



Des agents-modèles pour la modélisation et la simulation de systèmes complexes : application à l'écosystémique des pêches

Stéphane Bonneaud

► To cite this version:

Stéphane Bonneaud. Des agents-modèles pour la modélisation et la simulation de systèmes complexes : application à l'écosystémique des pêches. Autre [cs.OH]. Université de Bretagne occidentale - Brest, 2008. Français. NNT : . tel-00376168

HAL Id: tel-00376168

<https://theses.hal.science/tel-00376168>

Submitted on 16 Apr 2009

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

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

UNIVERSITÉ DE BRETAGNE OCCIDENTALE

— Thèse de doctorat —

Des agents-modèles pour la modélisation et la
simulation de systèmes complexes

Application à l'écosystémique des pêches

STÉPHANE BONNEAUD

Pour l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE BRETAGNE OCCIDENTALE

Mention INFORMATIQUE

LISYC – *Laboratoire d'Informatique des Systèmes Complexes – EA3883*

Soutenue publiquement le 14 octobre 2008, devant la commission d'examen :

<i>Président :</i>	Jacques TISSEAU	Professeur, LISYC, ENIB, Brest
<i>Rapporteuse :</i>	Zahia GUESSOUM	HDR, Maître de conférences, Laboratoire d'Informatique de Paris VI
<i>Rapporteur :</i>	René MANDIAU	Professeur, LAMIH, UMR CNRS 8530, Université de Valenciennes et du Hainaut-Cambrésis
<i>Examineur :</i>	Christophe CAMBIER	Maître de conférences, Laboratoire d'Informatique de Paris VI, Chargé de recherche à l'IRD
<i>Examineur :</i>	Vincent RODIN	Professeur, LISYC, UBO, Brest
<i>Directeur :</i>	Pierre CHEVAILLIER	HDR, Maître de conférences, LISYC, CERV, Brest
<i>Invité :</i>	Pascal REDOU	Maître de conférences, LISYC, CERV, Brest

Des agents-modèles pour la modélisation et la simulation de systèmes complexes

Application à l'écosystémique des pêches

Thèse de doctorat

STÉPHANE BONNEAUD

LISYC
Laboratoire d'Informatique des SYstèmes
Complexes
EA 3883 UBO-ENIB

STÉPHANE BONNEAUD
stephane.bonneaud@free.fr
[http ://stephanebonneaud.free.fr](http://stephanebonneaud.free.fr)



UBO
Université de Bretagne Occidentale

[http ://www.univ-brest.fr](http://www.univ-brest.fr)



CERV / ENIB
Centre Européen de Réalité Virtuelle
Ecole Nationale d'Ingénieurs de Brest

Technopôle Brest-Iroise
29280 Plouzané
[http ://www.cerv.fr](http://www.cerv.fr)
[http ://www.enib.fr](http://www.enib.fr)



Région Bretagne

[http ://www.region-bretagne.fr](http://www.region-bretagne.fr)
Ce travail a bénéficié du financement
de la région Bretagne.



Remerciements

Je remercie tout d'abord l'équipe du CERV pour tout ce qui, dans ce travail, appartient au groupe. En particulier, merci aux relecteurs, Pierre De Loor, Alexandra Fronville et Ronan Querrec, merci à mon co-directeur de thèse, Pascal Redou, et merci à Morgan.

Je remercie ensuite Christophe Cambier, j'ai été très heureux de le retrouver dans mon jury. Je remercie chaleureusement mes deux rapporteurs, Zahia Guessoum et René Mandiau. Merci à Jacques Tisseau, dont l'inspiration, les mots, gagnant en autonomie, sont venus habiter les différents chapitres de ce manuscrit.

Enfin, je remercie tout particulièrement et très chaleureusement mon directeur de thèse, Pierre Chevaillier. Son exigence, sa sagacité et sa confiance m'ont permis de prendre ma propre autonomie dans le plaisir de la recherche.

Table des matières

Table des figures	xi
Liste des tableaux	xv
1 Introduction	3
1.1 Contexte de la thèse	3
1.2 Problématiques et enjeux	4
2 Etat de l'art	9
2.1 Introduction	9
2.2 Les systèmes complexes	10
2.2.1 Qu'est-ce qu'un système complexe?	10
2.2.2 Quels types de systèmes?	12
2.2.3 Cadre théorique	13
2.2.3 - A Théorie générale des systèmes	13
2.2.3 - B Systèmes dynamiques et non-linéarité	14
2.2.4 Echelles d'analyse et d'observation	15
2.3 Le processus de modélisation	16
2.3.1 Qu'est-ce qu'un modèle?	16
2.3.2 Du modèle thématique au modèle computationnel	18
2.3.3 Cadre théorique et approches de modélisation	19
2.3.3 - A Modélisation agrégée	19
2.3.3 - B Modélisation centrée individu	21
2.3.4 Vers un modèle opérationnel	23
2.3.4 - A Discrétisation temporelle et spatiale	23
2.3.4 - B Implémentation de modèles agrégés	24

2.3.4 - C	Implémentation de modèles centrés individu	25
2.3.5	Multi-modélisation et couplage	27
2.4	Le processus de simulation	29
2.4.1	Qu'est-ce que la simulation ?	29
2.4.2	Expérimentation <i>in virtuo</i> : vers un laboratoire virtuel ?	30
2.4.3	La simulation : un processus complexe ?	32
2.4.3 - A	L'exploration des modèles	32
2.4.3 - B	Le modèle dans le processus de simulation	33
2.4.4	Rationaliser la simulation : l'approche POM	34
2.5	Approche agent et plates-formes de simulation	36
2.5.1	L'agent et l'individu	37
2.5.2	L'agent et son environnement	38
2.5.2 - A	Notion d' <i>artefact</i>	38
2.5.2 - B	Problématique de la granularité	39
2.5.3	Modélisation thématique : comment construire un modèle ?	40
2.5.4	Multi-modélisation : le couplage vu par les agents	42
2.5.5	Modélisation et simulation à l'aide d'agents	43
3	Positionnement	45
3.1	Fondements de la proposition	45
3.2	Proposition	46
4	Une société de modèles pour la modélisation de systèmes complexes	51
4.1	Introduction	51
4.2	Solution conceptuelle basée agent-modèle	52
4.2.1	E-nvironnement de modélisation : artefacts et patterns	52
4.2.1 - A	Notion d' <i>artefact</i>	52
4.2.1 - B	L' <i>artefact</i> comme support d'identification des patterns	52
4.2.1 - C	L' <i>artefact</i> comme support d'interaction des agents	53
4.2.1 - D	Une formalisation de l' <i>artefact</i>	54
4.2.1 - E	L' <i>artefact</i> comme support de l'analyse	56
4.2.2	A-gents : notion d' <i>agent-modèle</i>	56
4.2.2 - A	But et rationalité des <i>agents-modèles</i>	56
4.2.2 - B	Notion d' <i>opération</i> : vers une adaptation des agents	57
4.2.2 - C	Comportement d'un <i>agent-modèle</i> et types de modèles	59
4.2.3	I-nteraction : notion de <i>jonction</i>	62
4.2.3 - A	Synchronisation et autonomie comportementale	62
4.2.3 - B	Activités des agents et rythmes d'évolution hétérogènes	65
4.2.4	O-rganisation d'une société de modèles	67
4.2.5	U-tilisateur : modélisateur ou <i>agent-modèle</i> ?	69
4.3	Architecture du simulateur et modèle émergent	70
4.3.1	Architecture	70
4.3.2	Modèle émergent	71

4.3.2 - A	Un environnement phénoménologique	72
4.3.2 - B	Agents-individus	72
4.3.2 - C	Organisations des individus : peuplement et groupe	74
4.3.2 - D	Habitats	75
4.4	Méthodologie	75
4.4.1	Méthode basée POM	75
4.4.2	Question et cadre de modélisation	76
4.4.3	Phase de modélisation	78
4.4.4	Phase de simulation	79
4.5	Conclusion	80
5	Biais computationnels dans les modèles complexes de peuplements	83
5.1	Introduction	83
5.2	Problématique et méthode	84
5.3	Patterns	85
5.4	Modèle conceptuel	86
5.5	Comportements de l'agent dans un IBM	87
5.5.1	Problématique	87
5.5.2	Comportement déterministe	88
5.5.2 - A	Modèle computationnel	88
5.5.2 - B	Cas trivial	88
5.5.2 - C	Synchronicité de comportements indépendants	90
5.5.2 - D	Synchronisation de comportements dépendants	92
5.5.2 - E	Sensibilité des comportements aux états initiaux	93
5.5.3	Comportement stochastique	95
5.5.3 - A	Modèle computationnel	95
5.5.3 - B	Application	96
5.5.4	Analyse	99
5.6	Sensibilités d'un IBM aux discrétisations homogènes	100
5.6.1	Problématique	100
5.6.2	Sensibilité au nombre d'agents	101
5.6.2 - A	Modèle computationnel – comportements des agents	101
5.6.2 - B	Cas d'une population	102
5.6.3	Sensibilité à la discrétisation spatiale	106
5.6.3 - A	Modèle computationnel – comportements des agents	106
5.6.3 - B	Interaction avec l'environnement	110
5.6.3 - C	Interaction entre les agents	113
5.6.4	Sensibilité à la discrétisation temporelle	114
5.6.4 - A	Modèle computationnel – comportements des agents	114
5.6.4 - B	Cas d'une population	117
5.6.5	Analyse	120
5.7	Sensibilités d'un IBM aux discrétisations hétérogènes	121

5.7.1	Problématique	121
5.7.2	Sensibilité à la discrétisation en nombre d'agents	122
5.7.2 - A	Modèle computationnel – facteur d'échelle et comportements des agents	122
5.7.2 - B	Hétérogénéité du nombre d'agents	124
5.7.3	Sensibilité à la discrétisation spatiale	125
5.7.4	Sensibilité à la discrétisation temporelle	131
5.7.5	Analyse	136
5.8	Conclusion	139
6	Etude de modèles écosystémiques des pêches	143
6.1	Introduction	143
6.2	Contexte et problématiques	145
6.3	Objet d'étude et patterns	146
6.3.1	Problématique écologique	146
6.3.2	Problématique économique	148
6.4	Action 1 : Modélisation de peuplements marins	149
6.4.1	Question de modélisation et patterns	149
6.4.2	Cadre de modélisation	150
6.4.3	Expérimentation unitaire : choix de l'approche de modélisation	152
6.4.4	Ajout d'un agent sans impact sur le reste du système	154
6.4.5	Analyse	155
6.5	Action 2 : Modélisation EDP de l'écosystème du Golfe de Gascogne	156
6.5.1	Question de modélisation et patterns	156
6.5.2	Cadre de modélisation	157
6.5.3	Dimensionnement du modèle EDP pour deux espèces	158
6.5.4	Analyse	161
6.6	Action 3 : Modélisation IBM de l'écosystème du Golfe de Gascogne	162
6.6.1	Question de modélisation et patterns	162
6.6.2	Cadre de modélisation	163
6.6.3	Dimensionnement du modèle IBM sans hétérogénéités de discrétisation	163
6.6.4	Analyse	165
6.7	Action 4 : Couplage avec une pêcherie et hétérogénéités	166
6.7.1	Question de modélisation et patterns	166
6.7.2	Cadre de modélisation	166
6.7.3	Comparaison des modèles EDP et IBM du peuplement	168
6.7.4	Analyse	169
6.8	Action 5 : Modélisation économique du marché des navires de pêche	170
6.8.1	Question de modélisation et patterns	170
6.8.2	Cadre de modélisation	171
6.8.3	Processus de dépréciation de la valeur d'un navire de pêche	172
6.8.4	Scénarios d'incitation à la sortie de l'économie	173

6.9	Complexité, outillage et rationalisation	175
6.10	Conclusion	178
6.11	Perspectives à court terme	180
6.11.1	Influences sur le peuplement du réchauffement climatique et des activités de pêche	180
6.11.2	Couplage avec le <i>Virtual Laboratory Environment</i> (VLE) : système biodiversité-pêcheries	181
7	Conclusion générale	183
7.1	Bilan	183
7.2	Perspectives	185
	Références bibliographiques	189
A	Synchronisation de comportements dépendants – Preuve mathématique	205
A.1	Notations et hypothèses	205
A.2	Équations caractérisant $f^T(t)$	206
A.3	Évolution de la population globale	206
B	Projet Chaloupe	209
B.1	Problématique écologique : hypothèses et modèles	209
B.1.1	Environnement	209
B.1.2	Vieillessement	210
B.1.3	Croissance	210
B.1.4	Recrutement	211
B.1.5	Mortalité	212
B.1.6	Prédation	213
B.1.7	Occupation spatiale	214
B.2	Problématique économique : modèles	215
B.2.1	Cas d'un agent "acteur économique"	215
B.2.2	Bateau : unité de pêche	216
B.2.3	La pêche : agent "pêcheur"	216
B.2.4	Patron pêcheur : agent "acteur économique"	217
B.2.5	Environnement	221
B.3	Paramètres des modèles	222
C	Modélisation écologique à base d'équations aux dérivées partielles	225

Table des figures

1.1	Le développement durable des trois systèmes écologique, économique et social.	4
2.1	Exemple de pattern : le pattern classique des systèmes proies-prédateurs de type Lotka-Volterra.	34
2.2	Exemple de pattern identifié dans les données caractérisant l'économie de la pêche du GdG.	35
4.1	Illustration des types de données qui sont le plus souvent manipulés et visualisation de leur évolution au cours de la simulation.	53
4.2	Exemples de distributions de peuplements dans un environnement (2D). . . .	53
4.3	Exemple d'un <i>artefact</i> décrivant la structure spatiale d'un peuplement à un instant donné.	55
4.4	Exemple de notation d'un <i>agent-modèle</i>	57
4.5	Les trois types d' <i>opérations</i> qu'un agent peut manipuler.	59
4.6	Exemple de système d' <i>agents-modèles</i> : problème de lecture d'un <i>artefact</i> par un <i>agent-modèle</i> «pêcheries» et utilisation d'une <i>opération</i>	60
4.7	Illustration de la boucle d'exécution Read-eXecute-W rite.	63
4.8	Détails d'une boucle de dépendance entre deux <i>agents-modèles</i> – initialisation et premier pas de temps (exhibition de la boucle d'exécution R-X-W). . . .	64
4.9	Illustration du fonctionnement d'une <i>jonction</i>	66
4.10	Notation agent – exemple de système d'agents	69
4.11	Proposition méthodologique pour une modélisation et une simulation basées POM et <i>agent-modèle</i>	77
5.1	Graphe 5.1a : habitat des populations (affinité maximale : $a = 1$) ; Graphes 5.1b et 5.1c : évolution théorique d'une population suivant l'équation 5.1. . .	87
5.2	Evolution d'une population d'agents composés des comportements 2 et 3 : pour chaque agent, les comportements sont soit synchronisés soit désynchronisés. .	91

5.3	Evolution d'une population d'agents composés des comportements <i>DeterministicMate</i> et <i>DeterministicDie</i> partageant une même variable <i>age</i> (expérience 5.5.2 - D).	93
5.4	Sensibilité des comportements aux états initiaux – Etude des initialisations homogène ou hétérogène des variables <i>age</i> des comportements d'un agent. . .	95
5.5	Modèle stochastique : évolution de la moyenne dans le temps de 30 simulations et ajustement par un modèle linéaire.	98
5.6	Modèle stochastique : (a) voir tableau 5.6; (b) graphe des 30 résultats de simulation (1000 agents); (c) exemple d'une distribution en âge de la population (pas de temps quelconque, 1000 agents).	98
5.7	Discrétisation homogène en nombre d'agents avec K_{max} la densité maximale par case.	105
5.8	Discrétisation homogène en nombres d'agents – Evolution du coefficient de variation en fonction de K_{max}	106
5.9	Discrétisation homogène en nombre d'agents – Distributions spatiales du peuplement suivant K_{max} (t=2000).	107
5.10	Discrétisation spatiale homogène – Distributions spatiales du peuplement en fonction de la discrétisation spatiale.	112
5.11	Influence de la discrétisation spatiale, interaction entre agents – Evolution du nombre d'agents en fonction du temps (proies et prédateurs).	115
5.12	Influence de la discrétisation spatiale, interaction entre agents – Evolution du nombre d'agents en fonction du temps (proies et prédateurs).	116
5.13	Discrétisation homogène temporelle : évolution du nombre d'agents en fonction du temps.	119
5.14	Discrétisation homogène temporelle : coefficient de variation en fonction du pas de temps.	119
5.15	Hétérogénéité du nombre d'agents : évolution du nombre d'agents en fonction du temps (a_p : facteur d'échelle des proies, a_P : facteur d'échelles des prédateurs).	126
5.16	Hétérogénéité du nombre d'agents : évolution du nombre d'agents en fonction du temps (a_p : facteur d'échelle des proies, a_P : facteur d'échelles des prédateurs).	127
5.17	Hétérogénéité du nombre d'agents : évolution du nombre d'agents en fonction du temps (a_p : facteur d'échelle des proies, a_P : facteur d'échelles des prédateurs).	128
5.18	Diagramme d'agents-modèles – cas de deux agents-modèles en interaction trophique.	129
5.19	Hétérogénéité spatiale : distributions spatiales des proies et des prédateurs aux pas de temps 1, 1800 et 3600 en fonction de la discrétisation.	132
5.20	Hétérogénéité spatiale : distributions spatiales des proies et des prédateurs aux pas de temps 1, 1800 et 3600 en fonction de la discrétisation.	133
5.21	Hétérogénéité spatiale : distributions spatiales des proies et des prédateurs aux pas de temps 1, 1800 et 3600 en fonction de la discrétisation.	134
5.22	Diagramme d'agents-modèles – cas de deux agents-modèles en interaction trophique.	135
5.23	Discrétisation temporelle hétérogène – cas de deux populations – scénario 1. .	137

5.24	Discrétisation temporelle hétérogène – cas de deux populations – scénario 2. .	138
6.1	Diagramme d’agents-modèles – Action de modélisation 1 – Cadre de modélisation d’un système prédateurs-proies.	151
6.2	Diagramme d’agents-modèles – Cadre de modélisation pour expérimenter différentes approches de modélisation pour un peuplement de poissons avec un habitat fixé.	152
6.3	Action 1 – expérience 1 – section 6.4.3 : évolution en nombre du peuplement au cours de la simulation.	153
6.4	Action 1 – expérience 1 – section 6.4.3 : distribution spatiale du peuplement suivant la modélisation utilisée (pas de temps quelconque).	154
6.5	Action 1 – expérience 2 – section 6.4.4 : évolution dans l’espace des phases du nombre de prédateurs en fonction du nombre de proies.	155
6.6	Diagramme d’agents-modèles – Cadre de modélisation pour des scénarios d’étude des merlus et des sardines.	159
6.7	Expérience dimensionnement du modèle EDP au GDG– résultats obtenus en simulation (courbe avec les bulles) et données EVHOE observées (courbe en traits continus) pour les peuplements de merlus et de sardines.	160
6.8	Expérience dimensionnement du modèle EDP au GDG– résultats obtenus en simulation : exemples de distribution spatiale des deux espèces étudiées. . . .	160
6.9	Diagramme d’agents-modèles – Cadre de modélisation pour des scénarios d’étude d’écosystème marine (Pm : poids moyen global, Rm : Niveau trophique moyen, Lm : latitude préférentielle moyenne).	163
6.10	Action 3 – expérience 1 : résultats de simulation.	165
6.11	Diagramme d’agents-modèles – Cadre de modélisation pour des scénarios d’étude du peuplement marin du Golfe de Gascogne couplé à une pêcherie . .	167
6.12	Action 4 – expérience 1 : évolution de la distribution du peuplement suivant le modèle utilisé : 3 pas de temps sont présentés.	169
6.13	Diagramme d’agents-modèles – Cadre de modélisation pour des scénarios d’étude du marché des navires de pêches	171
6.14	Diagramme d’agents-modèles – Cadre de modélisation pour des scénarios d’étude du marché des navires de pêches	172
6.15	Action de modélisation du marché des navires de pêche – résultats scénario 1.	174
6.16	Action de modélisation du marché des navires de pêche – résultats scénario 2.	174
6.17	Illustration d’une organisation graphique des actions de modélisation, des différents modèles, des jeux de paramètres et des simulations.	177
B.1	La quantité pêchée par un pêcheur en fonction de la $CPUE$ (voir table B.1 pour les détails)	216
B.2	L’évolution de l’efficacité d’une unité de pêche avec réinvestissements réguliers du pêcheur.	218

