOA qui nous occupe ici, il est possible de distinguer certains de ces buts, liés aux différentes étapes de développement du modèle (Shannon, 1998). Une simulation sera ainsi observée afin de fournir à l’utilisateur, ou à un système automatisé d’analyse de simulations, des informations sur l’état du système pour :

- vérifier le simulateur,
- valider le modèle,
- valider une simulation ou certains paramètres de la simulation.

Par ailleurs, les méthodes d’observation sont elles aussi différentes. Les méthodes d’observation de SMA citées plus haut, ont été essentiellement conçues pour des systèmes composés d’agents fonctionnellement autonomes (aucun système externe à l’agent ne peut accéder à sa structure ou aux valeurs de ses propriétés). Dans le cas des SOA, il ne semble pas aberrant que des agents extérieurs à la simulation puissent accéder aux valeurs des propriétés des agents de la simulation. En d’autres termes, si les agents de la simulation sont fonctionnellement autonomes entre eux, rien n’impose qu’ils ne le soient des systèmes d’observation et d’exécution de la simulation. Ainsi, les plates-formes de SOA fournissent généralement des outils d’observation fondés sur la notion de « sondage » (probing), *i.e.*, d’accès direct aux propriétés à observer : un agent observateur ayant pour rôle le calcul d’un observable sonde l’ensemble des agents du système pour déterminer les agents devant être observés et agréger les états de ceux-ci.

L’utilisation de ces outils a bien évidemment un coût sur le temps de calcul de la simulation. Leur impact dépend de la complexité de l’algorithme d’agrégation, de la fréquence des observations et, selon la nature de l’observation, du nombre d’agents, de la complexité des organisations et des interactions ou encore de la taille de l’environnement. Il n’existe pas, à notre connaissance, de travail existant relatif à ce problème³¹.

³¹Les publications relatives à l’observation de SOA, e.g., Payet *et al.* (2005), traitent de la représentation des informations, non de leur calcul.

Dans cette section, basée sur Morvan *et al.* (2009c), nous introduisons deux méthodes d'observation directe de SOA et analysons de manière comparative leurs performances sur un cas d'étude simple. La première méthode, basée sur le concept d'auto-observation, consiste à intégrer des fonctions d'observation locale dans le cycle de vie des agents de la simulation. La seconde, largement utilisée dans les sciences humaines (e.g. le marketing, la sociologie ou encore la démographie), est basée sur la théorie des sondages. Toutes deux sont non-intrusives, dans le sens où la plate-forme n'a pas été modifiée pour les mettre en œuvre. Nous nous focalisons sur les simulations composées d'un très grand nombre d'agents, *i.e.*, pour lesquelles le temps de calcul nécessaire à l'observation de la simulation devient grand par rapport à celui de la simulation elle-même.

3.3.2 Description du cas d'étude

Nous considérons des SOA particulièrement simples, inspirées du *StupidModel* (Railsback *et al.*, 2005), afin de mettre en œuvre les méthodes décrites dans les sections suivantes. Ces SOA sont caractérisées par :

- un environnement toroïdal $\mathcal{E}$ à 2 dimensions composé de $100 \cdot 100$ cellules carrées (avec le voisinage de Moore³²),
- un ensemble $\mathcal{A}$ d'agents (on note $|\mathcal{A}| = N$ se déplaçant aléatoirement dans cet environnement durant 100 pas de temps,
- l'observation à chaque pas de simulation du nombre d'agents Z présents dans une zone de l'environnement $\mathcal{Z} \subseteq \mathcal{E}$, définie arbitrairement pour chaque simulation.

L'efficacité des différentes méthodes d'observation sera analysée en fonction de N et de $E(Z)$, l'espérance mathématique du nombre d'agents présents dans $\mathcal{Z}$. Nous considérerons ici que

$$E(Z) = N \cdot \frac{\text{surface de } \mathcal{Z}}{\text{surface de } \mathcal{E}}. \quad (3.7)$$

Ces simulations ont été implémentées sur la plate-forme MadKit/TurtleKit (Ferber et Gutknecht, 1998; Michel *et al.*, 2005). Chaque simulation est répétée 10 fois. Cependant, la dispersion des temps de calcul s'est révélée très faible. À titre d'exemple, la figure 3.2 compare les temps CPU nécessaires au calcul de telles simulations, pour $E(Z) = N/2$, en fonction de N , lorsqu'on les observe à l'aide de l'outil fourni dans la plate-forme (la méthode d'observation sur laquelle est basée cet outil, fondée sur l'observation de l'ensemble des agents, sera appelée « méthode naïve » par la suite) et lorsque l'on ne les observe pas.

Dans les sections suivantes, nous proposons des idées pour réduire l'impact des méthodes d'observation directe sur le temps de calcul des simulations.

3.3.3 Filtrage des informations

Dans le cas d'étude considéré, comme dans de nombreux cas, il n'est pas nécessaire de sonder tous les agents pour calculer la valeur de la propriété observée. Formellement, on

³²Dans le voisinage de Moore, chaque cellule possède 9 voisines : elle-même ainsi que les 8 cellules les plus proches.

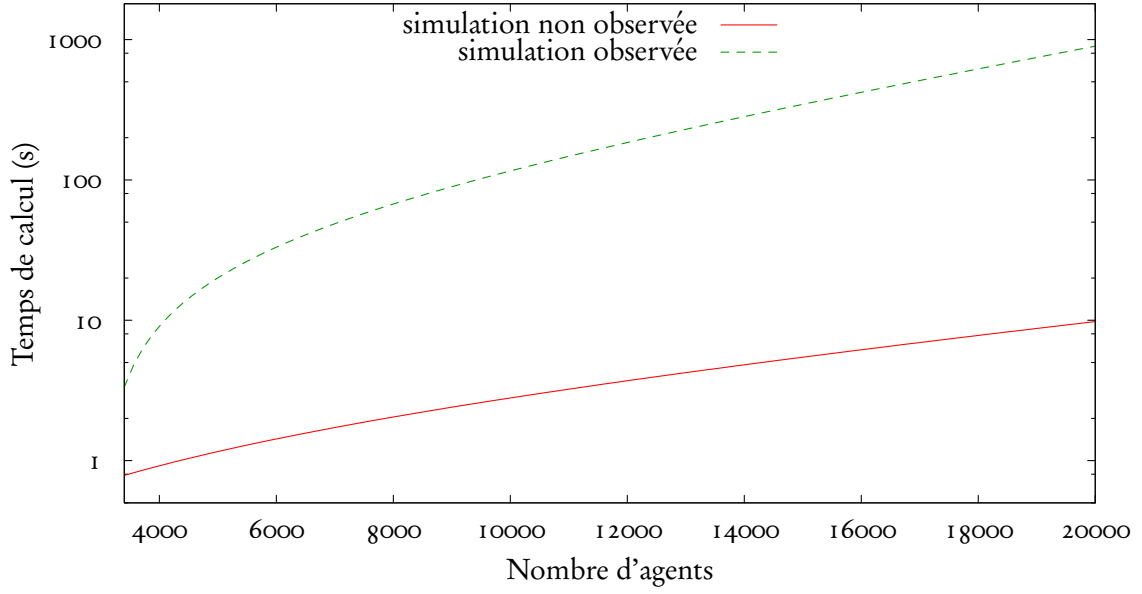


Fig. 3.2 – Temps CPU moyen utilisé pour le calcul de différentes simulations avec et sans observation pour $E(Z) = N/2$

considère une fonction d'observation

$$obs : 2^{\mathcal{A}} \rightarrow \mathcal{I}, \quad (3.8)$$

où $\mathcal{I}$ représente l'ensemble des valeurs pouvant être observées. On s'intéresse au sous-ensemble d'agents minimal $\mathcal{A}' \subseteq \mathcal{A}$ défini par un ensemble de contraintes (e.g., dans l'exemple précédent on considère une contrainte unique relative à la position de l'agent dans l'environnement) permettant de calculer obs correctement. En d'autres termes, on cherche à déterminer $\mathcal{A}'$ tel que

$$\begin{aligned} obs'(\mathcal{A}') &= obs(\mathcal{A}) \text{ et} \\ \nexists \mathcal{A}'' \subset \mathcal{A}' \mid obs'(\mathcal{A}'') &= obs(\mathcal{A}), \end{aligned} \quad (3.9)$$

où obs' est une fonction d'observation « simplifiée », i.e., pouvant être calculée plus rapidement que obs car spécifique à $\mathcal{A}'$:

$$\begin{aligned} \forall \mathcal{A}'' \neq \mathcal{A}' \subseteq \mathcal{A}, \quad obs'(\mathcal{A}') &= obs(\mathcal{A}') \\ \text{mais } obs'(\mathcal{A}'') &\neq obs(\mathcal{A}''). \end{aligned} \quad (3.10)$$

Ainsi, dans notre exemple, la fonction d'observation $obs(\mathcal{A})$ filtre les agents de $\mathcal{A}$ afin de ne compter que ceux situés dans $\mathcal{Z}$, alors que $obs'(\mathcal{A}')$ ne fait que compter les agents de $\mathcal{A}'$.

Afin de pouvoir utiliser obs' , il est nécessaire de considérer une fonction de filtrage capable d'identifier le sous-ensemble $\mathcal{A}'$ (dans notre exemple ce sous-ensemble ne contient bien évidemment que les agents situés dans $\mathcal{Z}$) :

$$\begin{aligned} filtre : 2^{\mathcal{A}} &\rightarrow 2^{\mathcal{A}} \mid \forall \mathcal{A}', \mathcal{A}'' \subseteq \mathcal{A}, \\ \text{si } filtre(\mathcal{A}') &= \mathcal{A}'', \text{ alors } \mathcal{A}'' \subseteq \mathcal{A}'. \end{aligned} \quad (3.11)$$

L'objectif est de définir et d'implémenter une telle fonction, tel que le temps nécessaire au calcul de l'observation soit réduit. Le présent travail repose sur des implémentations existantes d'outils d'observation. Nous nous intéresserons donc à la complexité empirique moyenne, *i.e.*, le temps CPU moyen observé pour l'exécution de simulations implémentant les différentes méthodes d'observation. Cette complexité, notée $\mathcal{C}$, dépend du nombre d'agents, $|\mathcal{A}|$, et de l'espérance du cardinal du sous-ensemble $\mathcal{A}'$ de $\mathcal{A}$ calculé par *filter* (*i.e.*, dans notre exemple de N et de $E(Z)$). On cherche donc à déterminer des méthodes de filtrage telles que :

$$\mathcal{C}(obs'(filter(\mathcal{A}))) < \mathcal{C}(obs(\mathcal{A})). \quad (3.12)$$

3.3.3.1 Auto-observation des agents

L'idée derrière cette méthode est d'implémenter la fonction de filtrage dans les agents de la simulation eux-mêmes. L'utilisation d'un modèle organisationnel permet alors d'identifier l'ensemble des agents devant être observés. Ainsi, un groupe est défini comme *l'ensemble des agents contenant les informations nécessaires et suffisantes au calcul d'une propriété*. En d'autres termes, un groupe définit pour une observation particulière, l'ensemble minimal des agents nécessaire à son calcul. Les agents s'auto-observent pour déterminer s'ils doivent rejoindre, quitter ou rester dans un groupe.

Un ensemble de règles (caractérisant *filter*) spécifié par le modélisateur/observateur, déterminant si un agent doit être observé pour calculer la valeur d'une propriété particulière, est évalué à chaque pas de simulation. Ainsi, dans notre cas d'étude, on considère le groupe G , contenant l'ensemble des agents présents dans $\mathcal{Z}$ et on associe à chaque agent l'ensemble de règles suivant :

- si l'agent se trouve dans $\mathcal{Z}$ et que l'agent n'appartient pas à G , alors le rejoindre,
- si l'agent ne se trouve pas dans $\mathcal{Z}$ et que l'agent appartient à G , alors le quitter.

Le système d'observation (obs') ne sondera que les agents appartenant au groupe G . La figure 3.3 représente la différence entre les temps de calcul de simulations observées à l'aide des méthodes basées sur l'auto-observation des agents et naïve en fonction du nombre d'agents dans la simulation N et du taux moyen d'agents observés $E(Z)/N$. La ligne en pointillée représente la ligne de niveau 0, *i.e.*, lorsqu'il n'y a pas de différence entre les deux méthodes d'observation. Ainsi, la zone située en dessous de cette ligne correspond aux cas où la méthode basée sur l'auto-observation des agents est la plus efficace. À l'inverse, la zone située au dessus de cette ligne correspond aux cas où la méthode naïve est la plus efficace.

3.3.3.2 Échantillonnage de la population

En réécrivant l'équation 3.9 de la manière suivante :

$$\begin{aligned} & obs(\mathcal{A}') \simeq obs(\mathcal{A}) \text{ et} \\ & \nexists \mathcal{A}'' \subset \mathcal{A}' \mid obs(\mathcal{A}'') \simeq obs(\mathcal{A}), \end{aligned} \quad (3.13)$$

i.e., en autorisant des observations approximatives, il devient possible de filtrer la population d'agents à observer sur des critères statistiques. On ne considère donc plus de fonction d'observation spécifique, les agents retournés par la fonction de filtrage n'étant pas nécessairement les agents à observer.

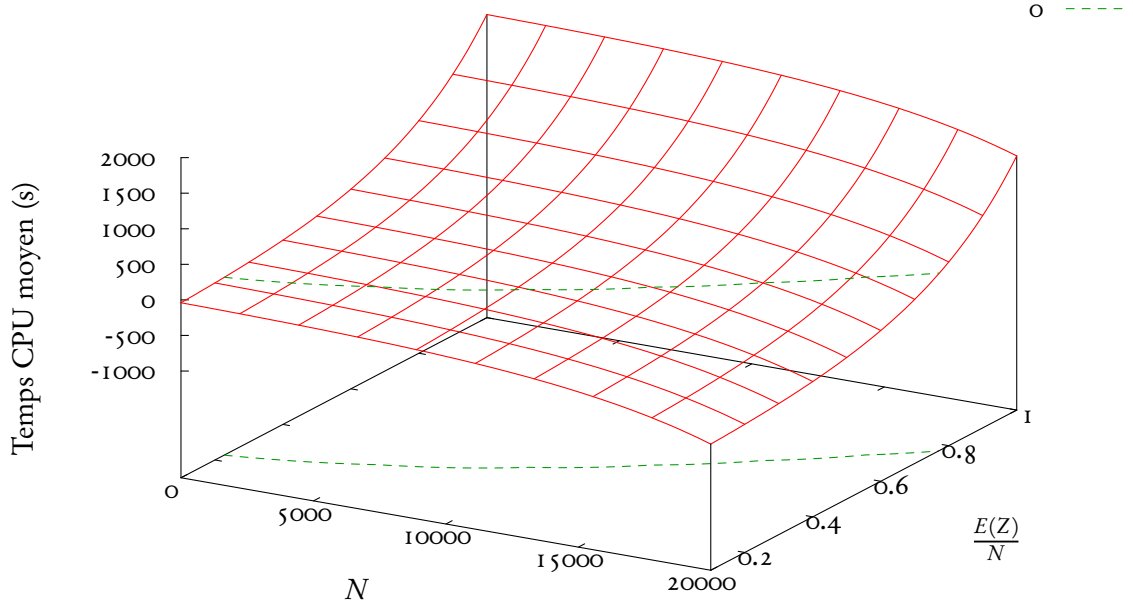


Fig. 3.3 – Différence entre les temps CPU moyens (s) de simulations observées à l'aide des méthodes basée sur l'auto-observation des agents et naïve

La théorie des sondages³³ fournit un cadre formel permettant de déterminer la taille de l'échantillon à observer. Considérons le cas de l'exemple précédent et notons n le nombre d'agents dans l'échantillon d'agents observés. Un estimateur de Z , noté $\hat{Z}$, peut ainsi être construit à partir de cet échantillon. E.g., l'estimateur de Horvitz-Thompson est défini par

$$\hat{Z} = \frac{N}{n} Z_n, \quad (3.14)$$

où Z_n représente le nombre d'agents de l'échantillon présents dans $\mathcal{Z}$. La qualité de cet estimateur, pour un échantillon donné, dépend de la variance V de $\hat{Z}$. Ici,

$$V(\hat{Z}) = \left(1 - \frac{n}{N}\right) \cdot \frac{S^2}{n} \simeq \frac{S^2}{n} \text{ si } N \gg n. \quad (3.15)$$

S^2 peut être estimé par :

$$S^2 \simeq \left(1 - \frac{E(Z)}{N}\right) \cdot \frac{E(Z)}{N}. \quad (3.16)$$

La précision absolue de $\hat{Z}$ est estimée par la demi-longueur de l'intervalle de confiance à 95% : ϵ . Si on suppose que la loi de $E(Z)$ est une loi normale alors :

$$\epsilon \simeq 2 \cdot \sqrt{V(\hat{Z})}. \quad (3.17)$$

³³Les résultats de cette théorie exposés ici ne seront pas démontrés, le lecteur intéressé peut se référer à Dussaix et Grosbras (1996) pour une présentation exhaustive.

La taille de l'échantillon est déterminée à partir de l'erreur absolue maximale d tolérée pour l'observation. On cherche donc à minimiser n sous la contrainte $\epsilon \leq d$. Ainsi :

$$n = \frac{1}{\frac{d^2}{4S^2} + \frac{1}{N}}. \quad (3.18)$$

La figure 3.4 montre le nombre d'agents à observer pour une erreur absolue tolérée de 0.08 et pour $E(Z) = N/2$. L'impact sur le temps de calcul est présenté dans la figure 3.5. La sémantique est la même que pour la figure 3.3 : la zone située à droite de la ligne en pointillés correspond aux cas où la méthode basée sur l'échantillonnage de la population d'agents est la plus efficace. À l'inverse, la zone située à gauche de cette ligne correspond aux cas où la méthode naïve est la plus efficace.

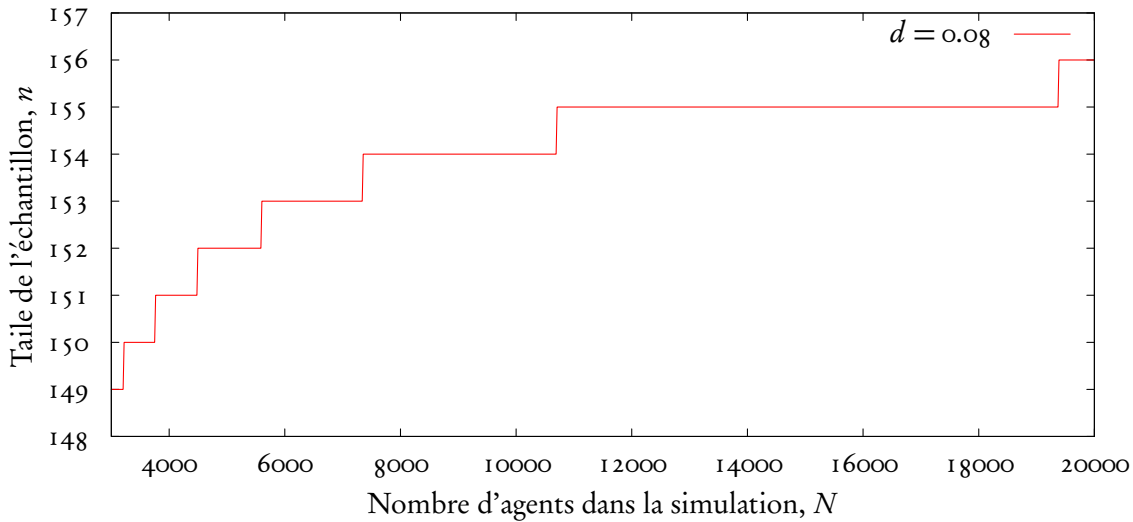


Fig. 3.4 – Taille de l'échantillon à observer, n , pour $d = 0.08$ et $E(Z) = N/2$

3.3.3.3 Bilan

Pour conclure, la figure 3.6 synthétise les résultats précédents en identifiant les conditions dans lesquelles il est préférable, en termes de temps de calcul, d'utiliser une méthode d'observation particulière. Bien évidemment, ces résultats sont propres au cas d'étude considéré ; ils permettent cependant de mettre en évidence que le choix d'une méthode d'observation n'a rien de trivial et que les performances de ces méthodes doivent être analysées sur un ensemble de simulations préalablement à l'exploitation du modèle. De plus, il semble très probable que la forme générale de la figure 3.6 soit, indépendamment de l'échelle, invariante quel que soit le problème considéré.

De manière générale, lorsque la proportion d'agents à observer est faible ou lorsque le nombre d'agents dans la simulation est important, la méthode naïve n'est pas la plus efficace et il devient alors intéressant de mettre en œuvre des méthodes plus évoluées comme celles présentées ici.

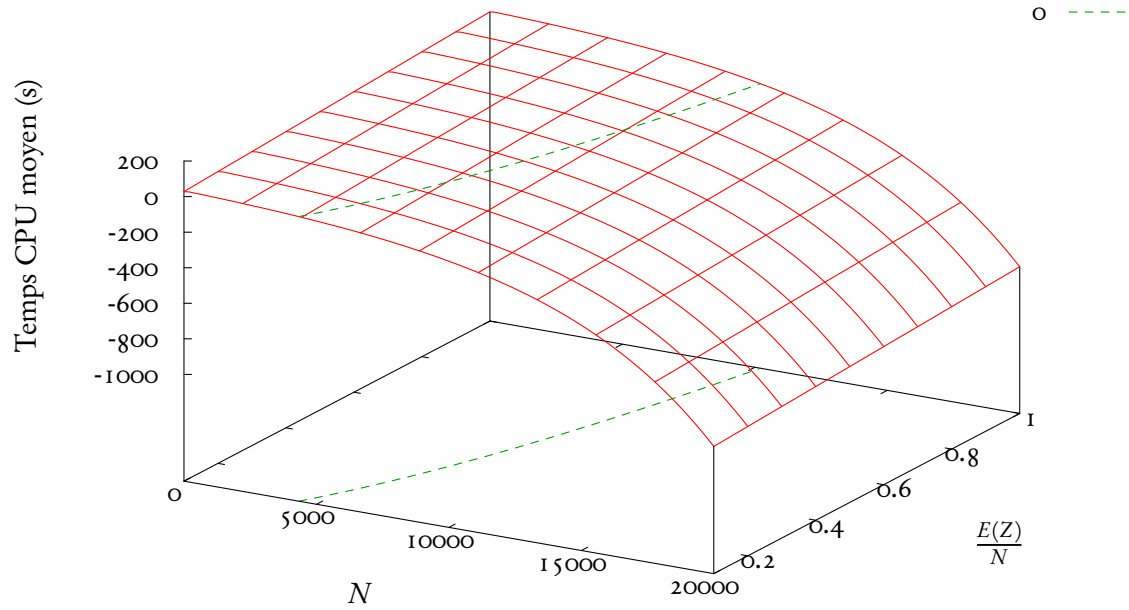


Fig. 3.5 – Différence entre les temps CPU moyens (s) de simulations observées à l'aide des méthodes basée sur l'échantillonnage de la population d'agents (avec $d = 0.08$) et naïve

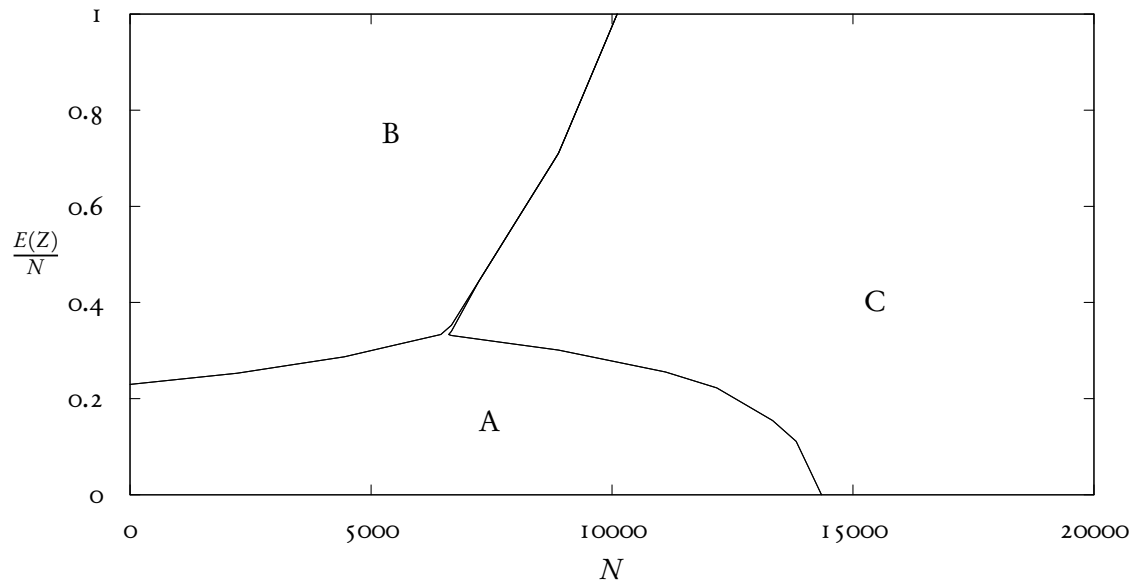


Fig. 3.6 – Conditions dans lesquelles les méthodes d'observations sont les plus efficaces ; A : auto-observation, B : naïve, C : échantillonnage

3.4 Résumé du chapitre

Dans ce chapitre nous avons présenté une architecture de système d'aide à la décision fondé sur l'exécution de simulations puis défini des métriques pour calculer la validité d'une simulation et d'un ensemble de paramètres utilisé pour exécuter des simulations. Ces métriques sont particulièrement strictes : pour être valide, une simulation ne doit violer aucune règle définie par l'expert ; de même, pour être totalement valide, un ensemble de paramètres doit permettre de valider plusieurs simulations. L'utilisation de ces métriques suppose de pouvoir observer, parfois en temps réel, les simulations. Or, observer une SOA est une tâche coûteuse en temps de calcul. Ainsi, nous avons proposé des méthodes d'observation permettant, dans certains cas identifiés sur un problème jouet, d'optimiser ce processus.

4

Modélisation et résolution de problèmes inverses basées sur des critères d'abduction

Sommaire

4.1	Description du modèle	52
4.1.1	Définitions	52
4.1.2	Objectifs de la résolution	53
4.1.3	Algorithme de résolution	53
4.2	Heuristiques	55
4.2.1	Critères de choix	55
4.2.2	Définition des heuristiques	57
4.3	Résultats expérimentaux	57
4.4	Résumé du chapitre	61

Les travaux présentés dans ce chapitre ont été motivés par différents facteurs. Tout d'abord, nous pensons que le problème inverse tel qu'il a été décrit au chapitre 2 n'est qu'une instance d'un problème plus général. Nous verrons que selon le contexte, il peut être pertinent de s'intéresser à d'autres instances de ce problème général. De plus, la difficulté majeure associée au problème inverse est la taille de l'espace de recherche. Le problème des ressources nécessaires à la simulation du modèle causal n'est pas considéré. En d'autres termes, le coût associé à la validation d'une hypothèse (c'est-à-dire d'un paramétrage du modèle) est considéré comme négligeable.

Les sections 4.1 à 4.3 sont basées sur Morvan *et al.* (2008a).

4.1 Description du modèle

4.1.1 Définitions

Soit Ω un ensemble d'observations réalisables sur un système (ou dans la cas d'un système simulé par un modèle m , l'ensemble des données pouvant être prédites par m) et Φ un ensemble d'hypothèses (*i.e.*, un ensemble d'explications ou paramétrages possibles pour m) pour Ω (cf. équations 2.1 et 2.3). Nous supposons qu'un espace métrique (Φ, d^Φ) peut être défini, en d'autres termes, qu'une métrique d^Φ peut être utilisée pour calculer la distance entre deux éléments de Φ .

Une mesure de cohérence entre une hypothèse ϕ de Φ et un ensemble d'observations $\Omega^* \subseteq \Omega$ est calculé à l'aide d'un modèle

$$m : \Phi \rightarrow 2^\Omega, \quad (4.1)$$

permettant d'associer une hypothèse à Ω^* , et d'une fonction

$$c : \Phi \times 2^\Omega \rightarrow [0, 1]. \quad (4.2)$$

Cette mesure de cohérence est notée c_{m,ϕ,Ω^*} et est calculée de la façon suivante :

$$c_{m,\phi,\Omega^*} = c(m(\phi), \Omega^*). \quad (4.3)$$

Le calcul de c_{m,ϕ,Ω^*} suppose l'utilisation de ressources. On associe donc au calcul d'une mesure de cohérence par un modèle m et une fonction c un coût.

Une hypothèse ϕ de Φ est validée pour un modèle m et une fonction c si et seulement si c_{m,ϕ,Ω^*} est connue. Soit $\Phi^v \subseteq \Phi$ l'ensemble des hypothèses validées et $\mathcal{C}_{m,\Phi^v,\Omega^*} : \Phi \times [0, 1]$ l'ensemble solution calculé par m et c :

$$\mathcal{C}_{m,\Phi^v,\Omega^*} = \bigcup_{\forall \phi \in \Phi^v} (\phi, c_{m,\phi,\Omega^*}). \quad (4.4)$$

De même, un espace métrique $(\mathcal{C}_{m,\Phi^v,\Omega^*}, d^\mathcal{C})$ est défini. Une mesure de cohérence estimée, notée $\widehat{c_{m,\phi,\Omega^*}}$, est associée à chaque hypothèse non validée, *i.e.*, à chaque ϕ de $\Phi - \Phi^v$. Cette mesure est estimée à partir de $\mathcal{C}_{m,\Phi^v,\Omega^*}$ par interpolation. L'ensemble solution estimé ainsi construit est défini comme suit :

$$\widehat{\mathcal{C}_{m,\Phi,\Omega^*}} = \bigcup_{\forall \phi \in \Phi} \begin{cases} (\phi, c_{m,\phi,\Omega^*}) & \text{si } \phi \in \Phi^v \\ (\phi, \widehat{c_{m,\phi,\Omega^*}}) & \text{sinon.} \end{cases} \quad (4.5)$$

Pour ne pas alourdir la notation et lorsque cela ne prête pas à confusion, les espaces métriques seront notés comme les ensembles correspondants. Durant le processus de raisonnement, les hypothèses à valider sont choisies itérativement par une méthode de sélection

$$\gamma : \widehat{\mathcal{C}_{m,\Phi,\Omega^*}} \rightarrow \Phi - \Phi^v. \quad (4.6)$$

Une mesure ϵ peut être définie pour déterminer la différence entre n ensembles solutions, typiquement entre un ensemble solution et l'ensemble solution estimé correspondant :

$$\epsilon : 2^{\mathcal{C}^n} \rightarrow \mathbb{R}. \quad (4.7)$$

Il est nécessaire de connaître l'ensemble résultat pour utiliser une telle mesure. Ainsi, elle est utile pour évaluer une implémentation de la méthode de résolution, et en particulier l'heuristique de choix d'hypothèses γ , non une solution particulière.