Liste des tableaux

5.1	Comportement de l'agent dans un IBM, cas trivial : valeurs des paramètres. .	89
5.2	Comportement de l'agent dans un IBM, synchronicité de comportements indépendants : valeurs des paramètres.	91
5.3	Comportement de l'agent dans un IBM, synchronisation de comportements dépendants : valeur des paramètres.	93
5.4	Comportement de l'agent dans un IBM, sensibilité des comportements aux états initiaux : valeur des paramètres.	94
5.5	Comportement de l'agent dans un IBM, comportement stochastique : valeur des paramètres.	96
5.6	Comportement de l'agent dans un IBM, comportement stochastique : simulations et résultats.	97
5.7	Sensibilité en nombre d'agents, cas d'une population : valeurs des paramètres.	104
5.8	Sensibilité en nombre d'agents, cas d'une population : simulations et résultats.	104
5.9	Sensibilité à la discrétisation spatiale, interaction avec l'environnement : valeurs des paramètres	111
5.10	Sensibilité à la discrétisation spatiale, interaction avec l'environnement : simulations et résultats.	113
5.11	Sensibilité à la discrétisation spatiale, interaction entre les agents : valeurs des paramètres.	113
5.12	Sensibilité à la discrétisation spatiale, interaction entre les agents : simulations et résultats.	114
5.13	Sensibilité à la discrétisation temporelle, valeurs des paramètres.	118
5.14	Sensibilité à la discrétisation temporelle : simulations et résultats.	118
5.15	Sensibilité à la discrétisation temporelle : écart d entre les valeurs (nombre d'agents) minimale et maximale suivant le pas de d'activation.	118
5.16	Hétérogénéité du nombre d'agents : valeurs des paramètres.	124
5.17	Hétérogénéité du nombre d'agents : simulations et résultats.	125

5.18	Hétérogénéité spatiale : valeurs des paramètres	130
5.19	Hétérogénéité spatiale : simulations et résultats.	131
5.20	Discretisation temporelle hétérogène : valeurs des paramètres	135
5.21	Hétérogénéité temporelle – scénario 1 : simulations	136
5.22	Hétérogénéité temporelle – scénario 2 : simulations	136
6.1	Action 1 – expérience 1 : valeurs des paramètres.	153
6.2	Action 1 – expérience 2 : valeurs des paramètres	155
6.3	$Q_{individus}$: nombre d’individus par espèce (données EVHOE-2004) – Q_{agents} : nombre d’agents espéré par espèce pour un nombre total de 10^5	164
B.1	Détails de la figure B.1	217
B.2	Problématique écologique : les paramètres	223
B.3	Problématique écologique : les variables, les variables liées et leur initialisation	223
B.4	Problématique économique : les paramètres	224
B.5	Problématique économique : les variables, les variables liées et leur initialisation	224

Chapitre 1

Introduction

1.1 Contexte de la thèse

Le travail présenté dans cette thèse s'appuie sur les travaux que nous avons effectués dans le cadre du projet ANR Chaloupe¹ au sein de l'équipe EBV, *Ecosystémique et Biologie Virtuelle*, du LISYC, *Laboratoire d'Informatique des SYstèmes Complexes*. Ce projet a alimenté en cas d'étude concrets notre questionnement et nos problématiques informatiques. Il part du constat écologique que les peuplements de poissons ont fortement évolué quantitativement et qualitativement durant les trente dernières années. Les deux facteurs exogènes identifiés par les experts comme responsables de ces modifications sont les activités de pêche et les changements environnementaux, en particulier le réchauffement climatique. La question posée est donc de pouvoir envisager divers scénarios d'évolution du système peuplements-pêcheries dans un environnement variable suivant différentes politiques de gestion de l'exploitation des ressources et différentes hypothèses sur le réchauffement climatique, afin de comprendre les multiples mécanismes en œuvre qui expliquent les dynamiques observées.

La modélisation et la simulation de systèmes complexes est un enjeu critique pour des domaines comme l'écologie et l'économie. Notre capacité à construire des outils permettant d'étudier et de manipuler des modèles d'écosystèmes, de marchés financiers ou encore de systèmes sociaux est alors une problématique essentielle. Cependant, l'étude du système peuplements-pêcheries est d'autant plus difficile qu'il est très hétérogène, dynamique et ouvert. Il est composé de nombreux éléments hétérogènes d'un point de vue structurel et dynamique : il existe des processus de natures différentes et des échelles multiples de description et d'analyse. Il constitue donc un bon exemple de système complexe. La

¹ <http://www.projet-chaloupe.fr/>, les partenaires sont l'IFREMER, l'IRD, l'ENIB, l'INRH, le CEDEM, l'ULCO, le CNRS/MNHN et le *World Fish Center*.

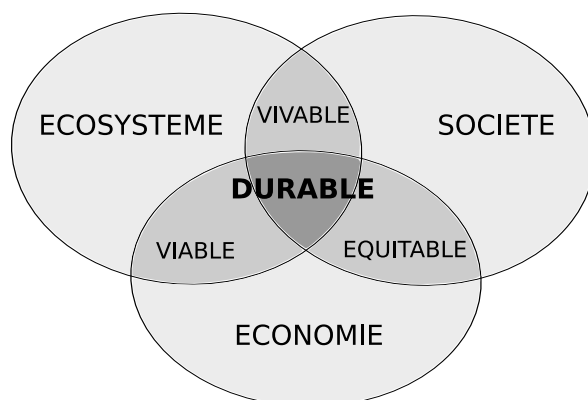


FIG. 1.1: Le développement durable des trois systèmes écologique, économique et social.

problématique est d'autant plus riche qu'elle se pose en termes de co-viabilité des différentes composantes du système dans le cadre paradigmatique du développement durable². La figure 1.1 présente les trois dimensions qui structurent le problème. L'enjeu est alors d'envisager la co-viabilité des peuplements et de leur exploitation équitable par des groupes sociaux. Ainsi, il est nécessaire *de facto* de prendre en compte non pas le point de vue d'une composante du système, mais de pouvoir envisager une modélisation qui permette aux différentes composantes – aux différents points de vue – d'être prises en compte de manière originale et autonome tout en permettant des interactions multiples entre les thématiques.

De ce fait, nous avons besoin d'un outil de modélisation qui permette la construction de représentations multiples. Il nous faut donc d'une part construire des modèles écosystémiques, et d'autre part aborder le problème de la plate-forme de simulation, donc des moyens informatiques théoriques et pragmatiques, à utiliser ou à construire pour mettre en œuvre nos modèles.

1.2 Problématiques et enjeux

La possibilité de répondre aux questions posées par Chaloupe tient en premier lieu à notre capacité à proposer un outil de modélisation qui autorise la construction d'une représentation d'un système complexe et cette représentation doit pouvoir elle-même être complexe. En effet, un système complexe, comme un système peuplements-pêcheries, est caractérisé par la

² Remarquons que pour certains, le développement durable est différent du concept de co-viabilité dans le sens où, dans une perspective de développement durable, les composantes écologiques sont considérées comme des ressources ou des stocks à gérer. A la différence, la co-viabilité n'implique pas forcément un tel asservissement du système écologique au système économique. Le lecteur pourra trouver plus d'informations sur la notion de développement durable dans (Laval 01).

multiplicité du nombre d'éléments et de processus qui le constituent, par leur hétérogénéité et par leurs interactions, ce que décrit le mot complexe pris au sens étymologique «complexus» et qui signifie «ce qui est tissé ensemble» (Benkirane 02). La question est donc systémique, et écosystémique dans notre cas (Fromentin 07), et il s'agit de réfléchir à la manière de construire un modèle de phénomènes hétérogènes d'entités en interaction et permettre de représenter séparément les différents phénomènes en les faisant «cohabiter» dans un même modèle.

Se pose également la question des processus de modélisation et de simulation. La modélisation doit permettre aux thématiciens de se construire des représentations, de les manipuler et d'apprendre d'elles en les élaborant. Cependant, la production et la mise en œuvre d'un modèle complexe fait appel à différentes compétences thématiques de par la nature multiple du système que l'on cherche à représenter. D'autre part, il est rare de travailler sur un seul modèle et l'étude d'un modèle entre alors dans un processus plus large de simulation qui doit permettre de construire, d'étudier et d'expérimenter différents modèles, construits de manière collaborative et associés à de multiples jeux de paramètres. Les processus de modélisation et de simulation sont alors eux-mêmes complexes. Ils requièrent par conséquent une méthodologie rigoureuse et un outil permettant de rationaliser la structure du modèle et les différents matériaux de modélisation et facilitant une modélisation incrémentale.

C'est pourquoi, il nous semble nécessaire de proposer un outil de simulation qui permette d'une part de clarifier les rôles des différents intervenants dans la modélisation et d'autre part de les assister dans certaines activités de modélisation et de simulation. Aussi bien pour la construction du modèle complexe, que pour l'identification de la valeur des paramètres dans le cas de modèles surparamétrés, la plate-forme de simulation doit pouvoir faciliter, voire guider, la démarche des modélisateurs. Dès lors, envisager des solutions d'automatisation des processus de modélisation et de simulation suppose de formaliser un certain nombre de processus et d'éléments du modèle, mais également de mieux identifier l'activité des thématiciens. C'est grâce à cela que nous pouvons proposer une méthodologie et un outil permettant la co-construction et la co-manipulation du modèle complexe par le système et les experts.

Par la suite, nous proposons une solution basée sur le concept d'*agent-modèle* afin de construire le modèle du système complexe comme une société de modèles en interaction. Nous nous sommes avant tout concentrés sur le processus de construction du modèle, car il conditionne l'ensemble des activités de simulation. Et nous proposons d'utiliser la solution agent pour nous permettre de «perdre» du contrôle sur le système en le distribuant dans des agents, afin de gagner *in fine* en maîtrise sur la modélisation et sur le modèle. La construction de modèles hétérogènes nous a obligé à proposer une solution de couplage des modèles autorisant l'hétérogénéité de modélisation et de description (en termes d'échelles en particulier). Ceci nous a conduit à adopter une démarche de modélisation centrée sur les données, en nous appuyant sur la modélisation orientée pattern.

En outre, il est également nécessaire de pouvoir garantir que toute représentation computationnelle est fidèle à la représentation conceptuelle que s'est faite le thématicien

et qu'il pense manipuler. En effet, il n'est en aucun cas de la responsabilité de l'informaticien de juger de la qualité du modèle construit par les thématiciens. Toutefois, il est nécessaire que le cadre computationnel soit parfaitement maîtrisé et distingué des problématiques conceptuelles des modèles. Les modèles centrés individu (IBM : *Individual Based Modelling*) en particulier sont souvent tellement complexes qu'il n'y a pas meilleure représentation de ces modèles que le code qui les met en œuvre, ce qui interroge sur la possibilité pour le modèle d'être un médiateur entre le réel et la théorie (Morgan 99). La question est donc de chercher à mieux étudier les biais dans les modèles computationnels dûs à des hypothèses computationnelles qui ne sont pas toujours explicites dans les plates-formes de simulation, ceci afin d'envisager une meilleure maîtrise des modèles en général et IBM en particulier. Nous avons par là-même étudié les biais computationnels dans les modèles de peuplements, modèles qui, de par les processus en jeu, permettent d'exhiber les différentes problématiques qui se posent. Ainsi, cette étude permet d'assurer, au niveau computationnel, une plus grande maîtrise des modèles et d'envisager de nouvelles solutions de contrôle par les agents de ces modèles.

Nous proposons par conséquent le plan de thèse suivant :

- ▷ Le premier chapitre replace ce travail dans son contexte. Préalable indispensable à la proposition, il introduit les concepts que nous manipulons et les travaux sur lesquels nous nous appuyons. La première section s'attache à définir la question des systèmes complexes comme elle s'est posée pour nous et à tenter d'en identifier les éléments principaux par rapport à notre problématique. Nous décrivons donc là les systèmes qui seront l'objet d'une modélisation. La deuxième section porte sur la place de l'informatique dans le processus de modélisation. Nous abordons ici la notion de modèle computationnel et la question particulièrement centrale de notre travail de la multi-modélisation et du couplage. La troisième section développe la question du processus de simulation, c'est-à-dire la mise en « mouvement » des modèles et leur expérimentation. Nous exhibons ici les problématiques qui se posent et nous introduisons la méthodologie de modélisation orientée pattern sur laquelle nous fondons notre proposition. Enfin, dans la dernière section, nous re-situons la question de l'approche agent dans notre problématique à la lumière des différentes questions de la modélisation et de la simulation que nous avons précédemment présentées. Cette section nous offre également l'occasion de positionner les plates-formes de simulation et d'en présenter les caractéristiques principales.
- ▷ Le deuxième chapitre formule notre positionnement par rapport au contexte que nous venons de présenter. Ainsi, nous reformulons trois éléments précis du contexte, l'approche orientée pattern, l'approche *in virtuo* et la modélisation et la simulation à l'aide d'agents, sur lesquels nous nous adossons pour formuler de manière succincte notre proposition.
- ▷ Le troisième chapitre présente notre proposition en termes de modèle, d'outil et de méthodologie. Il commence par la présentation de notre solution conceptuelle basée *agent-modèle* que nous décrivons suivant l'approche *Voyelles*. Nous précisons ainsi ce que nous entendons par environnement de modélisation et comment nous intégrons

des agents dans la modélisation, pour finir sur la question de la société de modèles et celle de l'utilisateur dans la modélisation et la simulation. Ensuite, dans la seconde section, nous abordons quelques points clés du simulateur que nous avons construit et qui nous paraissent nécessiter une présentation. Enfin, la dernière section décrit notre proposition méthodologique basée sur la méthodologie orientée pattern. Nous y présentons d'abord la méthodologie d'une manière globale, avant de détailler les différents éléments qui la structurent, la question et le cadre de modélisation, la phase de modélisation et la phase de simulation.

- ▷ Le quatrième chapitre est composé de six sections qui présentent une étude des biais computationnels dans les modèles IBM de peuplements. Ce chapitre est important car il permet d'abord de valider notre capacité à construire, avec notre outil et notre approche, des modèles de peuplements que nous maîtrisons. Il nous permet surtout d'aborder les principaux biais computationnels qui peuvent induire des dynamiques totalement différentes dans des modèles computationnels issus pourtant d'un même modèle conceptuel par certains points trivial.
- ▷ Le cinquième chapitre est construit autour de la présentation d'actions de modélisation que nous avons mises en œuvre dans le cadre du projet Chaloupe. Nos travaux de modélisation y sont résumés autour de différents axes d'étude dont l'objectif est de montrer la pertinence de notre proposition pour l'étude d'un cas réel, en faisant transparaître le travail effectué sur le cas d'étude.

Chapitre 2

Etat de l'art

2.1 Introduction

Nous étudions la modélisation et la simulation des systèmes complexes du type écosystème exploité. Remarquons que nous nous sommes plutôt focalisés sur la modélisation, c'est-à-dire la construction d'un modèle, parce que celle-ci conditionne toute l'activité de simulation. C'est pourquoi notre présentation de la modélisation est dans un premier temps dégagée des autres activités de la simulation, mais nous cherchons ensuite à la replacer dans le contexte de la simulation au sens large. L'état de l'art explicite par conséquent les trois éléments sur lesquels porte notre travail : 1° les systèmes complexes, 2° la modélisation et 3° la simulation.

Nous commençons alors par définir les systèmes complexes, non de manière absolue, mais en identifiant ce qui les caractérise et qui doit être pris en compte dans la modélisation. La complexité d'un système se définit avant tout par un point de vue sur ce système et la perception que l'on en a, ce qui n'empêche pas ce système d'être par nature complexe. Mais, dans une approche de modélisation, c'est bien le point de vue et la représentation que l'on se fait du système qui sont importants. Remarquons également que le terme «système complexe» représente en soi une multitude de systèmes et que, par conséquent, nous nous attachons à caractériser avant tout les systèmes de types écosystèmes exploités, afin ensuite, autant que possible, d'abstraire leurs caractéristiques.

Concernant le processus de modélisation, nous proposons dans un premier temps de mieux définir l'activité de construction d'un modèle. Nous dégageons dans ce processus, les questions de modélisation thématique liées aux systèmes étudiés d'une part, et les questions de modélisation opérationnelle qui nous intéressent en tant qu'informaticien d'autre part. Nous terminons sur la question de la multi-modélisation, qui répond à l'hétérogénéité des échelles

et des points de vue que nous évoquons au sujet des systèmes complexes. Nous abordons alors également la problématique du couplage de modèles hétérogènes, question qui est au cœur de notre proposition.

Concernant le processus de simulation, il s’agit de considérer la simulation au sens large d’abord, c’est-à-dire incluant le processus de modélisation, puis d’aborder la simulation en tant que processus indépendant de la modélisation, en exhibant le travail important et l’expertise que l’activité de simulation même requiert. Nous abordons alors les moyens existants d’analyse de simulation et d’exploration des modèles, qui nous amènent sur les méthodes de rationalisation du processus de simulation. Nous terminons cette section par la présentation de l’approche orientée pattern sur laquelle nous structurons notre proposition méthodologique et conceptuelle.

Enfin, nous terminons cet état de l’art en présentant nos problématiques du point de vue de l’approche agent. Nous identifions ainsi les solutions apportées jusque là par l’approche agent pour la modélisation et la simulation et nous terminons alors sur la modélisation et de la simulation à l’aide d’agents.

2.2 Les systèmes complexes

2.2.1 Qu’est-ce qu’un système complexe ?

“Saying that you are studying complex systems is like defining the bulk of zoology by calling it the study of “non-elephant animals”.” – S. Eubank

Le premier problème auquel nous sommes confrontés dans le cadre de ce travail est celui des systèmes complexes. Un système complexe a généralement peu ou pas d’organisation centrale, un graphe d’interaction non trivial et des interactions non linéaires. S’il avait une organisation centrale, on serait capable de limiter l’expression de sa dynamique à une de ces composantes. Si son graphe d’interaction était trivial, on serait peut-être davantage en mesure de traiter chaque composante de manière indépendante, en divisant pour comprendre. Et cette division serait d’autant plus simple si les interactions étaient linéaires. Un système complexe est souvent caractérisé de récurrent, donc composé de systèmes complexes, d’ouvert, c’est-à-dire dont les frontières sont difficiles à définir et qui est difficilement dissociable de son environnement. Il se distingue de plus par l’existence de boucles de rétroaction (notion issue de la cybernétique), qui peuvent avoir des dynamiques lentes et donc être difficilement identifiables. La question des boucles de rétroaction est, parmi les problématiques de la complexité, une des plus difficiles à estimer et à étudier. Un exemple en est le réchauffement climatique qui peut induire des dynamiques inattendues dans les peuplements de poissons. Ainsi, il peut avoir une influence négative sur les peuplements et indirectement sur la pêche. Mais, la diminution du nombre de prédateurs peut alors avoir un impact positif sur les

proies de ces prédateurs, ce qui peut alors pousser les pêcheurs à se porter sur les espèces de type proies. Ce report de l'effort de pêche sur les proies peut entraîner une amélioration de la situation des prédateurs et influencer négativement les proies et ainsi indirectement influencer à nouveau négativement les prédateurs.

Cependant, avec ces seuls éléments de description, le système complexe reste encore un objet abstrait. Essayons de compléter l'idée que l'on se fait de ce type de système en identifiant les notions conceptuelles qui s'y attachent. L'étude des systèmes complexes introduit en général aux notions d'auto-organisation (von Foerster 60; Ashby 04), ce que Morin nommait le Désordre organisateur (Morin 77), et d'émergence faible ou forte. Les notions de chaos (Li 75) et de robustesse locale¹ sont également fondatrices d'une certaine perception que l'on se fait des systèmes complexes. Que de notions qui sont l'objet chacune d'elles d'études, de recherches et de thèses, et qui rendent la proximité d'un système complexe légèrement intimidante. Classiquement cependant, on dira des systèmes complexes qu'ils sont irréguliers et irréductibles (s'ils étaient réguliers ou symétriques, ils seraient alors réductibles).