4.1.2 Objectifs de la résolution

Il est généralement admis que l'objectif d'un raisonnement abductif ou d'un problème inverse est la détermination de la meilleure explication possible à un ensemble d'observations. Cependant, dans un contexte d'aide à la décision, cette définition devrait être étendue.

Définition 3 (Problème inverse étendu) *L'objectif du problème inverse étendu est, étant donné un modèle m , une fonction de validation c et un ensemble d'observations $\Omega^* \subseteq \Omega$ réalisées sur un système source et un ensemble d'hypothèses $\Phi^* \subseteq \Phi$, de déterminer un ensemble solution estimé $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$ tel que*

$$\epsilon(\mathcal{C}_{m,\Phi^*,\Omega^*}, \widehat{\mathcal{C}_{m,\Phi,\Omega^*}}) \quad (4.8)$$

soit minimisé sous la contrainte C .

Φ^* représente le sous-ensemble des hypothèses pertinentes pour évaluer la qualité de la solution. Ainsi, l'objectif du problème dépend de la manière dont cet ensemble est défini. Il semble naturel de le faire à l'aide de contraintes. Dans le tableau 4.1, quatre objectifs nous semblant naturels sont identifiés. Bien sûr de nombreux autres objectifs pourraient être définis. Ainsi, dans sa définition la plus générale, la résolution d'un problème inverse peut être vue comme un processus d'identification (objectif *id.*), à l'inverse la définition de l'objectif *pr.* correspond à la définition classique du problème (cf. définition 2). Des métriques peuvent être définies pour déterminer comment les différents objectifs ont été atteints et ainsi la qualité de la solution. La tableau 4.2 présente, pour chaque objectif o , une métrique de qualité $q_{m,o}$ basée sur la « déviation de la valeur quadratique moyenne³⁴ » (RMSD) entre les ensembles solution réel et estimé.

4.1.3 Algorithme de résolution

L'algorithme 1 calcule un ensemble solution estimé, $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$, à l'aide d'un modèle m et d'une fonction de cohérence c . Cet ensemble solution peut ainsi être utilisé comme entrée d'un algorithme de traitement de l'information, e.g., pour fusionner plusieurs ensembles solutions.

L'arrêt de l'algorithme 1 dépend de l'évaluation d'une condition C . Cette condition peut porter sur le calcul de la solution ou sur la solution elle-même. Dans l'implémentation présentée dans le chapitre 7, la condition suivante est utilisée :

$$C := \left(\sum_{\forall r \in \mathcal{C}_{m,\Phi^*,\Omega^*}} \text{cost}(r) \right) \leq \text{maxCost}, \quad (4.9)$$

³⁴En anglais : Root mean square deviation

où $cost$ représente le coût (e.g., le temps de calcul) nécessaire pour sélectionner et valider une hypothèse et $maxCost$ est le coût maximal alloué à l'algorithme pour retourner une solution. Bien évidemment, l'algorithme se termine si toutes les hypothèses ϕ de Φ ont été validées. En utilisant une condition du type

$$C := c_{m,\phi,\Omega^*} \geq \mu, \quad (4.10)$$

où μ est un seuil de validité, l'algorithme 1 serait alors une forme du modèle hypothético-déductif (Lipton, 2004, ch. 4).

Cet algorithme est très simple et très général. Ainsi, il ne fait pas référence à l'objectif du problème ; cela dépend du choix de l'heuristique γ . Dans la section suivante, des idées permettant de construire une telle heuristique sont présentées. Nous nous concentrerons sur les objectifs *el.*, *prs.* et *id.* : pour l'objectif *pr.*, des heuristiques utilisées pour les processus d'optimisation (e.g., la recherche tabou ou le recuit-simulé) peuvent être utilisées (Pardalos, 2009). De plus, nous nous intéressons plus particulièrement aux cas pour lesquels l'ensemble d'hypothèses n'est pas forcément très grand mais le coût de validation d'une hypothèse est important.

Objectif	Définition	Contrainte associée à Φ^*
<i>pr.</i>	Déterminer les hypothèses les plus probables	$\forall \phi \in \Phi, \phi \in \Phi^*$ si et seulement si $c_{m,\phi,\Omega^*} = sup(c)$
<i>el.</i>	Éliminer les hypothèses improbables	$\forall \phi \in \Phi, \phi \in \Phi^*$ si et seulement si $c_{m,\phi,\Omega^*} = 0$
<i>prs.</i>	Déterminer les mesures de cohérence associées à toutes les hypothèses probables	$\forall \phi \in \Phi, \phi \in \Phi^*$ si et seulement si $c_{m,\phi,\Omega^*} > 0$
<i>id.</i>	Déterminer les mesures de cohérence associées à toutes les hypothèses	$\Phi^* = \Phi$

Tab. 4.1 – Objectifs d'un problème inverse

Objectif o	Métrique de qualité
<i>pr.</i>	$\forall \phi \in \Phi \mid c_{m,\phi,\Omega^*} = max(\mathcal{C}_{m,\Phi,\Omega}),$
<i>el.</i>	$\forall \phi \in \Phi \mid c_{m,\phi,\Omega^*} = 0,$
<i>prs.</i>	$\forall \phi \in \Phi \mid c_{m,\phi,\Omega^*} > 0,$
<i>id.</i>	$\forall \phi \in \Phi,$

$$q_{m,o} = \sqrt{\frac{\sum (\widehat{c_{m,\phi,\Omega^*}} - c_{m,\phi,\Omega^*})^2}{|\mathcal{C}_{m,\Phi,\Omega}|}}$$

Tab. 4.2 – Métriques de qualité associées aux objectifs d'un problème inverse

Algorithme 1 algorithme de résolution indirecte d'un problème inverse

Entrée: Un ensemble d'observations Ω **Entrée:** Une condition d'arrêt C **Entrée:** Une heuristique γ **Sortie:** Un ensemble solution estimé $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$

```

1: Déterminer  $\Phi$ 
2:  $\Phi^v \leftarrow \emptyset$ 
3:  $\mathcal{C}_{m,\Phi^v,\Omega^*} \leftarrow \emptyset$ 
4: Calculer  $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$ 
5: Tant que  $\neg(C \vee (\Phi = \Phi^v))$  Faire
6:    $\phi \leftarrow \gamma(\widehat{\mathcal{C}_{m,\Phi,\Omega^*}})$ 
7:   Calculer  $c_{m,\phi,\Omega^*}$ 
8:    $\Phi^v \leftarrow \Phi^v \cup \phi$ 
9:    $\mathcal{C}_{m,\Phi^v,\Omega^*} \leftarrow \mathcal{C}_{m,\Phi^v,\Omega^*} \cup (\phi, c_{m,\phi,\Omega^*})$ 
10:  Calculer  $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$ 
11: Fin tant que
12: Retourner  $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$ 

```

4.2 Heuristiques

4.2.1 Critères de choix

De nombreux critères pouvant aider un agent à choisir la meilleure hypothèse à valider ont été identifiés dans la littérature. Parmi ces derniers, la simplicité semble avoir été le plus utilisé par les informaticiens. La simplicité est généralement interprétée comme la simplicité logique. Pourtant, il paraît évident que pour un être humain la simplicité est un peu plus compliquée (Aliseda-Llera, 1998; Paul, 1993). Ici, nous considérons qu'une hypothèse ϕ est simple, s'il existe une méthode directe permettant d'associer ϕ à Ω . Plus formellement,

Définition 4 (Critère de simplicité) Une hypothèse ϕ est simple s'il existe un modèle rétrodictif m^{-1} tel que $m^{-1}(\Omega) = \phi$. De plus, si Φ est borné, ϕ peut être considérée comme simple si $\phi = \inf(\Phi)$ ou $\phi = \sup(\Phi)$.

Le critère de simplicité est utilisé pour prétraiter le problème. Ainsi, toutes les hypothèses reconnues comme simples seront validées prioritairement. D'autres critères comme le coût ou l'utilité ont par ailleurs été cités et peuvent être utilisés (Peirce, 1931, 1958; McGrew, 2003). Le critère de *coût* peut être défini informellement comme suit : l'hypothèse choisie devrait être, *ceteris paribus*, celle qui minimise le coût de validation. Ce critère peut être utilisé s'il est possible d'estimer (empiriquement ou sur des bases théoriques) la fonction de coût du modèle prédictif m , $cost_m : \Phi \rightarrow \mathbb{R}$. Cette fonction est définie formellement ci-dessous.

Définition 5 (Critère de coût) Soit $u_c : \Phi \rightarrow [0, 1]$ une fonction d'utilité pour le critère de coût. Pour tout ϕ de Φ

$$u_c(\phi) = 1 - \frac{cost_m(\phi)}{\max(cost_m)}. \quad (4.11)$$

Le critère d'utilité peut être interprété de différentes façons. Ici, nous le définissons informellement comme suit : l'hypothèse choisie devrait, *ceteris paribus*, celle qui maximise la connaissance de l'agent. Cela signifie que l'hypothèse choisie devrait être celle qui, une fois validée, maximise la qualité de la solution, relativement à une mesure de qualité. Bien évidemment, dans le cadre d'un problème réel, une telle hypothèse et les conséquences de sa validation sur la qualité de la solution ne peuvent être déterminées avec certitude. Cependant, les métriques de Φ ou de $\mathcal{C}_{m,\Phi^v,\Omega^*}$ peuvent être utilisées pour déterminer des fonctions d'utilité pour ce critère.

Afin de construire de telles fonctions, la notion de consécutivité est définie.

Définition 6 (Consécutivité) Soit (E, d) un espace métrique. Deux éléments e_i et e_j de E sont consécutifs dans (E, d) si et seulement si il n'existe pas d'élément e appartenant à E tel que

$$d(e, e_i) < d(e_i, e_j) \text{ et } d(e, e_j) < d(e_i, e_j). \quad (4.12)$$

La notion de distance maximale entre deux éléments consécutifs est alors définie.

Définition 7 (Distance maximale entre deux éléments consécutifs) Soit (E, d) un espace métrique et $d_{\max}((E, d))$ la distance maximale entre deux éléments consécutifs de (E, d) .

Soit e_i et e_j deux éléments consécutifs de E , $d_{\max}((E, d)) = d(e_i, e_j)$ si et seulement si il n'existe pas d'éléments consécutifs e_k et e_l de (E, d) tels que $d(e_k, e_l) > d(e_i, e_j)$.

Ainsi, au moins deux fonctions d'utilité du critère d'utilité peuvent être définies en utilisant les métriques de Φ (définition 8) et de $\mathcal{C}_{m,\Phi,\Omega}$ (définition 9). Ces fonctions d'utilité sont notées respectivement u_u^Φ et $u_u^\mathcal{C}$.

Définition 8 (Critère d'utilité défini par Φ) Soit $u_u^\Phi : \Phi \rightarrow [0, 1]$ une fonction d'utilité du critère d'utilité. Pour tout ϕ appartenant à Φ

$$u_u^\Phi(\phi) = \begin{cases} 0 & \text{si } \phi \in \Phi^v \\ \frac{d^\Phi(\phi, \phi_1)}{d_{\max}^\Phi(\Phi^v)} & \text{sinon,} \end{cases} \quad (4.13)$$

où l'hypothèse $\phi_1 \neq \phi$ est la plus proche de ϕ dans $\Phi^v \cup \{\phi\}$, i.e., il n'existe pas d'hypothèse $\phi_2 \neq \phi$ dans $\Phi^v \cup \{\phi\}$ telle que $d^\Phi(\phi, \phi_2) < d^\Phi(\phi, \phi_1)$.

Définition 9 (Critère d'utilité défini par $\mathcal{C}_{m,\Phi,\Omega}$) Soit $u_u^\mathcal{C} : \Phi \rightarrow [0, 1]$ une fonction d'utilité du critère d'utilité. Pour tout $r = (\phi, c_{m,\phi,\Omega^*})$ de $\mathcal{C}_{m,\Phi,\Omega}$ (c_{m,ϕ,Ω^*} n'étant pas nécessairement connue),

$$u_u^\mathcal{C}(\phi) = \begin{cases} 0 & \text{si } \phi \in \Phi^v \\ \frac{d^\mathcal{C}(r, r_1)}{d_{\max}^\mathcal{C}(\mathcal{C}_{m,\Phi^v,\Omega^*})} & \text{sinon,} \end{cases} \quad (4.14)$$

où l'hypothèse $\phi_1 \neq \phi$ associée à r_1 est la plus proche de ϕ dans $\Phi^v \cup \{\phi\}$ (cf. définition 8).

Pour tout ϕ de Φ , si ϕ appartient à Φ^v , $u_u^\Phi(\phi) = u_u^\mathcal{C}(\phi) = 0$. C'est une propriété souhaitable car il est inutile de valider plusieurs fois une hypothèse avec le même modèle prédictif.

4.2.2 Définition des heuristiques

Un opérateur d'agrégation est utilisé pour déterminer des fonctions d'utilité globales u^Φ et $u^\mathcal{C}$, basées respectivement sur u_u^Φ et $u_u^\mathcal{C}$. À titre d'exemple, la somme pondérée est utilisée ici.

Définition 10 (Heuristiques de choix d'hypothèses) Soit $u^\Phi : \Phi \rightarrow [0, 1]$ une fonction représentant l'utilité de valider une hypothèse de Φ . Pour tout ϕ de Φ

$$u^\Phi(\phi) = \alpha_\Phi \cdot u_c(\phi) + \beta_\Phi \cdot u_u^\Phi(\phi), \quad (4.15)$$

avec $\alpha_\Phi + \beta_\Phi = 1$.

Soit $u^\mathcal{C} : \Phi \rightarrow [0, 1]$ une fonction représentant l'utilité de valider une hypothèse de Φ . Pour tout $r = (\phi, c_{m,\phi,\Omega^*})$ de $\mathcal{C}_{m,\Phi,\Omega}$

$$u^\mathcal{C}(\phi) = \alpha_\mathcal{C} \cdot u_c(\phi) + \beta_\mathcal{C} \cdot u_u^\mathcal{C}(\phi), \quad (4.16)$$

avec $\alpha_\mathcal{C} + \beta_\mathcal{C} = 1$.

Deux heuristiques γ_ϕ et $\gamma_\mathcal{C}$ peuvent ainsi être définies :

Définition 11 Soit $\gamma_\phi, \gamma_\mathcal{C} : \mathcal{C}_{m,\Phi,\Omega} \rightarrow \Phi - \Phi^v$ deux fonctions retournant une hypothèse à valider.

Pour tout $r = (\phi, c_{m,\phi,\Omega^*})$ appartenant à $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$, $\gamma_\phi(\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}) = r$ si et seulement si il n'existe pas un résultat $r' = (\phi', c'_{m,\phi,\Omega^*})$ appartenant à $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$ tel que $u^\phi(\phi') > u^\phi(\phi)$ et ϕ n'appartient pas à Φ^v .

Pour tout $r = (\phi, c_{m,\phi,\Omega^*})$ appartenant à $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$, $\gamma_\mathcal{C}(\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}) = r$ si et seulement si il n'existe pas un résultat $r' = (\phi', c'_{m,\phi,\Omega^*})$ appartenant à $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$ tel que $u^\mathcal{C}(\phi') > u^\mathcal{C}(\phi)$ et ϕ n'appartient pas à Φ^v .

L'efficacité de ces heuristiques est évaluée sur un problème réel. Ce problème est décrit dans le chapitre 7. Les résultats sont présentés dans la section suivante. Notons qu'il est possible d'envisager des heuristiques « hybrides », i.e., basées sur une agrégation des différentes métriques. Cependant, les résultats expérimentaux obtenus n'étant pas satisfaisants (leurs performances sont toujours inférieures à celles des heuristiques γ_ϕ et $\gamma_\mathcal{C}$ pour les différents objectifs définis dans le tableau 4.1), elles ne sont pas présentées.

4.3 Résultats expérimentaux

Pour évaluer la pertinence de ces heuristiques pour un objectif donné, des ensembles solutions « réalistes », i.e., ayant la même forme générale les ensembles solutions issus de l'application présentée dans la partie suivante (figure 4.2), ont été générés avec $|\Phi| = 400$. La fonction de coût du modèle prédictif m a été approximée de la manière suivante : pour tout ϕ_i appartenant à Φ

$$\text{cost}_m(\phi_i) = \lambda \cdot \left(1 - \frac{i}{|\Phi| - 1}\right), \quad (4.17)$$

où λ le coût de validation de l'hypothèse ϕ_o .

Nous posons $\lambda = 1$. Ainsi, le coût de validation de l'ensemble des hypothèses de Φ est approximativement de 200. Le coût maximal alloué au calcul de la solution a été fixé arbitrairement à 20. 1000 ensembles solution ont été générés pour déterminer les valeurs optimales de α_ϕ and α_ϵ . Les résultats montrent que quel que soit l'objectif, $\alpha_\phi \simeq \alpha_\epsilon \simeq 0.001$. Une caractéristique intéressante de ce résultat est que la qualité de la solution est très mauvaise si le critère de coût n'est pas utilisé, *i.e.*, si $\alpha_\phi = \alpha_\epsilon = 0$, et optimale pour des valeurs de α_ϕ et α_ϵ très petites. Cependant, les résultats présentés dans la figure 4.1, laissent à penser que cette valeur dépend du coût maximal alloué au calcul de la solution. Cela est cohérent avec l'idée intuitive suivante : *plus le temps dont on dispose pour réaliser une tâche est important, moins le critère coût (temps) est important.*

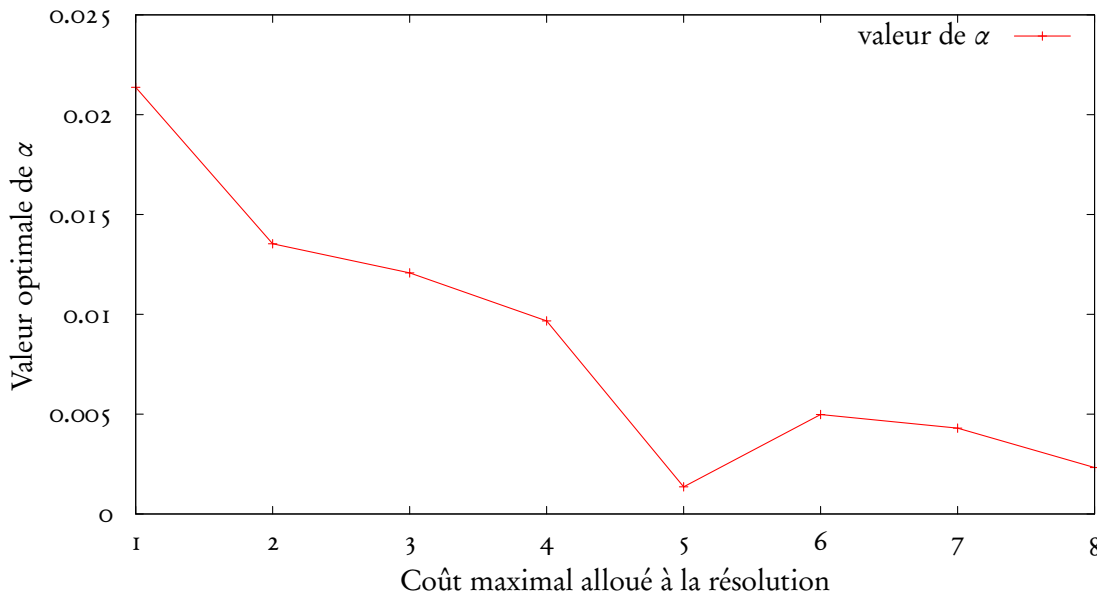


Fig. 4.1 – Valeur optimale de α_ϕ en fonction du coût maximal alloué pour calculer le résultat.

Les figures 4.3, 4.4 et 4.5 montrent les performances de γ_ϕ et γ_ϵ , respectivement pour les objectifs *id.*, *el.* et *prs.*, comparativement à des méthodes de choix aléatoire et d'échantillonnage de l'espace de recherche.

Pour les objectifs *id.* et *prs.*, γ_ϵ permet d'obtenir une solution quasi-optimale alors que les autres méthodes sont, à partir d'un certain coût, globalement inefficaces. L'analyse des résultats obtenus pour l'objectif *el.* doit être nuancée. En effet, si γ_ϕ donne de bons résultats au début de la recherche, γ_ϵ prend largement l'avantage sur la fin et permet d'obtenir ici encore une solution quasi-optimale. Par ailleurs, comme le note Lipton (2004) et bien que cela puisse paraître surprenant, le choix aléatoire n'est pas une si mauvaise méthode lorsqu'on ne dispose d'aucune information préalable sur la solution. En revanche, l'échantillonnage de l'espace recherche ne permet pas, du moins pour des coûts maximums autorisés faibles, de fournir de solutions de bonne qualité. γ_ϵ a ainsi été implémentée dans le système d'aide à la décision.

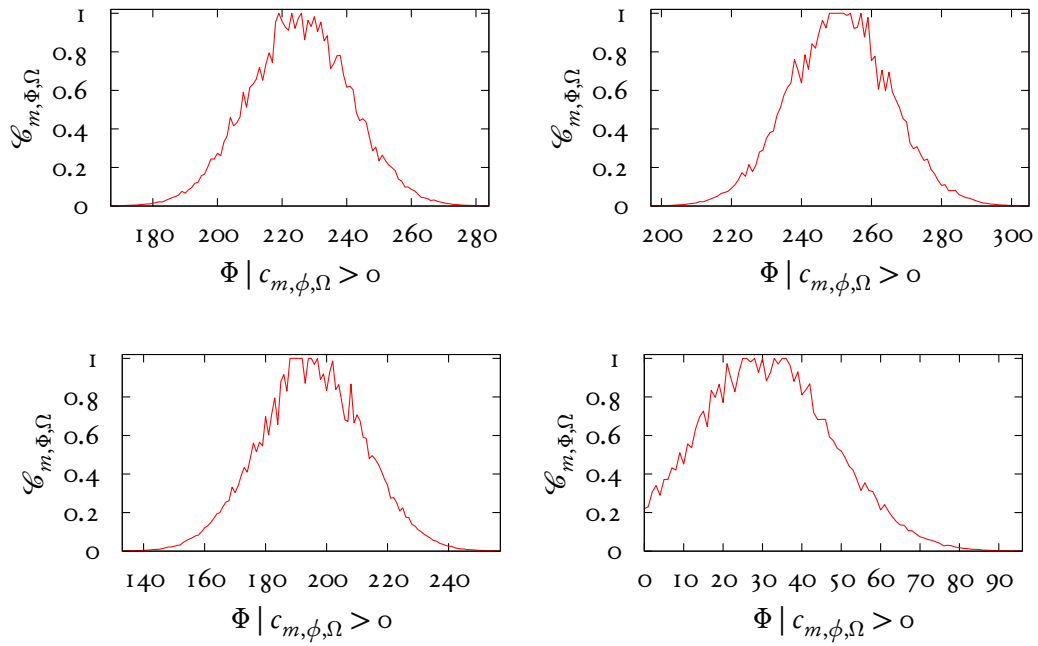


Fig. 4.2 – Exemples d'ensembles solutions générés pour évaluer les heuristiques

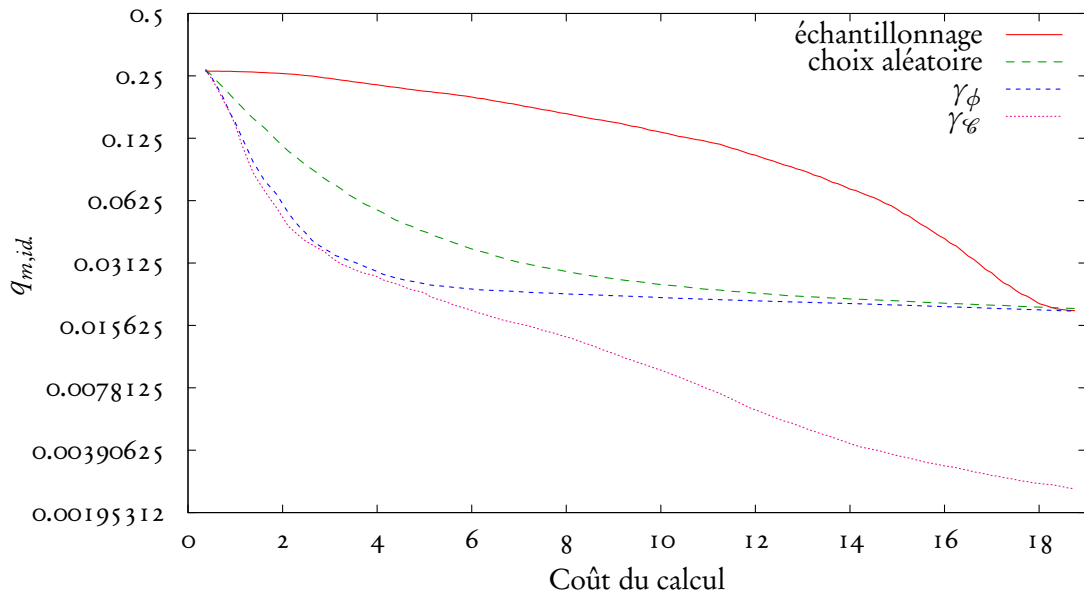


Fig. 4.3 – Analyse de la performance des heuristiques pour l'objectif *id*.

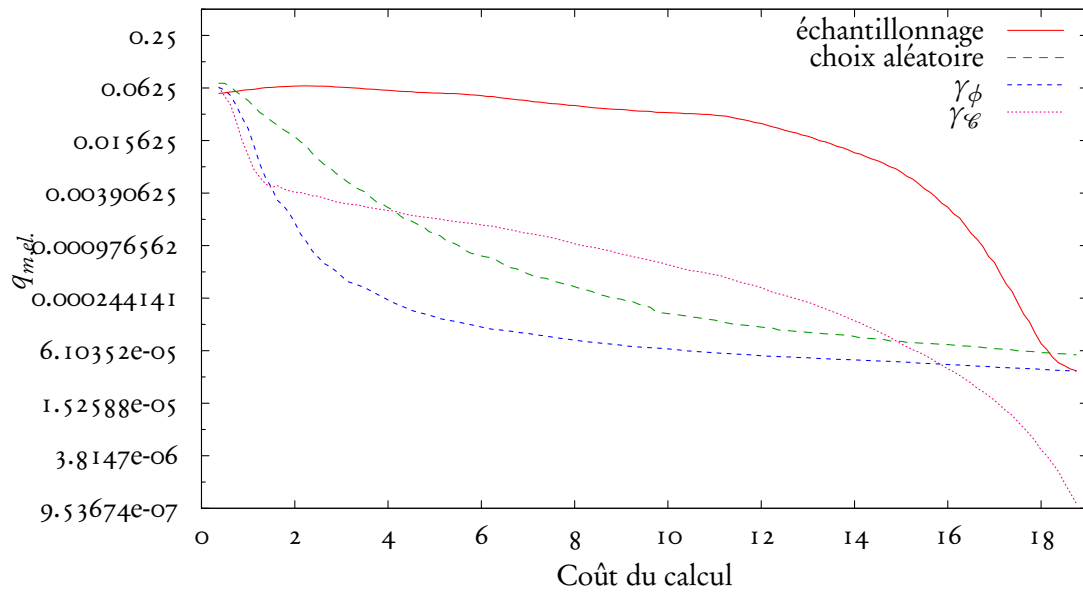


Fig. 4.4 – Analyse de la performance des heuristiques pour l'objectif el .

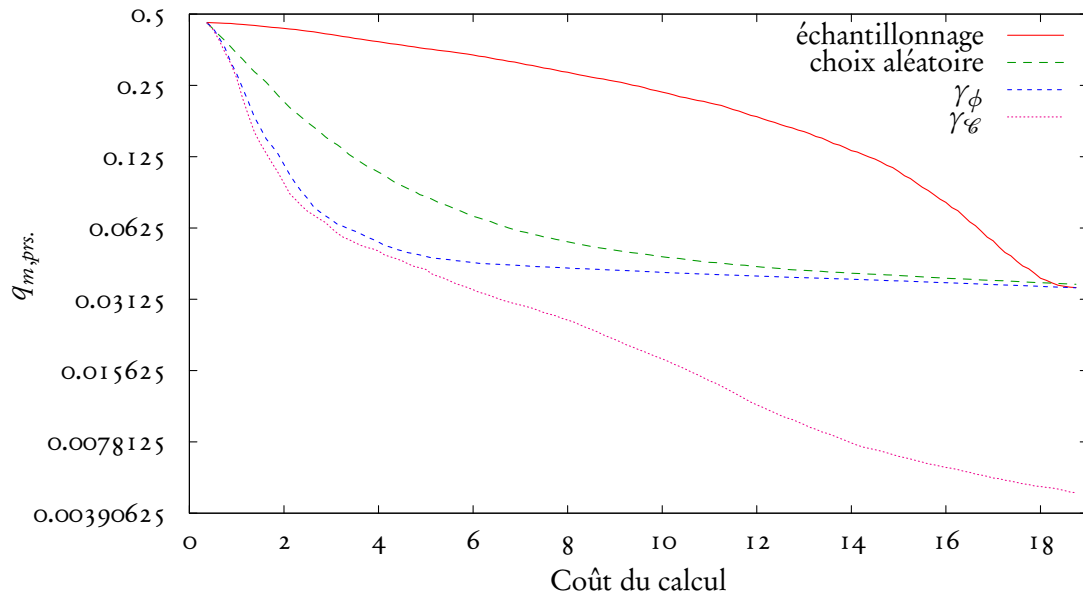


Fig. 4.5 – Analyse de la performance des heuristiques pour l'objectif prs .