On peut alors noter la remarque de Louis J. Gross, de l'Université du Tennessee, qui dit d'un système qu'il est complexe quand un scientifique bien entraîné et expert de la question ne peut pas rapidement savoir ou avoir le sentiment de savoir comment ce système va se comporter² (Gross 03). Sa remarque rejoint dans un premier temps la définition étymologique du mot *compliqué* qui signifie d'un problème qu'il faut du temps et du talent pour le comprendre. Surtout, la description du système complexe n'est plus celle de sa nature objective, mais du vécu subjectif que l'on a de son étude. Or, ce qui caractérise avant tout un système complexe, c'est son hétérogénéité, aussi bien du point de vue structurel que dynamique. C'est pourquoi, dans la perception que l'on a de lui s'exprime une multitude de niveaux d'observation, de processus différents identifiés à des échelles différentes. Et parler ici d'échelles différentes signifient que les processus étudiés peuvent s'exprimer à des niveaux de description spécifiques, avec des rythmes d'évolution qui leur sont propres et que Weber nomme des rythmes de variabilité (Weber 95). Ce sont par ailleurs, pour Weber, ces multiples rythmes de variabilité qui permettent la viabilité de l'organisme complexe – micro-organismes, cellules, fonctions biologiques, etc. On peut retrouver cette idée que la complexité est avant tout dans la perception de l'observateur, ce qui n'empêche pas d'ailleurs le système d'être effectivement complexe, dans la notion abordée précédemment d'auto-organisation. En effet, le degré d'auto-organisation d'un système dépend de l'observateur qui, dans les modèles mathématiques ou physiques, définit les propriétés, aspects et états du système, et par conséquent son niveau d'entropie ou d'ordre (Gershenson 03). Or, c'est ce niveau d'entropie

¹ La notion de robustesse locale permet essentiellement de dire qu'un système complexe est un système dont les composantes sont fortement interconnectées, ce qui lui permet d'être localement plus robuste aux perturbations.

² Louis J. Gross dit précisément ceci : "Complexity? A complex system is one in which a well trained scientist knowledgeable about the system of concern cannot rapidly intuit how the system will behave. By rapidly here, I mean with the aid of simple computational tools (pencil and paper, computer, calculator, etc.) in a few minutes."

et d’ordre du système qui permet de décrire son niveau d’auto-organisation. De plus, la perception de l’ordre ou du désordre, de l’organisation ou de la désorganisation, dépend en réalité du but du système qui est défini par l’observateur (Beer 66). Par conséquent, pour Gershenson et Heylighen, les systèmes auto-organisés sont moins un type de systèmes, qu’une perspective sur l’étude, la compréhension, la conception, le contrôle et la construction des systèmes, ce qui était en réalité déjà proposé par Ashby (Ashby 04). Pour conclure, c’est donc de la démarche de l’expert, de la mise en œuvre de son expertise, de son “talent”, dans le processus d’appréhension du système, qui passe sûrement par des représentations extérieures – des modèles –, que les tenants et les aboutissants du système vont pouvoir être petit à petit identifiés, et c’est là peut-être toute la problématique des systèmes complexes.

2.2.2 Quels types de systèmes ?

Avant d’étudier des systèmes complexes, nous étudions des écosystèmes exploités dans des environnements variables. Il est donc important que l’on définisse ce type de systèmes. Intéressons-nous d’abord à la notion d’écosystème. Un écosystème est l’unité écologique de base formée par le milieu (le biotope) et les myriades d’organismes individuels, animaux, végétaux et bactériens (le biocénose) qui y vivent, ce que Tansley a décrit comme l’association d’un environnement physico-chimique spécifique avec une communauté vivante (Ramade 03). L’écosystème est donc caractérisé par sa biomasse et le réseau d’interdépendances des constituants qui permet le maintien et le développement de la vie. Il est une unité fonctionnelle en tant que tel et ne peut être identifié au plan fonctionnel à la somme des propriétés inhérentes à chacun de ses constituants élémentaires comme la démarche classique “réductionniste” pourrait le faire conclure (Ramade 03). Ainsi, les propriétés émergentes des écosystèmes sont définies par trois caractéristiques (Nielsen 00) : 1° elles n’existent pas dans les sous-systèmes pris indépendamment ; 2° elles émergent à des niveaux supérieurs et résultent des interactions des sous-systèmes ; 3° de nouvelles propriétés apparaissent à un seul niveau du système et ne sont pas déductibles des observations faites sur les niveaux ou compartiments inférieurs du système. Au final, différents niveaux d’organisation, comme les notions d’individu et de population, permettent de développer différents éclairages sur ce que sont les écosystèmes. Ainsi, l’écosystème peut être perçu comme des individus qui s’organisent en groupes géographiques, en groupes d’âge, en associations inter-spécifiques (coopération entre espèces) ou autres, et en populations (Green 05a). Pour Andrewartha et Birch par exemple, les processus structurant l’écosystème sont au niveau des populations. L’écosystème ne se régule pas lui-même par ses propres mécanismes de réponse ou boucles d’interaction mais est dirigé par des processus stochastiques et ce sont les populations qui se régulent et qui structurent l’écosystème (Andrewartha 54).

Intéressons nous maintenant aux aspects économiques. Très généralement, l’économie étudie les activités humaines liées à la production, à la distribution, à l’échange et à la consommation de biens et de services. Une économie regroupe évidemment des biens matériels, des

actions, des institutions ou des marchés, mais derrière cela, se structure tout un ensemble de processus moins tangibles, comme les attentes subjectives, les anticipations et les hypothèses faites par les humains qui induisent des dynamiques complexes et changeantes dans l'économie globale (Arthur 95). L'économie d'un écosystème se caractérise par l'extraction d'un matériel brut, par la production d'une richesse injectée dans l'économie générale, par un capital disponible pour l'exercer et des coûts de fonctionnement. Cette économie est exercée par des groupes sociaux mouvants et répond à une démarche sociétale qui fait que, parfois, elle s'exerce en deçà du coût d'opportunité de son capital. On remarquera d'ailleurs que la composition d'une économie est indissociable d'une organisation sociale. Dans notre cas, l'exploitation d'un écosystème ou d'une ressource par une économie est indissociable du groupe social formé par les différents acteurs de cette économie. La viabilité ou la recherche de l'optimalité de l'économie est donc indissociable des acteurs sociaux qui l'exercent et de la ressource exploitée. Les échelles de description des systèmes économiques sont donc là aussi multiples, tout comme les échelles d'organisation et les processus peuvent s'exprimer au niveau des moyens de production (capital), des dynamiques individuelles d'exploitation (microéconomie) ou encore au niveau de l'organisation de l'exploitation par des politiques de gestion (gouvernance).

En conclusion, nous constatons qu'il y a de nombreuses échelles d'organisation au travers desquelles on peut étudier la co-viabilité d'un écosystème et de son exploitation, échelles d'organisation qui deviennent alors tour à tour l'échelle d'observation. Cependant, le fait d'observer un système et de l'expliquer à des niveaux différents d'abstraction ne change pas le système. Par contre, la perception que l'on a de ce système, et par voie de conséquence l'interprétation de l'évolution de ses dynamiques, dépendent alors de cette échelle d'observation. Si l'on observe par exemple un écosystème dans sa globalité, on peut percevoir la stabilisation d'une population donnée comme l'expression d'un mécanisme de capacité maximale du milieu ou de l'écosystème. Au niveau de la population, ou même individuel, cette stabilisation de la population pourra être expliquée par un comportement territorial des individus, comportement qui sert alors de mécanisme régulateur du nombre d'individus de la population.

2.2.3 Cadre théorique

2.2.3 - A Théorie générale des systèmes

La démarche classique pour appréhender un système est la démarche analytique qui repose sur le postulat de la causalité. Les phénomènes s'expliquent par un enchaînement de causalités et, si un phénomène est trop complexe, il suffit de le décomposer en ces sous-phénomènes pour en comprendre le fonctionnement. La Théorie Générale des Systèmes s'oppose à cette démarche, en s'appuyant au contraire sur la finalité des choses. Le phénomène complexe ne doit pas être décomposé, mais intégré dans son contexte comme

un phénomène et perçu comme une boîte noire (Wiener 65). On s’intéresse alors au comportement du phénomène et non à son fonctionnement et c’est l’étude des interactions des différents phénomènes qui permet d’en comprendre les comportements. C’est en percevant les phénomènes dans le système qu’ils forment de par leurs interactions que le rôle de ces phénomènes, et donc la dynamique du système, prend du sens. Par conséquent, la Théorie Générale des Systèmes nous dit que pour appréhender un système complexe, il n’est pas forcément nécessaire, au moins dans un premier temps, de comprendre le fonctionnement des phénomènes qui structurent ce système. Il s’agit avant tout de décrire comment ces phénomènes interagissent afin d’expliquer leurs comportements qui engendrent la dynamique globale du système.

L’étude des systèmes complexes de type écosystème exploité se prête bien à cette approche, comme le fait remarquer Bousquet pour l’étude des écosystèmes (Bousquet 01). L’écosystème est en effet un ensemble de parties en interactions d’où émergent des propriétés, et c’est également une organisation qui est plus que la somme de ses parties. Ainsi, étudier ce type de systèmes, c’est les appréhender dans leur complexité, comme le suggère (Desmeulles 06b) pour les systèmes biologiques, en ne cherchant pas forcément à décomposer les phénomènes, mais en les considérant comme une organisation de phénomènes dont les interactions expliquent la dynamique globale.

2.2.3 - B Systèmes dynamiques et non-linéarité

D’autres approches cherchent à appréhender les systèmes complexes. En particulier, la dynamique des systèmes et la théorie du chaos se sont intéressées aux systèmes dynamiques non linéaires avec pour objectif de comprendre comment les changements à l’intérieur de ces systèmes se produisent. Avant toute chose, rappelons tout de même qu’un système dynamique est un système classique, donc déterministe. Un système dynamique non linéaire est alors un système dont les comportements sont chaotiques.

La dynamique des systèmes est une méthode de modélisation (Forrester 69) dont l’objectif est de décrire les liens de causes à effets et de délier les différentes interactions, en identifiant les boucles de rétroaction. Pour cela, elle s’appuie sur les notions de stocks et de flux et cherche à déterminer les relations circulaires, inter-bloquées et retardées, entre les composants ainsi que le comportement des composants individuels eux-mêmes. C’est une démarche holistique qui s’intéresse donc au système dans son ensemble et qui part de l’hypothèse que toute cause produit un effet. Les composantes du système sont alors modélisées comme des nœuds représentant des variables du système et connectés par des arcs représentant les effets causaux. On associe alors à chaque changement de variable la modification d’une autre variable suivant une équation ou une règle données et on construit alors des diagrammes de boucles causales (CLD – Causal Loop Diagrams) qui identifient les variables importantes et la manière dont elles interagissent. On est donc dans une démarche très déclarative qui peut aisément s’écrire sur un simple tableur.

La théorie du chaos, dont le terme chaos a été introduit dans (Li 75), traite également des systèmes dynamiques. Les systèmes y sont rigoureusement déterministes, mais caractérisés par une forte sensibilité aux conditions initiales (Poincaré, fin du XIX^{eme}) et une forte récurrence, qui font qu'ils sont non prédictibles sur le long terme. Ce sont donc des systèmes aux comportements extrêmement désordonnés, dits chaotiques. Il est intéressant de constater que cette théorie proposa d'étudier un système non plus en résolvant ses équations, mais en cherchant les points fixes le caractérisant. Ainsi, la question est de savoir si un système va converger vers un état stationnaire à long terme ou non. C'est ce changement de point de vue qui nous intéresse et qui propose d'observer un système et sa dynamique dans son ensemble, sans chercher à en comprendre toutes les irrégularités. Dans cette perspective, la connaissance des attracteurs d'un système nous en apprend bien plus sur le système en question que la connaissance des composantes du système. Cette représentation du comportement d'un système dans l'espace des phases, sous forme d'attracteurs par exemple, nous permet alors de construire des patterns qualitatifs des comportements de systèmes complexes représentant un certain ordre géométrique dans le chaos. Ce point de vue a par ailleurs été utilisé dans l'étude des écosystèmes et Holling s'appuie sur les notions de points d'équilibre et d'attracteurs dans ses travaux (Holling 73).

2.2.4 Echelles d'analyse et d'observation

Pour terminer sur la question des systèmes complexes, nous abordons la question des échelles et donc de l'hétérogénéité qui est peut-être ce que l'on retiendra de la question de la complexité. Mais, avant tout, nous constatons que notre manière d'aborder un problème est en partie prisonnière de notre manière de représenter le contexte du problème. Le choix des échelles de description et d'observation a un impact sur la représentation que l'on se fait des choses. Or, la notion d'échelle se trouve partout dans notre rapport au réel et dans notre construction de représentations de ce réel. En particulier, appréhender un système complexe requiert de prendre en compte son hétérogénéité et, de ce fait, les échelles de description et d'analyse se multiplient dans la représentation du système (Parrott 06). Il faut par conséquent être capable de mêler les points de vue, les perspectives sur le système et les échelles de description et d'analyse et de pouvoir les associer dans une construction commune de représentations. Mais, cette complexification de la perception que l'on a du système entraîne par conséquent une représentation beaucoup plus riche et donc beaucoup moins accessible. Le risque est alors que la représentation globale du système soit illisible et impraticable. La question est donc de pouvoir considérer les différentes composantes du système dans leur originalité, de pouvoir manipuler les différentes échelles utilisées, tout en gardant la maîtrise sur le processus de description et d'analyse du système. Il s'agit donc de rationaliser la construction de la représentation globale que l'on se fait du système et de la mettre en rapport avec sa dynamique globale.

Les écosystèmes exploités entrent tout à fait dans cette problématique. De plus en plus

étudiés, ils sont par définition un objet d’étude pluridisciplinaire, abordés avec les points de vues de l’écologie, de l’économie et du social. Mais d’autres domaines sont également sollicités, comme les sciences sociales, les sciences de la nature et les sciences de la représentation des connaissances (Weber 95). Les représentations que l’on se fait de ces systèmes sont donc caractérisées par une multiplication des échelles d’organisation et d’observation, qui se placent au niveau de l’individu, du groupe ou du système global. Appréhender un tel système nécessite donc que cet ensemble de concepts et de représentations issus de contextes différents puisse être mis en cohérence.

2.3 Le processus de modélisation

2.3.1 Qu’est-ce qu’un modèle ?

*“De toutes les illusions, la plus périlleuse consiste à penser qu’il n’existe qu’une seule réalité.” –
(Watzlawick 84)*

Les modèles sont parmi les outils de premier plan de la science. On peut penser au modèle de Lorentz de l’atmosphère, au modèle de Poincaré du système solaire, aux équations de Navier-Stokes modélisant le mouvement des fluides dans l’approximation des milieux continus, au modèle de Lotka-Volterra sur l’interaction proie-prédateur, ou plus récemment aux modèles basés agents et évolutionnaires dans les sciences sociales. La multiplicité des modèles dans tous les domaines ne permet cependant pas encore de répondre au questionnement épistémologique que le modèle pose. D’après Frigg et Hartmann, ce questionnement est sémantique – quelle est la fonction de représentation d’un modèle ? –, ontologique – quel type d’objets sont les modèles ? – ou épistémologique – comment apprend-on des modèles ? –, et, évidemment, philosophique – comment les modèles sont-ils reliés à la théorie ? (Frigg 06). Sans aller au-delà du contexte de notre travail, peut-on déjà caractériser les modèles et leur fonction dans l’étude de systèmes à la lumière de ce questionnement épistémologique ?

On peut penser que les modèles permettent d’apprendre des choses sur le monde. Ils permettent d’explorer des hypothèses et d’étudier des questions sans expérimenter directement sur le monde réel, ce qui peut être utile pour des raisons de : 1^o reproductivité, on peut reproduire l’expérience plus facilement si l’on contrôle l’ensemble du système ; 2^o facilité, il est plus facile d’effectuer des expérimentations sur un modèle, comme dans le cas de modèles sociaux de comportements en situation de crise ; 3^o efficacité, les modèles étant moins riches que le système qu’ils représentent, il peut être plus aisé de tirer des conclusions, d’inventer de nouvelles solutions ou autre en travaillant sur un modèle. Les modèles en ce sens permettent une réflexion de remplacement ou *surrogate reasoning* (Swoyer 91). Ils permettent de réfléchir et d’interagir avec une “copie” plus ou moins abstraite ou plus ou moins détaillée du système auquel on s’intéresse.

Le modèle est-il alors un objet linguistique? Comme pour les structures mnésiques qui sont images pour certains et langages pour d'autres³, il n'y pas encore de réponse à cette question qui porte aussi bien sur le langage que sur la notion de représentation. Ontologiquement, un modèle peut être un objet physique, une description, une équation ou une structure informationnelle. Dans notre problématique, un modèle est une entité abstraite que l'on peut alors définir comme une construction théorique représentant quelque chose, avec un ensemble de variables et de relations logiques et quantitatives entre elles.

On peut alors se poser la question de savoir comment construire et manipuler un modèle dès lors qu'il est une entité abstraite. De nombreux ouvrages sur ce sujet existent et certains explorent la solution de l'expérimentation mentale du modèle (Brown 91). D'après Morgan (Morgan 99), on apprend d'un modèle quand on le construit et quand on le manipule. Il n'y a pas de règle fixe ou de recette pour construire un modèle et donc le fait de le construire nous permet de savoir ce qu'est le modèle et de le comprendre (Frigg 06). Dans notre problématique, les questions de la construction et de la manipulation se posent pour des modèles mathématiques et computationnels nécessitant l'utilisation de l'outil informatique. Il s'agit donc pour nous de réfléchir à la modélisation en tant que processus qui se divise en la construction et la manipulation d'un modèle. On peut même dès lors se demander si la question de la nature du modèle n'est pas secondaire au regard des processus même de construction et de manipulation.

Mais si on pose la question de la modélisation, on pose forcément la question du point de vue. La notion d'objectivité est en soi plus spéculative que réaliste. Pour Watzlawick, ce qui existe n'est qu'un ensemble de différentes versions de la réalité, parfois contradictoires, qui sont des effets de la communication (Watzlawick 84). C'est donc pour lui la communication, avant tout autre processus, qui crée ce que nous appelons la réalité qui est déjà une représentation subjective du réel. Remarquons enfin que pour Morrisson et Morgan, le modèle est un médiateur entre une certaine réalité et la théorie (Morgan 99). Donc, construire un modèle revient alors à construire une représentation subjective qui fait le lien entre nos réalités et une autre réalité plus abstraite que nous appelons théorie. La modélisation est alors la construction d'un objet extérieur sur lequel on essaie d'accrocher de manière cohérente des représentations qui mettent en relation des connaissances et une représentation implicite d'un système donné.

Enfin, nous avons rapidement abordé la question de la manipulation du modèle. Celle-ci est l'exécution du modèle, donc sa mise en œuvre dans une simulation. Il s'agit alors de projeter l'état du modèle, donc l'information qu'il porte, dans différents états qui se suivent temporellement. On parle de calculer le modèle et d'en observer l'évolution. On peut cependant se demander si la manipulation d'un modèle n'implique pas une expérimentation interactive de celui-ci. Nous tentons dans la suite de revenir sur cette question (voir section 2.4.1).

³ A la différence de ce que l'on pourrait laisser entendre ici, nombreux sont ceux qui pensent que l'on ne peut pas encore statuer définitivement sur la nature des structures mnésiques.

Pour conclure, il est possible de se poser beaucoup de questions sur ce qu’est un modèle. Seulement, l’essentiel pour nous n’est pas d’y répondre, mais de permettre à des thématiciens ou des modélisateurs d’exprimer leur propre conception d’un système au travers d’une représentation de ce système afin d’exercer une activité que nous ne sommes pas capables de bien définir. C’est là que se pose tout notre questionnement.

2.3.2 Du modèle thématique au modèle computationnel

S’il n’est pas de notre ressort, en tant qu’informaticien, de qualifier la nature du modèle en tant qu’objet épistémologique, il n’est pas non plus de notre responsabilité de qualifier les représentations que se construisent les experts. Cela implique cependant, dans le cadre de la modélisation informatique, de pouvoir garantir que le modèle implémenté est le plus proche possible du modèle conceptuel que l’expert a à l’esprit. Pour certains, il faut ainsi que tout utilisateur puisse étudier un modèle de la littérature indépendamment de la plate-forme, ce qui impose une meilleure séparation des composants du modèle et de ceux de la plate-forme (Ginot 02). Revenons par conséquent ici sur la question du modèle computationnel. Dans la suite des méthodologies de Fishwick (Fishwick 94) et de Gilbert et Troitzch (Gilbert 99), Drogoul identifie dans le processus de conception d’une simulation les trois rôles suivants : le thématicien, le modélisateur et l’informaticien (Drogoul 03b). Remarquons que Drogoul a articulé son travail dans le cadre de la simulation multi-agents, cadre dans lequel la traduction du modèle thématique n’est pas toujours un problème simple, et le travail de l’informaticien prend alors tout son sens.

Le thématicien définit l’intention du processus de simulation et manipule trois types d’information : les théories et les hypothèses, les observations (les données observées) et les questions.

Le modélisateur traduit le modèle thématique («domain model») en un modèle plus formel afin d’en supprimer les ambiguïtés et de spécifier le modèle conceptuel («design model»). Le modélisateur a donc pour fonction de mettre en rapport l’information du thématicien et les contraintes inhérentes à l’implémentation choisie et que l’informaticien impose. C’est également le modélisateur qui formalise les questions liées aux niveaux de granularité du système, de ses interactions et de ses limites.