4.4 Résumé du chapitre

Dans ce chapitre nous avons défini différents objectifs de résolution de problèmes inverses :

- déterminer la meilleure solution (définition « classique » du problème inverse),
- déterminer l'ensemble des solutions plausibles, *i.e.*, pour lesquelles la mesure de cohérence associée est non nulle,
- éliminer les solutions impossibles,
- déterminer la mesure de cohérence associée à chaque hypothèse (solution possible du problème).

Nous avons de plus associé une métrique à chacun de ces objectifs afin d'évaluer la qualité du processus de résolution. Bien sûr, la définition de ces métriques est subjective et de nombreuses autres métriques pourraient être définies.

Un algorithme de résolution ainsi que des heuristiques, inspirés par l'approche empirique de l'abduction, ont été définis. Les trois derniers objectifs du problème inverse sont tout particulièrement considérés. Des résultats expérimentaux ont enfin été présentés. Ils montrent que les différentes heuristiques proposées permettent de fournir efficacement des solutions de qualité.

Troisième partie

Application à l'entomologie médico-légale

Introduction à l'entomologie médico-légale

Sommaire

5.1	Principes généraux	66
5.1.1	Objectifs des expertises entomologiques	66
5.1.2	Colonisation d'un cadavre par les diptères nécrophages	66
5.2	Modélisation du développement des diptères nécrophages	68
5.2.1	La température, paramètre fondamental	68
5.2.2	Quelques modèles de développement	69
5.2.3	Variabilité de la vitesse de développement au sein d'une population	70
5.3	Calcul d'un intervalle post-mortem à l'aide de modèles de développement	72
5.3.1	Approche formelle	72
5.3.2	Problématique principale : l'estimation de la température	72
5.4	Résumé du chapitre	74

Pour estimer la date d'un décès, les médecins légistes et experts ont recours à différentes techniques. Lorsque l'intervalle post-mortem (IPM) est inférieur à soixante-douze heures, il est possible d'utiliser des méthodes issues de la médecine légale : analyse de la température rectale, observation des lividités etc. Si en revanche, l'IPM est supérieur à 72 heures, la seule technique utilisable en pratique procède de l'analyse des insectes retrouvés sur le corps. Cette branche de l'entomologie, appelée entomologie médico-légale, s'est considérablement développée ces dernières années. Dans ce chapitre, nous introduisons les bases théoriques de cette science et détaillons la méthodologie de détermination d'un IPM.

5.1 Principes généraux

5.1.1 Objectifs des expertises entomologiques

Dans le cadre d'une enquête liée à un décès, il est essentiel d'obtenir un maximum d'informations sur les conditions de la mort. De nombreuses méthodes ont vu le jour afin d'exploiter au mieux les indices présents. L'une d'elles, l'entomologie médico-légale, exploite les indices entomologiques (*i.e.*, les insectes ou restes d'insectes retrouvés sur ou à proximité du corps) afin d'estimer la date de décès de la victime³⁵. Cette technique soulève un intérêt croissant dans les polices scientifiques du monde entier. En effet, parmi l'ensemble des méthodes de datation de la mort, l'entomologie médico-légale est la seule utilisable en pratique lorsque l'intervalle post-mortem (IPM) est supérieur à soixante-douze heures.

Lors de la découverte d'un cadavre, les enquêteurs se rendant sur place prélèvent des échantillons représentatifs de l'entomofaune retrouvée sur ou à proximité de la victime. Ces échantillons, mis sous scellés, seront conservés à température contrôlée. Sur réquisition, ils seront transmis à un laboratoire où les entomologistes procéderont à la mise en élevage des spécimens et à leur identification (détermination de l'espèce à laquelle ils appartiennent) afin d'estimer le plus précisément leur âge.

Les méthodes modernes de datation en entomologie médico-légale reposent sur des modèles du développement des diptères nécrophages. Nous verrons dans les sections suivantes comment sont construits ces modèles et comment ils sont utilisés dans le cadre d'enquêtes criminelles.

5.1.2 Colonisation d'un cadavre par les diptères nécrophages

Un cadavre est une source importante de nutriments pour les organismes nécrophages, *i.e.*, se nourrissant de tissus en décomposition ou nécrophiles, *i.e.*, se nourrissant des organismes présents sur le cadavre. Parmi ceux-ci, les insectes et en particulier les diptères vont être les premières espèces à coloniser un corps pour y pondre des oeufs. Certaines espèces peuvent détecter un cadavre à de très grandes distances. Ainsi, quelques minutes seulement peuvent séparer le décès du début de la colonisation.

Cependant, toutes les espèces ne sont pas attirées par le même stade de décomposition. De nombreuses espèces de diptères vont ainsi se succéder sur le cadavre, suivies par certaines espèces coléoptères, d'hyménoptères et de lépidoptères (Méglin, 1894). Ainsi, certains auteurs ont parlé d'escouade pour désigner l'ensemble des espèces intéressées par une phase donnée de décomposition (cf. tableau 5.1). Il est important de noter que les processus de décomposition d'un corps et de sa colonisation par l'entomofaune sont très variables, dépendent de nombreux paramètres écosystémiques pour certains assez mal connus : « la dégradation d'un cadavre s'avère être un processus continu complexe bien plus qu'une suite d'étapes bien définies. La décomposition étant fortement liée aux caractéristiques du milieu et du corps, la succession des insectes est donc très variable » (Charabidze, 2008, p. 31).

³⁵L'entomologie médico-légale peut également être utilisée à d'autres fins comme par exemple la détection de stupéfiants ingérés avant le décès (Introna *et al.*, 2001) ou la présence de substances répulsives sur le corps (Charabidze *et al.*, 2005, 2009). Cependant, la datation de la mort reste l'objectif principal d'une expertise entomologique.

Escouade	Faune	État du corps	Âge du corps
1	<i>Calliphora vicina</i> (Dipt., Calliphoridae)	Frais	< 3 mois
	<i>C. vomitoria</i> (Dipt., Calliphoridae)		
	<i>Lucilia spp.</i> (Dipt., Calliphoridae)		
	<i>Musca domestica</i> (Dipt., Muscidae)		
	<i>M. autumnalis</i> (Dipt., Muscidae)		
	<i>Muscina stabulans</i> (Dipt., Muscidae)		
2	<i>Sarcophaga spp.</i> (Dipt., Sarcophagidae)	Forte odeur	
	[peut apparaître dans la 1ère vague] <i>Cynomia spp.</i> (Dipt., Calliphoridae)		
3	<i>Dermestes</i> (Col., Dermestidae)	Fermentation des graisses	
	<i>Anglossa</i> (Lep., Pyralidae)		
4	<i>Piophilidae casei</i> (Dipt., Piophilidae)	Fermentation caséique	3 – 6 mois
	<i>Madiza glabra</i> (Dipt., Piophilidae)		
	<i>Fannia</i> (Dipt., Fanniidae)		
	<i>Drosophilidae</i> (Dipt.)		
	<i>Sepsidae</i> (Dipt.)		
	<i>Sphaeroceridae</i> (Dipt.)		
	<i>Eristalis</i> (Dipt., Syrphidae)		
	<i>Teichomyza fusca</i> (Dipt., Ephydriidae) <i>Corynetes, Necrobia</i> (Col., Cleridae)		

Tab. 5.1 – Les quatre premières escouades du modèle de succession selon Smith. Le modèle entier comporte huit escouades (Smith, 1986).

Les larves de diptères pondues sur le corps vont s'y développer selon le cycle présenté dans la figure 5.1. On parle de stade pour désigner ces différentes phases de développement. Les œufs, une fois éclos, vont passer par trois stades larvaires (L1, L2, L3) durant lesquels ils vont se nourrir des tissus en décomposition. Ensuite, les larves vont quitter le corps pour s'empurger et se transformer en nymphe. A la fin du processus de métamorphose, l'imago, *i.e.*, l'insecte adulte, émergera de la pupa.

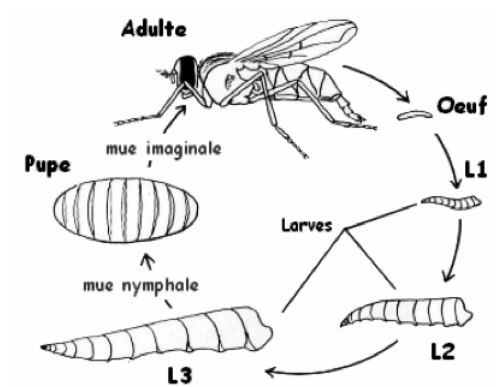


Fig. 5.1 – Cycle de développement holométabole des diptères Calliphoridae.

5.2 Modélisation du développement des diptères nécrophages

5.2.1 La température, paramètre fondamental

La vitesse de développement des diptères dépend principalement, comme pour de nombreuses autres espèces, animales ou végétales, de la température (Stinner *et al.*, 1974) :

$$\frac{da}{dt} = f(T(t)) \quad (5.1)$$

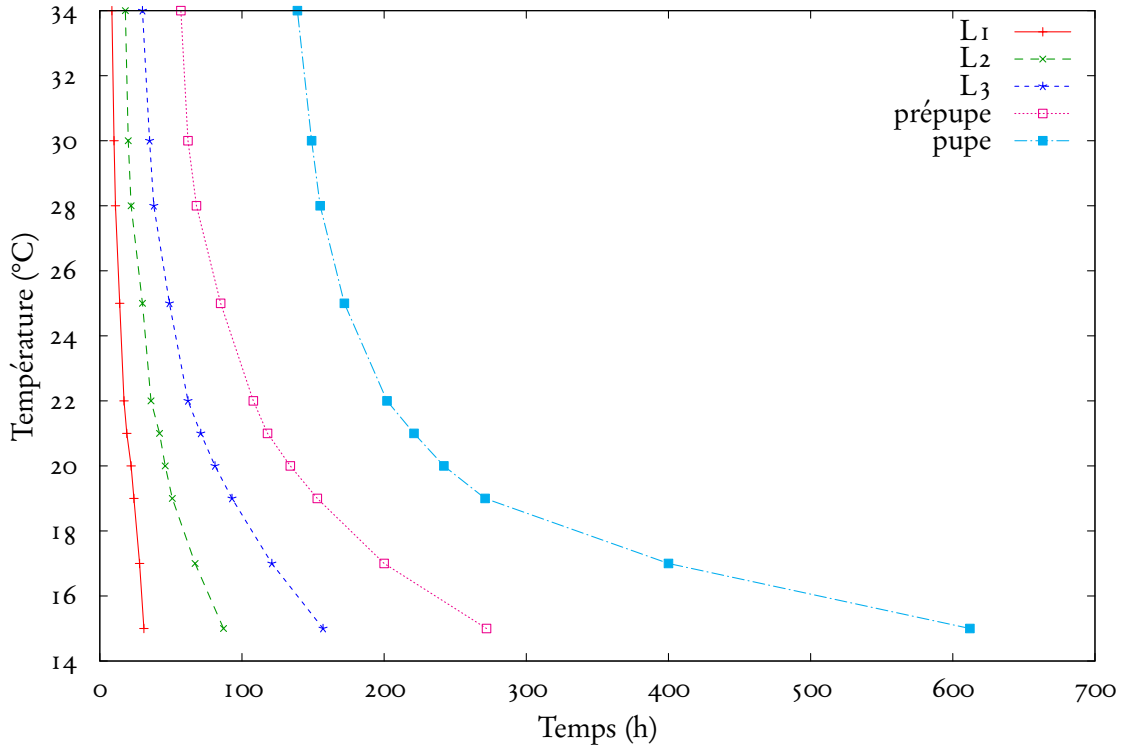
Où a représente le taux de développement accumulé (ou âge physiologique) de l'insecte ($\frac{da}{dt}$ représente donc sa vitesse de développement), $T(t)$ la température en fonction du temps t , et f un modèle de développement. De nombreux modèles de développement, plus ou moins fiables et faciles à utiliser en pratique, ont vu le jour. Certains d'entre eux seront décrits dans la section suivante. Le lecteur intéressé peut se référer à Wagner *et al.* (1984) pour une présentation exhaustive des modèles de développement utilisés en entomologie.

Il faut souligner que la précision de ces modèles est relative au contexte de leur utilisation. Ainsi, il a été démontré que certains modèles pouvaient être plus précis que d'autres dans certains cas et inversement (Higley et Haskell, 2000). Cependant, il n'est pas toujours possible d'identifier les conditions dans lesquelles il était préférable d'utiliser un modèle donné. De plus, des données relatives au développement des différentes espèces de diptères nécrophages, parfois fort différentes³⁶, ont été publiées en grand nombre depuis plusieurs décennies (Grassberger et Reiter, 2001; Greenberg, 1991; Kamal, 1958; Marchenko, 2001).

Il est possible de différencier deux types de modèles de développement : les modèles dits mathématiques et les modèles dits physiologiques (Wagner *et al.*, 1984). Les premiers sont de simples expressions mathématiques de la relation entre vitesse de développement et température et sont calibrés par régression. Les paramètres de ces modèles ne correspondent à aucune réalité biologique. Dans les modèles physiologiques, tous les paramètres sont relatifs à des constantes physiologiques, *e.g.*, la température d'activation d'une enzyme, de l'espèce considérée. En dépit de leur intérêt explicatif, ces modèles ne sont pas utilisés en entomologie médico-légale, les paramètres étant souvent inconnus et difficiles à déterminer avec précision. Dans la section suivante nous ne présenterons donc que deux modèles « mathématiques ».

Il est de plus possible de déterminer un modèle de développement spécifique pour chaque stade, basé sur le temps nécessaire à l'insecte pour atteindre un stade donné. Ainsi, on considérera n fonctions $f_1(T(t))$, $f_2(T(t))$, ... , $f_n(T(t))$, correspondant aux n stades de développement propres à l'espèce considérée (Grassberger et Reiter, 2001). Cela est particulièrement intéressant si l'on souhaite décrire la succession des différents stades. La figure 5.2 présente un diagramme isomorphe, donnant le temps moyen qu'un individu va passer dans chaque stade en fonction de la température, pour l'espèce *Lucilia sericata*.

³⁶Il semble ainsi que la vitesse de développement soit, pour une même espèce, dépendante de la latitude (Richards, 2007).


 Fig. 5.2 – Diagramme isomorphe de *Lucilia sericata*, issu de Grassberger et Reiter (2001)

5.2.2 Quelques modèles de développement

5.2.2.1 Le modèle des degrés jours (heures) accumulés

Les modèles des degrés jours accumulés (ADD) et des degrés heures accumulés (ADH) considèrent que l'augmentation du taux de développement est linéaire en fonction de la température dans un intervalle de températures $[T_{min}, T_{max}]$ et nul si la température est égale à T_{min} :

$$f(T(t)) = \begin{cases} \alpha \cdot T(t) + \beta & \text{si } T \in [T_{min}, T_{max}], \\ 0 & \text{si } T = T_{min}. \end{cases} \quad (5.2)$$

T_{min} et T_{max} représentent les températures minimales et maximales de développement. T_{max} est généralement proche de la température létale (Higley et Haskell, 2000). T_{min} peut être déduite de α et β . En effet,

$$T_{min} = -\frac{\beta}{\alpha}. \quad (5.3)$$

Ce modèle n'est utilisable que lorsque la température est, durant l'ensemble du développement, comprise entre T_{min} et T_{max} . En dépit de cela, il a été utilisé pour modéliser le développement de nombreux organismes vivants animaux ou végétaux, outre les diptères, comme le coton, le maïs ou encore certains pucerons (Allen, 1976). Il présente des intérêts pratiques évidents et est donc très utilisé par les experts entomologistes. De plus, en dépit de

sa simplicité, il peut se révéler plus précis que des modèles plus complexes, notamment quand la température de l'environnement n'est pas connue précisément (Higley et Haskell, 2000).

Afin d'estimer le temps de développement rapidement, les entomologistes associent à chaque espèce de diptère une valeur d'ADD. Cette unité représente le produit d'une durée (en jours) par une température (en °C). Par exemple, si une espèce X possède 200 comme valeur d'ADD et que sa température minimale de développement est 10°C, cela signifie qu'à 20°C, une larve d'une telle espèce aura besoin de 20 jours pour terminer son cycle de développement. En effet : $(20 - 10) \cdot 20 = 200$. De même à 15°C, elle aura besoin de 40 jours pour terminer son cycle de développement : $(20 - 15) \cdot 40 = 200$. On associe également à chaque espèce une valeur d'ADH que l'on utilise lorsque $T(t)$ est donnée avec suffisamment de précision.

5.2.2.2 Le modèle de Stinner

En dépit de leurs qualités les modèles des ADD et ADH peuvent se révéler imprécis lorsque la température ressentie par l'insecte est proche de T_{min} ou T_{max} . Des modèles dits curvilinéaires ont été développés pour pallier cela. Le modèle décrit par Stinner *et al.* (1974) établit la relation entre vitesse de développement et température de la façon suivante :

$$f(T(t)) = \begin{cases} \frac{c}{1 + e^{k_1 + k_2 \cdot T(t)}} & \text{si } T < T_{opt} \\ \frac{c}{1 + e^{k_1 + k_2 \cdot (-2 \cdot T(t))}} & \text{sinon.} \end{cases} \quad (5.4)$$

où c , k_1 et k_2 sont des paramètres du modèle identifiés par régression et T_{opt} représente la température optimale de développement. Ce modèle présente un compromis intéressant entre facilité d'utilisation et précision.

5.2.3 Variabilité de la vitesse de développement au sein d'une population

Tous les individus d'une même espèce ne se développent pas à la même vitesse. Deux méthodes ont été développées pour modéliser cette variabilité et sont décrites dans les sections suivantes.

5.2.3.1 Les Time Distributed Delays

La méthode des « Time Distributed Delays » (TDD) décrite par Severini *et al.* (2003), consiste à discrétiser le taux de développement accumulé. Ce dernier est ainsi défini comme un ensemble de H états,

$$a = \{a_1, a_2, \dots, a_H\}. \quad (5.5)$$

Un ensemble de H équations, représentant les flux de sortie d'un état sont déterminés afin d'estimer à chaque instant t , le nombre d'individus dans l'état final, $N_H(t)$, qui est une grandeur observable dans le système source. H est ainsi le paramètre déterminant la dispersion

des vitesses de développement au sein d'une population. Ces équations sont définies de la manière suivante :

$$\text{Pour } h = 1, 2, \dots, H, \frac{dF_h(t)}{dt} = A_h \cdot (F_{h-1}(t) - F_h(t)), \quad (5.6)$$

où A_h est un paramètre d'échelle,

$$A_h = \frac{1}{a_{h+1} - a_h}, \text{ et } F_h(t) = A_h \cdot N_h(t).$$

Un modèle de ce type est, dans le cas général, très difficile à calibrer, les flux intermédiaires, $F_{2,3,\dots,H-1}$, ne pouvant être observés. Il s'agit en effet de paramètres du modèle ne correspondant à aucune réalité biologique. Severini *et al.* (2003) proposent une méthode de calibrage utilisant un modèle de développement linéaire (ADD). Cependant ce modèle n'est utilisable que dans l'intervalle $[T_{min}, T_{max}]$.

5.2.3.2 La méthode de Régnière

La méthode développée par Régnière (1984) consiste à ajouter à l'expression de la vitesse de développement un coefficient de variabilité var , dépendant de la vitesse de développement de l'individu dans la population par rapport à la vitesse de développement médiane, *i.e.*, le quantile q , et de la température :

$$\frac{da}{dt} = var(q, T(t)) \cdot f(T(t)). \quad (5.7)$$

Régnière propose une forme originale de $var(q, T)$:

$$var(p, T) = 1 - \frac{\ln\left(\frac{q^{-k_1(T)} - 1}{0.5^{-k_1(T)} - 1}\right)}{k_2(T)}, \quad (5.8)$$

où $k_1(T)$ et $k_2(T)$ sont obtenus par régression. Notons que $var(q, T)$ est la fonction inverse de la fonction de répartition de la vitesse de développement des individus par rapport à la vitesse de développement médiane, *i.e.*, la fonction quantile de la distribution.

L'expression mathématique de ce modèle, adaptée au cas traité, n'est pas idéale pour une utilisation en entomologie médico-légale. En effet, il n'existe pas de relation mathématique entre médiane et moyenne. Nous avons réutilisé la méthode de Régnière afin de déterminer un modèle de variabilité de l'espèce *Lucilia sericata* basé sur la distribution de Weibull à trois paramètres (Weibull, 1951). $var(q, T)$ s'exprime alors sous la forme :

$$var(q, T) = (-\ln(1 - q))^{1/k_2(T)} \cdot k_1(T) + w, \quad (5.9)$$

où $k_1(T)$, $k_2(T)$ et w sont des paramètres du modèle, ici obtenus par régression à partir de données originales³⁷.

³⁷Ces données ont été fournies par M. Gosselin de l'Institut National de Criminalistique et de Criminologie (Belgique).

Il semble de plus que pour un même individu q varie en fonction de la température. En d'autres termes, un individu est « optimal » à une température donnée. Les modèles de ce type sont donc difficilement utilisables à températures variables pour les individus les plus rapides et les moins rapides de la population (Charabidze, 2008).

5.3 Calcul d'un intervalle post-mortem à l'aide de modèles de développement

5.3.1 Approche formelle

Il est très simple, *dans un environnement contrôlé*, de prédire le temps nécessaire au développement d'un insecte et à l'inverse de dater l'instant t_1 auquel il a été pondu à partir de son taux de développement accumulé a à l'instant t_2 :

$$a(t_2) = \int_{t_1}^{t_2} f(T(t)) dt \quad (5.10)$$

En pratique, cette équation est discrétisée et résolue numériquement :

$$a(t_2) = \sum_{t_1}^{t_2} f(T(t)) \Delta t, \quad (5.11)$$

où Δt représente la durée séparant deux mesures de température (e.g., un jour ou une heure pour les stations météorologiques de Météo France).

Il est ainsi possible de déduire, à partir de t_1 , un IPM minimum et d'estimer la date de la mort. La précision et la qualité de cette estimation sont très variables car dépendant de nombreux paramètres (comportement de ponte, échantillonnage, effet de masse, etc.).

5.3.2 Problématique principale : l'estimation de la température

Dans un écosystème complexe, comme un corps humain dans un milieu naturel, il devient très difficile d'estimer la température ressentie par les insectes, et ce pour trois raisons majeures qui sont détaillées dans les sections suivantes : l'existence de micro-climats, l'inertie thermique du cadavre et l'augmentation de température due aux masses de larves. De ce fait, l'estimation d'un intervalle post-mortem peut être parfois inexacte.

5.3.2.1 L'existence de micro-climats

Les criminels tuent rarement leur victime au pied d'une station météorologique. Ainsi, la température de la scène de crime peut différer notablement de la température relevée par la station météorologique la plus proche. Il est possible de relever les températures sur le lieu de découverte du corps et de les corrélérer avec les températures relevées par la station météorologique. Selon Gosselin *et al.* (2006), même lorsque la corrélation n'est pas très bonne, cela permet de préciser l'estimation de l'IPM. Cette méthode demeure cependant controversée.

5.3.2.2 L'inertie thermique du corps

L'inertie thermique d'un cadavre peut être importante notamment dans les heures qui suivent le décès. La figure 5.3 représente la forme générale que suit la décroissance de la température corporelle ; à température variable, les températures du corps et de l'environnement diffèrent tout au long de la décomposition. De plus, les températures en différents points du corps varient selon la nature des tissus et les conditions dans lesquelles le cadavre est placé. Ce facteur semble donc difficile à prendre en compte dans le cadre d'une expertise entomologique classique (Charabidze, 2008).

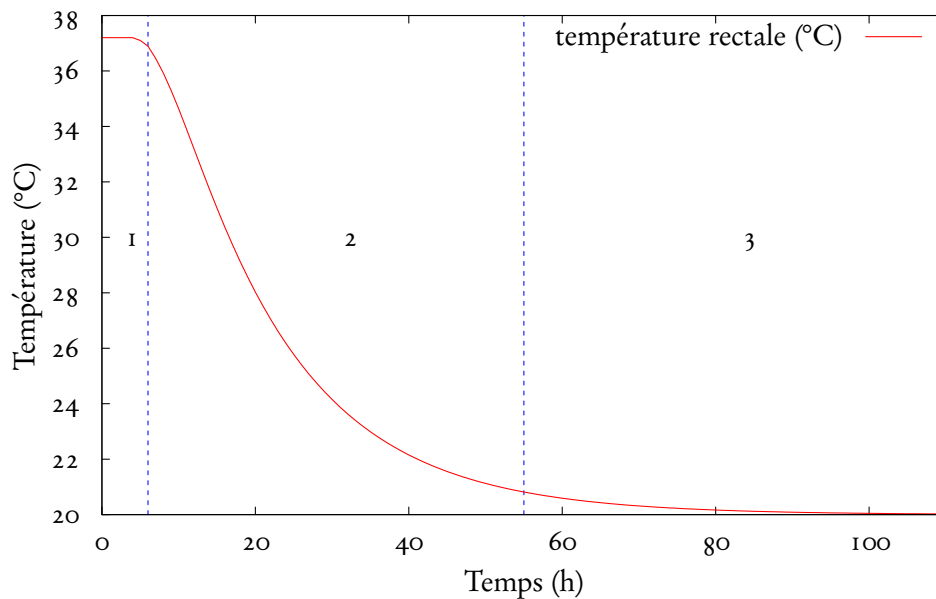


Fig. 5.3 – Forme générale du refroidissement d'un cadavre à température constante (20°C).
1 : plateau thermique initial / 2 : décroissance linéaire / 3 : phase « exponentielle »

5.3.2.3 L'effet des masses de larves

Les larves de diptères ont un comportement grégaire conduisant à la formation de larges masses. De nombreux auteurs ont noté que la température au sein de ces masses pouvait différer notablement de la température de l'environnement (Campobasso *et al.*, 2001; Deonier, 1940; Gruner *et al.*, 2007; Hewadikaram et Goff, 1991; Joy *et al.*, 2002, 2006; Richards et Goff, 1997). Greenberg (1991) relève ainsi des augmentations pouvant aller jusqu'à 18°C. Ainsi, l'influence de ce phénomène, appelé dans la littérature « effet des masses de larves » ("*maggot mass effect*"), sur le développement des diptères semble importante (Higley et Haskell, 2000; Marchenko, 2001; Goodbrod et Goff, 1990; Ireland et Turner, 2006; Kaneshrajah et Turner, 2004; Saunders et Bee, 1995).

Pourtant, peu d'auteurs se sont attachés à comprendre et à modéliser ce comportement grégaire et ses conséquences sur le développement des larves. Un projet de recherche financé par le département de la justice américain propose un modèle mathématique de l'effet des

masses de larves (Slone *et al.*, 2005). Il s'agit d'une analyse quantitative globale et non d'une approche comportementale. Le résultat principal est l'expression d'une relation entre le volume d'une masse de larves et la température maximale relevée en son sein (Slone et Gruner, 2007). Cette étude fait suite à un premier travail de modélisation dont l'objet était de prendre en compte de nombreux paramètres écosystémiques (Byrd et Allen, 2000).

Dans le chapitre 6, nous proposons un modèle orienté agent du comportement des larves de diptères permettant d'estimer l'effet des masses de larves.

5.4 Résumé du chapitre

Nous avons présenté dans ce chapitre le fondement de l'entomologie médico-légale : la modélisation du développement des diptères nécrophages. Nous avons ensuite vu comment utiliser ces modèles pour estimer la date d'un décès. Enfin, nous avons présenté les sources d'imprécision liées à cette technique.

6

Modélisation et simulation orientées agent du développement des Diptères nécrophages

Sommaire

6.1	Introduction	76
6.1.1	Description du système étudié	76
6.1.2	Objectifs de la modélisation	78
6.1.3	Choix du méta-modèle	80
6.2	Description du modèle conceptuel	81
6.2.1	L'environnement	81
6.2.2	Populations et comportements de ponte des espèces	84
6.2.3	Larves de diptères	85
6.3	Implémentation	89
6.3.1	Couplage Modèles/Agents	89
6.3.2	Agents informatiques	89
6.4	Validation du modèle et résultats de simulation	91
6.4.1	Validation des modèles de développement et de variabilité	91
6.4.2	Validation du modèle thermique du corps	91
6.4.3	Calibrage et validation du modèle comportemental	93
6.5	Résumé du chapitre	96

Nous avons identifié dans le chapitre précédent plusieurs sources d'imprécision relevant d'une simplification du système modélisé, dans le calcul d'un IPM par la méthode entomologique. Dans ce chapitre, nous introduisons un modèle à base d'agents, dont l'objectif est de prendre en compte deux de ces sources d'imprécisions : l'inertie thermique du corps et le comportement thermique des larves de diptères. Ce chapitre est organisé selon la méthodologie de conception de simulations orientées agent présentée dans la section 1.3.2. Le système sera ainsi décrit successivement selon les points de vue du thématicien, du modélisateur et de l'informaticien. Enfin, des résultats de simulations seront présentés.

6.1 Introduction

6.1.1 Description du système étudié

Le système étudié est un milieu nutritif (cadavre humain ou environnement utilisé en laboratoire³⁸) se transformant au cours du temps, situé dans un écosystème pouvant être ouvert et dont la température peut être variable, dans lequel des larves de diptères interagissent et se développent.

Nous considérons trois types d'acteurs : les diptères femelles gravides (c'est-à-dire portant des œufs), les larves de diptères et l'environnement, espace actif d'interactions. Les deux premiers types d'acteurs appartiennent à une espèce donnée et ont ainsi leurs caractéristiques propres. Nous considérons que l'ensemble des interactions entre acteurs, et en particulier entre les larves de diptères, sont médiatisées par l'environnement. En d'autres termes, nous supposons qu'il n'y a pas d'interaction directes entre acteurs mais entre ceux-ci et l'environnement, dont l'état est le résultat de leurs actions.

6.1.1.1 L'environnement

L'environnement constitue le milieu dans lequel les larves de diptères vont se développer et interagir. Il s'agit d'un milieu actif, dont les caractéristiques évoluent au cours du temps. Nous considérons deux types de transformation : la perte de masse due à l'activité des insectes et les changements de température. Ainsi, la transformation de l'environnement résultant de l'activité microbienne sera négligée dans le modèle et ce pour deux raisons : le manque de données quantitatives et la faible influence de ce paramètre³⁹.

Notons que dans le cadre du corps humain, les différents types de tissus dont il est composé ainsi que sa topologie vont induire un schéma de colonisation particulier. En effet, les œufs sont principalement pondus aux alentours des orifices naturels et des plaies et se nourrissent prioritairement des tissus « mous » (cerveau, viscères).

De plus, tout corps est soumis aux lois de la thermodynamique. Ainsi, la température de l'environnement dépend, outre l'activité des larves, de la température et du type du milieu dans lequel il est situé.

6.1.1.2 Les diptères femelles gravides

Nous supposons que les interactions entre diptères adultes et l'environnement se limitent à la ponte : *i.e.*, à l'introduction de larves dans le système. Cet acteur est modélisé d'un point de vue populationnel, la dynamique de ponte de l'ensemble des individus pouvant être simulée sans considérer les individus eux-mêmes. Nous distinguerons de plus les dynamiques de populations et de ponte. La dynamique d'une population dépend essentiellement de paramètres climatiques, variables selon les espèces.

Nous considérons que la dynamique de ponte dépend essentiellement de la température de l'écosystème, de l'état de décomposition du milieu et du cycle nycthémeral⁴⁰.

³⁸Il s'agit de milieux nutritifs homogènes composés de gellose et de sang.

³⁹Dans le cadre de la décomposition d'un cadavre dans un environnement ouvert, 80% de la perte de masse est liée à l'activité des insectes nécrophages (Payne, 1965).

⁴⁰Le cycle nycthémeral désigne la variation de paramètres climatiques ayant une influence sur le comporte-

6.1.1.3 Les larves de diptères

Les larves de diptères sont pondues dans l'environnement et s'en nourrissent. Le développement des larves est rythmé par une succession de stades, mobiles ou non (cf. tableau 6.1).

Stade	Déplacement	Interactions avec l'environnement
œuf	non	aucune
L ₁	oui	déplacement, consommation, augmentation de la température
L ₂		
L ₃		
prépupe		déplacement rectiligne
pupe	non	aucune

Tab. 6.1 – Succession des stades durant le développement des larves de diptères.

Le comportement des larves durant certains stades actifs (L₁, L₂, L₃) est caractérisé par un gréganisme induisant une auto-régulation de la température au sein d'agrégats suffisamment importants (Anderson et VanLaerhoven, 1996; Deonier, 1940; Turner et Howard, 1992). Bien qu'apparemment simple, la structuration de ces masses est en réalité relativement complexe. Ainsi, le comportement d'une larve et sa place dans l'agrégat dépendent à la fois de l'environnement et du « niveau de remplissage » de son jabot⁴¹. Ainsi, sur la figure 6.1, représentant une masse de diptères, on observe « une répartition spatiale des individus impliquant l'expression de comportements distincts » (Charabidze, 2008, p. 144). Il est ainsi possible de distinguer, sur ce critère comportemental, trois groupes :

- les larves qui se nourrissent, agrégées verticalement, en contact direct avec le milieu nutritif (Hobson, 1931),
- les larves dont le jabot est vide ou presque vide, cherchant à se nourrir, *i.e.*, à rejoindre le groupe des larves qui se nourrissent,
- les larves dont le jabot est plein, se déplaçant à proximité de la masse (Hafez, 1948).

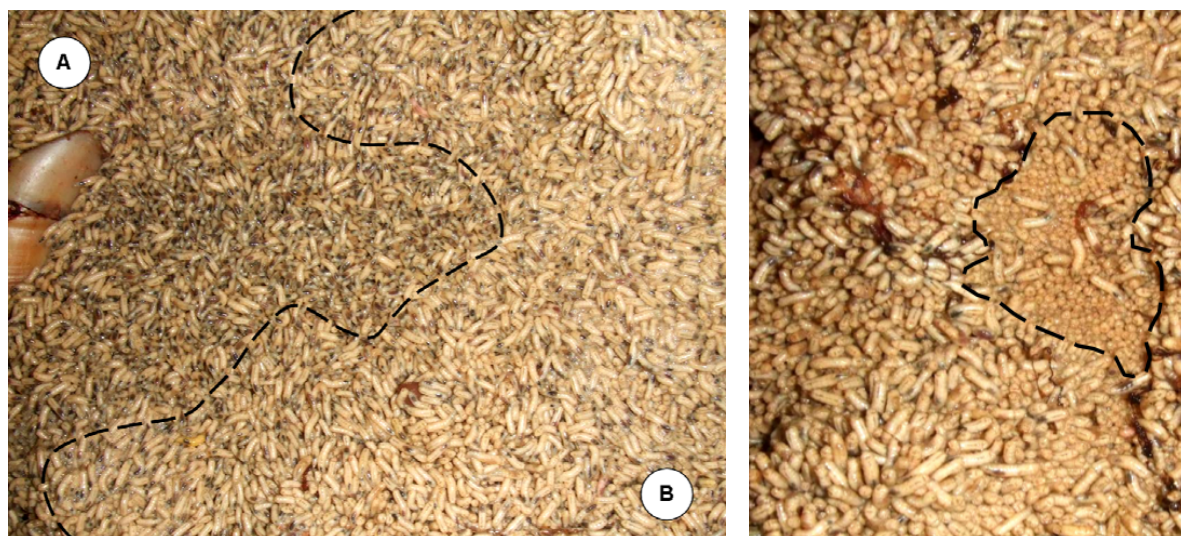
Sur la base d'observations réalisées en laboratoire et en milieu non contrôlé, des taxies⁴² sont associées aux deux derniers groupes. Dans le premier cas, nous supposons que les larves qui se nourrissent émettent un signal attractif⁴³ pour les larves à la recherche de nourriture ; ainsi ces dernières suivent le gradient de ce signal d'agrégation. Dans le second cas, nous supposons que les larves en phase de digestion recherchent un environnement dans lequel la température soit optimal pour se développer : elles suivent donc un gradient de température. De plus, la perception des stimuli (température et signal d'agrégation) n'est pas parfaite (Hafez, 1948). Enfin, nous associons au premier groupe un comportement alimentaire : manger jusqu'à saturation du jabot.

ment des individus comme la luminosité ou la température et liée à l'alternance jour/nuit.

⁴¹Le jabot est une structure digestive visible par transparence lorsqu'elle est pleine.

⁴²Une taxie est un comportement orientant le déplacement d'un individu vers la source de stimulations (Ferber, 1995).

⁴³Bien que la nature exacte de ce signal n'ait pas été identifiée, il semble qu'il soit chimique (Liu et O'Flaherty, 2006; Maddox *et al.*, 2006).



(a) A : larves en phase de digestion. B : larves à la recherche de nourriture
(b) Larves se nourrissant, agrégées verticalement

Fig. 6.1 – Différents groupes de larves identifiés dans un agrégat (Charabidze, 2008, p. 145)

Les comportements décrits plus hauts sont observables lorsque les larves sont placées en conditions viables. Dans le cas général, la définition de la viabilité est très complexe. Dans le type d'environnement que nous considérons, nous supposons que cette condition dépend uniquement de la température : en effet, lorsque celle-ci devient trop élevée, on observe une fuite désordonnée et la reformation d'un ou plusieurs agrégats dans des zones viables. Nous associons à ce comportement de fuite un déplacement aléatoire. C'est, selon nous, le passage de l'un à l'autre comportement qui permet la régulation de la température au sein des masses, comme le suggère la figure 6.2. Ce comportement semble associé à l'activité des individus. Ainsi, on n'observe pas, ou peu, d'augmentation de température dans la masse lorsque les larves ne sont pas en présence de nourriture (figure 6.3) ou lorsque la nourriture est disposée de telle sorte que les individus peuvent y accéder tout en restant agrégés, *i.e.*, d'une manière générale lorsque l'activité des larves est faible.

Les comportements des larves durant les autres stades sont beaucoup plus simples. Durant les stades inactifs, œuf et pupe, les larves ne font que se développer. Dans le cas du stade prépupe, la larve recherche un emplacement abrité pour s'empurger. Ainsi, lorsque les larves se développent sur un cadavre, elles sortent du corps durant ce stade. Nous assimilons ce comportement à un déplacement rectiligne vers l'un des bords de l'environnement.

6.1.2 Objectifs de la modélisation

Objectifs généraux Nous avons vu dans le chapitre précédent que la température était un paramètre fondamental pour prédire le temps nécessaire au développement d'un diptère et inversement pour dater sa ponte. L'objectif principal de ce travail est d'estimer plus précisément la température ressentie par les insectes se développant sur un cadavre, en simulant le

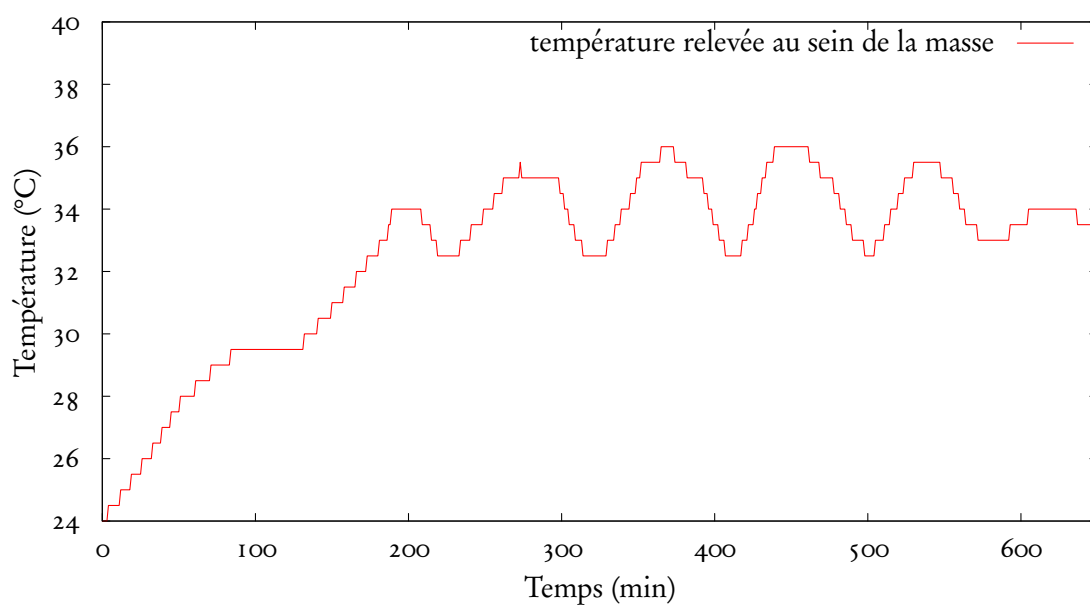


Fig. 6.2 – Oscillations de la température relevée au sein d’une masse de 200g de larves de second stade de *P. terraenovae* placée à 30°C en présence de nourriture

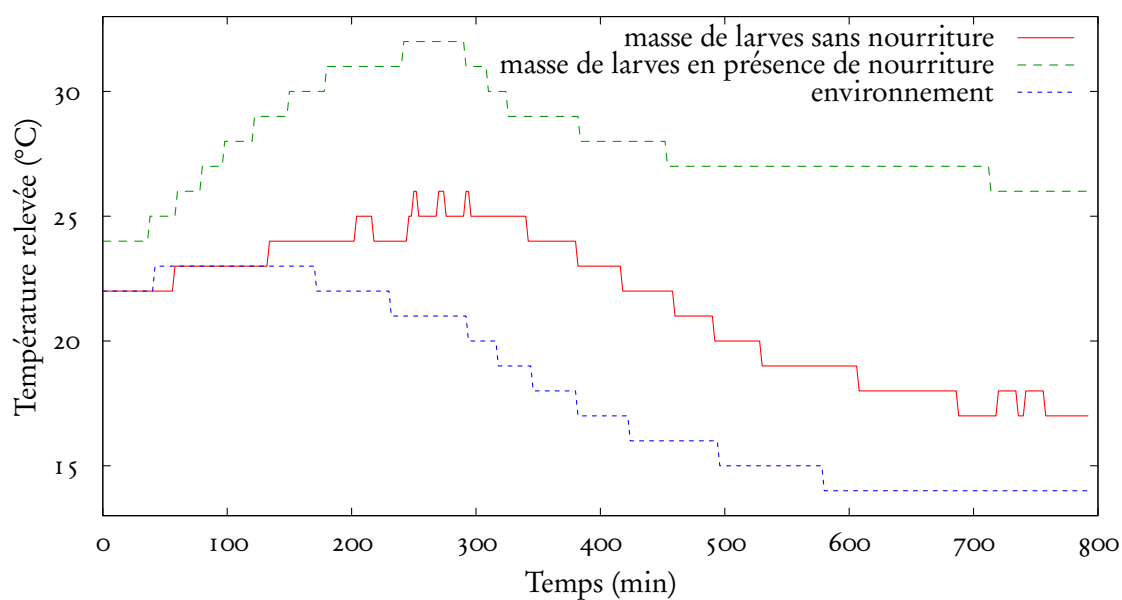


Fig. 6.3 – Températures relevées dans des masses (500g) de larves avec et sans nourriture

comportement thermique du corps et le comportement grégaire des larves de diptères, afin de réaliser de meilleures prédictions du temps de développement des diptères. Par ailleurs, ce modèle peut être utilisé comme « laboratoire virtuel » afin d'analyser les comportements collectifs des insectes dans des cas complexes.

Modélisation de l'environnement Plusieurs types d'environnements peuvent être simulés : le corps humain et différents environnements de laboratoire. L'objectif général de la simulation de ces environnements est la reproduction de leur dynamique thermique. Nous distinguons deux objectifs secondaires pour chacun des environnements simulés. Dans le cadre du corps humain, il s'agit de reproduire la topologie du corps humain avec suffisamment de précision, de sorte que le schéma général de colonisation soit respecté. Parallèlement, le modèle doit être le plus simple possible pour être simulé efficacement. Dans le cadre de l'environnement de laboratoire, il doit être possible d'inclure des objets (sources de chaleur, discontinuités, etc.) avec lesquels les agents pourront interagir.

Modélisation des larves de diptères L'objectif majeur de la modélisation du comportement individuel et des interactions des larves de diptères est la reproduction de patterns stables (aussi bien qualitatifs que quantitatifs) liés à l'auto-régulation de la température au sein d'une masse de larves. Dans un second temps, ce modèle comportemental est couplé à des modèles de développement afin de simuler le développement de populations hétérogènes de diptères dans un environnement complexe. Il s'agit de plus de tester l'influence des différents couplages possibles, en utilisant notamment différents modèles de développement, sur les prédictions de temps de développement.

6.1.3 Choix du méta-modèle

Plusieurs facteurs ont motivé le choix du méta-modèle agent :

La complexité du système étudié Le système étudié peut être qualifié de complexe en raison du grand nombre d'acteurs interagissant simultanément, de la richesse de leurs interactions, de l'importance de la topologie de l'environnement sur le comportement du système et de l'existence de boucles de rétroaction.

La complexité des comportements collectifs Les interactions des larves de diptères permettent au niveau du groupe, l'émergence d'agrégations, favorisant une auto-régulation de la température efficace difficile à quantifier car dépendant de nombreux paramètres quantitatifs (nombre d'individus, température extérieure, caractéristiques thermiques de l'environnement) et qualitatifs (topologie de l'environnement, comportement des individus).

L'hétérogénéité des modèles existants Les modèles existants (développement, population, etc.) sont très différents de part leur forme et les échelles de temps (de la minute à la semaine) ou d'observation (individu, population ou espèce) considérées.

L'échec de l'approche macroscopique L'approche macroscopique développée par Slone *et al.* (2005) semble avoir donnée des résultats décevants et difficilement exploitables dans le cadre d'une expertise entomologique.

6.2 Description du modèle conceptuel

6.2.1 L'environnement

6.2.1.1 Caractéristiques générales

Nous considérons ici un environnement bidimensionnel, représentant une « tranche » de corps humain, ou plus généralement tout type de milieu nutritif, dans lequel les agents vont interagir. L'environnement est découpé en cellules carrées. Deux échelles de discrétisation sont définies. Dans la première, des cellules de quatre centimètres de côté sont utilisées. C'est à cette échelle que le comportement thermique du corps est modélisé. La deuxième échelle utilise des cellules de deux centimètres de côté. C'est à cette échelle que les interactions entre agents sont modélisées. A chaque cellule, on associe un ensemble de variables que les agents pourront, selon les effecteurs dont ils disposent, manipuler. Chaque variable d'environnement possède trois paramètres : sa valeur, un coefficient de diffusion et un coefficient d'évaporation. Le coefficient de diffusion représente la proportion de la valeur de la variable d'environnement qui sera répartie équitablement entre les voisins de la cellule au pas d'exécution suivant. Le coefficient d'évaporation représente la proportion de la valeur de la variable d'environnement (après diffusion) qui sera perdue par la cellule au pas d'exécution suivant. Le tableau 6.2 récapitule les différentes variables d'environnement ainsi que leurs caractéristiques. Les valeurs initiales des variables dépendent de la simulation.

Nom	Coefficient de diffusion	Coefficient d'évaporation
Température T (°C)	0	0
Masse (kg)	0	0
Signal d'agrégation g	1	0.6

Tab. 6.2 – Variables d'environnement définies dans le modèle

Le comportement de la variable « température » est déterminé par un modèle spécifique, basé sur la notion d'automate cellulaire. Le réseau d'automates cellulaires sera défini selon deux axes : structurel et fonctionnel ⁴⁴. Les règles structurelles correspondent à l'aspect topologique de l'automate : structure spatiale, nombre de dimensions, arrangement des cellules et mode de détermination du voisinage. Les règles fonctionnelles concernent l'aspect dynamique du réseau : nombre d'états et règles de transition.

6.2.1.2 Règles Structurelles

Tout d'abord, nous distinguons les cellules appartenant à l'environnement extérieur et les cellules du corps proprement dites (cf. figure 6.4). De plus, pour les cellules du corps,

⁴⁴Les sections suivantes sont basées sur (Veremme *et al.*, 2008)

nous différencions les classes de cellules, selon le type de tissu qu'elles représentent (muscle, cerveau, organe, ou peau), chacune ayant ses caractéristiques propres (*masse*, *largeur*, *chaleur massique* et *conductivité*). Une cellule de type *environnement* peut appartenir à la classe des *fluides* ou des *solides*. Si une cellule est de la classe *fluide*, elle possèdera les caractéristiques suivantes : *viscosité dynamique*, *densité*, *nombre de Prandtl* et *conductivité*. Si elle est de type *solide*, elle ne possèdera que la *conductivité* comme caractéristique. A chaque cellule, on associe une variable : la température.

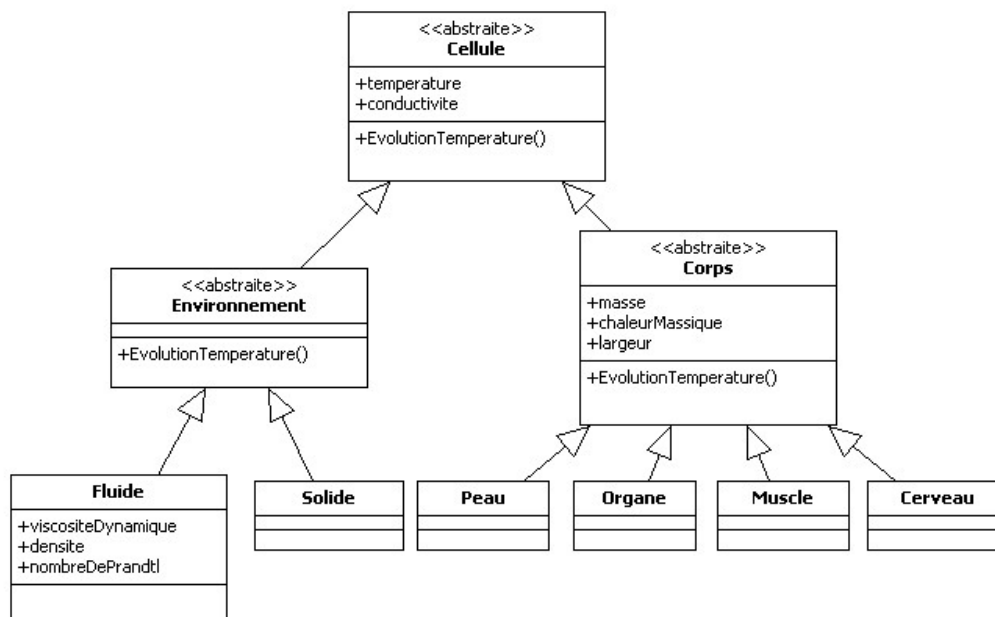


Fig. 6.4 – Classes des cellules de l'automate

L'ensemble des caractéristiques est connu et nous nous basons sur différents travaux existants (tableau 6.3). Le réseau d'automates est structuré en prenant en compte les proportions (masse et taille) du corps humain issues de la littérature.

Tissus	Propriété	conductivité k	chaleur massique c	densité ρ
	Unité	$W.m^{-1}.^{\circ}C^{-1}$	$J.kg^{-1}.^{\circ}C^{-1}$	$kg.m^{-3}$
Peau		0.47	3680	1085
Muscle		0.51	3800	1085
Cerveau		0.49	3850	1080
Organe		0.46	3346	933

Tab. 6.3 – Propriétés thermiques des tissus (Werner et Buse, 1988) : conductivité k , chaleur massique c , densité ρ

6.2.1.3 Règles Fonctionnelles

Les règles de transition correspondent aux règles définissant, pour l'ensemble des cellules, le passage d'un pas de temps p au pas de temps suivant $p + 1$. L'automate étant composé de deux classes principales de cellules, ces règles ou méthodes sont redéfinies en fonction de la classe. Ainsi pour évoluer, (1) à chaque unité de temps, une cellule du corps de l'automate devra parcourir l'ensemble de ses voisins, (2) pour chaque voisin, appliquer la loi thermique correspondante en fonction du type *fluide* ou *solide* de ce dernier (équation 6.1). Enfin (3), pour obtenir sa nouvelle température au pas de temps suivant, elle fera la moyenne des températures obtenues de l'échange avec ses huit voisins. La taille des cellules de l'automate étant considérée comme minime par rapport à l'ensemble du réseau, nous pouvons considérer les échanges indépendants les uns des autres. La simple moyenne suffit donc à obtenir la température instantanée issue de l'échange avec ses voisines. En ce qui concerne les cellules de l'environnement, celles-ci évoluent selon l'environnement considéré. Lors d'une simulation sous température fixe, aucune règle de transition n'est définie pour ces cellules. Sous températures variables, à chaque pas de temps, une cellule de l'environnement réinitialise sa température par rapport à celle de l'environnement.

Les transferts thermiques peuvent s'effectuer selon les trois modes suivants (Incropera *et al.*, 2006) :

- Convection : transfert de chaleur dans un fluide ou d'un fluide à un solide.
- Conduction : transfert de chaleur de proche en proche dans les solides.
- Radiation : transfert de chaleur dans le vide.

Dans ce modèle, nous négligeons le mode de transfert radiatif et ainsi seules les lois de convection et de conduction sont implémentées. Les cellules du corps humain étant solides, les échanges thermiques se feront :

- Par convection avec les cellules de l'environnement de type *fluides* (si le corps est retrouvé à l'air libre ou dans l'eau par exemple).
- Par conduction avec les cellules du type *corps* ou les cellules de type *environnement solide* (si le corps est retrouvé enterré par exemple).

L'équation caractéristique de l'échange thermique entre un corps c et un environnement e est définie comme suit :

$$T_c(p + 1) = T_e(p) + (T_c(p) - T_e(p)) \cdot e^{(-\sigma \cdot p)}, \quad (6.1)$$

où T_c et T_e représentent respectivement les températures (°C) du corps et de l'environnement. La valeur σ diffère en fonction des caractéristiques du corps et de l'environnement (fluide(s) ou solide(s)). Pour un échange par convection

$$\sigma = h \cdot S, \quad (6.2)$$

où h est le coefficient d'échange thermique ($W \cdot m^{-2} \cdot ^\circ C^{-1}$) et S , la surface (m^2) de contact entre le corps et l'environnement.

Dans le cas d'un échange par conduction

$$\sigma = \frac{1}{K}, \quad (6.3)$$

où K représente le coefficient de conductivité ($W \cdot m^{-1} \cdot ^\circ C^{-1}$) du corps.

Le lecteur intéressé peut se référer à Mall et Eisenmenger (2005a,b) s'il souhaite d'avantage d'informations concernant les calculs des différentes valeurs citées.

6.2.2 Populations et comportements de ponte des espèces

Les modèles de population et de ponte pour une espèce donnée sont encapsulés dans un agent particulier dénommé *Layer*. Cet agent détermine à chaque pas de simulation, si des agents, représentant des larves de l'espèce considérée, doivent être introduits dans l'environnement et où. Le modèle de population utilisé ici est basé sur le travail de Goulson *et al.* (2005). Les données sont basées sur des expériences de piégeage réalisées à l'Institut de Médecine Légale de Lille (Charabidze, 2008). Concrètement, ce modèle considère la variation de population sur une semaine en fonction de deux paramètres climatiques dépendant de l'espèce. L'équation générale permettant de définir la population Pop à $t + 1$ à partir de la population à t (t est exprimé en semaines) est définie comme suit :

$$Pop(t + 1) = a + b_o \cdot Pop(t) + b_1 \cdot W_1 + b_2 \cdot W_2, \quad (6.4)$$

où W_1 et W_2 représentent les valeurs des paramètres climatiques considérés, et a , b_o , b_1 et b_2 représentent des paramètres obtenus par régression.

Le modèle précédent permet de déterminer la population des diptères pour une espèce donnée dans un écosystème. Il est ensuite nécessaire de déterminer la population de femelles gravides Pop_G . On considère que 80% d'une population piégée est composée de femelles et que parmi ces dernières, un tiers sont gravides (Fisher *et al.*, 1998). Ainsi,

$$Pop_G(t) = \frac{4}{15} \cdot Pop(t). \quad (6.5)$$

Selon Byrd et Allen (2000), la probabilité qu'une femelle gravide ponde sur un corps à un instant t dépend de l'attractivité du corps pour cette espèce, du cycle nyctéméral et de la température. Il considère de plus une fonction d'attractivité unique, ce qui n'est pas irréaliste si l'on considère seulement des espèces précoces. Soit $at(t)$ cette fonction. Il s'agit d'une distribution normale centrée sur $24h$, avec un écart type de $52.8h$:

$$at(t) = e^{(t-24)^2 / (2 \cdot 52.8^2)}, \quad (6.6)$$

où t est exprimé en heures et $t = 0$ représente l'instant de la mort.

Les auteurs définissent de plus une fonction d'oviposition, $ov(t)$, modélisant les préférences de pontes en fonction du cycle nyctéméral. L'expression originale de cette fonction considère que le pic de ponte a lieu à midi. Or, ce n'est pas le cas pour toutes les espèces. Nous généralisons donc la fonction sous la forme :

$$ov(t) = 0.8 \cdot \sin\left(\frac{\pi}{12} \cdot (t + 6 - ov_p)\right), \quad (6.7)$$

où t est exprimé en heures et ov_p représente le pic d'oviposition.

Ainsi, la probabilité P qu'un insecte ponde durant l'heure t est donnée par la fonction :

$$P(t) = \begin{cases} at(t) \cdot ov(t) & \text{si } ov(t) > 0 \text{ et } T(t) > T_{ponte} \\ 0 & \text{sinon,} \end{cases} \quad (6.8)$$

où T_{ponte} représente la température minimale d'activité pour l'espèce considérée et T la température de l'environnement. Chaque individu ne pond qu'une seule fois, et dépose en moyenne 223 ± 12 œufs (Wall, 1993). Deux zones d'oviposition équiprobables, correspondant aux zones proches des orifices naturels, sont définies : la face et le haut du bassin.

6.2.3 Larves de diptères

La modélisation du comportement et du développement des diptères nécrophages implique le couplage de nombreux modèles qui seront détaillés dans cette section. L'agent encapsulant l'ensemble de ces modèles est appelé *Larva*. Nous étudierons tout d'abord les modèles liés à l'évolution des différentes caractéristiques de l'agent dans le temps (stade, vitesse de déplacement, etc.). Dans un second temps, nous détaillerons le modèle comportemental, permettant aux agents d'interagir à travers l'environnement⁴⁵. À chaque pas de temps, ces différents modèles sont exécutés séquentiellement : interaction puis évolution.

6.2.3.1 Modèles d'évolution

Les agents évoluent au cours du temps. Nous nous intéressons essentiellement aux propriétés suivantes : taux de développement accumulé pour chacun des stades, stade, vitesse de déplacement, longueur, élévation de température sur une période de temps donnée liée à l'activité métabolique. De plus, la mortalité des individus sera considérée d'un point de vue populationnel.

Stade et développement Le stade de l'individu est déterminé par un ensemble de modèles de développement définis pour chacun des stades. Nous proposons à l'utilisateur les modèles de Stinner et des degrés heures accumulés. Cependant, de nouveaux modèles peuvent être définis très facilement. De plus, le modèle de variabilité de développement présenté dans la section 5.2.3.2 a été implémenté. L'algorithme 2 explicite le calcul des taux de développement accumulé et du stade. $a_i(p)$ représente le taux de développement accumulé par l'agent au pas de simulation p pour le stade i . $T(p)$ représente la moyenne des températures ressenties par l'agent durant p q le quantile associé à l'agent, var le modèle de variabilité utilisé et s le stade courant de l'agent.

Vitesse de déplacement Le modèle permettant de déterminer la vitesse de déplacement d'une larve est issu de Charabidze *et al.* (2008a). Ce modèle définit la vitesse V ($\text{cm} \cdot \text{min}^{-1}$) d'un individu comme une fonction de la température ressentie T ($^{\circ}\text{C}$) et de sa longueur l (mm) :

$$V(T, l) = 5.45 \cdot \log_{10}(l) + 0.66 \cdot T - 12.8. \quad (6.9)$$

⁴⁵Cette section est basée sur Morvan *et al.* (2008b)

Algorithme 2 Algorithme de développement de l'agent larve

```

1: Soit un ensemble de modèles de développement  $f_1, f_2, \dots, f_n$ .
2: Pour  $i = s$  to  $n$  Faire
3:    $a_i(p+1) \leftarrow a_i(p) + var(q, T(p)) \cdot f_i(T(p))$ 
4:   Si  $a_i(p+1) \geq 1$  Alors
5:      $s \leftarrow s + 1$ 
6:   Fin si
7: Fin pour

```

Longueur La longueur de l'agent, l (mm), est calculé à partir des taux de développement accumulés pour le stade L_3 , a_3 , à p (le pas de temps actuel) et du pas de temps auquel la larve a éclo p_0 . On considère, à titre simplificatif, que l'augmentation de taille est linéaire (Charabidze, 2008) :

$$l(a_3(p), a_3(p_0)) = (l_{max} - l_{min}) \cdot \frac{a_3(p) - a_3(p_0)}{1 - a_3(p_0)} + l_{min}, \quad (6.10)$$

avec l_{min} et l_{max} représentant respectivement les longueurs minimale et maximale de l'agent en mm.