L’informaticien doit ensuite collaborer avec le thématicien au moyen d’un modèle opérationnel («operational model») et du modèle conceptuel du modélisateur afin de construire le modèle computationnel («Computational System») en tant que tel. L’informaticien aborde et introduit les questions liées à la gestion du temps (horloge/événements, synchrone/asynchrone), à la gestion de l’espace et des comportements.

Nous avons vu précédemment que les systèmes que nous étudions sont fortement hétérogènes et complexes et que les modèles conceptuels qui en sont des représentations peuvent être tout aussi complexes. Par conséquent, on peut se poser un certain nombre de

questions sur le modèle computationnel et sa construction. 1° Est-on capable de garantir que le modèle computationnel est équivalent au modèle conceptuel? Et si oui, comment le vérifie-t-on? 2° Construire un modèle computationnel, c'est décrire le modèle conceptuel dans un algorithme exécutable qui contient tous les détails qui permettent de le mettre en œuvre dans un simulateur : description du temps, de l'espace, de l'utilisation d'agents éventuellement, etc. Peut-on décrire ces informations au niveau conceptuel? Et sinon, quelles hypothèses doit-on rajouter au niveau du modèle computationnel? 3° Enfin, le thématicien peut-il encore apprendre sur son modèle thématique en mettant en œuvre dans des simulations le modèle computationnel? Ce dernier n'introduit-il pas des biais? Pour le moment, il n'y a pas de réponse générale à ces questions et il est nécessaire pour chaque modèle d'effectuer les analyses nécessaires pour y répondre.

2.3.3 Cadre théorique et approches de modélisation

Nous avons détaillé la question du modèle et introduit les notions de modèles thématique, conceptuel et computationnel. Il nous faut maintenant nous intéresser plus particulièrement aux types de modèle que nous voulons étudier et mettre en œuvre. Nous proposons de construire cette présentation autour des paradigmes de modélisation classique et centrée individu. En effet, les approches de modélisation sont partagées par l'axe épistémologique classique qui distingue la vision holiste de la vision réductionniste. On parle alors de modélisations "top-down" et "bottom-up", formulations qui dépeignent bien le processus mental qui permet de passer respectivement de la théorie aux données, ou des données à la théorie. L'opposition holisme / réductionnisme est bien évidemment abstraite, mais elle structure cependant depuis longtemps notre manière de penser. En particulier, elle caractérise les deux paradigmes majeurs qui nous intéressent, le paradigme classique basé sur des modèles agrégées, dont l'approche est holiste et les échelles de description et d'observation sont au niveau de la population, et le paradigme individu-centré ou *IBM* (*Individual-Based Modelling*) (DeAngelis 94), qui considère explicitement les composantes individuelles du système et dont l'échelle de description est celle du microscopique alors que l'échelle d'observation est au niveau macroscopique.

2.3.3 - A Modélisation agrégée

La modélisation dite agrégée s'appuie sur le point de vue global et l'idée est d'exprimer certaines grandeurs importantes du système et leurs relations au sein d'équations mathématiques. Les modèles sont alors des systèmes d'équations différentielles (*ED*) ou d'équations aux dérivées partielles (*EDP*) que l'on sait résoudre analytiquement ou numériquement, avec un coût de calcul très faible et une très bonne précision. Les deux grandes forces de cette approche sont sa maniabilité et son formalisme qui permet aux thématiciens de chercher dans le cadre formel des équations mathématiques des lois univer-

selles décrivant leur réalité. Ainsi, l’approche agrégée s’intègre naturellement dans la démarche classique de modélisation et dans des domaines comme l’écologie et l’économie. Entrant en résonnance avec le point de vue holiste classique des thématiciens, pour lesquels le niveau macroscopique (populations, groupes) est le premier niveau d’observation et de description, l’approche a permis de développer de nombreux modèles qui structurent encore aujourd’hui notre perception et notre appréhension des écosystèmes et des systèmes économiques. Le modèle de Lotka-Volterra (Volterra 31) est un exemple important de cette approche. Celui-ci abstrait complètement un système de proies-prédateurs, il est alors indépendant des espèces qu’il peut représenter tout en exprimant un phénomène structurel fondamental qui est la relation d’interdépendance trophique de deux espèces. Ce modèle est d’ailleurs la source d’un très grand nombre de modèles qui en sont dérivés (Shin 00).

Cependant, ce qui fait la force de l’approche fait aussi sa faiblesse et la modélisation agrégée est souvent dénoncée comme trop éloignée du système réel, les variables et paramètres que l’on manipule n’ayant pas toujours de pertinence thématique. Pour Amblard (Amblard 03) par exemple, une approche macroscopique est limitée du point de vue de : 1° l’étanchéité du niveau d’analyse, les variables d’un modèle global sont toutes au même niveau d’analyse (Ferber, 1995) ; 2° la complexité et le manque de réalisme des paramètres utilisés ; 3° la difficulté à modéliser les comportements ; et enfin, 4° ce que Amblard appelle la carence qualitative et qui signifie qu’il est très difficile avec des paramètres quantitatifs de prendre en compte les observations des thématiciens issues directement du terrain. Comme le fait remarquer Weber dans (Weber 95), le modèle de Shaeffer (1954) par exemple permet d’appréhender l’exploitation d’une ressource dans le cas d’un accès libre à cette ressource mais induit des représentations fausses. D’un intérêt didactique certain, les hypothèses sont cependant irréalistes – environnement constant, stock à “l’équilibre”, prix constants, etc. – et la dynamique même du modèle conduit à des représentations erronées du système.

Remarquons enfin que l’approche agrégée est à la base d’autres approches de modélisation qui se sont construites dans les domaines de l’écologie, de l’économie ou du social. L’écologie reprit par exemple dans un premier temps les outils mathématiques de la systémique, outils avec lesquels la description d’un système se fait en termes de compartiments reliés entre eux par des flux et des contrôles (Costanza 93). Cette approche prit d’autant plus d’importance à partir des années 50 dans les sciences de la vie et les sciences humaines que l’informatique permit alors de résoudre numériquement certaines équations qui ne pouvaient être abordées analytiquement. On peut alors également penser aux logiciels de modélisation Ecopath, Ecosim et Ecospace (Pauly 00) qui sont utilisés pour étudier l’impact des pêcheries sur les écosystèmes (Christensen 04).

Enfin, l’approche agrégée et la dynamique des systèmes sont également à l’origine de la dynamique des populations qui est spécifique à l’écologie et qui a fourni de nombreux modèles agrégés. La dynamique des populations est l’étude des changements marginaux et sur le long terme en nombres, poids individuels et en structures d’âge des individus dans une ou plusieurs populations et des processus biologiques et environnementaux influençant ces

changements. Ce qui caractérise la dynamique des populations est qu'elle s'articule autour de la notion de stabilité des populations ou de populations en équilibre, même si des modèles de populations dites cycliques existent, mais ce sont alors des scénarios de migration ou de fluctuations saisonnières qui sont étudiés et la notion de population en équilibre dans un environnement donné n'est pas remise en cause. La dynamique des populations se caractérise également par l'utilisation de systèmes à base d'équations ou de modèles de stocks et de flux dans lesquels le système est représenté par des compartiments et des flux d'énergies entre ces compartiments. Ce domaine permis alors la naissance de nombreux modèles de processus comme le modèle de croissance Malthusienne, le modèle de croissance de Verhulst qui suit une fonction logistique en réponse à la croissance exponentielle du modèle de Malthus, le modèle de Gompertz (ou loi de mortalité de Gompertz-Makeham).

En conclusion, nous constatons que cette approche structure les théories écologiques et économiques et de nombreux modèles qui servent aujourd'hui de référence permettent à beaucoup d'aborder la réalité des systèmes au travers de ces modèles. Le risque est ainsi de croire que le modèle et le réel se confondent, mais cela nous permet de penser que cette approche fonctionne cependant très bien en permettant de construire des modèles qui deviennent alors médiateurs entre la théorie et le réel. Il est par conséquent nécessaire pour nous de pouvoir envisager dans la construction d'un modèle complexe de permettre aux experts d'utiliser des EDP.

2.3.3 - B Modélisation centrée individu

L'approche centrée individu, ou modélisation individu-centrée, traduction de l'expression anglaise *Individual-Based Modeling* a donné l'acronyme IBM couramment utilisé, participe de la démarche de plus en plus fréquente chez les thématiciens de poser comme élément central l'individu, et avec lui ses comportements et ses interactions avec les autres éléments du système (individus et environnement). Approche réductionniste, le principe est de prendre un point de vue microscopique pour décrire le système, ce qui se traduit par un modèle des individus du système dans leur environnement. On cherche alors à faire émerger les propriétés du système à partir des éléments le constituant et de leurs interactions (Shin 00). L'approche IBM part ainsi de l'hypothèse que les dynamiques d'un écosystème peuvent être modélisées utilement par 1° une modélisation des caractéristiques de l'environnement du système qui affectent les individus, 2° une modélisation des traits des individus qui influencent le comportement des individus et comment ils interagissent entre eux et avec leur environnement, 3° une simulation de ces interactions (Grimm 05a). Cependant, alors que le niveau de description est microscopique, le niveau d'observation reste macroscopique. En effet, la notion d'émergence est centrale dans cette approche (Stockler 91) et les caractéristiques des niveaux supérieurs peuvent être déterminés comme des propriétés émergentes des niveaux inférieurs (Breckling 05). Donc, un modèle IBM est un modèle de comportements individuels qui est utile pour expliquer des phénomènes au niveau de la population dans un contexte particulier (ce contexte étant caractérisé par l'environnement biotique et abiotique, ce qui

inclut parfois l’état des individus) (Grimm 05a; Railsback 02).

L’approche IBM répond à la difficulté d’utiliser des modélisations agrégées classiques, voire l’impossibilité de formuler globalement l’évolution de certains systèmes. Elle permet alors de considérer des entités autonomes et leurs interactions dans leur unicité et leur localisation spatiale et temporelle. Plus généralement, l’approche IBM permet aux modélisateurs de s’intéresser plus précisément aux détails des systèmes qu’ils étudient et d’exprimer des hypothèses moins restrictives (Railsback 01b). Les modélisateurs et les thématiciens peuvent en effet naturellement décrire le niveau individuel qu’ils perçoivent dans le système réel. La modélisation IBM permet ainsi d’apporter des réponses en autorisant une description hétérogène, et chaque population, communauté, ou groupe peut avoir ses propres particularités ce qui est essentiel pour comprendre leur dynamique (Kirby 04). En d’autres termes, il est alors possible de décrire le système au travers de ses règles de fonctionnement local (Huston 88), là où la modélisation classique décrit uniquement le comportement global du système.

Cependant l’approche IBM pose un certain nombre de problèmes qui rendent l’étude et la manipulation des modèles difficiles. Ainsi, Amblard (Amblard 03) isole trois sources principales de complexité dans ce type de modèles : 1° ils sont souvent fortement stochastiques, la solution des générateurs aléatoires pour modéliser la variabilité individuelle est fréquente et peut-être préférable (Pitchford 01) ; 2° ils ont une grande expressivité et par conséquent il est tentant de vouloir rendre compte, notamment dans le modèle des individus, de trop de phénomènes qui peuvent obscurcir l’étude globale ultérieure ; et 3° le modèle de simulation centrée individu est souvent complexe par essence (interactions et agrégations des entités pour la production de comportements collectifs observables). Grimm et d’autres rejoignent cette dernière critique en suggérant que le frein principal à la construction de théories à partir de modèles IBM est le fait que les IBM sont, pour la plupart, des modèles complexes (Grimm 99b; Axtell 99). Il est alors difficile de les développer, de les communiquer et de les analyser. Or une des premières caractéristiques d’un système complexe est son irréductibilité. Par conséquent, la meilleure, voire la seule, manière de communiquer un IBM est le programme lui-même qui est l’unique description correcte et complète de l’IBM (Lorek 99), ce qui limite fortement leur étude. Plus encore, dans un IBM, les individus interagissent dans des organisations dynamiques complexes qui sont construites sur les relations entre comportements individuels et comportements collectifs qui miment bien souvent efficacement les comportements réels. L’ensemble des comportements possibles de l’IBM en simulation est alors infiniment plus difficile à caractériser que ses règles de fonctionnement.

L’approche IBM est au final puissante et problématique : puissante, elle permet d’étudier les changements organisationnels qui s’opèrent sur un système suivant des conditions et hypothèses locales précises (Grimm 02) ; problématique, il est difficile de définir les états du système global. Autant, on peut penser qu’un modèle IBM, parce qu’il est décrit au niveau microscopique, est davantage capable qu’un modèle agrégé par exemple de reproduire des dynamiques globales non linéaires, autant il est alors difficile de le prouver. En réalité, la

modélisation IBM répond avant tout à des besoins pragmatiques : la réalité des systèmes étudiés est hétérogène et les données observées issues des systèmes réels sont globales mais également souvent individuelles. En effet, les thématiciens qui travaillent directement sur le comportement des individus dans leur environnement peuvent alors trouver dans l'approche IBM un outil de modélisation naturelle. Donc, à cette souplesse qu'offre l'approche IBM s'oppose alors la nécessité de développer la théorie pour implémenter, communiquer et analyser les IBM (Grimm 99b).

Cela n'empêche pas cependant l'approche IBM d'être de plus en plus utilisée dans différents domaines d'application. L'écologie moderne en particulier a été à l'origine du développement de l'IBM en voulant prendre en compte la variabilité individuelle. L'étude de la variabilité individuelle est ainsi essentielle pour l'écologie et pour l'étude de la dynamique des populations (Huston 88; Huse 02; Tyler 94). De ce fait, l'approche IBM permet d'aborder plus de systèmes que la modélisation classique et procure une approche pour l'étude écologique multi-dimensionnelle (Breckling 02). L'approche IBM est d'ailleurs pour Grimm plus qu'un simple outil (*just another tool*) mais le moyen d'une nouvelle perspective, car les outils que l'on utilise fournissent l'information avec laquelle on construit ses perspectives. Pour Grimm, il y a ainsi la place pour un nouveau domaine d'application des IBM, l'Ecologie Virtuelle, qui a pour objet la comparaison d'ensemble de données avec d'autres obtenues par des écologistes virtuels (c'est-à-dire simulés). On pourra trouver plus de détails sur l'approche *Virtual Ecology* dans (Berger 99). Remarquons que l'économie a également trouvé dans l'approche IBM un moyen d'étudier des modèles micro-économiques, et en particulier d'étudier les liens entre composantes microscopiques et environnement économique macroscopique (Antona 98).

2.3.4 Vers un modèle opérationnel

Nous avons vu précédemment la question des approches de modélisation, nous nous intéressons maintenant à la problématique de l'implémentation de modèles thématiques ou conceptuels dans un langage exécutable sur un ordinateur. Cette implémentation en termes de modèle opérationnel se traduit en particulier par l'introduction d'hypothèses sur le temps et l'espace. De plus, il existe des méthodes particulières pour les approches agrégées et centrées individu que nous proposons également de présenter.

2.3.4 - A Discrétisation temporelle et spatiale

Construire un modèle opérationnel, c'est avant toute chose aborder la question de la discrétisation du modèle et de la description de l'espace et du temps du modèle. Les aspects spatiaux et temporels sont en effet fondamentaux dans la question de la modélisation et mettre en œuvre le modèle, c'est l'animer dans le temps et éventuellement dans l'espace. Cette discrétisation dépend du modèle conceptuel, c'est-à-dire de l'approche de modélisation,

et donc du point de vue. On distingue ainsi trois catégories de modèles : 1° les modèles à évolution continue, 2° les modèles à pas de temps discret et les modèles par événements discrets.

Dans les modèles à pas de temps discret, le temps est une variable entière⁴ et l’état du système est calculé pour chaque valeur de cette variable. La distribution des pas de temps n’est pas forcément homogène, même si en général, elle l’est. Et enfin l’essentiel est que l’on considère chaque changement d’état du système comme instantané.

Dans les modèles à évolution continue, le temps est associée à une variable réelle dont toute valeur représente un état du modèle (dans un intervalle de temps donné – le temps de la simulation). Il n’est cependant pas possible de résoudre numériquement des modèles à évolution continue sur un ordinateur, il est nécessaire pour cela de discrétiser l’environnement suivant l’espace et le temps.

Enfin, les simulations à événements discrets proposent de changer de point de vue et de s’intéresser à ce qui conduit à un changement d’état du système (Zeigler 76). On n’organise plus le temps *a priori*, mais ce sont les composantes actives du modèle qui savent quand elles ont besoin d’être activées. Il suffit alors d’ordonner temporellement ces activations avant de les exécuter. L’avantage est de ne ”perdre du temps” que quand cela est nécessaire, car l’ordonnanceur ”saute” d’événement en événement. De plus, on peut réaliser des simulations dont la représentation est continue ou discrète.

Dans tous les cas, le temps et l’espace sont par définition discrétisé dans un modèle opérationnel. Ce processus de discrétisation est alors un processus qui ajoute ou modifie des hypothèses du modèle. Nous avons ici en particulier souligné les différentes approches de discrétisation suivant la question temporelle. Remarquons que la discrétisation spatiale est tout aussi importante et qu’elle peut avoir un impact sur les résultats, comme le montre (Anneville 98).

2.3.4 - B Implémentation de modèles agrégés

Pour mettre en œuvre un modèle agrégé dans un ordinateur, et parce que c’est un modèle continu, il est nécessaire d’utiliser des méthodes numériques permettant d’approcher les valeurs du modèle continu. Ces méthodes numériques se basent sur le calcul différentiel, ou calcul infinitésimal, qui se base lui-même sur une variable de temps continu. Mais pour mettre en œuvre le modèle dans un système numérique, il est alors nécessaire de discrétiser le temps et il faut que les pas de temps soient aussi petits que possibles pour assurer la stabilité numérique.

⁴ elle peut également être un nombre rationnel par ailleurs, c’est simplement qu’on peut la ramener à une variable entière.

L'objectif des méthodes numériques est alors de minimiser l'erreur entre la valeur approchée et la valeur du modèle continu. Cependant, plus on cherche à gagner en précision, plus on accroît le temps de calcul, et il est nécessaire d'établir un équilibre entre la précision et un temps de calcul raisonnable. Les principales méthodes que l'on peut utiliser sont : Euler explicite, Euler implicite et Runge-Kutta.

La méthode d'Euler explicite est la méthode de base que permet d'approcher une équation différentielle ordinaire ou une équation aux dérivées partielles en calculant son développement limité au voisinage de t . On calcule alors à nouveau une approximation de la solution en $t + 1$ et ainsi de suite. Il est possible d'augmenter le degré de la méthode d'Euler en augmentant l'ordre du développement, ce qui diminue l'erreur, mais augmente le temps de calcul. On voit tout de suite que le problème de cette méthode est que plus la dérivée seconde de la solution est grande, plus on s'éloigne vite de la solution et l'erreur est grande. On peut alors également adapter le pas d'intégration en fonction de l'erreur commise et utiliser une méthode de contrôle du pas. La différence maintenant avec les méthodes implicites est que les méthodes explicites calculent l'état du système au pas de temps suivant en fonction de l'état du système à l'instant courant.

Les méthodes implicites consistent à résoudre une équation basée sur l'état du système au pas de temps courant et sur celui au pas de temps suivant. En d'autres termes, les méthodes implicites calculent le développement limité en t en fonction de $t + 1$. La méthode d'Euler implicite est en général plus stable et les erreurs de calcul s'y propagent moins vite. Mais, le coût de calcul est très supérieur à la méthode d'Euler explicite.

Enfin la méthode de Runge-Kutta repose sur le principe de l'itération, c'est-à-dire que l'on calcule en t une première approximation de la solution en $t + 1$ qui va nous servir à calculer une deuxième approximation plus précise et le rang de méthode correspond au nombre d'approximations que l'on effectue.

2.3.4 - C Implémentation de modèles centrés individu

Arrivée dans les années 50, une des premières formes de modélisation à se centrer sur l'individu, c'est-à-dire sur l'élément microscopique du système étudié, est la microsimulation (Orcutt 07). C'est une approche statistique dont le processus de calcul est markovien, l'objectif étant de prédire des distributions statistiques futures des individus d'une population sous certaines hypothèses. Le principe est de manipuler un échantillon d'individus d'une population et de faire évoluer cet échantillon dans le temps, c'est-à-dire de faire vieillir les individus, à l'aide de fonctions probabilistes de transition. L'originalité de l'approche est de considérer explicitement les individus, de les caractériser par un âge, ou une variable qui lui est liée, et d'associer à chaque individu un ensemble de variables et donc un état. Il suffit alors de déterminer la probabilité à chaque instant que l'individu change d'état et d'incrémenter également son âge. Cette approche est critiquée parce qu'elle ne permet pas de modéliser les

interactions entre individus, que l’évolution de chaque entité est stochastique et ne prend en compte que son état courant et sa probabilité de transition vers les états possibles suivants. Par conséquent, elle ne permet pas non plus de modéliser un individu de manière plus fine, avec des modèles cognitifs par exemple. Elle est toutefois utilisée afin d’étudier et d’envisager les impacts sur les individus de différentes politiques économiques ou sociales par exemple (Merz 91).