Mortalité La modélisation de la mortalité des individus est problématique. S'il est avéré que la température et la densité sont des facteurs influant fortement sur la survie des individus, leur impact reste difficile à quantifier (Wall *et al.*, 2001; Moe *et al.*, 2002; Dent, 1997). La figure 6.5 montre ainsi les résultats expérimentaux de taux de survie en fonction de la densité obtenus par différents auteurs pour différentes espèces. Ces résultats sont, en l'espèce, difficilement exploitables. C'est pourquoi nous reprenons l'approche globale de Byrd et Allen (2000) qui postule que, pour une population importante, le taux de mortalité ne dépend que du stade des individus. Cette approche statistique est intéressante car très simple à mettre en œuvre et adaptée à notre problématique : en effet, nous ne cherchons pas à déterminer quels individus en particulier vont survivre mais à estimer le nombre d'individus en vie. Il est ainsi possible d'associer à chaque stade une probabilité horaire de décès (tableau 6.4).

Stade	Probabilité horaire
œuf	0.015
L1	0.01
L2	0.005
L3	0.001
pupe	

Tab. 6.4 – Probabilité de décès horaire d'une larve de diptère en fonction du stade (Byrd et Allen, 2000)

Émission de chaleur Nous postulons que l'élévation de température T_e (°C), résultant de l'activité d'un individu sur un pas de temps p , est proportionnelle à son taux de développe-

ment accumulé par l'individu pour atteindre le stade prépupe a_4 :

$$T_e(p) = \begin{cases} \lambda \cdot a_4(p) & \text{si } j > 0 \\ 0 & \text{sinon,} \end{cases} \quad (6.11)$$

où j représente le niveau de remplissage du jabot de l'individu (cf. équation 6.13) et λ un paramètre à déterminer et dépendant du pas de temps choisi.

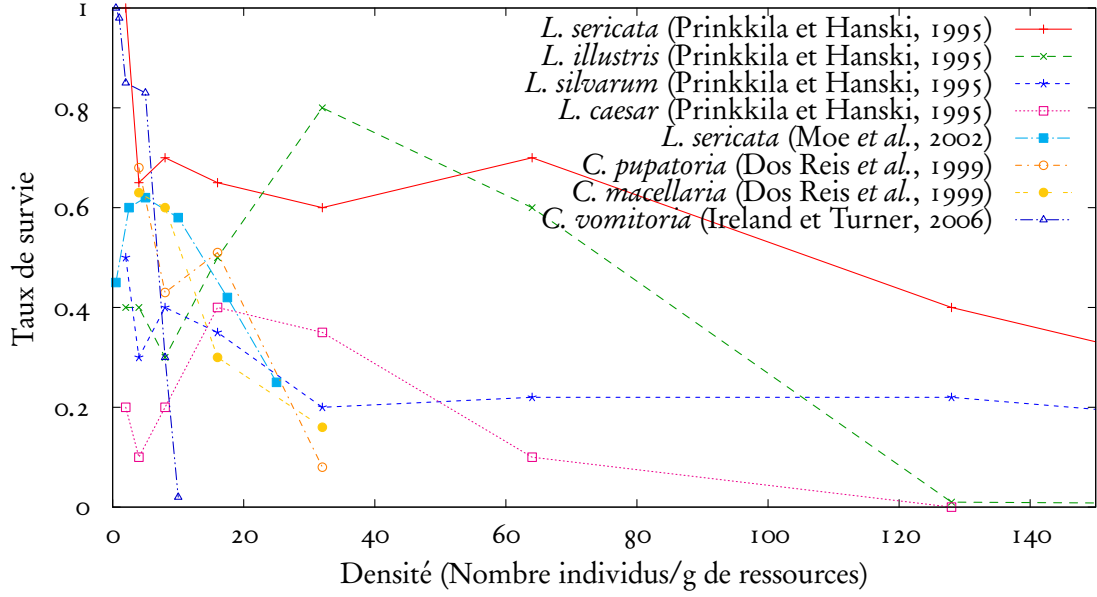


Fig. 6.5 – Taux de survie de larves de diptères en fonction de la densité d'individus

6.2.3.2 Modèle comportemental

Le modèle comportemental des agents larves peut être représenté comme un automate à deux états (figure 6.6).

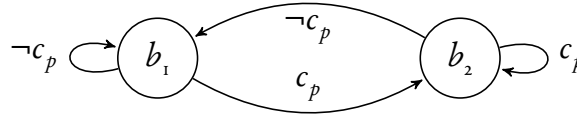


Fig. 6.6 – Modèle comportemental. b_1 : comportement normal, b_2 : comportement de panique

L'état b_1 représente le comportement d'un agent dans des conditions viables et l'état b_2 son comportement dans des conditions non viables. La transition entre ses deux états dépend d'une condition c_p :

$$c_p := T > T_{max}, \quad (6.12)$$

où T représente la température ressentie par l'agent et T_{max} la température maximale supportée par une larve.

Dans l'état b_2 , l'agent se déplace de manière aléatoire dans l'environnement. Le comportement normal est lui un peu plus compliqué et est défini comme un automate à trois états, chaque état représentant l'un des comportements fondamentaux d'un agent (figure 6.7). Une variable j est introduite pour représenter l'état du jabot de l'agent ($j = 0$ si le jabot est vide, $j = 1$ s'il est rempli). De plus, nous définissons la vitesse de remplissage et de vidage du jabot en fonction du temps t (min) comme une fonction de la température T (°C), i.e.,

$$\frac{dj}{dt} = \pm \begin{cases} 0 & \text{si } T \leq T_{min} \\ 1 & \text{si } T \geq 25 + T_{min} \\ \frac{T - T_{min}}{25} & \text{sinon,} \end{cases} \quad (6.13)$$

où T_{min} est la température minimale à laquelle une larve est active.

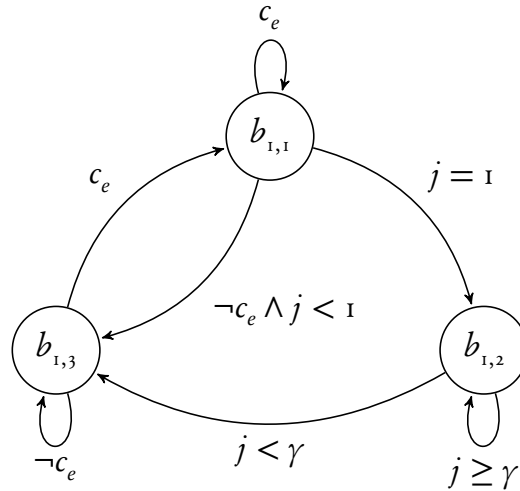


Fig. 6.7 – Modèle comportemental d'une larve en conditions normales. $b_{1,1}$: alimentation, $b_{1,2}$: optimisation de la température, $b_{1,3}$: recherche de nourriture

L'état $b_{1,1}$ représente le comportement alimentaire : l'agent est immobile et consomme les ressources de la cellule de l'environnement sur laquelle il se trouve. Le modèle permettant de déterminer la quantité de ressource consommée $r(g)$ est issu de Godoy *et al.* (1996) :

$$\frac{dr}{dt} = \frac{m \cdot f_4(T(t))}{0.39}, \quad (6.14)$$

où m représente la masse d'un individu au stade prépupe et f_4 la vitesse de développement pour atteindre le stade prépupe. Un agent se nourrit sur une cellule si une condition c_e est vraie. c_e est définie comme suit :

$$c_e := (n \geq n_{min}) \wedge (j < 1) \wedge (T < T_{max}) \wedge ((r \geq r_{min}) \vee (j = 0)), \quad (6.15)$$

où n représente la quantité de ressources nutritives disponibles dans la cellule, n_{min} une quantité minimale de nourriture pour laquelle la cellule est attractive, r le nombre d'agents larve sur la cellule et r_{min} un nombre minimal d'agents permettant de définir une agrégation sur une cellule. Cette condition peut être interprétée de la façon suivante : *un agent se nourrit sur une cellule si les conditions de viabilité sont respectées et si une agrégation est située sur la cellule ou si le jabot de l'agent est vide*. De plus, les agents dans l'état $b_{1,1}$ émettent un signal attractif g caractérisé par un coefficient de diffusion g_d et un coefficient d'évaporation g_e (tableau 6.2).

L'état $b_{1,2}$ est un comportement d'optimisation de la température : l'agent recherche la cellule lui permettant d'optimiser sa vitesse de développement. L'algorithme sur lequel repose ce comportement est un suivi de gradient. L'agent reste dans l'état $b_{1,2}$ jusqu'à ce que la valeur de son jabot soit inférieure à une certaine valeur i.e., $j < \gamma$.

L'agent passe alors dans l'état $b_{1,3}$ représentant un comportement de recherche de nourriture. Dans cet état, l'agent cherche à s'agréger à d'autres larves. De même que dans l'état $b_{1,2}$, l'algorithme sur lequel repose ce comportement est un suivi de gradient.

La perception de l'agent n'est pas optimale. Ainsi des seuils différentiels de perception sont déterminés pour les variables d'environnement utilisées dans l'algorithme de suivi de gradient. E.g., pour une variable d'environnement donnée, soit v sa valeur et Δv son seuil différentiel de perception. Cela signifie que l'agent ne sera pas capable de différencier v et $v \pm \Delta v$. Les niveaux de perception pour la température T et le signal d'agrégation g sont appelés respectivement ΔT et Δg .

6.3 Implémentation

6.3.1 Couplage Modèles/Agents

Le code associé aux agents a été totalement découplé du code des modèles que les agents encapsulent. Un agent ne dispose donc que de primitives lui permettant d'exécuter différents types de modèles (e.g., un modèle de développement) mais ignore quels modèles en particulier il encapsule (e.g., le modèle des ADH). Cette architecture permet ainsi de définir très facilement de nouveaux modèles. Concrètement, on associe à chaque agent un ensemble d'interfaces JAVA représentant les méthodes associées aux modèles.

On considère deux types d'agents : *Layer*, représentant une population de diptères gravides et *Larva*, représentant une larve de diptère.

6.3.2 Agents informatiques

6.3.2.1 L'agent *Layer*

L'agent *Layer* crée des agents *Larva*, leur associe un ensemble de modèles (comportement, développement, etc.) propres à l'espèce qu'il représente et enfin les situe dans l'environnement. Les pas de simulation auxquels les agents *Larva* sont « pondus » sont calculés hors ligne (avant la simulation) à partir des données climatiques fournies et des modèles de population et de ponte associés à chaque *Layer*.

6.3.2.2 L'agent *Larva*

Une approche particulièrement basique pour implémenter le modèle comportemental de l'agent *Larva* aurait été d'associer à chaque pas de simulation un comportement atomique. Cela pose pourtant plusieurs problèmes. Tout d'abord, l'agent peut changer de comportement au cours d'un même pas de simulation. Deuxièmement, les interactions entre les agents et l'environnement dépendent, notamment, de leur taux de développement. Ainsi un agent au stade L3 se déplace et se nourrit plus rapidement qu'un agent au stade L2. La notion de comportement atomique n'aurait donc pas été la même selon le taux de développement des agents. La solution adoptée consiste à associer, pour chaque agent un nombre d'actions fondamentales n_a , que l'agent aura le droit d'exécuter durant un pas de simulation. n_a dépend de la vitesse de déplacement de l'agent et est réévalué à chaque pas de simulation :

Les actions associées à chacun des comportements sont ainsi définis comme suit :

- comportements de déplacement ($b_2, b_{1,2}, b_{1,3}$) : l'agent se déplace d'une cellule.
- comportement de nourrissage (b_1) : l'agent consomme une quantité de ressource Δr de la cellule dans laquelle il se trouve égale à

$$\Delta r = \frac{m \cdot f_4(T(t)) \cdot \Delta t}{0.39}, \quad (6.16)$$

avec

$$\Delta t = \frac{\Delta p}{n_a}, \quad (6.17)$$

où Δp représente la durée d'un pas de simulation.

La valeur de la variable j , représentant l'état du jabot est évaluée à la fin de chaque action fondamentale :

$$\Delta j = \pm \begin{cases} 0 & \text{if } T \leq T_{min} \\ \Delta t & \text{if } T \geq 25 + T_{min} \\ \Delta t \cdot \frac{T - T_{min}}{25} & \text{sinon.} \end{cases} \quad (6.18)$$

j est ainsi décrémenté dans les états $b_2, b_{1,2}$, et $b_{1,3}$ et incrémenté dans l'état b_1 . L'algorithme 3 est exécuté par chaque agent *Larva* à chaque pas de simulation.

Algorithme 3 Algorithme exécuté à chaque pas de simulation par les agents *Larva*

- 1: **Si** L'agent est dans un stade actif **Alors**
 - 2: **Pour** $i = 1$ to n_a **Faire**
 - 3: Effectuer une action fondamentale
 - 4: Mettre à jour la valeur de j
 - 5: **Fin pour**
 - 6: **Fin si**
 - 7: Evoluer
-

Le principal problème lié à cette approche est l'apparition d'artefacts de simulation liés à la séquentialité des actions. Un processus de calibrage a été réalisé afin de déterminer une durée de pas de simulation, significative pour l'utilisateur, et la plus grande possible, telle que l'on

n'observe pas d'artefact de simulation. Concrètement, une analyse qualitative de différentes simulations fut conduite par le thématicien, les facteurs observés étant la structuration et de la dynamique des masses et de la consommation de l'environnement. Ainsi, pour l'ensemble des résultats présentés par la suite, les simulations ont été réalisées avec $p = 1$ min.

6.4 Validation du modèle et résultats de simulation

6.4.1 Validation des modèles de développement et de variabilité

Les données et les modèles de développement utilisés étant issus de la littérature, nous avons avant tout cherché à valider le modèle de variabilité (cf. section 5.2.3.2) et l'implémentation des différents modèles. Les résultats obtenus montrent que le modèle reproduit avec une grande précision la variabilité intraspécifique de la vitesse de développement observée dans différentes conditions. Des exemples à températures différentes sont présentés dans la figure 6.8 pour l'espèce *Lucilia sericata*.

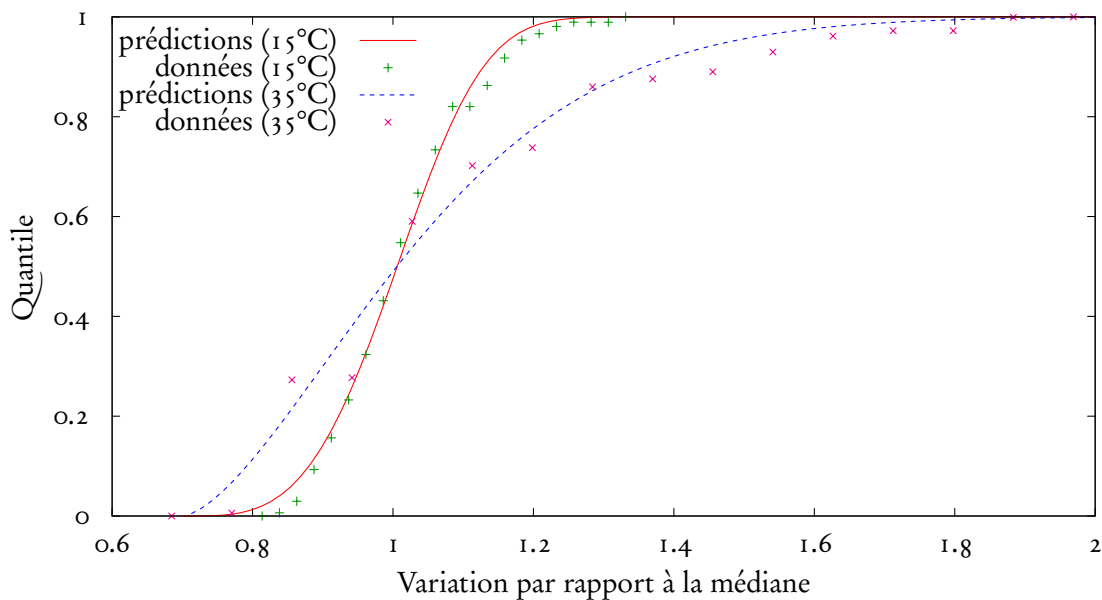


Fig. 6.8 – Variabilité intraspécifique de la vitesse de développement observée et prédite pour l'espèce *Lucilia sericata* à 15 et 35°C

6.4.2 Validation du modèle thermique du corps

Le principal problème lié à la validation du modèle thermique est la difficulté à conduire des expériences sur des cadavres, la législation étant très stricte en France. Ainsi, dans un premier temps, nous considérons que le modèle de Henssge⁴⁶ est, sous température fixe, valide. Nous avons ainsi confronté les prédictions de notre modèle et de celui de Henssge. La

⁴⁶Le modèle de Henssge, décrit dans de nombreuses publications, e.g., Henssge (1988, 1992, 2004), permet de prédire l'évolution de la température d'un corps humain sous température fixe. Il est généralement considéré par

figure 6.9 donne les écarts maximaux observés entre les prédictions des deux modèles en fonction de la masse du corps, dans l'intervalle de validité du modèle de Henssge. Les écarts sont, relativement à la précision attendue, très faible et permettent de valider notre modèle sous température fixe.

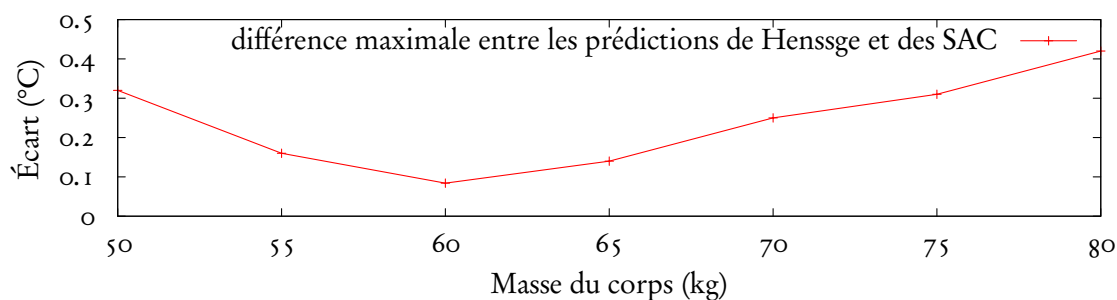


Fig. 6.9 – Différence maximale entre les prédictions de Henssge et des SAC

La validation du modèle sous température variable est basée sur des expériences réalisées sur un modèle porcin dans un environnement non contrôlé. De la difficulté à conduire de telles expériences résulte un nombre de répliquats très faible. Toutefois, les résultats obtenus sont très encourageants comme le montre la figure 6.10.

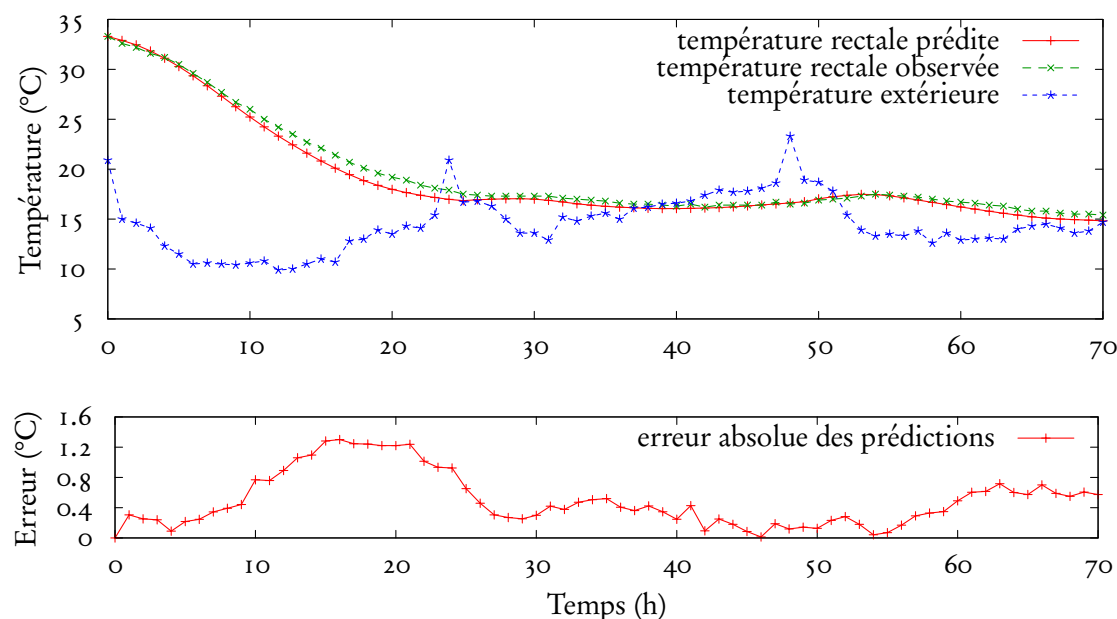


Fig. 6.10 – Évolution de la température rectale (observation et prédiction) d'un porc de 30 kg sous températures variables et erreur absolue des prédictions

les médecins légistes comme le plus fiable des modèles permettant de déterminer un IPM à l'aide de la méthode thermométrique (Verica *et al.*, 2007).

Outre son intérêt prédictif évident, ce modèle a permis de montrer que le plateau thermique observé dans les premières heures après la mort (figure 5.3) est très certainement une conséquence des lois de conduction et convection, et n'est donc pas lié à l'activité microbienne post-mortem comme cela avait pu être supposé (Veremme *et al.*, 2008). Par ailleurs, ce modèle a été utilisé pour quantifier la qualité des estimations d'IPM réalisés à l'aide du modèle de Henssge sous température fixe (Morvan *et al.*, 2009a). Nous avons ainsi montré que lorsque la température varie de manière cyclique, l'utilisation du modèle de Henssge pouvait induire des erreurs importantes (figure 6.11).

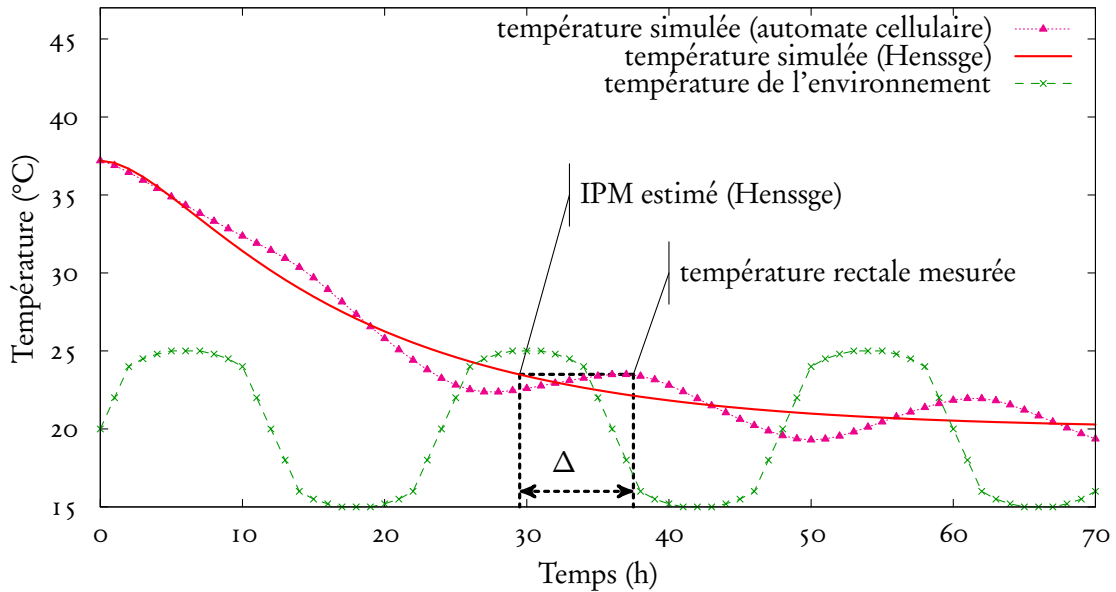


Fig. 6.11 – Différence entre l'IPM réel et calculé à l'aide du modèle de Henssge lorsque la température de l'environnement varie (amplitude 10°C, période 24 h)

6.4.3 Calibrage et validation du modèle comportemental

Les modèles décrits aux sections 6.2.3.2 et 6.2.3.1 comportent un certain nombre de paramètres devant être calibrés :

- le niveau de remplissage du jabot déclenchant le comportement de recherche de nourriture γ ,
- l'émission de chaleur λ ,
- le nombre minimal d'agents déterminant une agrégation r_{min} ,
- la quantité minimale de ressources nutritives sur une cellule « intéressante » n_{min} ,
- la température maximale pour laquelle le comportement d'une larve est normal T_{max} ,
- les seuils différentiels de perception pour la température et le signal d'agrégation, respectivement ΔT et Δg .

Ce modèle a été calibré en deux temps. Tout d'abord, nous avons cherché à déterminer pour chaque paramètre l'ensemble des valeurs permettant de reproduire des comportements d'aggrégations de larves, notamment en présence de sources de chaleur, observés en laboratoire. Une majorité des paramètres du modèle comportemental ont été calibrés ainsi. La se-

conde validation est quantitative. Nous avons cherché à reproduire un phénomène stable associé à la régulation de la température sur le corps : l'élévation maximale de température relevée dans l'agrégation durant l'ensemble du développement des larves en fonction de la température extérieure T_{ext} et du nombre de larves dans la masse r (Slone et Gruner, 2007; Charabidze, 2008). Les données sont issues d'expériences réalisées *in vitro* à l'institut de médecine légale de Lille. Slone et Gruner (2007) considèrent, d'après des expériences réalisées en conditions naturelles, que la température maximale ne dépend, pour des masses suffisamment importantes, que du volume de la masse. Ce paramètre reste difficile à quantifier précisément et n'a donc pas été utilisé.

6.4.3.1 Calibrage qualitatif

L'objectif de cette validation est de reproduire le comportement d'une agrégation observée *in vitro*. On considère différents scénarios expérimentés dans un environnement de test de 80 · 80 cm. Dans un premier temps, on considère un environnement homogène. L'objectif est principalement de réduire l'ensemble des solutions possibles en identifiant les « valeurs limites » des paramètres permettant de reproduire certains patterns simples (formation d'agrégat, déplacement dans l'environnement, etc.)

Dans un second temps, on considère un environnement dans lequel des sources de chaleur sont disposées à des emplacements donnés puis activées durant l'expérience. Les sources de chaleur sont placées de sorte que les larves puissent atteindre les différentes sources de chaleur. Voici un exemple d'expérience réalisée :

- une première source de chaleur produisant une température locale de 25°C est ainsi activée 24 heures après la ponte d'environ 200 œufs,
- une seconde source, produisant une température locale de 32°C et située à environ 50 cm de la première est activée 24 heures plus tard.

Ce dispositif permet d'observer en laboratoire la colonisation successive des sources peu de temps après leur activation. Le tableau 6.5 récapitule les données relatives à l'implémentation des sources de chaleur dans les simulations.

	Position	Température (°C)	Activation (h)
Source 1	(28, 8)	25	24
Source 2	(32, 64)	32	48

Tab. 6.5 – Caractéristiques des sources de chaleur

Nous considérons trois observations que les simulations devraient reproduire :

- une agrégation initiale compacte,
- la colonisation de la source 1,
- la colonisation de la source 2.

À chacune de observations, on associe une appréciation parmi :

- parfaitement reproduite,
- partiellement reproduite,
- non reproduite.

Différents jeux de paramètres sont ainsi testés itérativement, l'objectif étant de reproduire parfaitement l'ensemble des observations. Le processus de recherche est guidé par les notes associées aux observations des simulations précédentes. Certaines simulations sont par ailleurs annotées par le thématicien lorsque celui-ci considère que des informations supplémentaires devraient être considérées.

À la suite de ce processus, nous avons pu identifier ou restreindre le domaine de validité de certains paramètres. Les résultats sont synthétisés dans le tableau 6.6.

Paramètre	Valeurs	Unité
γ	0.87	–
r_{min}	[10, 20]	–
n_{min}	$9 \cdot 10^{-3}$	kg
ΔT	0.02	°C
Δg	$95 \cdot 10^{-9}$	–
T_{max}	[30, 45]	°C

Tab. 6.6 – Paramètres du modèle identifiés par calibrage qualitatif

6.4.3.2 calibrage quantitatif

Dans un second temps, nous avons confronté un ensemble de simulations à des données quantitatives caractérisant un phénomène considéré comme stable, l'élévation maximale de température relevée dans l'agrégation en fonction de la température extérieure T_{ext} et du nombre de larves dans la masse r afin de déterminer les valeurs optimales de λ , T_{max} et r_{min} .

Le principal problème lié à cette validation fut l'analyse des données. En effet, la difficulté à conduire des expériences de cette nature en conditions contrôlées, n'a pas permis de réaliser un nombre de répliques suffisant pour s'assurer que les résultats agrégés convergeaient bien vers la moyenne. De plus, il n'a pas été possible de réutiliser des données publiées, les protocoles expérimentaux étant trop peu détaillés pour être simulés. En résumé, les données utilisées pour le calibrage sont caractérisées par une forte variabilité. Ainsi, l'utilisation seule de métriques « classiques » comme l'erreur quadratique moyenne (MSE) ou l'erreur moyenne (ME) ne semble pas suffisante pour déterminer les paramètres optimaux (Casella et Berger, 2001). Ainsi, nous avons également considéré l'erreur systématique associée aux paramètres de la simulation, ici T_{ext} et r . Il s'agit de projeter sur les axes des différents paramètres les erreurs moyennes et de déterminer par régression une relation linéaire entre le paramètre et l'erreur moyenne. Les figures 6.12 et 6.13 montrent l'erreur quadratique moyenne et l'erreur moyenne en fonction de λ et T_{max} mesurées à partir de la comparaison des résultats expérimentaux et des simulations. Le paramètre r_{min} a peu d'influence sur les résultats : les courbes représentant les niveaux $r_{min} = 20$ et $r_{min} = 10$ sont, à de très légères différences près, superposées. Il est ainsi possible d'identifier une région optimale de l'espace de recherche mais non un singleton. L'analyse de l'erreur systématique associée aux paramètres de la simulation permet en revanche d'identifier les paramètres $\lambda = 0.0003$ et $T_{max} = 40$ comme optimaux. Ces résultats, présentés dans les figures 6.14 et 6.15 permettent de montrer que le modèle

sous-estime légèrement l'élévation de température liée à l'activité des insectes pour des températures élevées ($\gtrsim 30^{\circ}\text{C}$) et des petites populations ($\lesssim 2000$ individus) ; en revanche elle est légèrement surestimée pour des températures faibles et moyennes ($\lesssim 30^{\circ}\text{C}$) et des populations importantes ($\gtrsim 2000$ individus).