Un autre type de modèle opérationnel individu-centré est l’approche par automates cellulaires (Neumann 66), formalisée par Wolfram (Wolfram 83), qui est en réalité apparue avant la microsimulation, dans les années 40. C’est une approche maintenant classique qui a été longuement étudiée et dont le premier exemple est le ”Jeu de la vie” de Conway (Gardner 70). Un automate cellulaire est une grille régulière de cellules qui sont caractérisées par des états et des fonctions de changement d’état qui dépendent du voisinage de chaque cellule. Le temps est discrétisé en pas de temps et l’état global de l’automate est calculé localement par chaque cellule à chaque pas de temps et les changements d’états des cellules sont réalisés de manière synchrone. Epistémologiquement réductionniste, les automates cellulaires ont fortement marqués par leur simplicité de construction que contrastait la complexité des objets géométriques ou conceptuels qu’ils étaient capables de produire. A une époque où l’informatique était en train de naître, les systèmes auto-reproducteurs (ou auto-réplicatifs) de Von Neumann laissaient envisager la description de systèmes complexes, qui rappelaient les systèmes vivants, à partir d’un minimum de règles locales. Pourtant, les règles rigides de discrétisation de l’espace et le point de vue lui-même qui impose ce maillage d’états, ainsi que le synchronisme du système, contraignent fortement la modélisation. On pourra remarquer les travaux sur l’universalité des automates cellulaires et la quête de l’automate universel qui serait capable de simuler n’importe quel autre automate et la machine de Turing (Wolfram 84; Banks 70). Enfin, les automates cellulaires ont été étudiés comme des systèmes dynamiques, voire chaotiques, dont le comportement global peut être indécidable comme par exemple (Kari 92).

Enfin, nous abordons l’approche multi-agents. Les systèmes multi-agents (SMA) sont donc issus de l’approche individu-centrée, mais aussi de l’intelligence artificielle et de l’informatique distribuée. La première motivation des SMA est peut-être la volonté d’étudier et de reproduire les systèmes naturels qui, sans contrôle centralisé, mettent en œuvre des tâches collectives fort complexes dans des environnements dynamiques. Les travaux sur les colonies d’insectes (MANTA (Drogoul 92)) sont par exemple issus directement de cette inspiration. L’approche agent est donc une approche locale, individu-centrée, qui fait l’hypothèse qu’il est possible de faire accomplir des tâches complexes par un ensemble d’entités simples qui interagissent et coopèrent et, surtout, qui n’ont qu’une connaissance partielle de leur environnement ou de la dynamique globale du système dans lequel elles évoluent. Le système global est alors perçu comme émergent de l’activité des agents et de leurs interactions. L’approche agent permet alors d’aborder, de comprendre et d’expérimenter les mécanismes d’auto-organisation collective et d’émergence et plus largement d’aborder les problématiques de la complexité (Ferber 99). Elle permet ainsi de ne pas simplifier à l’extrême une réalité

complexe (Bousquet 99).

L'agent du point de vue computationnel est souvent associé au concept objet de la programmation orientée objet. L'approche objet offre effectivement le support naturel à l'écriture de systèmes multi-agents et permet de rendre les agents explicites grâce au mécanisme de l'encapsulation. Mais, l'agent est bien plus qu'un objet dans le sens où l'objet permet de décrire le "comment", l'objet actif le "comment" et le "quand", alors que l'agent définit le "comment", le "quand" et le "pourquoi" (Muller 02). Dans le comportement de l'agent se trouvent alors explicités les mécanismes calculant dynamiquement le "quand" et le "comment".

On peut trouver de nombreuses ontologies d'agents dans la littérature, comme par exemple, les agents intelligents, rationnels ou cognitifs (Wooldridge 95). Ces différentes catégories se basent le plus souvent sur la rationalité de l'agent, qui peut être simple comme dans les agents réactifs (Brooks 91) ou sophistiquée comme dans les agents cognitifs, qui peuvent être basés sur une architecture BDI (*Belief-Desire-Intention*) (Georgeff 87) par exemple. En effet, on caractérise souvent l'agent par la description de son état interne et ses capacités de décision, c'est-à-dire le modèle qui lui permet de construire son comportement. Ce modèle qui construit la décision⁵ de l'agent est d'ailleurs un domaine de recherche qui appréhende des notions aussi fondamentales que les représentations, les buts ou les intentions. Pour nous, l'agent est, avant toute chose, un modèle autonome et hétérogène, qui se manipule comme une boîte noire. Il peut donc en particulier être récurrent et être composé lui-même d'un SMA (Servat 98b) ou être contrôlé par un humain et permettre des simulations participatives (Schuler 93). Ensuite et plus généralement, nous définirons et évaluerons nos agents en fonction de trois "capacités" que l'on prête généralement à un agent : 1° l'autonomie, 2° l'adaptation et 3° l'apprentissage.

Nous avons pu voir par exemple que l'approche IBM est elle-même fondamentalement bivalente et on peut distinguer dans la construction de modèles IBM une approche *top-down* et une approche *bottom-up* (Conte 92) : 1° les phénomènes microscopiques individuels sont dérivés de situations macroscopiques ; 2° les hypothèses sont posées au niveau individuel et le comportement collectif peut alors être comparé aux phénomènes macroscopiques observés pour évaluer les hypothèses utilisées au niveau microscopique (Epstein 96).

2.3.5 Multi-modélisation et couplage

Nous avons abordé la question du modèle, puis distingué les problématiques conceptuelles d'une part, en présentant les paradigmes de modélisation classique et IBM, et les problématiques d'implémentation d'autre part, en exhibant les techniques principales d'implémentation basées sur ces paradigmes de modélisation. Nous constatons cependant

⁵ La décision prend place dans la boucle de contrôle classique "perception-décision-action" de l'agent.

que la dichotomie dans les faits entre les différentes approches n’est pas forcément bénéfique à la qualité de la modélisation. Comme nous l’avons vu, la modélisation IBM est appropriée pour des descriptions qui requièrent un haut degré de localisation, alors que la modélisation agrégée est plus adaptée à des systèmes qui peuvent être modélisés globalement, mais le vrai enjeu est de pouvoir associer les outils de modélisation et les résultats, et la comparaison entre les approches de modélisation peut permettre de mieux comprendre les approximations et les imperfections de chacune des approches, donc des modèles particuliers (Parunak 98). Ainsi les études qui cherchent à intégrer les différents moyens de modélisation, dont font partie les approches de modélisation, se multiplient (Duboz 04b). On peut constater alors que les différents modèles, basés sur différentes approches avec chacun leur point de vue et leur échelle de description, permettent d’enrichir l’expérience de modélisation et d’avoir une meilleure compréhension des résultats de simulation (Duboz 03; Edwards 04; Fahse 98). La question des moyens à utiliser pour aborder les systèmes complexes ne se pose plus alors seulement par rapport à une approche de modélisation donnée. En effet, construire une représentation multi-formes d’un système hétérogène pousse de plus en plus de travaux vers des solutions mixtes (Villa 01) et nous constatons que par définition il est difficile de limiter un modèle complexe dans un cadre uniforme de conception.

Au final, nous constatons, avec d’autres (Duboz 01), que l’étude d’un système complexe peut justifier d’utiliser plusieurs approches de modélisation. La multi-modélisation permet à des experts de différents domaines – l’écologie, l’économie et les sciences sociales par exemple – de construire un modèle avec des matériaux théoriques qui n’ont pas toujours été faits pour co-exister. Or, l’hétérogénéité même des systèmes que l’on étudie oblige à appréhender le modèle comme un système complexe et, dans la pratique, les approches de modélisation sont souvent combinées. Il est alors important de permettre aux différents experts d’appréhender la modélisation dans un cadre ouvert avec leurs propres représentations et leurs propres références que l’on cherche ensuite à mettre en commun dans le modèle. C’est pourquoi il faut des outils pour décrire les différents points de vue et échelles du modèle, outils qui sont entre autres les approches de modélisation et les formalismes. Le formalisme DEVS a été proposé pour justement formuler dans un même modèle tous les autres formalismes (Zeigler 76). L’avantage de DEVS est de garantir la cohérence temporelle du modèle global en permettant de coupler des modèles hétérogènes et de les synchroniser (Quesnel 05). Seulement, l’hypothèse de départ de DEVS est de dire que pour faire communiquer des modèles hétérogènes, il faut qu’ils ”parlent la même langue”, d’où un formalisme extrêmement codifié et un cadre rigide de modélisation. Pour nous, l’intérêt principal de la multi-modélisation est d’une part de construire des modèles individuellement simples, pouvant être validés de manière incrémentale, ce qui est le cas pour la modélisation IBM (Epstein 96) ; d’autre part, elle laisse aux modélisateurs le choix du type de modèle le plus adapté en fonction du questionnement et des données disponibles, en d’autres termes, le type de modèle le plus apte à rendre compte des dynamiques identifiées.

2.4 Le processus de simulation

*“La notion de modèle en tant que représentation du réel repose sur deux métaphores, l’une artistique et l’autre juridique. La métaphore juridique de délégation suggère la notion de remplacement : le modèle tient lieu de réalité. La métaphore artistique de réalisation suggère la notion de présence : le modèle est une réalité. L’expérimentation *in virtuo* d’un modèle numérique lui assure une véritable présence et ouvre ainsi de nouveaux champs d’exploration, d’investigation et de compréhension du réel.” – (Tisseau 01a)*

2.4.1 Qu’est-ce que la simulation ?

Au sens large, la simulation inclut le processus de modélisation et c’est pour Fishwick la succession des étapes de conception du modèle, de son exécution et de l’analyse de son exécution (Fishwick 94). C’est alors une méthode consistant à créer un artefact, le modèle, et à en observer le comportement quand il est soumis à des variations, en particulier temporelles. Donc, dans un sens restreint, la simulation est la mise en œuvre ou le calcul d’un modèle, c’est-à-dire la construction de sa dynamique, dans un système informatique. Cela consiste à faire évoluer le modèle dans le temps. La simulation regroupe donc un certain nombre d’activités et mettre en œuvre une simulation implique des activités d’expérimentation du modèle, de production et d’analyse des résultats, de vérification et de validation du modèle de simulation qui, pour Kleijnen, font de la simulation un art autant qu’une science (Kleijnen 95).

Toutefois, la simulation est centrée sur l’expérimentation du modèle. Expérimenter le modèle, c’est interagir avec lui afin de vérifier que le modèle est cohérent au regard des questions de modélisation, des données réelles et du comportement théorique que le modèle devrait avoir. Cette démarche expérimentale permet surtout d’observer le comportement d’un système, dont les règles de fonctionnement ne permettent pas de déduire le comportement global. Les expérimentateurs peuvent alors appliquer une démarche scientifique sur un objet dont ils connaissent toutes les règles de fonctionnement comme s’il s’agissait d’un phénomène naturel, ce qui peut permettre alors d’établir une connaissance plus compacte du fonctionnement du système étudié (Deffuant 03). Il s’agit donc d’induire les comportements et les mécanismes du système réel à partir de ceux du modèle.

Cependant, la phase d’expérimentation d’un modèle se traduit en réalité par différentes activités qui requiert diverses compétences et responsabilités. En effet, le thématicien qui expérimente un modèle étudie en réalité le modèle conceptuel qu’il se représente au moyen du modèle computationnel. L’informaticien (qui est ici non thématicien) qui expérimente un modèle cherche à valider son modèle computationnel par rapport au modèle conceptuel. Plus encore, le thématicien met en œuvre trois activités différentes lors de l’étude et l’expérimentation de son modèle : 1° la validation de la représentation qu’il s’est construite du système, activité qui implique construction et modification du modèle ; 2° la recherche des valeurs des paramètres, le thématicien ne cherche plus à valider la structure de son modèle, mais à dimensionner les différents processus et paramètres ; 3° la production de résultats.

Remarquons alors que la place du modèle évolue au cours de ces différentes activités et, alors que pour la première le modèle est en construction, pour la dernière celui-ci doit au contraire être "figé" afin de pouvoir être exécuté un certain nombre de fois suivant différentes hypothèses et surtout être analyser au travers des résultats de simulation obtenus.

Ainsi, une fois le modèle conceptuel construit, il y a plusieurs activités différentes dans le processus de simulation, qui requièrent des responsabilités différentes. De ce fait, le processus de simulation n'est pas trivial et il est malheureusement encore peu formalisé. Il est par conséquent nécessaire de davantage rationaliser et de faciliter ces différentes activités. Ceci est d'autant plus vrai que le modèle est complexe et que de ce fait les jeux de paramètres sont importants, les processus à valider sont multiples et hétérogènes, les données produites sont volumineuses et diverses, etc.

2.4.2 Expérimentation *in virtuo* : vers un laboratoire virtuel ?

Le développement de l'informatique et particulièrement de la simulation multi-agents et de la réalité virtuelle nous offre de nouvelles perspectives sur les possibilités de percevoir et d'interagir avec nos systèmes informatiques et en l'occurrence nos modèles. Ces nouvelles possibilités d'expérimentation à l'aide de simulations numériques permettent d'envisager un nouveau type d'investigation : l'expérimentation *in virtuo* (Tisseau 01a). L'expérimentation *in virtuo*, c'est une expérimentation conduite dans un environnement virtuel de modèles numériques auquel l'homme participe. Expérimenter son modèle, c'est alors le construire dynamiquement en le perturbant localement ou en modifiant telle ou telle composante du modèle. Le principe est de s'appuyer sur la visualisation dynamique du modèle pour observer le déroulement de la simulation et placer le thématicien au cœur d'un laboratoire virtuel dans lequel il aurait à disposition une sorte de paillasse virtuelle pour étudier, modifier localement ou globalement son modèle, arrêter ou relancer la simulation. Cette approche se développe en particulier dans le domaine de la biologie (Desmeulles 06a), ce qui répond par ailleurs à un besoin des modélisateurs (Endy 01).

Or, travailler à la construction de modèles de systèmes complexes requiert, de par la diversité des composants, des interactions et des niveaux d'abstraction à assembler, de pouvoir rationaliser cette construction. Il n'existe cependant pas de théorie permettant de formaliser la complexité de ces systèmes, et aucune preuve formelle permettant de vérifier la validité de ce que l'on construit. De ce fait, il est nécessaire de s'appuyer sur une démarche expérimentale qui permette de s'approprier le modèle, de le construire éventuellement par tâtonnements, par essai-erreur par exemple, et par construction incrémentale du modèle qui permet alors de valider *a posteriori* le modèle. C'est dans cette perspective expérimentale qui permet d'observer le modèle en exécution, mais également de l'animer dynamiquement que l'expérimentation des modèles prend un sens tout particulier.

Nous avons par ailleurs vu que la construction même du modèle permettait l'acquisition

de connaissances sur le modèle et sur le système étudié et, qu'au delà du modèle, c'était sa construction qui était féconde. Dans une approche *in virtuo* qui augmente les interactions entre le système et le modélisateur, on peut penser que cette activité de modélisation est favorisée, parce que l'expérimentation *in virtuo* d'un modèle numérique lui assure une véritable présence et implique un véritable vécu de l'expérimentation que ne suggère pas la simple analyse de résultats numériques (Tisseau 01a). On peut alors se poser la question de la meilleure méthode ainsi que du meilleur l'outil pour améliorer et développer la manipulation et l'étude de ce type de modèles complexes dans une approche *in virtuo*. Plus encore, dans une approche multi-experts, qui nécessite donc une collaboration entre ces experts, la mise en commun des représentations de chacun est une problématique à laquelle il s'agit de répondre.

L'expérimentation *in virtuo* et le concept de laboratoire virtuel sont séduisants et nous sommes persuadés qu'il faut aller plus loin dans ces approches qui sont de plus en plus envisagées ou abordées (Tisseau 01a; Ramat 03; Amblard 03). Seulement, l'approche nous pousse à réfléchir davantage sur les activités de modélisation et de simulation de systèmes complexes. Il y a tout d'abord la question de l'observation. Le principe de l'approche est de donner accès à l'utilisateur à une véritable paillasse virtuelle, contenant outils virtuels de calcul, de mesure et de visualisation. Mais, cela implique de savoir ce qu'il faut regarder et il n'est pas évident d'appréhender *a priori* les multiples points de vue pertinents pour observer les évolutions structurelles et organisationnelles du système. Dans les modèles IBM en particulier, les individus sont dans des organisations complexes qui peuvent évoluer. Le cadre d'expérimentation doit alors permettre d'appréhender la richesse de comportement du modèle ou de comparer des dynamiques d'un modèle avec d'autres modèles ou avec le système réel. Face à la multitude d'échelles et de points de vue dans la description, la question de l'observation n'est pas trivial et il est nécessaire de permettre de rationaliser le rapport de l'utilisateur avec le modèle global.

Ensuite, que signifie dans le cadre de notre problématique de perturber un modèle en cours d'exécution ? L'expérimentation *in virtuo*, c'est interagir avec le modèle afin de le modifier localement ou d'introduire de nouvelles entités. Dans le cas de simulations de réactions chimiques, le principe fonctionne bien et la construction d'une paillasse virtuelle peut remplacer la paillasse réelle (Rodin 04). Mais, est-on capable, pour tout modèle complexe, de modifier celui-ci de manière interactive tout en gardant le contrôle sur le modèle ? On voit bien alors que pour la construction d'un modèle, l'interaction avec celui-ci dans un cadre virtuel qui permette à l'utilisateur de mieux vivre son activité de modélisation est une solution pertinente. Il faut cependant alors séparer l'activité de modélisation, c'est-à-dire de construction d'un modèle, et l'activité de simulation pour laquelle l'ensemble du modèle doit être maîtrisé. En effet, l'outil de simulation doit permettre d'associer tout résultat à un modèle précis et à un jeu de paramètres donné, ce qui implique que le processus de simulation soit verrouillé. C'est uniquement ainsi que l'on peut analyser sereinement les résultats de simulations et induire ou déduire précisément des connaissances sur un modèle donné et donc sur le système que l'on étudie.

En conclusion, l’expérimentation d’un modèle ne peut pas être uniquement une affaire d’interaction et d’expérimentation à la volée. Il est nécessaire que cette démarche expérimentale entre dans un cadre rigoureux qui rationalise et organise la modélisation, l’expérimentation du modèle et les expérimentations en général.

2.4.3 La simulation : un processus complexe ?

2.4.3 - A L’exploration des modèles

Plus un modèle est complexe, plus il peut se révéler difficile à étudier en simulation et n’est pas forcément alors plus utile qu’un modèle moins complexe (Lindenschmidt 06). Il existe, pour aider les modélisateurs dans le processus de simulation, un certain nombre d’analyses qui permettent de mieux comprendre ses modèles en les explorant par simulations (Ginot 06c)⁶(Santner 03; Ginot 06b; Sacks 89). Les analyses présentées par Ginot et Monod sont nombreuses : la cohérence numérique, les analyses de sensibilité et d’incertitude, les analyses de sensibilité aux conditions initiales et à la propagation d’erreur, les analyses de stabilité, l’analyse qualitative du comportement, la recherche des comportements ”robustes” et enfin la ”méta” modélisation.

Les analyses les plus classiques sont les trois premières. Le problème de la cohérence numérique est lié aux biais computationnels sur lesquels nous avons insisté. Il s’agit avant toute chose de garantir que la granularité imposée par l’implémentation du modèle n’influence les résultats que de manière négligeable ou du moins maîtrisée. Il s’agit donc également de mieux connaître les problématiques liées à l’implémentation des modèles. Les analyses de sensibilité doivent permettre de déterminer l’influence des paramètres sur les sorties du modèle. Elles sont extrêmement utiles pour faciliter l’acquisition de connaissances sur la dynamique générale du modèle que l’on a construit.

Ces analyses sont cependant à mettre en œuvre dans un plan de simulation et même éventuellement d’expérience, ce qui implique d’organiser le processus de simulation. Plus un modèle est complexe, plus en général le nombre de paramètres est élevé. Or, si l’on souhaite mettre en œuvre les analyses dont on a parlé à l’instant, si l’on souhaite explorer les dynamiques du modèle suivant la valeur de ses différents paramètres, il faut pouvoir explorer les espaces possibles de paramètres de manière efficace. En effet, soit on a une idée précise de la valeur de tel paramètre, parce qu’il représente une réalité thématique que l’on sait qualifier, soit on a besoin de tester l’influence d’un certain nombre de valeurs de ce paramètre sur le système afin de choisir la ”meilleure”. Idéalement, on teste toutes les valeurs possibles de tous les paramètres que l’on ne connaît pas⁷. Concrètement, cela représente

⁶ Notons que nous empruntons bien évidemment le titre de cette sous-section à l’article de Ginot et Monod.

⁷ A moins bien évidemment que l’on ait une méthode analytique qui permette de connaître la dynamique

pour un modèle constitué de dix paramètres pouvant prendre chacun dix valeurs un nombre total de 10^{10} simulations. Si chaque simulation mettait seulement une heure à s'exécuter, il faudrait alors 1157407 années pour effectuer toutes les simulations. Le plan de simulation est donc un premier pas dans la formalisation des simulations. Toutes les simulations doivent alors suivre la même démarche et méthodologie et les paramètres doivent être clairement établis. La notion de plan d'expérience propose d'aller bien plus loin dans la formalisation de la planification. C'est une méthode statistique qui permet de rationaliser l'exploration de l'espace des paramètres. Il existe pour cela plusieurs types de plan d'expérience, nous citons ici les plus fréquemment rencontrés : le plan non factoriel (en étoile), le plan factoriel complet, le plan factoriel fractionnaire, les plans composites centrés et les plans Box et Behnken. Pour plus d'informations, le lecteur pourra se documenter avec (Dagnelie 03) ou encore (Box 05).

2.4.3 - B Le modèle dans le processus de simulation

Nous proposons ici de replacer le travail sur un modèle dans le cadre d'un processus de simulation qui intègre de nombreuses actions de modélisation. En effet, nous remarquons qu'il est rare de ne travailler que sur un modèle, en particulier quand celui-ci est complexe. Et la complexité du processus de simulation tient également dans la gestion de ces différents modèles. En effet, analyser et expérimenter un modèle, c'est multiplier les expériences, éventuellement dans un processus incrémental de modélisation, ce qui se traduit par un jeu de modèles qu'il est nécessaire d'explorer.

La multiplication des modèles se justifie par la nécessité de multiplier les analyses et les vérifications pour limiter les biais liés à un modèle ou les erreurs d'interprétation. Pour Uchmanski (Uchmanski 99), on apprend d'ailleurs plus d'une suite de modèles que d'un seul unique modèle (Grimm 99a). Cette suite de modèles peut alors permettre d'aborder un modèle complexe dans une démarche de modélisation incrémentale. Etudier des modélisations simplifiées d'un modèle complexe permet alors d'identifier les processus, variables et paramètres qui jouent un rôle essentiel dans le modèle (Nes 05; Murray 01). Elle permet également d'expérimenter différentes approches de modélisation. Le changement de formalisme ou de technique de modélisation peut ainsi être un grand avantage (Grimm 99a; Duboz 03) pour la compréhension du phénomène que l'on modélise (Amblard 03; Amblard 01). Différents travaux trouvent un avantage à étudier une suite de modèles plutôt qu'un seul, comme (Salvanes 01) ou encore Wyszomirski et al. pour qui une suite de modèles leur permet de mettre en perspective leurs considérations et hypothèses et de prendre du recul par rapport à certains facteurs parasites (Wyszomirski 99). Pour Fulton et al., il est également préférable de travailler sur la comparaison et la validation de plusieurs modèles moins réalistes que sur un seul modèle "ultime" (Fulton 03).

Par conséquent, la quantité de données produites par tous ces modèles est rapidement

globale du système, mais nous avons vu qu'il n'en existait pas pour le type de modèles que l'on manipule.

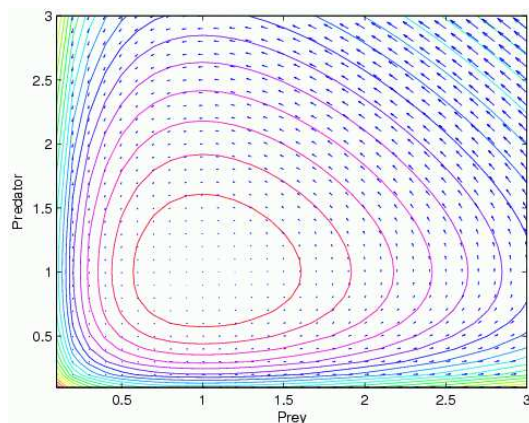


FIG. 2.1: Exemple de pattern : le pattern classique des systèmes proies-prédateurs de type Lotka-Volterra.

très élevée. Un modèle peut déjà, si l’on en explore longtemps les différents jeux de paramètres possibles, générer une grande quantité de données. Pour pouvoir mettre en rapport les données produites, pour pouvoir les analyser et les communiquer, il est alors nécessaire de connaître l’état précis du modèle, du jeu de paramètres et les hypothèses utilisées pour produire ces données. C’est le processus de simulation dans son ensemble qu’il s’agit alors de maîtriser, de rationaliser et de structurer en le mettant en rapport avec les actions de modélisation et les modèles utilisés.

2.4.4 Rationaliser la simulation : l’approche POM

Nous proposons maintenant d’étudier une méthodologie de modélisation qui permet de rationaliser la structure du modèle et la manière de le concevoir et ainsi le processus de modélisation lui-même. La modélisation orientée-pattern (*Pattern-Oriented Modelling – POM*) cherche à rendre la modélisation plus rigoureuse et compréhensive (Grimm 96; Grimm 05b). La démarche de modélisation doit être motivée par une (ou des) problématique(s) sur un objet d’étude clairement identifié : il y a des questionnements et donc des hypothèses sur cet objet que l’on caractérise par des traits remarquables ou *patterns*.

Un pattern est l’observation d’une structure non aléatoire émergente. C’est, pour Grimm, un indicateur des processus et des composants structurants du système étudié. Il contient des informations sur l’organisation interne du système et sur les mécanismes qui le font émerger (Grimm 05b). Ce sont par conséquent des structures de haut-niveau qui traduisent une organisation non stochastique de processus sous-jacents en interaction. Ce sont des dynamiques multi-échelles et multi-formes dans les données qui expriment une réalité structurelle et dynamique d’un système. La figure 2.1 présente un premier exemple

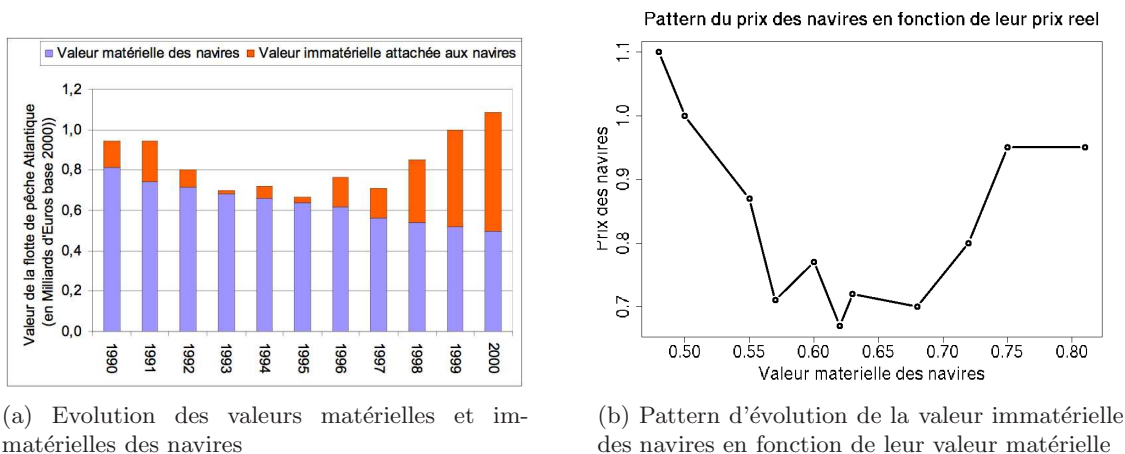


FIG. 2.2: Exemple de pattern identifié dans les données caractérisant l'économie de la pêche du GDG.

de pattern : on y voit les projections, suivant différentes conditions, dans l'espace des phases du nombre d'individus d'une population de prédateurs en fonction du nombre d'individus d'une population de proies. Le pattern "proie-prédateur" est un schéma d'évolution de deux populations théoriques bien connu et la projection du nombre de proies et du nombre de prédateurs dans l'espace des phases permet de faire resurgir ici une interdépendance entre les deux populations. Ce pattern est classique et issu d'un système de type Lotka-Volterra. Les figures 2.2b et 2.2a présentent un autre exemple de pattern. On voit dans la figure 2.2a, le pattern de l'évolution au cours du temps du prix réel moyen des navires de pêche du GDG en fonction de la valeur matérielle moyenne de ces navires. La valeur immatérielle d'un navire est la valeur que prend le navire dans le marché des navires de pêche. Elle représente alors aussi bien la valeur du matériel, que celle plus abstraite de la place de ce navire dans l'économie. Elle est donc l'expression d'un certain nombre de processus qui structurent cette économie. De manière plus générale, les patterns s'inscrivent dans des structures spatio-temporelles : évolution de la diversité spécifique, dynamique des structures démographiques.

Il est très difficile de caractériser formellement les patterns, tout comme il est difficile de caractériser un phénomène émergent. Remarquons que la forme géométrique du pattern du système de type Lotka-Volterra est saisissante et qu'elle pourrait nous faire penser que les patterns sont très géométriques. Cependant, il est issu d'un système théorique et abstrait, dont la formulation se prête parfaitement bien à des dynamiques extrêmement régulières, et il est normal d'avoir un tel résultat. En général, les patterns observés dans des systèmes réels sont bruités et particuliers tout comme celui de l'évolution du prix des navires. Ils sont influencés par de nombreux phénomènes, voire micro-phénomènes, mais ce sont les principaux, ceux qui expliquent les grandes tendances de la dynamique du pattern, qui font l'objet de la modélisation. Donc, caractériser et formaliser un pattern, c'est caractériser et

formaliser l’influence sur les données des phénomènes que l’on cherche à modéliser. Il n’est donc pas possible de les formaliser *a priori*. C’est aux thématiciens de reconnaître ces traits remarquables comme l’expression d’une dynamique particulière. La question de modélisation est alors en quelque sorte reformulée dans les patterns qui la structurent.

Le principe de la *POM* est de partir des patterns observés pour établir un modèle structurel du système. Le modèle est ainsi directement attaché à l’organisation interne du système réel (Grimm 05a). La démarche se décompose alors en 1° l’observation de patterns caractérisant le système et ses dynamiques et 2° le choix des variables et processus qui peuvent expliquer l’émergence de ces patterns (Grimm 05b). La modélisation devient orientée donnée et les mécanismes sélectionnés le sont pour leur capacité à influencer un ou des patterns. Les patterns sont alors, non seulement au centre de la modélisation, mais également de l’observation et de l’analyse du système. Tout comme on a cherché à caractériser les systèmes chaotiques que l’on n’arrivait pas à décrire dans le détail par des points fixes ou des structures géométriques dans les dynamiques de ces systèmes, caractériser la dynamique d’un système complexe par des patterns permet d’identifier et de qualifier l’essentiel de cette dynamique. Les patterns permettent alors de rationaliser la modélisation. Remarquons, enfin, que l’approche POM permet aux différents thématiciens construisant un modèle de s’entendre sur la dynamique globale du système en dehors de tout formalisme et de toute hypothèse particulière hormis celles concernant la question de modélisation.

2.5 Approche agent et plates-formes de simulation

Nous proposons dans cette section de revenir sur l’approche agent. Nous avons déjà présenté l’approche, mais nous n’avons pas abordé toutes les problématiques qui se posent dans le contexte de notre travail par rapport aux agents et à leur utilisation pour et dans la modélisation. En particulier, nous revenons sur la notion d’agent par rapport à celle d’individu des modèles IBM. L’objectif ici est de réinscrire l’agent comme un moyen computationnel pour implémenter un individu d’un IBM. Ensuite, se pose la question de l’environnement de l’agent. Nous identifions ainsi ce qui compose l’environnement de l’agent et les hypothèses de granularité qui se posent avec la mise en œuvre d’agents dans un environnement. Dans la suite, nous proposons un aperçu de quelques plates-formes agent de simulation, en exhibant leur démarche en termes de modélisation. Cela nous permet de mettre en perspective différents points de vue sur la modélisation. Dans la suite, nous revenons sur le problème de la multi-modélisation et sur le couplage dans une approche agent. Enfin, nous terminons en ouvrant notre éclairage sur l’approche agent pour la simulation avec le problème de la modélisation et de la simulation à l’aide d’agents. Les agents ne sont donc plus vus comme des supports computationnels à des modèles conceptuels, mais comme des solutions facilitant la modélisation et la simulation.

2.5.1 L'agent et l'individu

L'approche multi-agents est souvent identifiée à l'approche individu-centré, on peut citer l'exemple de (Kirby 04) mais de nombreux travaux font cet amalgame. L'approche agent est par définition une approche individu-centrée, donc l'assimilation des deux termes agent et individu est par essence naturel. L'approche agent est alors un moyen computationnel naturel à l'implémentation de modèles IBM. Seulement, dans une approche de modélisation mixte, il peut être nécessaire de distinguer les deux termes.

L'individu des IBM est évidemment un agent thématique, voire conceptuel, que Drogoul distingue de l'agent opérationnel ou computationnel des SMA (Drogoul 03b). On peut en effet distinguer ontologiquement et épistémologiquement les deux termes. Ontologiquement, l'individu et l'agent désignent tous deux une entité structurellement définie en opposition avec ce qui n'est pas elle et dynamiquement autonome. Seulement, l'individu fait davantage référence à une espèce spécifique. L'agent lui fait plutôt référence à la notion d'autonomie comportementale et aux capacités réflexives d'une entité, plutôt qu'à son existence. Epistémologiquement, l'individu représente l'entité atomique d'un système perçu avec un éclairage paradigmatique particulier : on est ici presque uniquement dans le descriptif du niveau d'abstraction et on ne dit rien du fonctionnement de l'individu. Le mot agent souligne lui, entre autre chose, l'architecture computationnelle de l'entité : on ne présuppose rien ici de la nature de l'agent dans le système réel, ce qui compte c'est le rôle d'agent – l'agentification – de cette entité.

De manière plus générale, dans un modèle IBM mis en œuvre à l'aide d'un SMA, il n'y a pas forcément de bijection entre l'ensemble des agents et celui des individus du modèle conceptuel, et encore moins entre les agents et les individus du système étudié. Par ailleurs, cette bijection n'est même pas immédiate entre les individus conceptuels et les individus du système étudié. Ainsi, par exemple, pour certains auteurs, on peut distinguer les modèles basés individu de ceux qui sont orientés individu (Uchmanski 96). Les premiers sont pour Uchmanski et Grimm forcément composés d'individus explicites. Les seconds, par contre, sont des modèles de processus physiologiques basés équation qui prennent place au niveau de l'individu (i.e. digestion, croissance). On peut penser également aux modèles basés sur la notion de super-individu (Scheffer 95). Le super-individu regroupe plusieurs individus et son comportement caractérise la dynamique de ces individus. Cela pose un certain nombre de questions et de problèmes, comme par exemple : 1° comment créer des super-individus ? On part généralement de modèles centrés individus pour construire des super-individus, ce qui pose le problème du regroupement des individus en des super-individus. 2° Comment décrire les interactions entre super-individus et avec l'environnement ? Ou encore 3° comment modéliser la mortalité d'un super-individu (Rose 93) ? Par exemple, si un individu ou un groupe d'individus (la moitié du super-individu par exemple) meurent, le super-individu doit-il mourir entièrement ? Enfin, 4° réciproquement, peut-on déduire le comportement d'un individu à partir du comportement du super-individu qui le contient ?

Pour conclure, on constate que, dans un IBM, l'agent computationnel n'a pas davantage

à représenter un individu thématique d’une population qu’autre chose. Au contraire, il est important de faire la différence entre les deux notions, car le système multi-agent est un moyen naturel pour implémenter et mettre en œuvre un IBM, mais ce n’est qu’un moyen. Et en tant que moyen, il peut entraîner ses propres biais qu’il est alors nécessaire de connaître et de maîtriser. De plus, dans un modèle complexe qui requiert alors de nombreux moyens de représentations, il est important de pouvoir aborder la solution basée agent de la manière la plus large possible, afin d’envisager des solutions permettant de mieux reproduire telle ou telle composante du système complexe, en affinant la solution agent utilisée.

2.5.2 L’agent et son environnement

Mettre en œuvre une simulation multi-agents nécessite de considérer les agents, mais aussi leur environnement. Ainsi, on peut se demander dans un premier ce qui caractérise cet environnement, ce qui nous amène à la notion d’*artefact*. Ensuite, la simulation multi-agents impose un certain nombre d’hypothèses et en particulier elle suppose une certaine granularité du système. Nous abordons alors ce problème de granularité qui s’exprime avant tout dans la discrétisation temporelle et dans la nécessité d’ordonnancer l’activation des différents agents.

2.5.2 - A Notion d’*artefact*

Pour Omicini et al., la question de la construction et de la manipulation d’outils par l’humain est aussi fondamentale que celle du langage (Omicini 06). Plus encore, elle est en relation avec le langage dans l’évolution des capacités cognitives des humains au cours de l’Histoire. Ainsi, le rapport de soi avec l’extérieur, les autres, passe par le langage et par les outils. Basés sur ce constat, de nombreux travaux étudient l’environnement, qui est de ce fait une problématique centrale dans l’approche agent et qui permet d’étudier les activités de collaboration (Susi 01). L’environnement définit ainsi les contraintes spatio-temporelles des agents et la manière dont les agents interagissent. On pourra alors caractériser l’environnement par ce qui le constitue, c’est-à-dire les agents, des objets informationnels et des *artefacts* qui sont les outils qui permettent et contraignent toute activité de collaboration entre agents (Omicini 06) (Molesini 06).

La notion d’*artefact* dans le cadre des SMA a été proposée et formalisée par Omicini et al., dans (Omicini 06), qui la décrivent comme permettant et conditionnant toute activité (intelligente) des agents (voir section 2.5.2). En tant qu’outils de médiation, les *artefacts* permettent et étendent les actions de l’agent, mais ils les conditionnent également de par leur propre limitation. Ce qu’il faut retenir surtout, c’est que les *artefacts* sont des objets passifs, qui n’ont pas de but mais qui permettent aux agents d’atteindre les leurs, et qu’ils fournissent : une interface d’utilisation (les opérations que l’agent peut invoquer), la procédure de manipulation dans le temps de l’*artefact*, et une description du service qui permet aux

agents de les sélectionner dans l'environnement en fonction du but qu'il cherche à atteindre. Enfin, Omicini et al. proposent plusieurs types d'*artefacts* : les *artefacts* individuels, sociaux et de ressource. Remarquons que les *artefacts* sociaux permettent alors de médier les interactions entre les agents et fournissent un service qui est, avant tout, destiné à atteindre un but du SMA. Nous nous appuyons ainsi sur cette notion d'*artefact* et nous proposons dans la suite de voir plus en détails comment nous l'appliquons à notre problématique.

2.5.2 - B Problématique de la granularité

Nous proposons maintenant d'aborder le problème de la granularité dans les modèles agents, en abordant la question de l'espace et celle du temps. Nous voulons ainsi exhiber un certain nombre d'hypothèses et de définitions qui caractérisent les modèles agents ou IBM. Avant toute chose, la question de la granularité se pose sur l'environnement des agents.

L'approche agent ne spécifie pas de granularité particulière de l'espace au contraire des automates cellulaires par exemple. Par contre, comme nous l'avons déjà vu, la simulation informatique, et en particulier la simulation multi-agents, introduit une discrétisation du temps. Il est alors nécessaire d'aborder le problème de l'activation des agents et l'ordonnancement est alors un problème crucial de la simulation multi-agents (Michel 00).

Il y a deux grandes familles d'activation : 1^o synchrone et 2^o asynchrone. Une activation synchrone permet d'approcher la nature «parallèle» du système réel sur une machine séquentielle. Dans un écosystème réel par exemple chaque individu a son propre rythme de vie. Sur une machine séquentielle, il est nécessaire d'exécuter les comportements individuels des agents l'un après l'autre. Le principe de l'approche synchrone est alors de garantir que pour un pas de temps donné, tous les agents perçoivent le même état du monde et décident ensuite de l'action à effectuer⁸. Les actions de tous les agents sont mises en œuvre à la fin du pas de temps, lorsque tous les agents ont été activés. Ainsi, dans l'approche synchrone, un pas de temps correspond à un instant de temps. C'est d'ailleurs la modélisation la plus naturelle du temps au regard de ce que l'on entend par discrétisation temporelle. Il n'y a donc pas de problème d'ordonnancement et les agents peuvent être activés dans un ordre quelconque. Cependant, parce que les actions sont mises en œuvre à la fin, il peut être nécessaire d'avoir des politiques de gestion de conflit (Michel 04). Et, suivant le système modélisé et les interactions possibles entre les agents et l'environnement, les conflits entre des actions effectuées par des agents différents sur le même objet du monde peuvent être très difficiles à gérer.