6.5 Résumé du chapitre

Dans ce chapitre nous avons présenté un modèle à base d'agent du développement des diptères nécrophages dans un environnement complexe. Ce modèle permet, à la différence des modèles utilisés classiquement en entomologie médico-légale, de simuler la température locale à laquelle se développent les diptères et ce grâce à un modèle thermique du corps humain basé sur la notion d'automate cellulaire et un modèle à base d'agent du comportement des diptères nécrophages.

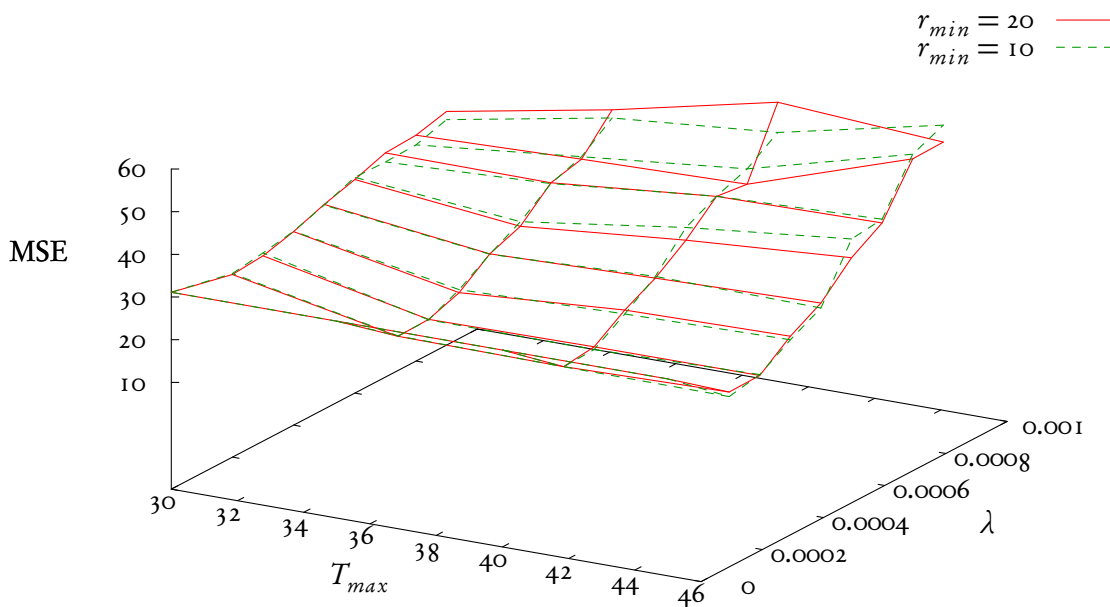
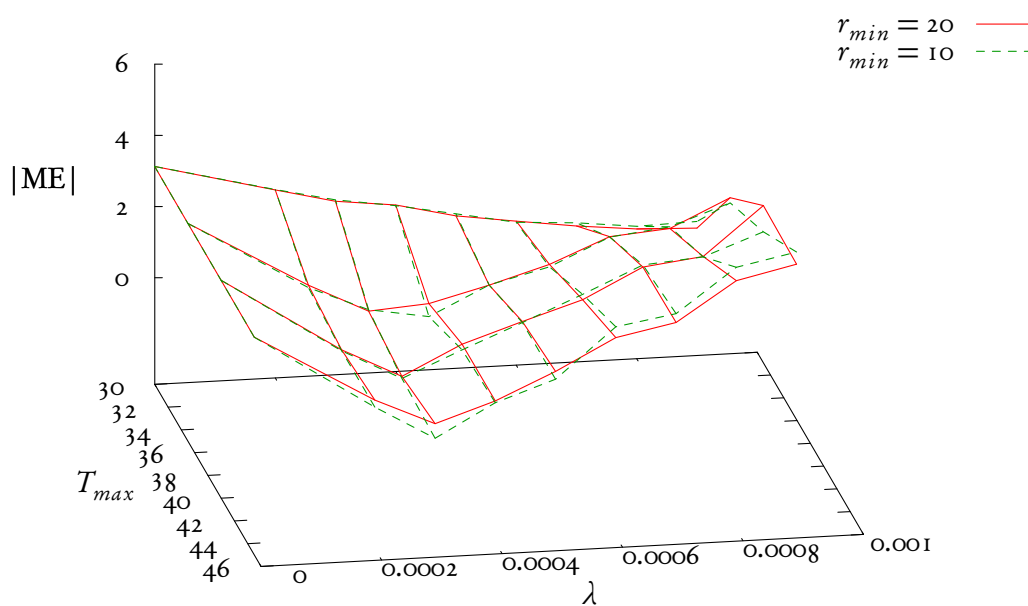
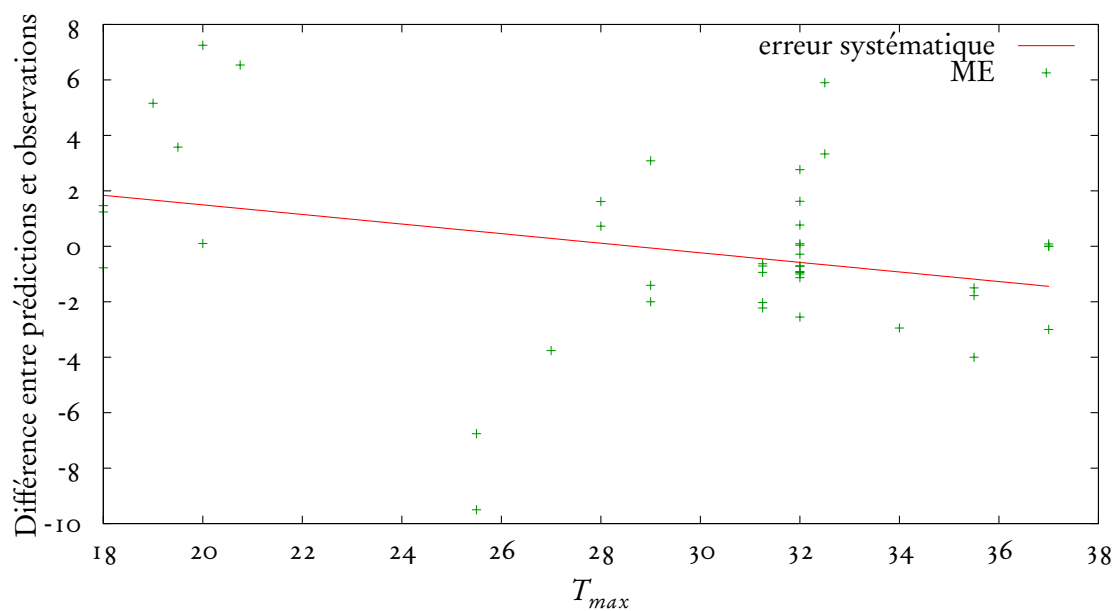


Fig. 6.12 – Erreur quadratique moyenne (MSE) en fonction de λ et T_{max}

Fig. 6.13 – Valeur absolue de l'erreur moyenne ($|ME|$) en fonction de λ et T_{max} Fig. 6.14 – Erreur systématique associée à T_{ext} pour $\lambda = 0.0003$ et $T_{max} = 40$

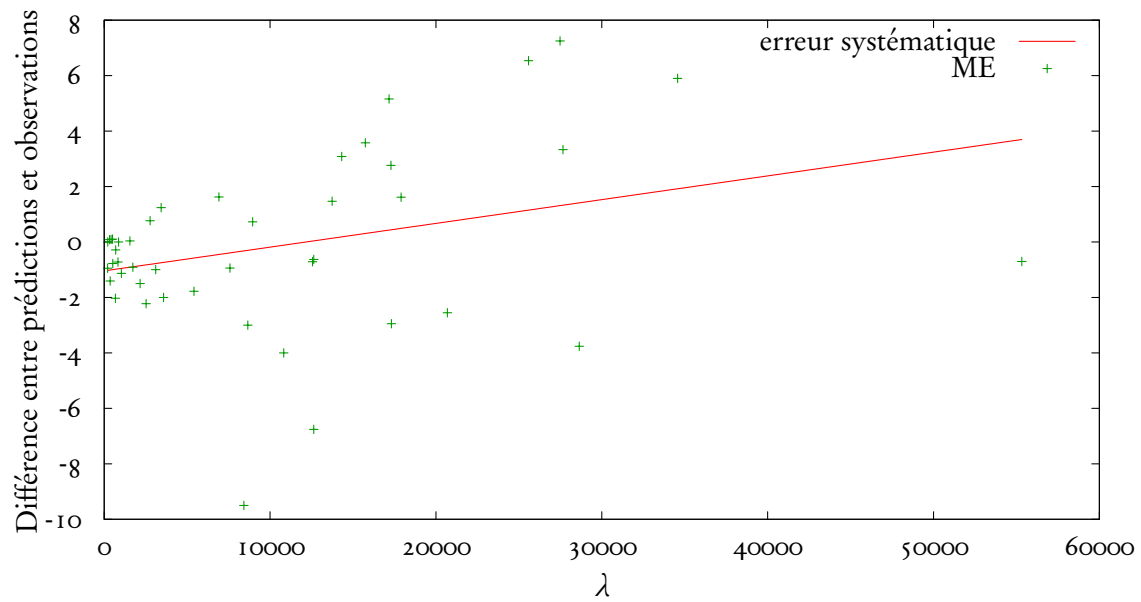


Fig. 6.15 – Erreur systématique associée à r pour $\lambda = 0.0003$ et $T_{max} = 40$

7

Entomologie médico-légale et aide à la décision

Sommaire

7.1	Architecture du système d'aide à la décision	100
7.1.1	Stockage des données et construction des simulations	100
7.1.2	Résolution de problèmes inverses	101
7.2	Règles de décision	104
7.2.1	Taux de développement accumulé	106
7.2.2	Stade de développement	107
7.2.3	Espèces retrouvées sur le corps	108
7.3	Résultats	108
7.3.1	Premier exemple	108
7.3.2	Second exemple	109
7.4	Résumé du chapitre	109

Dans ce chapitre, nous présentons l'implémentation de l'architecture du système d'aide à la décision (SIAD) décrite dans le chapitre 3 d'un point de vue technique et applicatif. Cette implémentation repose sur un ensemble de composants logiciels indépendants publiés sous licence libre. La figure 7.1 représente l'ensemble des composants utilisés et leurs interactions. Nous détaillons par la suite les technologies utilisées pour le stockage des données, la persistance, la construction des modèles et des règles de décision et l'implémentation de la méthode de résolution de problèmes inverses décrite dans le chapitre 4. L'implémentation du simulateur a été décrite dans la section 6.3. Nous présentons enfin des résultats d'expertises entomologiques réalisées avec le système.

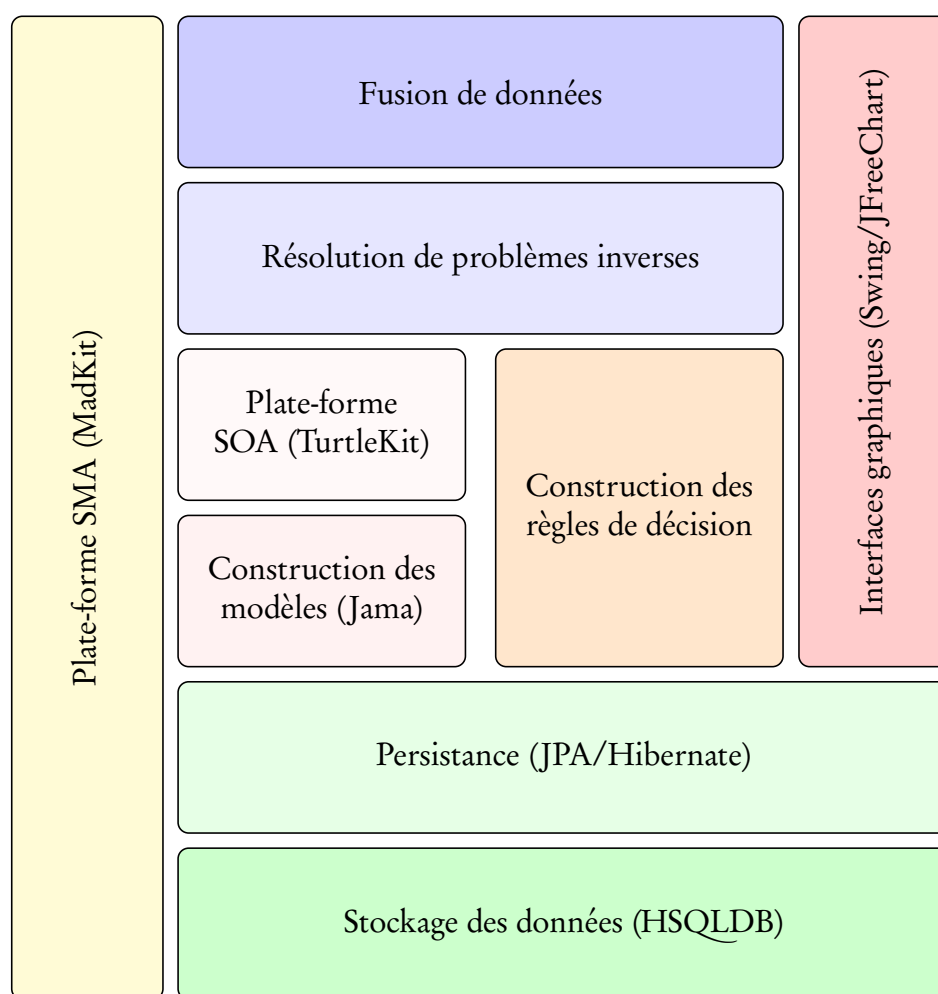


Fig. 7.1 – Composants logiciels utilisés pour l’implémentation du SIAD ; le nom des composants tiers est donné entre parenthèses

7.1 Architecture du système d’aide à la décision

7.1.1 Stockage des données et construction des simulations

Les données nécessaires à l’exécution du logiciel et en particulier des simulations (données expérimentales, liées aux expertises, aux modèles, etc.) sont stockées dans une base de données relationnelle embarquée. Le système de gestion de base de données est HSQLDB⁴⁷, choisi pour sa rapidité et sa petite taille. Une couche de persistance basée sur l’implémentation de JPA (Biswas et Ort, 2006) par Hibernate⁴⁸ permet de sauvegarder et de restaurer facilement des données en base.

⁴⁷<http://hsqldb.org>

⁴⁸<https://www.hibernate.org>

L'utilisateur a la possibilité de choisir indépendamment les modèles (e.g., le modèle de développement) et les données expérimentales qui seront utilisées pour calculer les paramètres des modèles utilisés dans les simulations. Plusieurs jeux de données issus de différentes publications ou expériences peuvent être utilisés simultanément. De même différents modèles peuvent être choisis : des expertises seront réalisées avec chacun d'entre eux. Les paramètres des modèles sont calculés par une implémentation de l'algorithme de Levenberg-Marquardt (Marquardt, 1963) basée sur Jama ⁴⁹, un package de calcul matriciel.

Afin de déterminer la validité des simulations et des paramètres (dans notre application, nous considérons un paramètre unique : l'heure du décès) de la simulation, il est nécessaire de construire des règles de décision à partir des informations fournies par l'expert (cf. section 3.2). Ces règles de décision sont détaillées dans la section 7.2.

7.1.2 Résolution de problèmes inverses

La méthode de résolution de problèmes inverses présentée dans le chapitre 4 a été implémentée de sorte que l'algorithme de résolution ainsi que les heuristiques associées puissent être définies indépendamment de la nature des hypothèses et du modèle utilisé pour leur validation. La figure 7.2 présente les classes et interfaces JAVA utilisées pour représenter les notions d'hypothèses, de distances entre hypothèses, d'espace de recherche et d'ensemble solution. Ces interfaces doivent être implémentées pour être utilisées par l'algorithme de résolution. Des classes abstraites, implémentant ces interfaces ont été développées afin de faciliter le processus d'intégration de nouveaux types d'hypothèse en définissant certaines méthodes permettant d'ajouter, d'enlever des hypothèses ou encore de calculer la distance entre deux hypothèses.

7.1.2.1 Implémentation objet de l'algorithme de résolution

L'algorithme de résolution a tout d'abord été implémenté dans une classe *AbductiveInquiry*. Là aussi, les objets utilisés par les algorithmes (heuristique, modèle, condition d'arrêt, méthode d'interpolation) sont définis de manière générique à l'aide d'interfaces (figure 7.3). Une méthode de cette classe permet de comparer les différentes heuristiques utilisées pour le choix des hypothèses à valider selon l'objectif. Des rapports peuvent être ainsi générés aux formats L^AT_EX et pdf, si les logiciels requis (gnuplot, pdftex, TikZ/Pgf) sont présents sur le système, ou au format texte. C'est cette implémentation qui a été utilisée pour calculer les résultats présentés dans le chapitre 4.

7.1.2.2 Implémentation agent de l'algorithme de résolution

Bien qu'elle utilise les mêmes interfaces que l'implémentation objet, cette implémentation de l'algorithme, sur lequel repose le système d'aide à la décision, est quelque peu différente. L'algorithme est implémenté dans un agent MadKit (*ForenSeekExpertise*) et un simulateur TurtleKit (*ForenSeekLauncher*), implémentant la classe *Model*, est défini. L'algorithme a été modifié de sorte que plusieurs simulations puissent être lancées simultanément. Cependant, cela peut entraîner une dégradation de la qualité de la solution : il est en effet impossible

⁴⁹<http://math.nist.gov/javanumerics/jama>

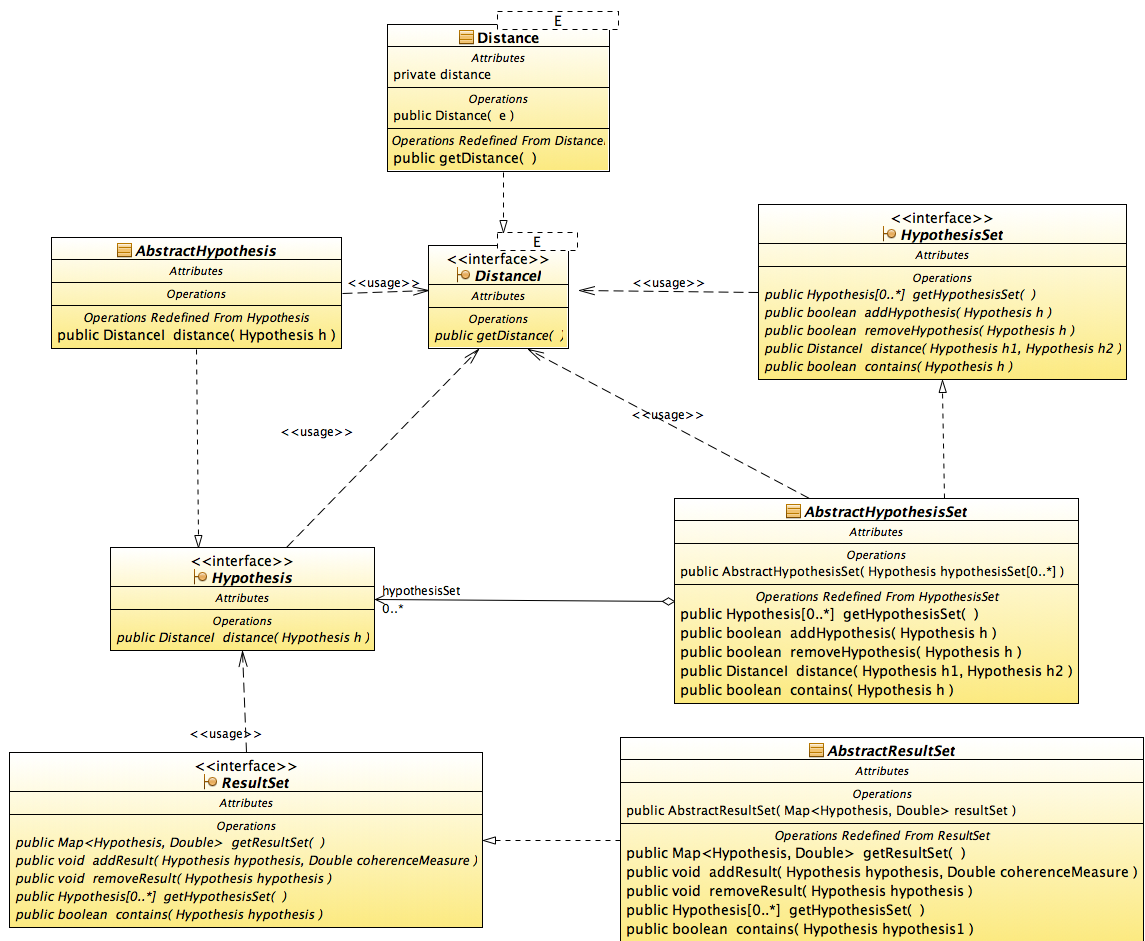


Fig. 7.2 – Diagramme de classes représentant les notions d'hypothèse et d'ensemble solution

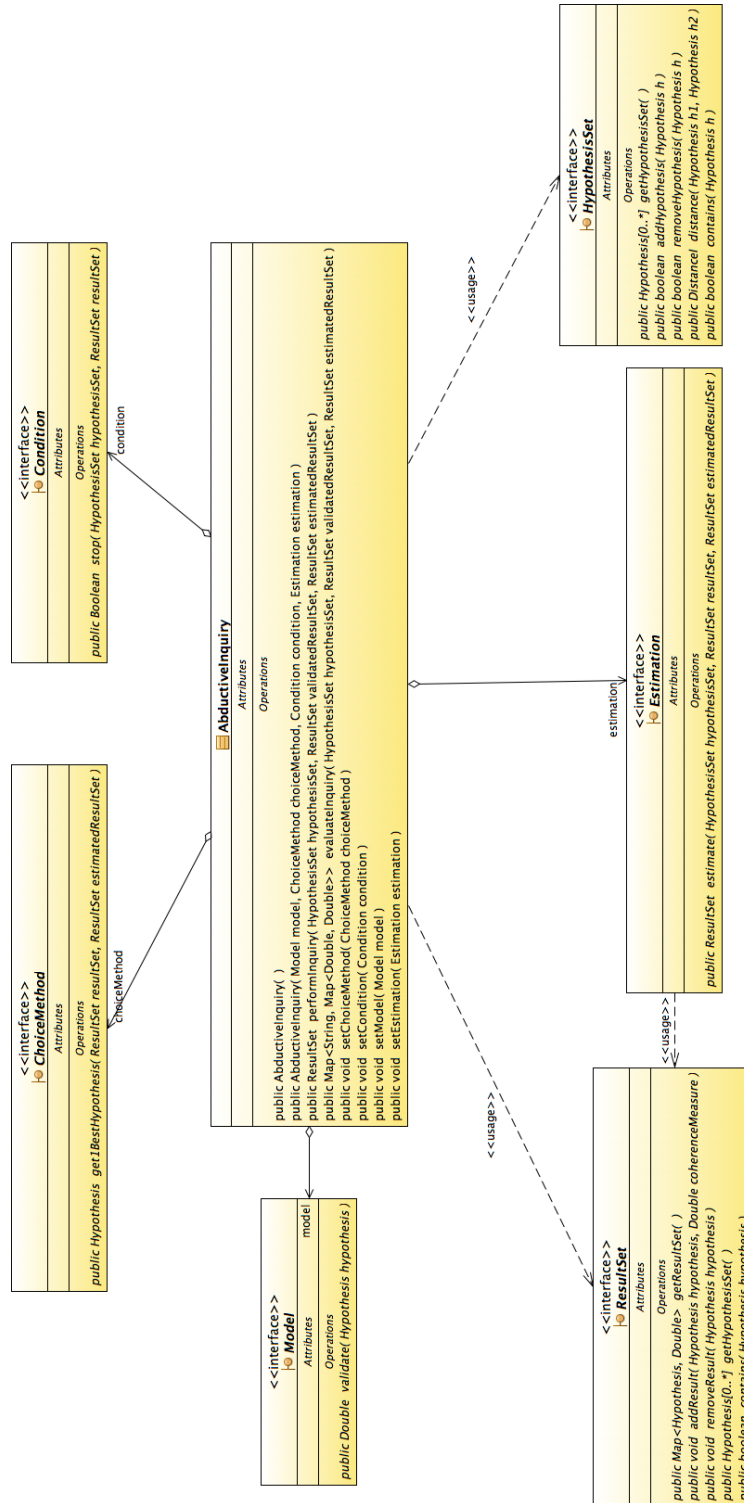


Fig. 7.3 – Diagramme de classes associé à l'implémentation objet de l'algorithme de résolution

de déterminer précisément la mesure de cohérence associée à une hypothèse avant la fin du processus de validation. Rappelons que l'on appelle :

- Φ l'ensemble des hypothèses,
- Φ^v l'ensemble des hypothèses validées,
- c_{m,ϕ,Ω^*} la mesure de cohérence associée à une hypothèse ϕ de Φ ,
- $\mathcal{C}_{m,\Phi^v,\Omega^*}$ l'ensemble des mesures de cohérence c_{m,ϕ,Ω^*} ,
- $\widehat{c_{m,\phi,\Omega^*}}$ la mesure de cohérence estimée associée à une hypothèse ϕ de Φ
- $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$ l'ensemble des mesures de cohérence estimées.

L'agent *ForenSeekExpertise* détermine successivement k hypothèses qui seront validées simultanément par autant d'agents *ForenSeekLauncher*. Le choix optimal de k dépend notamment de l'architecture matérielle de l'ordinateur sur lequel s'exécute le système. Au début de la validation de chaque hypothèse ϕ de Φ , celle-ci est ajoutée à Φ^v . De plus, la mesure de cohérence estimée de ϕ avant sa validation, $\widehat{c_{m,\phi,\Omega^*}}$, est ajoutée à l'ensemble solution $\mathcal{C}_{m,\Phi^v,\Omega^*}$. Durant la validation de ϕ , l'agent *ForenSeekLauncher* met à jour régulièrement $\widehat{\mathcal{C}_{m,\Phi^v,\Omega^*}}$, en modifiant c_{m,ϕ,Ω^*} . *ForenSeekExpertise* met quant à lui à jour, par interpolation, $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$. Lorsque un agent *ForenSeekLauncher* termine la validation de ϕ , celui-ci affecte la valeur définitive de c_{m,ϕ,Ω^*} à $\mathcal{C}_{m,\Phi^v,\Omega^*}$, puis envoie un message à *ForenSeekLauncher* pour l'informer qu'il a fini de valider ϕ et que celui-ci peut donc recalculer $\widehat{\mathcal{C}_{m,\Phi,\Omega^*}}$ et choisir une nouvelle hypothèse.

On peut ainsi noter quelques différences notables avec l'algorithme 1 :

- les hypothèses à valider sont ajoutées à Φ^v au début du processus de validation, et non à la fin pour que la même hypothèse ne puisse être validée plusieurs fois par différents *ForenSeekLauncher* ; Φ^v ne représente donc plus l'ensemble des hypothèses validées, mais l'ensemble des hypothèses validées ou en cours de validation.
- le choix des nouvelles hypothèses à valider, qui n'est plus basé sur des mesures de cohérences calculées mais sur des estimations raffinées itérativement par un agent *ForenSeekLauncher*,
- consécutivement, les mesures de cohérences de $\mathcal{C}_{m,\Phi^v,\Omega^*}$ peuvent être des estimations ; l'ensemble solution est donc réellement calculé à la fin du calcul des différentes mesures de cohérence.

La figure 7.4 résume les interactions entre les différents types d'agents et les ensembles solutions et d'hypothèses manipulés durant la résolution du problème.

Une interface réalisée avec JFreeChart permet d'observer en temps réel l'état d'avancement de la résolution du problème pour chacun des modèles utilisés (figure 7.5). L'interface est totalement paramétrable (titres, échelle, couleurs, etc.) et les résultats sont exportables sous la forme d'images.

7.2 Règles de décision

Afin de déterminer la mesure de cohérence associée à une hypothèse, des règles sont construites à partir des informations fournies par l'utilisateur du système. Rappelons que pour être valide, une simulation ne doit violer aucune règle (équation 3.4). Dans le cadre de l'application à l'entomologie médico-légale, nous considérons essentiellement deux règles, relatives au développement des larves de diptères. D'autres règles sont également définies afin de

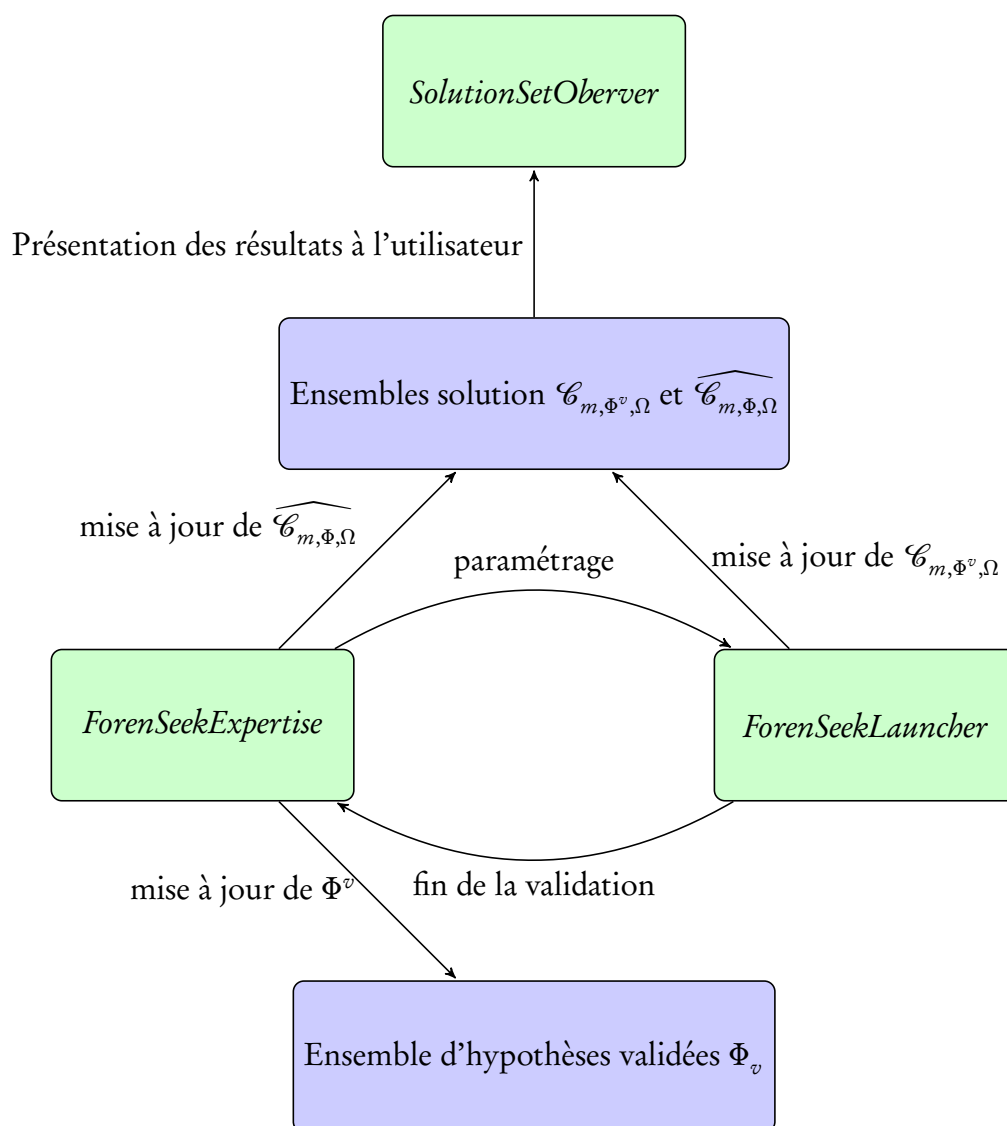


Fig. 7.4 – Interactions entre les différents types d'agents et les ensembles manipulés durant la résolution du problème



Fig. 7.5 – Interface de visualisation de l'état d'avancement de la résolution d'un problème

répondre éventuellement à d'autres questions : le corps a-t-il été inaccessible après le décès ? a-t-il été déplacé ? Nous cherchons ainsi à confronter des données plus générales (présence ou absence de certaines espèces) aux simulations. Par ailleurs, différents types de règles sont implémentées. Certaines règles sont évaluées à chaque pas de simulation (l'observation des agents repose alors sur la méthode décrite à la section 3.3.3.1 : les agents s'auto-observent pour déterminer s'il doivent être observés), alors que d'autres sont évaluées à la fin de la simulation. Nous adoptons la convention suivante :

- les règles évaluées durant la simulation sont identifiées par des lettres majuscules (e.g., R),
- les règles évaluées à la fin de la simulation sont identifiées par des lettres majuscules ornées d'un point (e.g., $\dot{R}$)

Les règles évaluées durant la simulation ne peuvent permettre d'établir la validité d'une simulation. Elles sont en revanche utilisées pour établir la non validité de la simulation avant son terme et optimiser ainsi le temps de calcul.

7.2.1 Taux de développement accumulé

La principale donnée utilisée par les entomologistes pour déterminer un IPM est, pour chaque espèce, le taux de développement accumulé par l'individu le plus âgé retrouvé sur ou près du corps (équation 5.11). Cette donnée est connue avec une certaine imprécision liée aux conditions de conservation des échantillons et à la précision des mesures. Elle est généralement donnée sous la forme d'un intervalle. Ainsi, on notera $a^e = [a_{min}^e, a_{max}^e]$ le taux

de développement accumulé par le ou les individus les plus développés de l'espèce e . On peut donc définir, pour chaque espèce retrouvée, la règle suivante, évaluée à la fin de la simulation :

$$\dot{R}_a^e = \begin{cases} \top & \text{si pour l'agent } Larva \ i \text{ le plus développé de } e \ a_i \in a^e \\ \perp & \text{sinon,} \end{cases} \quad (7.1)$$

où a_i représente le taux de développement accumulé de l'individu i .

Il est souvent possible de déterminer si cette règle sera violée avant la fin de la simulation. Ainsi, si aucun agent *Larva* de l'espèce e ne peut atteindre a_{min}^e en supposant qu'il se développe à sa température optimale T_{opt}^e , alors cela veut dire que la simulation est fausse. De plus, si le taux de développement accumulé d'un agent devient supérieur à a_{max}^e , alors la simulation est fausse :

$$R_a^e = \begin{cases} \perp & \text{s'il existe un agent } Larva \ i \text{ de } e \text{ tel que } a_i > a_{max}^e \text{ ou} \\ & \text{si pour tout agent } Larva \ j \text{ de } e, \\ & a_j + (p_n - p) \cdot var^e(q_j, T_{opt}^e) \cdot f^e(T_{opt}) < a_{min}^e \\ \top & \text{sinon,} \end{cases} \quad (7.2)$$

où $(p_n - p)$ représente le nombre de pas restant avant la fin de la simulation et var^e et f^e représentent respectivement les modèles de variabilité de développement et de développement associés à l'espèce e . $a_i + (p_n - p) \cdot var^e(q_i, T_{opt}^e) \cdot f^e(T_{opt})$ représente le taux de développement accumulé par un agent i après le dernier pas de temps p_n si la température est optimale durant le reste de la simulation.

7.2.2 Stade de développement

Parfois, il est impossible de déterminer avec suffisamment de précision le taux de développement des individus prélevés pour les raisons suivantes :

- les conditions de conservation des prélèvements sont inconnues,
- les prélèvements ont été fixés,
- des pupes vides ont été retrouvés (des individus ont donc émergé).

Il est alors nécessaire d'utiliser une autre information : le stade s^e de l'individu le plus âgé de l'espèce e (si des individus ont émergé, on considère qu'ils sont au stade adulte). Les règles de décision sont déterminés de la même manière que pour le taux de développement accumulé. On considère ainsi la règle suivante, évaluée à la fin de la simulation pour chaque espèce :

$$\dot{R}_s^e = \begin{cases} \top & \text{si pour l'agent } Larva \ i \text{ le plus développé de } e \ s_i = s^e \\ \perp & \text{sinon,} \end{cases} \quad (7.3)$$

où s_i représente le stade de développement de l'individu i .

On considère de plus la règle suivante, évaluée à chaque pas de simulation :

$$R_s^e = \begin{cases} \perp & \text{s'il existe un agent } Larva \ i \text{ de } e \text{ tel que } s_i > s^e \text{ ou} \\ & \text{si pour tout agent } Larva \ j \text{ de } e, \\ & s(a_j + (p_n - p) \cdot var^e(q_j, T_{opt}^e) \cdot f^e(T_{opt})) < s^e \\ \top & \text{sinon,} \end{cases} \quad (7.4)$$

où $s(a)$ représente le stade de développement associé au taux de développement accumulé a . Cette règle est donc violée si aucun agent *Larva* de l'espèce e ne peut atteindre le stade s^e en se développant à température optimale.

7.2.3 Espèces retrouvées sur le corps

La présence ou l'absence de certaines espèces sur le corps peut être un élément intéressant pour valider une simulation bien que son utilisation soit controversée (Charabidze, 2008). Nous avons définies deux règles liées à la présence d'espèces sur le corps :

$$R_e^e = \begin{cases} \frac{1}{T} & \text{si } e \text{ n' pas été retrouvée sur le corps et } N_e > 0 \\ 1 & \text{sinon.} \end{cases} \quad (7.5)$$

$$\dot{R}_e^e = \begin{cases} \frac{1}{T} & \text{si } e \text{ a été retrouvée sur le corps et } N_e = 0 \\ 1 & \text{sinon,} \end{cases} \quad (7.6)$$

où N_e représente le nombre d'agents *Larva* de l'espèce e .

7.3 Résultats

Il n'a malheureusement pas été possible de tester le système d'aide à la décision sur des cas réels⁵⁰. Nous avons ainsi défini des cas de tests basés sur des données entomologiques et climatiques fictives mais vraisemblables. Ainsi, s'il n'a pas encore été possible de démontrer que ce système calculait des estimations d'IPM plus ou aussi précises que les estimations d'experts, nous avons établi qu'il donnait des résultats cohérents. Nous considérons ainsi deux exemples dans lesquels sont mises en œuvre les règles de décision portant sur le taux de développement accumulé et le stade de développement des insectes retrouvés sur le corps.

7.3.1 Premier exemple

Considérons un premier exemple, présenté initialement dans Morvan *et al.* (2008d). On suppose qu'une victime est retrouvée 140 heures après sa disparition. L'analyse de l'entomofaune retrouvée sur le corps permet d'identifier des larves de l'espèce *Lucilia sericata* dont le taux de développement accumulé est compris entre 0.3 et 0.4 ; les règles de décision portant sur le taux de développement accumulé sont donc utilisées.

Les températures relevées par la station météorologique la plus proche sont présentées dans la figure 7.6. Différents modèles de développement (ADD et Stinner) ont été construits à partir de données issues de Grassberger et Reiter (2001) (ces modèles sont appelés « ADD 1 » et « Stinner 1 ») et de ces mêmes données modifiées de manière arbitraire par nos soins afin de confronter les résultats obtenus à l'aide de différents modèles. L'ensemble des hypothèses est discrétisé heure par heure (on considère donc 141 hypothèses). Les résultats sont présentés, par type de modèle, dans les figures 7.7 et 7.8.

⁵⁰Le laboratoire d'entomologie médico-légale de l'Institut de recherche criminelle de la gendarmerie nationale (IRCGN) nous a transmis des données liées à d'anciennes expertises. Cependant, la totalité des informations nécessaires à leur utilisation n'a pas pu être rassemblée jusqu'ici.

Dans Morvan *et al.* (2008d), nous avons utilisé le modèle des croyances transférables (MCT) (Smets et Kennes, 1994; Smets, 1998) pour fusionner les ensembles solutions estimés déterminés par les différents modèles. Sans entrer dans les détails, ce modèle repose sur une interprétation subjective de la théorie des fonctions de croyance (Dempster, 1968; Shafer, 1976). Il permet de prendre en compte les imperfections des informations, aussi bien les incertitudes que les imprécisions. Dans le cadre du MCT, deux niveaux sont distingués :

- le niveau *crédal*, où sont représentées (partie statique) et manipulées (partie dynamique) les informations disponibles,
- le niveau *pignistique* où la décision est prise.

Le passage du niveau *crédal* au niveau *pignistique* permet de définir une distribution de probabilités sur Φ . La probabilité associée à chaque heure possible de la mort représente, dans notre cas, la probabilité que la victime soit décédée à ce moment. La figure 7.9 représente la distribution des probabilités pignistiques associée à Φ . Il est ainsi possible d'identifier l'hypothèse ϕ_{78} comme étant l'heure la plus probable de la mort et de considérer l'intervalle de confiance $\{\phi_{74}, \dots, \phi_{84}\}$.

7.3.2 Second exemple

On suppose dans ce second exemple que des pupes vides de l'espèce *Lucilia sericata* ont été retrouvées à proximité d'un corps ; les règles de décision portant sur le stade de développement sont donc utilisées. L'espace des hypothèses est ici plus grand que pour la premier exemple (571 heures, soit 23 jours et 19 heures, entre la disparition et la découverte de la victime). Les données de températures utilisées pour l'expertise sont présentées dans la figure 7.10. Les modèles de développement ADD et Stinner construits à partir de données issues de Grassberger et Reiter (2001) sont utilisés. Les résultats sont présentés dans la figure 7.11. Les données utilisées étant très imprécises, il est seulement possible de définir un IPM minimal : 292 heures pour le modèle de Stinner et 232 heures pour ADD. Les différences entre les résultats fournis par les deux modèles s'expliquent (comme pour le premier exemple) par la variation importante des températures (de 9.9°C à 27.3°C). En effet, le modèle ADD est moins performant à ces extremums.

7.4 Résumé du chapitre

Dans ce chapitre, nous avons présenté l'implémentation informatique de l'architecture du système d'aide à la décision décrite dans la seconde partie de cette thèse. Cette implémentation est basée sur des logiciels libres dont la plate-forme SMA MadKit. Dans le cadre de l'application à l'entomologie médico-légale, une expertise réalisée avec le système a été présentée. Les résultats montrent que le système permet d'identifier un ensemble cohérent d'hypothèses comme solutions possibles. Cependant, pour valider le système et l'utiliser en conditions réelles, il sera nécessaire de disposer de données d'expertise pour laquelle la vérité, *i.e.*, l'heure de la mort, est connue. La collaboration avec le laboratoire d'entomologie médico-légale de l'Institut de recherche criminelle de la gendarmerie nationale initiée en 2009 va ainsi dans ce sens.

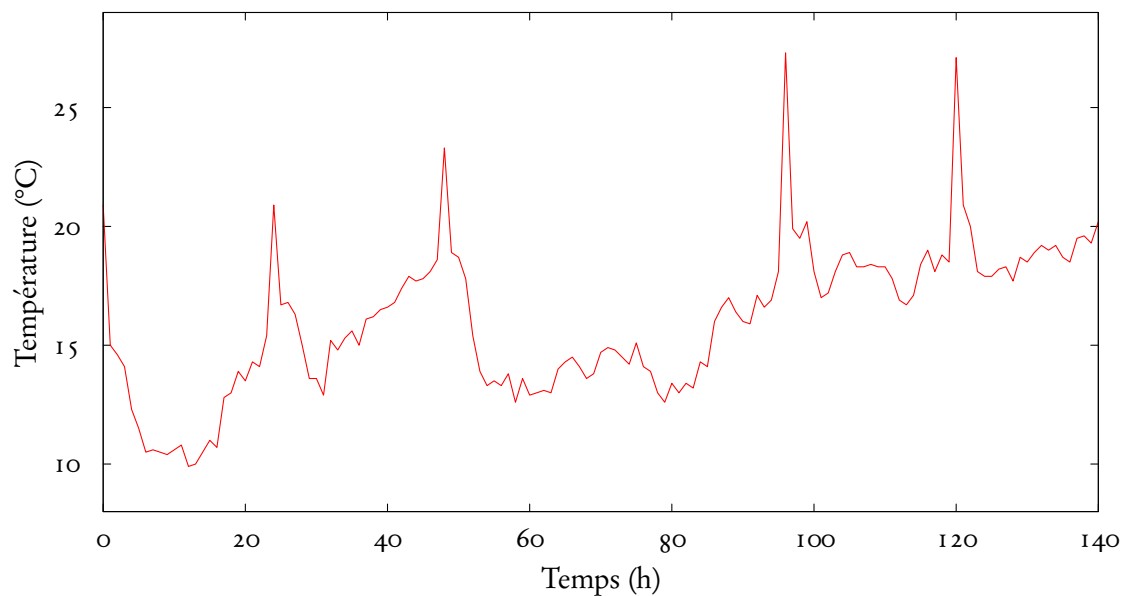


Fig. 7.6 – Températures utilisées pour le premier exemple

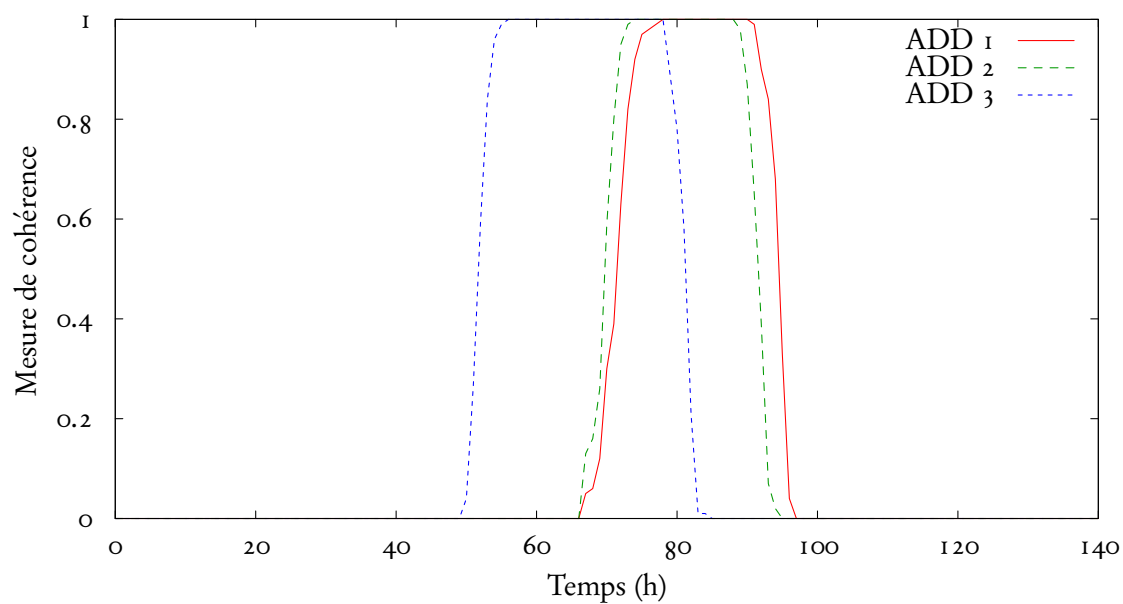


Fig. 7.7 – Résultats de l'expertise (premier exemple) pour les différents modèles ADD utilisés

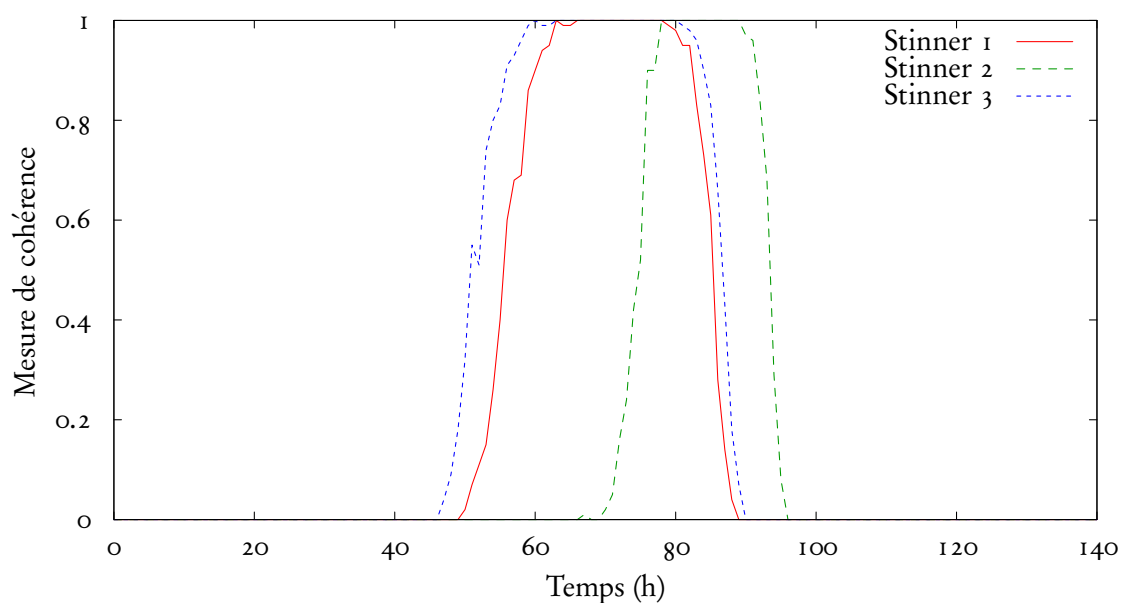


Fig. 7.8 – Résultats de l'expertise (premier exemple) pour les différents modèles de Stinner utilisés

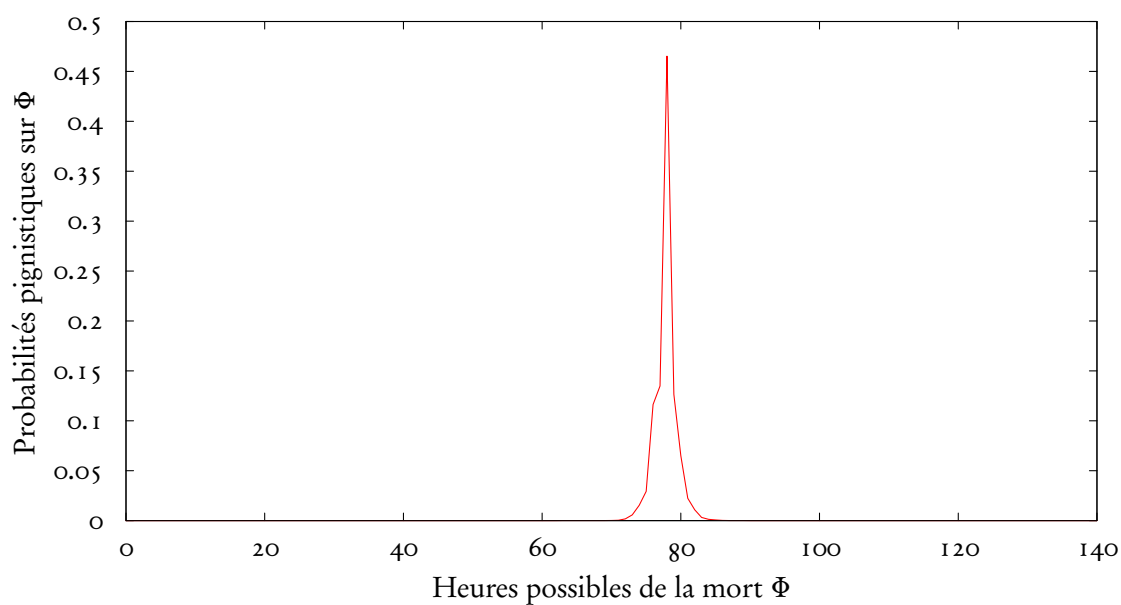


Fig. 7.9 – Distribution des probabilités pignistiques sur l'ensemble des heures possibles du décès pour le premier exemple

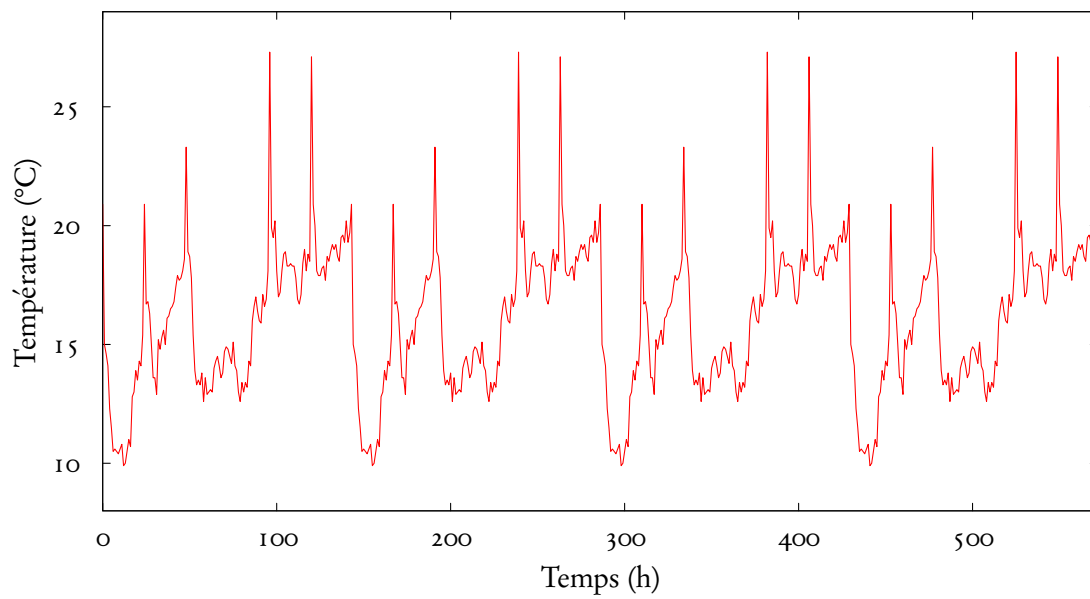


Fig. 7.10 – Températures utilisées pour le second exemple

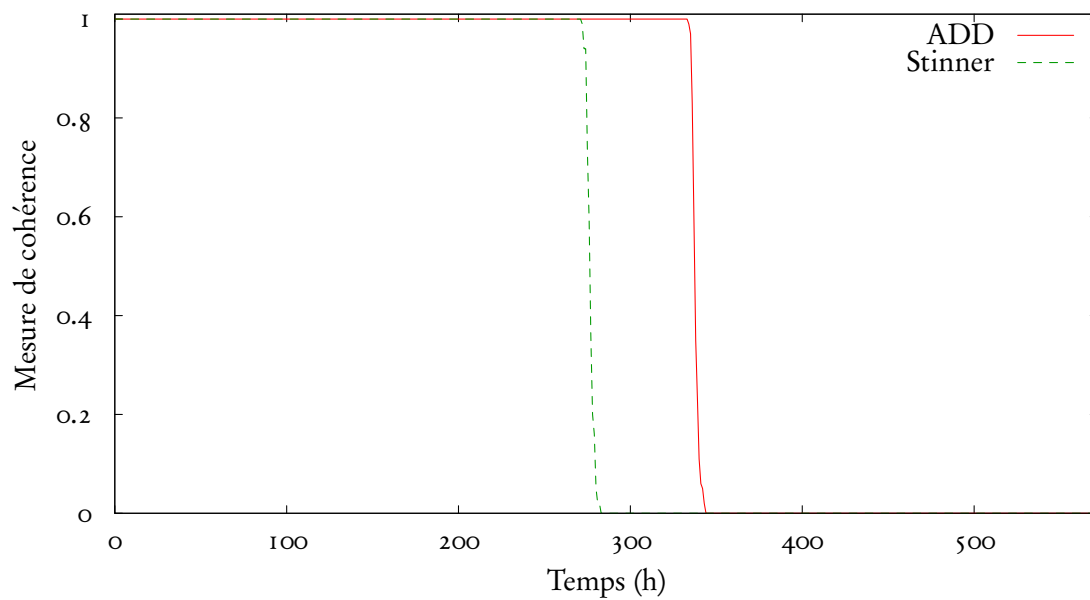


Fig. 7.11 – Résultats de l'expertise (second exemple) pour les deux modèles utilisés

Conclusion et perspectives

Synthèse des travaux

Nous avons développé dans cette thèse un certain nombre d'idées afin de construire un système d'aide à la décision basé sur des simulations orientées agent. L'entomologie médico-légale s'est révélé être un champ d'application intéressant pour mettre en œuvre ces idées puisqu'il implique la modélisation d'un système complexe (un cadavre colonisé par plusieurs centaines voire plusieurs milliers d'insectes en interaction) et la mise en œuvre d'une méthode de résolution de problèmes inverses.

Chronologiquement, nous nous sommes tout d'abord intéressés à la problématique de modélisation. Débuté dans le cadre d'un stage de Master recherche (Morvan, 2006), ce travail se poursuit encore aujourd'hui. En effet, les différents modèles présentés aux entomologistes ont soulevé dans leur communauté un certain nombre de questions encore ouvertes. Il a de plus été nécessaire de réaliser des expériences afin de valider certaines hypothèses (e.g., sur la structuration des masses de larves) et de disposer de données pour calibrer les modèles (vitesse de déplacement, augmentation de température dans la masse). Les échanges constants entre « thématiciens » et « modélisateurs » ont ainsi été particulièrement enrichissants et ont permis aux uns et aux autres de s'initier à des techniques bien éloignées de leurs préoccupations initiales (développement informatique pour les premiers, expérimentations sur des larves de diptères pour les seconds). Les résultats obtenus montrent que la prise en compte des comportements individuels pour modéliser le système étudié était une piste de recherche intéressante qui inspire aujourd'hui d'autres équipes (Richards, 2009).

Nous avons ensuite cherché à développer une méthode de raisonnement en tenant compte des contraintes propres à l'application :

- existence de nombreux modèles,
- recherche d'un ensemble de solutions cohérentes avec les informations fournies par les experts et non de la « meilleure solution »,
- simulations très gourmandes en temps de calcul.

Les recherches bibliographiques réalisées sur l'abduction (dans le domaine de l'intelligence artificielle) et la résolution de problèmes inverses nous ont convaincu que les méthodes existantes étaient inadaptées à notre problématique. C'est ainsi que nous nous sommes intéressés à l'approche pragmatique de l'abduction développée en épistémologie (Lipton, 2004). Nous avons sur cette base développé une méthode de résolution de problèmes inverses fondée sur des critères (coût, utilité, simplicité) permettant de guider le raisonnement abductif.

Enfin, partant du constat que la simulation de notre modèle consommait énormément de ressources (en particulier du temps CPU), nous avons cherché à optimiser ce processus :

- en construisant des règles de décision permettant d'arrêter les simulations avant leur terme,
- en développant des méthodes d'observation de simulations « intelligentes ».

Ce dernier travail, encore embryonnaire, nous paraît particulièrement important. En effet, les avancées à la fois théoriques (méta-modèles) et pratiques (plates-formes de simulation) dans le domaine de la simulation orientée agent permettent aujourd'hui la modélisation de systèmes toujours plus complexes, *i.e.*, formés de composants en interaction toujours plus nombreux. Cependant, l'exécution et l'analyse des systèmes artificiels résultants posent un certain nombre de problèmes d'ingénierie non résolus. Ce champ de recherche semble donc prometteur.

Perspectives

L'un de nos objectifs aujourd'hui est d'étendre l'application de nos travaux à d'autres domaines. Des collaborations ont ainsi été initiées avec différents laboratoires pour travailler sur :

- la problématique de l'agrégation chez les espèces pré-sociales⁵¹
- le contrôle distribué des systèmes électriques embarqués dans les avions de ligne⁵².

Ces deux champs d'application, très différents, posent tout deux des problématiques de modélisation et d'aide à la décision.