Dans l'approche asynchrone, les agents perçoivent, décident et agissent chacun à leur tour. Les conflits entre deux actions de deux agents différents sont alors impossibles, car les agents modifient le monde au moment de leur activation et les agents, activés après lui au même pas de temps, perçoivent le monde ainsi modifié. L'approche asynchrone correspond

⁸ Rappelons que les agents sont caractérisés classiquement par la boucle de contrôle «perception-décision-action».

davantage à l'idée que le système est fortement distribué et non organisé et synchroniser les agents implique déjà un contrôle centralisé sur le monde. Cependant, le pas de temps n'est plus strictement instantané et l'approche asynchrone pose alors le problème de l'ordonnancement des agents. Classiquement, les agents sont ordonnancés de manière pseudo-aléatoire, ce qui permet de minimiser les biais que peut entraîner un ordre spécifique d'activation.

2.5.3 Modélisation thématique : comment construire un modèle ?

Nous proposons maintenant de voir un panel de plate-formes agent de modélisation et de simulation. L'objectif n'est pas d'être exhaustif, mais de proposer quelques éléments de repère. Nous avons choisi de les catégoriser suivant la manière dont ces plates-formes aborde la construction de modèles, mais il existe d'autres éléments de comparaison (le lecteur pourra lire par exemple (Railsback 06)). Les approches spécifiques, d'abord, proposent des plates-formes spécialisés pour des problématiques spécifiques : ce sont des plates-formes dédiées. Il y a les approches semi-spécifiques, qui sont centrées sur une approche ou un paradigme de modélisation, par exemple l'approche IBM, et qui se basent sur certaines hypothèses fortes, mais qui permettent tout de même de construire une grande variété de modèles. Enfin, il y a les approches dites génériques qui visent à proposer des plates-formes de modélisation et de simulation permettant de construire n'importe quel SMA.

L'intérêt des plates-formes dédiées est essentiellement de permettre de se concentrer sur la modélisation thématique. On peut penser à MANTA (Drogoul 92) évidemment, à BacSim (Kreft 98), un simulateur pour la modélisation individu-centrée de la croissance de colonies de bactéries, ou à des modèles basés sur Swarm comme Arborscapes (Savage 99) qui propose un modèle de dynamique sylvestre, et enfin à Osmose (Shin 01). Osmose, pour *Object-oriented Simulator of Marine Ecosystems Exploitation*, a été développé par Shin et Cury pour évaluer les dynamiques des communautés de poissons et, plus qu'une plate-forme, c'est un modèle. Individu-centré, l'approche se caractérise par sa définition de la prédation entre poissons et groupes de poissons qui dépend uniquement de la taille et de l'âge d'un individu. Les espèces peuvent occuper différents niveaux trophiques suivant leur taille, les migrations et effets saisonniers ne sont pas pris en compte. Les groupes de poissons sont explicitement représentés, constituent les entités en interaction et sont des super-individus. Les processus modélisés sont la croissance, la reproduction, la prédation et la famine.

Les approches semi-spécifiques ont l'avantage de fournir moins de contraintes sur la modélisation et sont davantage des plates-formes que des modèles. Elles sont semi-spécifiques parce qu'elles se concentrent sur un domaine ou un type d'application thématique la plupart du temps. Modydic⁹ (Ginot 98), par exemple, est un outil de modélisation et de simulation

⁹ <http://www.avignon.inra.fr/mobidyc>

dédié aux modèles individu-centrés pour l'étude de la dynamique des communautés. La démarche de Mobydic est donc de se concentrer sur un domaine d'étude thématique, ce qui a permis aux constructeurs de Mobydic de se baser sur une deuxième hypothèse fondamentale pour leur système. Mobydic propose en effet aux utilisateurs – thématiciens ou informaticiens – de construire tous les comportements de leurs agents à partir de comportements atomiques (ou élémentaires). A partir de ces comportements, l'utilisateur est censé pouvoir décrire n'importe quel agent (animat, agent cellule ou agent non situé) et donc pouvoir composer les comportements de ces agents à partir des primitives fournies et paramétrables. Basé sur SmallTalk, le code des agents est en réalité toujours accessibles et n'importe quelle primitive peut être ajoutée à la bibliothèque de primitives existantes. L'intérêt premier est alors de permettre très rapidement et très simplement la construction de modèles IBM. Les limitations principales de l'approche sont liées à la discrétisation spatiale en cellules qui est basée sur un système d'automates cellulaires. Cependant, cet outil a participé largement au questionnement méthodologique et, en se positionnant sur un domaine d'application particulier (qui va tout de même de l'écologie à la biologie), il permet de réfléchir sur la question des peuplements en termes de primitives fondamentales structurant ces systèmes.

La plate-forme de modélisation et de simulation Cormas (Bousquet 98a) est également une plate-forme semi-spécifique et qui se concentre sur l'étude de la gestion des ressources naturelles. Basée sur des automates cellulaires, Cormas représente une méthodologie et un point de vue sur la modélisation. L'espace est discrétisé en cellules, justement parce que l'environnement est porteur de ressources (Bousquet 01). Remarquons par ailleurs que les données réelles sur les ressources sont elles aussi le plus souvent discrétisées de manière uniforme et homogène. Donc, Cormas prend finalement une réalité des données et de la description thématique comme contrainte de modélisation. De plus, Cormas permet l'utilisation d'agents non situés, appelés agents communicants, donc la modélisation n'est pas limitée à l'utilisation d'automates cellulaires. Le désavantage est évidemment tout de même lié aux automates cellulaires qui forcent la discrétisation spatiale par un maillage de cellules homogènes (topologie régulière et voisinage homogène). De la même manière, la gestion du temps dans les automates cellulaires est synchrone et donc l'état des cellules est modifié de manière «atomique» à la fin de chaque cycle de simulation.

Enfin, on remarquera également la plate-forme agent inVitro (McDonald 05), qui permet de construire des modèles écologiques agrégés ou IBM. Elle permet également l'utilisation de super-individus et de modèles orientés individu. Les modèles agrégés peuvent ainsi représentés des groupes ou des populations et être manipulés à l'intérieur d'un agent.

Enfin, il existe quelques plates-formes qui abordent de manière très générale la problématique de la modélisation agent. Elles se basent essentiellement sur le paradigme de la modularité et proposent de construire et de rassembler des bibliothèques d'outils réutilisables. L'avantage est qu'elles n'imposent aucun point de vue particulier et place le travail de réflexion sur le langage de programmation de modèles et de simulations, sur les outils génériques d'observation et d'analyse d'un système, etc. Nous présentons ici Swarm et MI-

MOSA.

Swarm¹⁰ est typiquement un progiciel construit sur la base d’une architecture de bibliothèques donnant accès à un certain nombre de services donc la fonction principale est de permettre la simulation de collections d’agents interagissants de manière concurrente (Minar 96; Hiebeler 94). Ces services vont de la gestion des agents (cycles de vie, ordonnancement), à la gestion de la communication (messages) ou à l’observation des systèmes (mécanismes de sondes) et à sa visualisation, etc.

Nous terminons en nous intéressant à la plate-forme MIMOSA¹¹ (Muller 04; Muller 05). Face à la concurrence outre-atlantique, une partie de la communauté agent française a voulu s’organiser pour proposer une plate-forme de modélisation et de simulation individu-centrée et plus largement agent au travers du projet MIMOSA. L’objectif ambitieux est de construire une plate-forme structurée en trois couches : le noyau multi-agents de simulation, le niveau de modélisation et le niveau de mise en œuvre des simulations. Donc, le projet s’articule autour de ces trois axes qui sont aussi bien technologiques que méthodologiques et avec la perspective de permettre une réflexion de fond sur les différentes problématiques de la modélisation et de la simulation IBM.

2.5.4 Multi-modélisation : le couplage vu par les agents

Comme nous l’avons vu dans la partie 2.3.5, la multi-modélisation permet d’appréhender l’hétérogénéité des systèmes que l’on étudie. Nous avons également abordé le formalisme DEVS (*Discrete Event System Specification*) (Zeigler 89) qui offre un cadre conceptuel formel dans lequel on peut exprimer tous les autres formalismes et en particulier celui des modèles de type IBM et des modèles à base d’équations différentielles. La plate-forme de simulation VLE, pour *Virtual Laboratory Environment*¹² (Ramat 03), est basée sur le formalisme DEVS. Il est alors possible dans VLE de coupler des modèles hétérogènes et de les mettre en œuvre dans un même modèle global. VLE fournit de plus les éléments de base permettant de définir des agents, un ensemble d’instruments de mesure pour suivre l’évolution des grandeurs du système et un moyen pour modifier dynamiquement le système et les agents. Cette plate-forme permet de construire des modèles hétérogènes dans le même modèle global et d’éviter ainsi de voir la multi-modélisation comme le couplage de simulateurs différents.

L’approche agent permet également de «faire évoluer» ensemble des modèles hétérogènes. Ainsi, l’approche agent ne présuppose rien sur le mécanisme interne des agents. Par rapport à la microsimulation ou aux automates cellulaires, il est alors possible d’aborder la complexité des systèmes que l’on étudie sans les contraindre *a priori*. Ainsi, tout la question du

¹⁰ <http://www.swarm.org>

¹¹ <http://sourceforge.net/projects/mimosa>

¹² <http://vle.univ-littoral.fr>

couplage de modèles portés par des agents se pose dans leur capacité d'interagir ensemble. Si les interactions entre agents sont directes, cela implique l'utilisation d'un mécanisme de communication commun. Classiquement, les agents communiquent à l'aide de messages. Si les interactions entre agents sont indirectes, les agents communiquent *via* l'environnement, et en particulier via les *artefacts* de l'environnement. On peut ainsi construire un couplage faible via l'environnement sans que les différents agents soient contraints à partager un cadre formel strict.

2.5.5 Modélisation et simulation à l'aide d'agents

Nous avons jusque là parlé de la modélisation et la simulation à base d'agents. Mais les agents peuvent, non pas représenter des concepts thématiques, mais aider à la construction de modèles et au processus de simulation. Comme nous l'avons vu, nous sommes confrontés à l'hétérogénéité et à la richesse des modèles eux-mêmes, qui deviennent des multi-modèles, et à la complexité des processus de construction et d'expérimentation. Il est nécessaire également de pouvoir explorer les modèles dans différents espaces de paramètres, de mettre en œuvre plusieurs modèles dans de nombreuses simulations. Les agents représentent alors une solution permettant d'augmenter la maîtrise de ces différents processus. En distribuant le contrôle, en permettant un point de vue local et distribué, à n'importe quel niveau d'abstraction, les agents sont un moyen permettant de localiser des heuristiques de contrôle et de recherche à tous les niveaux clés des processus de modélisation et de simulation. Peu de travaux explorent cette problématique. On peut cependant citer Pereira et al. (Pereira 04b; Pereira 04a) qui ont surtout travaillé sur la question de la recherche automatique des paramètres. Ils proposent pour cela un agent calibration qui doit optimiser les résultats de simulation suivant une fonction de fitness. Ils envisagent pour cela le langage ECOLANG (Pereira 05) qui permet aux agents de se communiquer des informations sur le système simulé. L'idée est ensuite d'explorer, suivant différentes heuristiques, les espaces de paramètres et d'effectuer des simulations automatiques.

Pour conclure, nous remarquons qu'une solution à base d'agents pour la modélisation et la simulation ouvre des perspectives sur la manière d'envisager la construction et l'étude d'un modèle. Il est cependant nécessaire alors de pouvoir aborder une telle solution avec un outil de modélisation et de simulation qui permette de l'envisager de manière rationnelle.

Chapitre 3

Positionnement

3.1 Fondements de la proposition

Le chapitre précédent nous a permis de présenter l’approche POM, *in virtuo* et agent sur lesquelles nous nous appuyons pour formuler notre proposition. Dans ce chapitre, nous explicitons les éléments de ces approches qui nous ont inspiré pour proposer une solution fondée sur la co-construction basée *agent-modèle* de modèles hétérogènes. Ainsi, nous faisons apparaître non seulement «l’endroit où nous sommes», mais également «le chemin que nous empruntons».

Approche POM : Comme nous l’avons présenté dans la section 2.4.4, l’approche orientée pattern met au centre de la modélisation, de l’observation et de l’analyse les traits remarquables dans les données que sont les patterns. L’objectif pour nous est donc de construire notre approche autour de ces objets qui apparaissent dans les données et de centrer les processus de modélisation et de simulation autour des données en général. Notre démarche s’inscrit alors dans une méthodologie construite par des thématiciens et centrée sur les objets que les thématiciens manipulent et étudient. Il s’agit enfin pour nous de spécifier les aspects opérationnels qu’il faut alors donner à la POM pour la mettre en œuvre dans un simulateur informatique.

Approche in virtuo : Parce que les systèmes que l’on étudie sont complexes et que les modèles de ces systèmes sont la plupart du temps complexes, il est nécessaire de pouvoir expérimenter ces systèmes. L’informatique et les domaines de la simulation multi-agents et de la réalité virtuelle nous permettent d’envisager de nouvelles possibilités de construction, d’expérimentation et d’exploration des modèles. L’expérimentation *in virtuo* est alors l’expérimentation conduite dans un univers virtuel de modèles numériques, univers dans lequel l’humain est pris en compte et participe. Il n’y a

cependant en l'état pas de méthodes ni d'outils qui permettent d'expérimenter des modèles complexes et de les explorer avec suffisamment de rigueur pour pouvoir envisager d'utiliser une approche interactive de ce type. En d'autres termes, il n'est pas évident qu'un thématicien puisse se contenter d'interagir avec le système dans un laboratoire virtuel. Il est donc nécessaire pour nous d'identifier ce qui dans la simulation de nos systèmes peut nécessiter une telle interaction avec le modèle qui tient lieu de réalité, mais qui est également une réalité (Tisseau 01a).

Modélisation et simulation à l'aide d'agents : Nous avons vu en quoi l'approche *in virtuo* peut être séduisante et permet d'entrevoir la modélisation comme une expérience dans le cadre d'un laboratoire virtuelle. Au centre de ce laboratoire se trouvent les modèles et les humains qui les construisent. Or, le modèle est un agent autonome capable de médier le travail de mise en relation du réel et de la théorie (Morgan 99). En s'appropriant les termes de Morgan et de Morrisson et en faisant du modèle d'un phénomène un agent autonome au sens de l'approche agent, nous voulons permettre la construction de modèles hétérogènes. Les systèmes multi-agents autorisent en effet à coupler des entités hétérogènes dont le fonctionnement interne est inconnu. On peut alors construire la simulation dans la perspective méthodologique de la POM et le comportement du système – les patterns – émerge de l'interaction des différents agents – les modèles – et de leurs comportements individuels.

Il n'est pas nécessaire que les agents respectent un formalisme particulier pour interagir, parce que leurs interactions peuvent être indirectes, médiées par l'environnement, et en particulier par les données. D'autre part, l'approche agent cherche à construire des systèmes complexes, en se basant sur les boucles de rétroaction, les mécanismes d'émergence et d'auto-organisation. Or, dans le cadre des modèles écologiques et d'exploitation économique, on cherche à comprendre le fonctionnement de tels systèmes en se basant sur les mêmes concepts. Il est alors naturel de réfléchir non pas à l'agent comme support de l'individu thématique, mais comme support des phénomènes qui se trouvent à différentes échelles d'abstraction, qui ont un rôle dans la construction du modèle et son étude et dont le point de vue est local.

3.2 Proposition

Partant de notre positionnement, notre proposition se construit autour des points suivants.

Un modèle : Nous proposons de construire les modèles de systèmes complexes en utilisant les notions d'artefacts (Omicini 06) et d'*agents-modèles* et de structurer la modélisation en vue de constituer une société de modèles. Il s'agit pour nous d'articuler le modèle du système complexe autour des artefacts qui sont les supports des données partagés par les agents. Les agents, naturels ou artificiels (Chicoisne 02), produisent ou perçoivent les données au travers des artefacts et sont par là-même contraints dans leurs interactions et

donc dans leur collaboration avec les autres agents pour produire la dynamique globale. Les artefacts sont enfin le support de l'analyse et le principe est de ne percevoir les dynamiques du modèle qu'au travers de l'évolution des artefacts, donc des données, et en particulier de reconstruire sur ces évolutions, les dynamiques des patterns structurant la question de modélisation. Car ce sont les patterns qui permettent *in fine* de discriminer les simulations et de valider les résultats obtenus.

La notion d'*agent-modèle* sert de support au modèle d'un phénomène donné qui constitue alors le comportement de l'agent. Les rôles de l'agent sont les suivants : 1° il est responsable du couplage du modèle avec l'environnement. L'agent-modèle rend compte de la dynamique du phénomène modélisé. Il est responsable de l'exécution du modèle et donc de la lecture des artefacts dont le modèle a besoin et de la production des artefacts pour lesquels il s'est engagé. 2° L'agent est également responsable du «bon» comportement du modèle. Il peut ainsi contrôler les dynamiques de son modèle au regard d'un domaine de variabilité des variables de sortie du modèle, domaine spécifié par les experts 3° Enfin, l'agent peut servir de médiateur entre les contraintes du modèle global et celles de son modèle interne.

Nous proposons également d'utiliser deux autres types d'agents qui se sont révélés pertinents au cours des processus de modélisation et de simulation. Les agents-observateurs construisent des observables à partir des artefacts. Ils jouent un rôle central dans l'approche orientée pattern en distinguant ce qui relève des données, des modèles et des patterns. Les agents-contrôleurs permettent de mettre en relation les dynamiques de plusieurs agents-modèles. Ils servent de support à des algorithmes de contrôle et d'observation permettant de construire des contraintes entre plusieurs agents et donc entre plusieurs phénomènes.

Nous proposons avec ce modèle d'aborder la question d'une société de modèles en permettant d'appréhender la construction du modèle complexe comme un système d'entités autonomes. La dynamique du modèle global, exhibant les patterns recherchés, émergent alors des comportements et des interactions des agents du système. Les questions de la distribution de la maîtrise sur le système passent alors par l'analyse du rôle de l'agent dans le système, des problématiques de manipulation d'un modèle par l'agent et de ses capacités à faciliter le contrôle et l'analyse du modèle.

Un outil : Nous avons construit un outil de simulation afin : 1° d'aborder toutes les problématiques computationnelles liées à l'utilisation d'un simulateur agent ; et 2° de préciser et comprendre les problématiques de l'outil basé agent-modèle, car de la même manière que l'on apprend d'un modèle en le construisant, nous avons appris de notre outil pour la construction de multi-modèles en le développant.

Notre simulateur permet une modélisation centrée sur les données au moyen des artefacts, solution que nous avons formalisé et qui permet de décrire toutes les données de nos modèles ainsi que de fournir les moyens de leur manipulation. Il est possible de coupler des modèles hétérogènes au moyen du système d'agent que nous proposons. Ces agents nous permettent également en l'état actuel du simulateur de construire des modèles IBM, dits *émergents*, des modèles à base d'EDP, dits *intensionnels*, et

des modèles de données, dits *extensionnels*. Les modèles *extensionnels* permettent par ailleurs d'expliciter les forçages des modèles qui sont critiques dans les simulations en général.

Les agents sont construits comme des boîtes noires et nous proposons des solutions pour leur permettre de percevoir et d'agir sur le monde, c'est-à-dire les artefacts. Afin de préserver la cohérence temporelle des modèles thématiques, nous proposons une synchronisation des agents-modèles et un système de marquage temporel automatique des artefacts qui permet aux agents de détecter automatiquement les incohérences temporelles. Cette synchronisation introduit une nouvelle boucle dans le pas de temps et la notion de jonction qui permet de lire et écrire les artefacts de manière automatique. Nous proposons d'utiliser des *opérations* qui peuvent être directement manipulées par les agents afin de leur permettre de modifier leurs perceptions et actions sur l'environnement.

Une méthodologie : Nous proposons finalement une méthodologie basée sur la notion d'*agent-modèle* et sur la POM pour la construction de modèles hétérogènes. Notre proposition méthodologique est alors celle-ci :

1. formulation de la question de modélisation ;
2. construction du cadre de modélisation, c'est-à-dire, identification des patterns, puis description des artefacts, choix des *agents-modèles* (phénomènes), et enfin choix des agents-observateurs et agents-contrôleurs du système ;
3. phase de modélisation : construction, éventuellement incrémentale, des modèles des *agents-modèles*, puis implémentation et validation des modèles computationnels, et enfin choix des paramètres ;
4. phase de simulation : production de résultats – le système est exécuté sans être interrompu¹ suivant un plan de simulation rigoureux – et analyse des données.