Le travail sur l'outil d'aide à la décision dédié à l'entomologie médico-légale est poursuivi aujourd'hui dans le cadre d'une thèse consacrée à la fusion d'informations. Nous espérons ainsi proposer à l'utilisateur un logiciel facilement utilisable, ne nécessitant pas de connaissance informatiques particulière. Les sources de ce logiciel sont librement accessibles⁵³. Actuellement en développement, une première version devrait être disponible fin 2009. Nous devrions très prochainement pouvoir réaliser des expertises à partir de données réelles fournies par le laboratoire d'entomologie médico-légale de l'Institut de Recherches Criminelles de la Gendarmerie Nationale (IRCGN) et confronter ainsi les estimations du système à celles des experts d'une part et à la vérité (*i.e.*, l'heure réelle de la mort) d'autre part. Cependant, un travail important de collecte de données devra être réalisé afin de proposer à l'utilisateur des modèles pour l'ensemble des espèces pouvant coloniser un corps⁵⁴.

L'entomologie médico-légale étant un domaine en plein essor, de très nombreux articles sont publiés chaque année et pourraient permettre développer à la fois le modèle et les règles

⁵¹Il s'agit d'une collaboration entre plusieurs équipes belges et françaises travaillant dans le domaine de l'entomologie (le laboratoire d'entomologie médico-légale de l'Institut de Médecine Légale de Lille, l'unité d'écologie sociale de l'Université Libre de Bruxelles, l'unité d'entomologie de la Faculté Universitaire des Sciences Agronomiques de Gembloux, l'Université de Louvain, le laboratoire environnement et santé de la Faculté Libre des Sciences et Technologies de Lille) et de l'informatique (Le LAGIS de l'École Centrale de Lille, le pôle Ingénierie et Sciences du vivant de l'école des Haute Études d'Ingénieur (HEI), et le LGI2A de l'Université d'Artois).

⁵²Cette collaboration a débuté cette année avec un laboratoire spécialisé dans le génie électrique : le L2EP (Université des Sciences et Technologies de Lille, Arts et Métiers ParisTech, École Centrale de Lille, Hautes Études d'Ingénieur) dans le cadre d'un stage de Master recherche.

⁵³<http://sourceforge.net/projects/forenseek/>

⁵⁴Nous avons jusqu'ici travaillé sur deux espèces particulièrement représentées en France : *Lucilia sericata* et *Protophormia terraenovae*.

de décision. Citons par exemple le travail de Megyesi *et al.* (2005) qui propose d'utiliser le modèle des degrés jours accumulés (section 5.2.2.1) pour déterminer l'âge d'un cadavre. Il est ainsi possible de lier les températures de l'environnement et le stade de décomposition du corps. L'intégration de cette méthode dans notre modèle pourrait permettre :

- de définir de manière dynamique les fenêtres de ponte des différentes espèces (celles-ci étant liées au stade de décomposition),
- d'intégrer des observations sur l'état du cadavre dans les règles de décision.

D'un point de vue plus fondamental, différents problèmes qui se sont posés durant cette thèse nous conduit à travailler sur une méthode de modélisation multi-niveaux, *i.e.*, permettant d'associer dans un même modèle les connaissances issues des différents niveaux d'observation du système (micro-connaissances et macro-connaissances). L'approche proposée dans Morvan *et al.* (2009b), fondée sur le paradigme agent, consiste à modéliser, du point de vue des agents, la façon dont une organisation émerge à partir de leurs interactions et, du point de vue du système, les contraintes imposées aux agents par l'organisation ainsi que les propriétés émergentes. La dynamique des organisations est ainsi spécifiée à différents niveaux d'observation. Il serait très intéressant de formaliser la notion de « structure organisationnelle dynamique » qui sous tend ce travail et jusque là définie de manière intuitive. La problématique de l'adaptation est aussi essentielle. En effet, dans cette approche, les agents représentant le niveau microscopique adaptent leurs comportements aux contraintes imposées par les agents représentant le niveau macroscopique et réciproquement. Cette méthode a été initialement développée pour répondre à un problème de modélisation dans le domaine de la biologie (Morvan *et al.*, 2008c). Cependant nous l'avons appliquée à la problématique du contrôle des systèmes complexes ; un exemple lié au routage de véhicules autoguidés est présenté dans Morvan *et al.* (2009b).

D'une manière générale, la problématique de l'aide à la décision pour les systèmes complexes est riche en perspectives et applications potentielles. Nous espérons que cette thématique sera fructueuse pour les équipes avec lesquelles nous avons travaillé et plus généralement pour la communauté scientifique.

Bibliographie

- Aliseda-Llera, A. (1998). *Seeking Explanations : Abduction in Logic, Philosophy of Science and Artificial Intelligence*. Thèse de doctorat, Stanford University, Department of Computer Science.
- Allen, J. (1976). A modified sine wave method for calculating degree days. *Environmental Entomology*, 5(3):388–396.
- Anderson, G. et VanLaerhoven, S. (1996). Initial studies on insect succession on carrion in southwestern british columbia. *Journal of Forensic Sciences*, 41(4):617–625.
- Aristote (2008). *La Métaphysique*. Flammarion.
- Armatte, M. (2009). Crise financière : modèles du risque et risque de modèle. *Mouvements*, 58(2):160–176.
- Baas, N. (1992). Emergence, hierarchies and hyperstructures. *Artificial Life*, 3:515–537.
- Biswas, R. et Ort, E. (2006). The java persistence api - a simpler programming model for entity persistence. Rapport technique, Sun Microsystems, Inc.
- Bonabeau, E., éditeur (1999). *Artificial Life, Special Issue on Stigmergy*, volume 5. MIT Press, Cambridge.
- Botía, J., Hernansaez, J. et Gómez-Skarmeta, A. (2006). *Programming Multi-Agent Systems*, volume 4411 de *Lecture Notes in Computer Science*, chapitre On the Application of Clustering Techniques to Support Debugging Large-Scale Multi-Agent Systems, pages 217–227. Springer Berlin Heidelberg.
- Byrd, J. et Allen, J. (2000). *Forensic Entomology : The Utility of Arthropodes in Legal Investigations*, chapitre Computer modeling of insect growth and its application to forensic entomology. CRC Press.
- Campobasso, C., Di Vella, G. et Introna, F. (2001). Factors affecting decomposition and diptera colonization. *Forensic Sci. Int.*, 120:18–27.
- Casella, G. et Berger, R. (2001). *Statistical Inference*. Duxbury Press, 2nd édition.
- Cellier, F. (1991). *Continuous System Modeling*. Springer Verlag.

- Charabidze, D. (2008). *Étude de la biologie des insectes nécrophages et application à l'entomologie médico-légale*. Thèse de doctorat, Université de Lille 2.
- Charabidze, D., Bourel, B. et Gosset, D. (2005). Behavioural analysis of the repellent effect of various substances on necrophagous insects using olfactometry. *In Proceedings of the 3rd Meeting of the European Association for Forensic Entomology, Lausanne, Switzerland*.
- Charabidze, D., Bourel, B., Hedouin, V. et Gosset, D. (2009). Repellent effect of some household products on fly attraction to cadavers. *Forensic Science International*, 189(1-3):28-33.
- Charabidze, D., Bourel, B., Leblanc, H., Hedouin, V. et Gosset, D. (2008a). Effect of body length and temperature on the crawling speed of *protophormia terraenovae* larvae (robineau-desvoidy) (diptera calliphoridae). *Journal of insect physiology*, 54:529-533.
- Charabidze, D., Bourel, B., Morvan, G., Hedouin, V. et Gosset, D. (2008b). Forenseek, un programme de simulation et d'expertise dédié à l'entomologie médico-légale. *Journal de médecine légale droit médical*, 51(3):131-140.
- Charabidze, D., Morvan, G., Bourel, B., Jolly, D., Hedouin, V. et Gosset, D. (2007). Modélisation of maggot activity on carrion. *In Proceedings of the 5th Meeting of the European Association for Forensic Entomology, Bruxelles, Belgium*.
- Charabidze, D., Morvan, G., Dupont, D., Gosset, D. et Bourel, B. (2008c). Forenseek : un programme de simulation du développement des insectes nécrophages dédié à l'entomologie médico-légale. *Annales de la société entomologique Française*, 44(3):385-392.
- Chopard, B. et Droz, M. (2004). *Cellular Automata Modeling of Physical Systems*. Cambridge University Press.
- Coquillard, P. et Hill, D. (1997). *Modélisation et simulation d'écosystèmes*. Masson, Paris.
- David, D. et Courdier, R. (2008). Emergence as metaknowledge : refining simulation models through emergence reification. *In Proceedings of ESM'2008, Le Havre, France*, pages 25-27.
- David, D. et Courdier, R. (2009). See emergence as a metaknowledge. a way to reify emergent phenomena in multiagent simulations? *In Proceedings of ICAART'09*, pages 564-569, Porto, Portugal.
- Dempster, A. (1968). A generalization of bayesian inference. *Journal of Royal Statistical Society*, pages 205-247.
- Dent, D. (1997). *Methods in Ecological and Agricultural Entomology*, chapitre Quantifying Insect Populations : Estimates and Parameters. CAB International.
- Deonier, C. (1940). Carcass temperature and their relation to winter blowfly populations and activity in the southwest. *Journal of Economic Entomology*, 33:166-170.

-
- Derveeuw, J. (2006). *Artificial Economics, Agent-Based Methods in Finance, Game Theory and Their Applications*, volume 564 de *Lecture Notes in Economics and Mathematical Systems*, chapitre Market Dynamics and Agents Behaviors : a Computational Approach, pages 15–26. Springer Berlin Heidelberg.
- Deutsch, A. et Dormann, S. (2004). *Cellular Automaton Modeling of Biological Pattern Formation : Characterization, Applications, and Analysis*. Modeling and Simulation in Science, Engineering and Technology. Birkhauser Boston.
- Dorigo, M., Maniezzo, V. et Colorni, A. (1996). Ant system : optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics*, 26(1):29–41.
- Dos Reis, S., Von Zuben, C. et Godoy, W. (1999). Larval aggregation and competition for food in experimental populations of *chrysomya putoria* (wied.) and *cochliomyia macellaria* (f.) (dipt., calliphoridae). *Journal of applied entomology*, 123(8):485–489.
- Drogoul, A., Vanbergue, D. et Meurisse, T. (2002). Multi-agent based simulation : Where are the agents? In *Proceedings of MABS*, Lecture Notes in Computer Science, pages 89–104. Springer Verlag.
- Drogoul, A., Vanbergue, D. et Meurisse, T. (2003). Simulation orientée agent : où sont les agents? In *Actes des Journées de Rochebrune, Rencontres interdisciplinaires sur les systèmes complexes naturels et artificiels*, Megève, France.
- Dubois, D. et Prade, H. (1995). Fuzzy relation equations and causal reasoning. *Fuzzy Sets and Systems*, 75(2):119–134.
- Dussaix, A.-M. et Grosbras, J.-M. (1996). *Les sondages : Principes et méthodes*. Que sais-je? Presses Universitaires de France.
- Epstein, J. (2008). Why model? *Journal of Artificial Societies and Social Simulation*, 11(4):12.
- Ferber, J. (1995). *Les Systèmes Multi-Agents : Vers une Intelligence Collective*. InterEditions.
- Ferber, J. et Gutknecht, O. (1998). A meta-model for the analysis and design of organizations in multi-agent systems. In *Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS98)*, pages 128–135.
- Ferber, J., Gutknecht, O. et Michel, F. (2004). From agents to organizations : an organizational view of multi-agent systems. In *Agent-Oriented Software Engineering (AOSE)*, volume 4, pages 214–230.
- Ferber, J., Michel, F. et Bàez-Barranco, J. (2005). Agre : Integrating environments with organizations. In *Post-proceedings of the First International Workshop on Environments for Multiagent Systems*.
- Fichwick, P. (1997). Computer simulation : growth through extension. *Transactions of the Society for Computer Simulation International*, 1(14):13–23.

- Fisher, P., Wall, R. et Ashworth, J. (1998). Attraction of the sheep blowfly *Lucilia sericata* (Diptera : Calliphoridae) to carrion bait in the field. *Bulletin of Entomological Research*, 86:611-616.
- Godoy, W., Von Zuben, C., Dos Reis, S. F. et Von Zuben, F. (1996). Theoretical estimates of consumable food and probability of acquiring food in larvae of *Chrysomya putoria* (Diptera : Calliphoridae). *Memorias do Instituto Oswaldo Cruz*, 91:257-264.
- Goodbrod, J. et Goff, M. (1990). Effects of larval population density on rates of development and interactions between two species of *Chrysomya* (Diptera : Calliphoridae) in laboratory culture. *Journal of Medical Entomology*, 27(3):338-343.
- Gosselin, M., Karapetian, J., Braet, Y., Bourguignon, L. et Hubrecht, F. (2006). Forensic entomology : beyond simple temperature measurements. In *Proceedings of the 4th Meeting of the European Association for Forensic Entomology, Bari, Italia*.
- Goulson, D., Derwent, L., Hanley, M., Dunn, D. et Abolins, S. (2005). Predicting calyptrate fly populations from the weather, and probable consequences of climate change. *Journal of Applied Ecology*, 42:795-804.
- Grassberger, M. et Reiter, C. (2001). Effect of temperature on *Lucilia sericata* (Diptera : Calliphoridae) development with special reference to the isomegalen and isomorphen diagram. *Forensic Science International*, 120:32-36.
- Greenberg, B. (1991). Flies as forensic indicators. *Journal of Medical Entomology*, 28:565-577.
- Gross, D. et McMullin, B. (2001). Is it the right ansatz? *Artificial Life*, 7:355-365.
- Gruner, S., Slone, D. et Capinera, J. (2007). Forensically important calliphoridae (Diptera) associated with pig carrion in rural north-central Florida. *Journal Medical Entomology*, 44(3):509-515.
- Gutknecht, O. et Ferber, J. (1997). Madkit : Organizing heterogeneity with groups in a platform for multiple multi-agent systems. Rapport technique 97188, LIRMM, 161, rue Ada - Montpellier - France.
- Gutknecht, O. et Ferber, J. (1999). Vers une méthodologie organisationnelle de conception de systèmes multi-agents. Rapport technique, LIRMM.
- Hafez, M. (1948). On the behaviour and sensory physiology of the house-fly larva, *Musca domestica* L. feeding stage. *Parasitology*, 40:215-236.
- Henssge, C. (1988). Death time estimation in case work - the rectal temperature time of death nomogram. *Forensic Science International*, 38:209-236.
- Henssge, C. (1992). Rectal temperature time of death nomogram : dependance of corrective factor on the body weight under stronger thermic insulation conditions. *Forensic Science International*, 54:51-66.

-
- Henssge, C. (2004). *Estimation of the time since death in the early post-mortem period*. Hodder Arnold Publication.
- Hewadikaram, K. et Goff, M. (1991). Effect of carcass size on rate of decomposition and arthropod succession patterns. *American J. Forensic Med. Pathology*, 12(3):235-240.
- Higley, L. et Haskell, W. (2000). *Forensic Entomology : The Utility of Arthropodes in Legal Investigations*, chapitre Insect development and forensic entomology, pages 288-295. CRC Press.
- Hintikka, J. (2001). Is logic the key to all good reasoning? *Argumentation*, 15:35-57.
- Hobson, R. (1931). Studies on the nutrition of blow-fly larvae. *Journal of Experimental Biology*, 8(2):128-138.
- Incropera, F., DeWitt, D., Bergman, T. et Lavine, A. (2006). *Fundamentals of Heat And Mass Transfer*. John Wiley & Sons Inc, 6th revised edition édition.
- Introna, F., Campobasso, C. et Goff, M. (2001). Entomotoxicology. *Forensic Science International*, 120:42-47.
- Ireland, S. et Turner, B. (2006). The effects of larval crowding and food type on the size and development of the blowfly, calliphora vomitoria. *Forensic Science International*, 159(2): 175-181.
- Joy, J., Herrell, M. et Rogers, P. (2002). Larval fly activity on sunlit versus shaded raccoon carrion in southwestern west virginia with special reference to the black blowfly (diptera : Calliphoridae). *J Med Entomol.*, 39(2):392-397.
- Joy, J., Liette, N. et Harrah, H. (2006). Carrion fly (diptera : Calliphoridae) larval colonization of sunlit and shaded pig carcasses in west virginia, usa. *Forensic Science International*, 164(2-3):183-192.
- Kamal, A. (1958). Comparative study of thirteen species of sarcosaprophagous calliphoridae and sarcophagidae (diptera) i. bionomics. *Annals of the Entomological Society of America*, 51(261-271).
- Kaminka, G., Pynadath, D. et Tambe, M. (2002). Monitoring teams by overhearing : A multi-agent planrecognition approach. *Journal of Artificial Intelligence*, 17:83-135.
- Kaneshrajah, G. et Turner, B. (2004). Calliphora vicina larvae grow at different rates on different body tissues. *Int J Legal Med.*, 118(4):242-244.
- Langlois, M. (1997). *Automates cellulaires : Application à la simulation urbaine*. Lavoisier.
- Lempert, R. (2002). A new decision sciences for complex systems. *PNAS*, 99(3):7309-7313.
- Leszczyna, R. (2004). Evaluation of agent platforms. Rapport technique, European Commission, Joint Research Centre, Institute for the Protection and security of the Citizen, Ispra, Italy.

- Lipton, P. (2004). *Inference to the Best Explanation*. International Library Of Philosophy, Routledge, London, 2nd édition.
- Liu, S. et O'Flaherty, R. (2006). Determinants of maggot aggregation behavior in carrion. <http://ysp.ucdavis.edu/Research06/lius/default.html>.
- Maddox, N., Wong, B., Kimsey, R., Bullard, R. et Armijo-Prewitt, T. (2006). Determinants of maggot behavior in carrion.
- Mall, G. et Eisenmenger, W. (2005a). Estimation of time since death by heat-flow finite-element model. part i : method, model, calibration and validation. *Legal Medecine*, 7:1-14.
- Mall, G. et Eisenmenger, W. (2005b). Estimation of time since death by heat-flow finite-element model. part ii : application to non-standard cooling conditions and preliminary results. *Legal Medicine*, 7:69-80.
- Marchenko, M. I. (2001). Medicolegal relevance of cadaver entomofauna for the determination of the time of death. *Forensic Science International*, 120:89-120.
- Marquardt, D. (1963). An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11:431-441.
- Maturana, H. R. et Varela, F. J. (1980). *Autopoiesis and Cognition The Realization of the Living*, volume 42 de *Boston Studies in the Philosophy of Science*. Springer Berlin Heidelberg.
- McGregor, S. et Fernando, C. (2005). Levels of description : A novel approach to dynamical hierarchies. *Artificial Life*, 11(4):459-472.
- McGrew, T. J. (2003). Confirmation, heuristics, and explanatory reasoning. *British Journal for the Philosophy of Science*, 54:553-567.
- Mégnin, P. (1894). *La faune des cadavres*. G. Masson.
- Megyesi, M., Nawrocki, S. et Haskell, N. (2005). Using accumulated degree- days to estimate the postmortem interval from decomposed human remains. *Journal of forensic sciences*, 50:618-626.
- Michel, F. (2002). Introduction to turtlekit. Rapport technique 02215, LIRMM.
- Michel, F., Beurier, G. et Ferber, J. (2005). The turtlekit simulation platform : Application to complex systems. In *Proceedings of the First International Conference on Signal-Image Technology and Internet Based Systems*, pages 122-127.
- Minsky, M. (1965). Matter, minds and models. In *International Federation of Information Processing Congress*, volume 1, pages 45-49.
- Moe, S., Stenseth, N. et Smith, R. (2002). Density dependence in blowfly populations : experimental evaluation of non-parametric time-series modelling. *Oikos*, 98(3):523-533.
- Morin, E. (1977). *La méthode Tome 1*. Éditions du Seuil.

-
- Morin, E. (1980). *La méthode Tome 2*. Éditions du Seuil.
- Morvan, G. (2006). Modélisation du développement des diptères nécrophages par système multiagents. Mémoire de D.E.A., Université d'Artois.
- Morvan, G., Dupont, D. et Kubiak, P. (2008a). A simulation-based model of abduction. *In Proceedings of ESM'2008, Le Havre, France*, pages 183–187.
- Morvan, G., Jolly, D. et Charabidze, D. (2008b). Thermoregulation in *p. terraenovae* aggregations, an agent-based approach. *In Proceedings of the ESM'2008, Le Havre, France*, pages 417–422.
- Morvan, G., Jolly, D., Dupont, D. et Kubiak, P. (2007). A decision support system for forensic entomology. *In Proceedings of the 6th EUROSIM congress, Ljubljana, Slovenia*.
- Morvan, G., Jolly, D., Veremme, A., Dupont, D. et Charabidze, D. (2008c). Vers une méthode de modélisation multi-niveaux. *In Actes de la 7^{ème} Conférence de Modélisation et Simulation MOSIM, Paris, France*, volume 1, pages 167–174.
- Morvan, G., Veremme, A., Charabidze, D., Bourel, B. et Gosset, D. (2009a). Modélisation de l'évolution post mortem de la température corporelle lorsque la température de l'environnement varie. Communication à la société française de médecine légale.
- Morvan, G., Veremme, A., Dupont, D. et Jolly, D. (2009b). Modélisation et conception multiniveau de systèmes complexes : stratégie d'agentification des organisations. *Journal Européen des Systèmes Automatisés*, 43:381–406.
- Morvan, G., Veremme, A., Dupont, D. et Jolly, D. (2009c). Stratégies d'observation de simulations orientées agent. *In Actes des 17^{èmes} Journées Francophones sur les Systèmes Multi-Agents (JFSMA'09)*, pages 233–236.
- Morvan, G., Veremme, A., Lefèvre, E. et Mercier, D. (2008d). Application du modèle des croyances transférables dans le cadre d'expertises en entomologie médico-légale. *In 5^{ème} Atelier "Fouille de données complexes dans un processus d'extraction des connaissances", Sofia Antipolis, France*.
- Müller, J.-P., Ratzé, C., Gillet, F. et Stoffel, K. (2005). Modeling and simulating hierarchies using an agent-based approach. *In Proceedings of the MODSIM 2005 International Congress on Modelling and Simulation*, pages 1631–1638.
- Paavola, S. (2003). Abduction as a logic and methodology of discovery : the importance of strategies. *Foundations of Science*, 9(3):267–283.
- Pardalos, C. F. P., éditeur (2009). *Encyclopedia of Optimization*, volume 1–7. Springer Verlag.
- Paul, G. (1993). Approaches to abductive reasoning. *Artificial Intelligence Review*, 7(2):109–152.

- Payet, D., Medoc, J., Ralambondrainy, T., F. et Courdier, R. (2005). Outils d'observation et d'analyse de simulations multi-agents : l'expérience de la plate-forme geamas/biomas. In *CABM-HEMA. Conference on Multi-agent modelling for environmental management, Bourg Saint Maurice - Les Arcs, France*.
- Payne, J. (1965). A summer carrion study of the baby pig *sus scrofa linnaeus*. *Ecology*, 46:592-602.
- Peirce, C. S. (1931). *Collected papers of Charles Sanders Peirce*, volume 1-6. Cambridge, Harvard University Press.
- Peirce, C. S. (1958). *Collected papers of Charles Sanders Peirce*, volume 7-8. Cambridge, Harvard University Press.
- Prinkkila, M. et Hanski, I. (1995). Complex competitive interactions in four species of *lucilia* blowflies. *Ecological entomology*, 20(3):261-272.
- Railsback, S., Lytinen, S. et Grimm, V. (2005). Stupidmodel and extensions : A template and teaching tool for agent-based modeling platforms. <http://condor.depaul.edu/slytinen/abm/StupidModelFormulation.pdf>.
- Railsback, S., Lytinen, S. et Jackson, S. (2006). Agent-based simulation platforms : Review and development recommendations. *Simulation*, 82(9):609-623.
- Ralambondrainy, T., Médoc, J.-M., Courdier, R. et Guerrin, F. (2007). Tools to visualise the structure of multiagents' conversations at various levels of analysis. In *MODSIM 2007 International Congress on Modelling and Simulation. Modelling and Simulation Society of Australia and New Zealand*, pages 3098-3104.
- Ratzé, C., Gillet, F., Müller, J.-P. et Stoffelb, K. (2007). Simulation modelling of ecological hierarchies in constructive dynamical systems. *Ecological Complexity*, 4(1-2):13-25.
- Reggia, J., Nau, D. et Wang, P. (1983). Diagnostic expert systems based on a set covering model. *International journal of man-machine studies*, 19(5):437-460.
- Régnière, J. (1984). A method of describing and using variability in development rates for the simulation of insect phenology. *The Canadian entomologist*, 116(10):1367-1376.
- Richards, C. (2007). Latitude-related variation in the development rates of the green blow-fly *chrysomya albiceps* (diptera : Calliphoridae). In *Proceedings of the 5th Meeting of the European Association for Forensic Entomology, Bruxelles, Belgium*.
- Richards, C. (2009). Estimating the temperatures experienced by blow flies (diptera :calliphoridae) during larval development : Part 1. In *Proceedings of the 7th Meeting of the European Association for Forensic Entomology, Uppsala, Sweden*.
- Richards, E. et Goff, L. (1997). Arthropod succession on exposed carrion in three contrasting tropical habitats on hawaii island, hawaii. *J. Med. Entomol.*, 34(3):328-339.

-
- Ricordel, P. et Demazeau, Y. (2000). *Engineering Societies in the Agents World*, chapitre From Analysis to Deployment : A Multi-agent Platform Survey, pages 93–105. Lecture Notes in Computer Science. Springer Berlin Heidelberg.
- Russel, S. et Norvig, P. (1995). *Artificial Intelligence : A Modern Approach*. Prentice-Hall, Englewood Cliffs, NJ.
- Saunders, D. et Bee, A. (1995). Effect of larval crowding on size and fecundity of the blow fly, calliphora vicina (diptera : Calliphoridae). *Eur. J. Entomol.*, 92:615–622.
- Schelling, T. C. (1971). Dynamic models of segregation. *Journal of Mathematical Sociology*, 1:143–186.
- Servat, D., Perrier, E., Treuil, J.-P. et Drogoul, A. (1998). *When Agents Emerge from Agents : Introducing Multi-scale Viewpoints in Multi-agent Simulations*, volume 1534 de *Lecture Notes in Computer Science*, pages 183–198. Springer-Verlag Berlin Heidelberg.
- Severini, M., Baumgärtner, J. et Limonta, L. (2003). Parameter estimation for distributed delay based population models from laboratory data : egg hatching of oulema duftschmiedi redthenbacher (coleoptera, chrysomelidae) as an example. *Ecological Modelling*, 167(3):233–246.
- Shafer, G. (1976). *A Mathematical Theory of Evidence*. Princeton University Press.
- Shanahan, M. (1989). Prediction is deduction but explanation is abduction. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence, Detroit, Michigan, USA*, pages 1055–1060.
- Shannon, R. (1998). Introduction to the art and science of simulation. In *Proceedings of the 30th conference on Winter simulation*, pages 7–14.
- Slone, D. et Gruner, S. (2007). Thermoregulation in larval aggregations of carrion-feeding blow flies (diptera : Calliphoridae). *Journal of Medical Entomology*, 44(3):516–523.
- Slone, D., Gruner, S. et Allen, J. (2005). Assesing error in pmi prediction using a forensic entomological computer model. Rapport technique, U.S. Department of Justice.
- Smets, P. (1998). *The Transferable Belief Model for Quantified Belief Representation*, pages 267–301. Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Smets, P. et Kennes, R. (1994). The transferable belief model. *Artificial Intelligence*, 66(2):191–234.
- Smith, K. (1986). *A manual of forensic entomology*. Trustees of the British Museum (Natural history), London.
- Stinner, R. E., Gutierrez, A. P. et Butler Jr, G. D. (1974). An algorithm for temperature-dependant growth rate simulation. *The Canadian Entomologist*, 106:519–524.

- Tarantola, A. (2008). *Inverse Problem Theory And Methods For Model Parameter Estimation*. Society for Industrial & Applied Mathematics.
- Treuil, J.-P., Drogoul, A. et Zucker, J.-D. (2008). *Modélisation et simulation à base d'agents*. Dunod.
- Turner, B. et Howard, T. (1992). Metabolic heat generation in dipteran larval aggregations : a consideration for forensic entomology. *Medical and Veterinary Entomology*, 6(2):179-181.
- Ueda, K., Kitob, T. et Fujii, N. (2006). Modeling biological manufacturing systems with bounded-rational agents. *CIRP Annals - Manufacturing Technology*, 55(1):469-472.
- Ueda, K., Markus, A., Monostorib, L., Kalsc, H. et Arai, T. (2001). Emergent synthesis methodologies for manufacturing. *CIRP Annals - Manufacturing Technology*, 50(2):535-551.
- Varela, F. J. (1989). *Autonomie et Connaissance*. Éditions du Seuil.
- Veremme, A., Dupont, D., Morvan, G., Jolly, D. et Charabidze, D. (2008). Modélisation de la température d'un corps par automates cellulaires. *In Actes de la 7ème Conférence de Modélisation et Simulation MOSIM, Paris, France*, volume 1, pages 500-507.
- Verica, P., Janeska, B., Gutevska, A. et Duma, A. (2007). Post mortem cooling of the body and estimation of time since death. *Soudní lékařství*, 52(4):50-56.
- Wagner, T., Wu, H.-I., Sharpe, P., Schoolfield, R. et Coulson, R. (1984). Modeling insect development rates : A literature review and application of a biophysical model. *Annals of the Entomological Society of America*, 77(2):208-225.
- Wall, R. (1993). The reproductive output of the blowfly, *lucilia sericata*. *Journal of insect physiology*, 9:743-750.
- Wall, R., Pitts, K. et Smith, K. (2001). Pre-adult mortality in the blowfly *lucilia sericata*. *Medical & Veterinary Entomology*, 15(3):328-334.
- Weibull, W. (1951). A statistical distribution function of wide applicability. *Journal of Applied Mechanics*, 18(3):293-297.
- Werner, J. et Buse, M. (1988). Temperature profiles with respect to inhomogeneity and geometry of the human body. *Journal of Applied Physiology*, 65:1110-1118.
- Wilkins, D., Lee, T. et Berry, P. (2003). Interactive execution monitoring of agent teams. *Journal of Artificial Intelligence Research*, 18:217-261.
- Wooldridge, M. (2002). *An Introduction to MultiAgent Systems*. John Wiley & Sons.
- Zeigler, B., Kim, T. et Praehofer, H. (2000). *Theory of Modeling and Simulation*. Academic Press, 2nd édition.