Il s'agit ainsi de rationaliser la simulation en s'appuyant sur la solution d'*agent-modèle*. Remarquons que nous suggérons de distinguer le processus de modélisation, qui peut éventuellement être interactif dans la perspective d'un laboratoire virtuel, et le processus de simulation qui doit privilégier la rigueur et la production de résultats. De plus, nous proposons qu'à la question de modélisation, qui fonde et dirige toute action de modélisation, soit associée un cadre de modélisation qui décrit complètement tous les agents et artefacts impliqués dans le modèle global. Ce cadre de modélisation doit ainsi permettre de reproduire les patterns identifiés avec la question de modélisation et doit faciliter les collaborations entre thématiciens, sans préjuger, ni définir le fonctionnement interne des agents.

¹ Les modèles participatifs sont par contre autorisés et le système ne doit pas être interrompu par les expérimentateurs uniquement.

Chapitre 4

Une société de modèles pour la modélisation de systèmes complexes

4.1 Introduction

Nous présentons dans cette partie notre proposition pour la modélisation et la simulation basée *agent-modèle*. Cette partie se structure en trois sections : 1° la proposition conceptuelle d'*agent-modèle* et de société de modèles, 2° l'outil de simulation construit sur le modèle au travers de certains éléments clés de la proposition et 3° la méthodologie pour la mise en œuvre de simulations à base d'*agents-modèles*.

Concernant la proposition conceptuelle d'abord, elle constitue le point d'orgue de notre proposition. Nous développons sa présentation en prenant pour guide l'approche «voyelles» (Demazeau 01; Demazeau 95). Concernant l'outil, nous présentons le simulateur qui a été réalisé dans le cadre de la thèse. Seulement, nous n'abordons que les problématiques qui ne sont pas présentées dans la section décrivant la proposition conceptuelle. Les autres sont alors implicitement validées par les résultats de simulation obtenus et présentés dans ce mémoire. Nous nous concentrons ainsi essentiellement sur les moyens que l'on a construit pour permettre des modélisations IBM dans notre simulateur. Pour finir, nous présentons notre proposition méthodologique en détaillant la construction du cadre de modélisation, la phase ou expérience de modélisation et enfin la phase de simulation.

4.2 Solution conceptuelle basée agent-modèle

Dans cette section, nous développons tout d'abord les notions d'E-nvironnement de modélisation et d'*artefacts*. Nous décrivons ensuite les A-gents de notre solution et leur rôle dans l'environnement de modélisation. Puis, nous abordons la question des I-nteractions entre ces agents, interactions indirectes médiées par l'environnement et qui se basent sur la notion de *jonction*. Nous pouvons ainsi décrire ce que nous entendons par une société de modèles qui forme l'O-rganisation des agents dans leur environnement. Ceci nous amène à proposer dans la suite un certain nombre de caractéristiques de la société de modèles, c'est-à-dire du modèle global. Enfin, nous étudions la question de l'U-tilisateur et sa place dans la modélisation.

Notons que nous utilisons une notation graphique pour décrire nos systèmes d'*agents-modèles*. Cette notation graphique est une extension du langage UML (Rumbaugh 99). Nous introduisons alors la notation de chaque concept au moment de le présenter, ce qui permet de progresser au même rythme dans la description formelle et dans la description graphique de notre modèle.

4.2.1 E-nvironnement de modélisation : artefacts et patterns

4.2.1 - A Notion d'*artefact*

L'application de la modélisation orientée pattern à la construction du modèle se traduit par la suite d'actions suivante : 1° exhiber les données à partir desquelles sont observées les patterns du système étudié ; 2° proposer une composition de sous-systèmes en interaction qui modélisent les phénomènes et qui reproduisent la dynamique des données, desquelles émergent les patterns. Il nous faut par conséquent, dans l'environnement de modélisation, identifier un moyen pour porter les données et nous proposons pour cela la notion d'*artefact*. Nous nous intéressons particulièrement aux *artefacts* sociaux (Omicini 06) qui permettent alors de médier les interactions entre les agents.

4.2.1 - B L'*artefact* comme support d'identification des patterns

La notion d'*artefact* nous autorise à structurer l'environnement de modélisation autour des objets qui permettent et rationalisent toutes les interactions entre agents. Mais avant cela, un *artefact* est, pour nous, le support d'identification des patterns. Il porte les données dans lesquelles les thématiciens ont observés les patterns et peut avoir de 1 à n dimensions. La figure 4.1 présente une illustration de l'évolution au cours du temps des données portées par trois exemples d'*artefacts* dont la dimension est respectivement 0, 1 et 2. De cette manière, un pattern est déterminé par un *artefact*, qui en est de ce fait un instantané temporel, ou

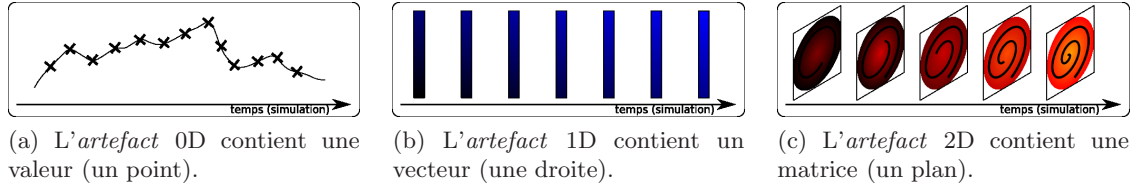


FIG. 4.1: Illustration des types de données qui sont le plus souvent manipulés et visualisation de leur évolution au cours de la simulation.

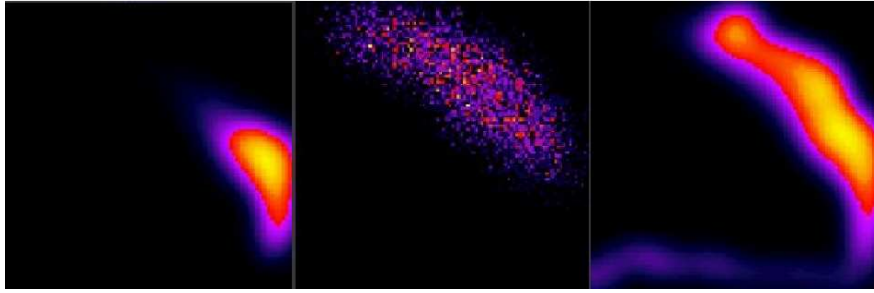


FIG. 4.2: Exemples de distributions de peuplements dans un environnement (2D).

par plusieurs *artefacts*, qui doivent être alors «observés ensemble» pour pouvoir accéder au pattern. En résumé, un *artefact* soutient une quantité mesurable, soit une donnée ou un ensemble structuré de données. La figure 4.2 donne un exemple de données de trois *artefacts* de dimension 2 pris à un pas de temps quelconque. On y voit des distributions spatiales de peuplements. Prenons un autre exemple, dans la section 2.4.4, nous avons présenté le pattern caractéristique des systèmes de type Lotka-Volterra. Il permet d'identifier l'interdépendance entre les populations de proies et de prédateurs. Ce pattern pourrait alors être observé grâce à un *artefact* portant le nombre de proies et un *artefact* portant le nombre de prédateurs au cours du temps, *artefacts* qui sont de dimension 0.

4.2.1 - C L'*artefact* comme support d'interaction des agents

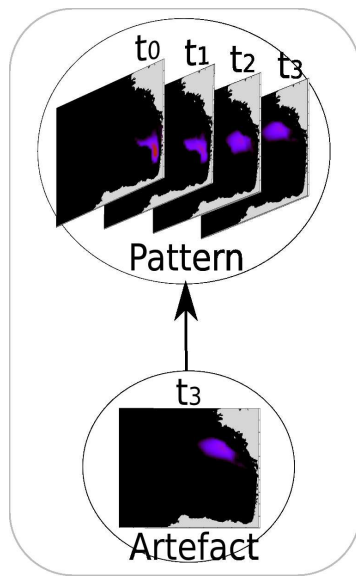
Ensuite, un *artefact* est aussi un support d'interaction entre agents. Comme les *artefacts* sociaux de Omicini et al., les *artefacts* permettent de médier les interactions entre agents et exhibent les informations nécessaires pour pouvoir être utilisés. Ces informations décrivent les données que les *artefacts* portent, informations que nous formalisons par une sémantique et une forme qui sont accessibles et connues par les agents. La sémantique donne à l'agent la capacité de savoir si l'*artefact* lui permet d'atteindre son but, elle représente la nature de la donnée portée par l'*artefact*. La sémantique est ainsi proche de la notion de "description de service" qui caractérise les *artefacts* de Omicini et al. La forme d'un *artefact* est la définition de son occupation spatiale et temporelle. En effet, une donnée, par exemple le

nombre d'individus d'une population, peut être globale ou spatialisée dans l'environnement. Cette donnée peut alors avoir plusieurs formes : soit elle est globale et de dimension 0, soit elle occupe plusieurs dimensions et la forme décrit la distribution de la donnée dans l'espace suivant des axes de modélisation. Ainsi, l'*artefact* occupe toutes ou certaines des dimensions de l'environnement de modélisation et cette occupation est finie, c'est-à-dire qu'elle existe dans un intervalle précis. Enfin, l'*artefact* a une estampille temporelle qui permet d'éviter qu'un agent manipule un *artefact* qui n'est pas à jour ou, plus généralement, qui viole le principe de causalité. Nous nous sommes pour cela inspirés des travaux de Lamport (Lamport 78). Pour résumer, l'agent peut connaître le type de la donnée de l'*artefact*, c'est-à-dire le service de l'*artefact*, et savoir s'il lui est utile de manipuler l'*artefact*. Et l'agent peut donc savoir comment l'*artefact* est structuré et comment il peut alors le manipuler.

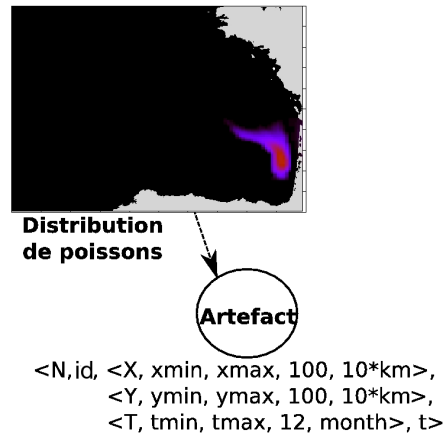
4.2.1 - D Une formalisation de l'*artefact*

Un *artefact* est caractérisé par les propriétés suivantes : $\langle S, f(S), O, t_{last} \rangle$ avec S la sémantique de l'*artefact*. $f(S)$ est une fonction, par exemple la fonction logarithme, et cela nous permet de considérer le changement d'échelle (par exemple, à l'échelle logarithmique) d'une donnée sans en perdre la sémantique (par défaut f est l'identité). t_{last} est l'estampille temporelle de l'*artefact*. Enfin, O est l'ensemble des axes spatio-temporels que l'*artefact* occupe. O n'est jamais vide et contient au moins la dimension temporelle. Une occupation O se formalise ainsi : $\langle S, v_{init}, v_{final}, \delta v, unite \rangle$. S est la sémantique de l'axe considéré, donc T pour le temps par exemple, v_{init} est la position initiale de l'*artefact* sur l'axe, donc pour le temps, la date du premier pas d'actualisation de l'*artefact*. v_{final} est de la même manière la position finale de l'*artefact* sur l'axe et δv le pas de discrétisation minimum. Pour l'occupation temporelle, c'est le pas de temps minimum d'actualisation de l'*artefact*. Enfin, $unite$ est l'unité dans laquelle sont représentées les données portées par l'*artefact*.

Les figures 4.3a et 4.3b présentent un exemple d'*artefact*. La figure 4.3a est une illustration permettant de visualiser l'observation d'un pattern dans un *artefact*. Le pattern en question est une évolution spatiale remarquable et l'*artefact* en est l'instantané temporel. Nous remarquons que dans cet exemple, la donnée portée par l'*artefact* est de dimension 2, elle occupe donc deux axes de modélisation. La figure 4.3b présente la formalisation de cet *artefact*. Dans cet exemple, x_{min} , x_{max} , y_{min} , y_{max} , t_{min} , t_{max} sont des paramètres et l'environnement de modélisation doit au moins contenir les dimensions ou axes de modélisation X , Y et T . Les discrétisations et unités choisies illustrent les valeurs possibles. On voit également dans cette dernière figure que l'on note graphiquement les *artefacts* comme un cercle avec une étiquette détaillant leur sémantique et leur forme.



(a) Exemple schématique d'un *artefact* et du pattern observé à partir de cet *artefact*



(b) Détails de l'*artefact* et notation graphique

FIG. 4.3: Exemple d'un *artefact* décrivant la structure spatiale d'un peuplement à un instant donné.

4.2.1 - E L'*artefact* comme support de l'analyse

Finalement, un *artefact* est un support d'analyse et, en tant que modélisateur du système, notre observation de celui-ci s'appuie sur les *artefacts*. Il nous est possible ainsi d'observer les patterns du système simulé. De plus, l'observation du système est rationalisée, parce qu'il ne s'agit plus d'observer n'importe où dans le modèle, mais seulement les informations structurant le modèle. Le comportement des agents est alors observé indirectement via les données et les agents, en tant que boîtes noires, peuvent avoir n'importe quelle structure et n'importe quelle complexité. On rejoint là un éclairage systémique de l'observation du modèle. Notons que nous introduisons, dans la section 4.2.4, la notion d'agent-observateur dont le rôle est de construire des observables à partir des *artefacts*.

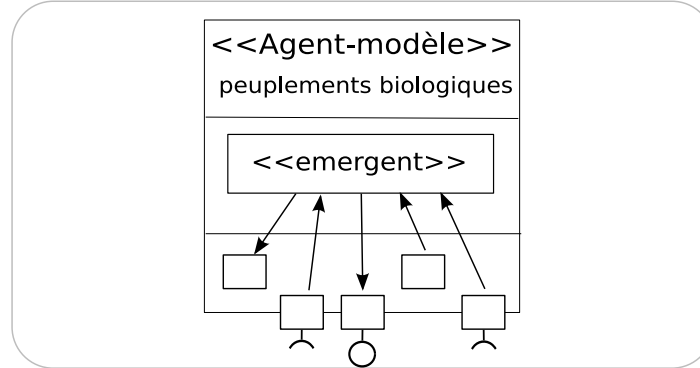
4.2.2 A—agents : notion d'*agent-modèle*

Remarque Avant de commencer, précisons les termes d'agent et d'agent thématique. Le mot agent est consacré ici aux agents de notre environnement de modélisation. Les agents thématiques sont les agents-individus qui correspondent aux individus conceptuels des IBM.

Suivant la méthodologie POM, après avoir exhiber les données dans lesquelles sont observées les patterns, il est nécessaire de proposer une composition de sous-systèmes en interaction. Chaque sous-système représente un phénomène que l'on veut modéliser et qui influence la dynamique globale et donc l'émergence des patterns. Nous proposons alors que le modèle global soit conçu comme une société de modèles en interaction et que, dans cette société de modèles, chaque agent mette en œuvre un de ces modèles, c'est-à-dire un des phénomènes que l'on veut modéliser. Nous proposons ainsi la notion d'*agent-modèle* qui met en œuvre le modèle d'un phénomène. Ainsi, se pose un certain nombre de questions que nous abordons maintenant : le but de l'agent en modélisation, sa rationalité et les moyens qu'il a pour percevoir, raisonner sur des *artefacts* et les manipuler.

4.2.2 - A But et rationalité des *agents-modèles*

La rationalité des agents est structurée par les *artefacts* lus ou écrits. L'agent perçoit l'environnement de modélisation et donc tous les *artefacts* qui le composent. Cette perception des *artefacts* lui permet de connaître leur description et en particulier leur sémantique et leur forme. L'agent qui prend en entrée un *artefact* avec une sémantique et une forme données peut alors chercher dans l'environnement cet *artefact* pour le manipuler. L'agent agit dans son environnement en écrivant des données dans des *artefacts*. Pour pouvoir faire cela, il doit vérifier auparavant que les données qu'il écrit respectent la sémantique et la forme de l'*artefact*. Ainsi, les agents contrôlent localement leur influence sur les *artefacts* qui sont

FIG. 4.4: Exemple de notation d'un *agent-modèle*

complètement formalisés. De plus, si un agent ne peut pas manipuler un *artefact*, il peut en informer les modélisateurs et ainsi signaler des problèmes de structuration du modèle.

La rationalité de l'agent, c'est plus généralement sa capacité à calculer les modifications de ses variables internes ou de son environnement (les *artefacts*). Cette rationalité dépend alors du modèle de l'*agent-modèle*. Un agent est donc défini par des *propriétés* qui peuvent être des variables, paramètres et constantes qui sont manipulés par le modèle interne de l'agent qui constitue le comportement de l'*agent-modèle*. Ces *propriétés* sont soit des *propriétés* d'entrées-sorties par lesquelles l'agent interagit avec les *artefacts*, soit des *propriétés* internes à l'agent. Les variables et les paramètres peuvent être définis comme appartenant à un domaine particulier de valeurs. Cela permet à l'agent : 1° de maintenir des observateurs locaux sur les variables de son modèle et les enregistrer quand celles-ci sortent de leur domaine de valeurs ou le signaler aux modélisateurs ; 2° de mettre en œuvre des heuristiques particulières pour initialiser les paramètres de son modèle.

La figure 4.4 présente un exemple d'*agent-modèle* sous la forme d'un classeur structuré. Le nom de l'agent figure dans la partie haute du classeur. Le stéréotype définit le rôle de l'agent, qui est ici un *agent-modèle*. On fait figurer en-dessous du nom, dans la case centrale, le comportement de l'agent et donc ici le modèle de l'*agent-modèle*. Enfin, on fait apparaître en-dessous les *propriétés* de l'*agent-modèle*. Les flèches permettent de décrire la manière dont le modèle interagit avec les *propriétés*. Enfin, les *propriétés* peuvent éventuellement être entrantes ou sortantes, elles sont alors visibles de l'extérieur du classeur et on utilise la notation UML des ports auxquels sont associés des interfaces (entrée ou sortie).

4.2.2 - B Notion d'*opération* : vers une adaptation des agents

Il se trouve des situations dans lesquelles un *artefact* utilisé par un agent a la bonne sémantique mais pas la bonne forme, ceci nécessite donc une adaptation de l'agent à l'*artefact*.

Afin d'assurer une bonne maîtrise de la modélisation, nous avons choisi de rendre explicite cette adaptation en introduisant la notion d'*opération*. En effet, changer la forme de l'*artefact* n'est pas toujours souhaitable et «noyer» la transformation dans le modèle fait perdre en visibilité et spécialise inutilement le modèle par rapport à un environnement. Par suite, un *agent-modèle*, qui prend en entrée un *artefact* avec une sémantique donnée et qui trouve dans l'environnement un *artefact* avec la même sémantique mais une forme qui ne correspond pas à celle requise par l'agent, peut utiliser une *opération* pour changer la forme de l'*artefact* et ainsi avoir une *propriété* qui en est une transformation. Les *opérations* permettent ainsi d'effectuer des transformations sur les données perçues ou écrites dans les *artefacts* par les agents. Si cette transformation n'est pas possible, l'*agent-modèle* envoie alors un message aux modélisateurs pour leur signaler son incapacité à fonctionner.

Une *opération* est un modèle mathématique qui est décrit comme la transformation d'une forme d'entrée en une forme de sortie. Il peut s'agir de transformations géométriques ou analytiques (projection, moyenne, somme, etc.) sur la forme de l'*artefact* et des opérations analytiques de changement d'échelle (logarithmique par exemple) sur la sémantique. L'agent peut ainsi appliquer une *opération* sur la donnée qu'il perçoit ou qu'il écrit dans un *artefact*. D'un point de vue méthodologique, la notion d'*opération* permet d'externaliser les modèles mathématiques et de ne pas les confondre avec le comportement de l'agent. On distingue alors les *opérations* de : 1^o projection, 2^o discrétisation et 3^o réduction (voir figure 4.5).

Opération de projection : elle permet de réduire le nombre d'axes de modélisation qu'une donnée occupe, c'est-à-dire de réduire sa dimension (voir figure 4.5a). Pour une donnée exprimée sur un axe, donc de dimension 1, la projeter revient à ne plus la spatialiser.

Opération de discrétisation : elle permet de réduire ou d'augmenter la discrétisation d'une donnée (voir figure 4.5b). La contrainte forte de cette opération est qu'il n'est possible de transformer une discrétisation donnée en une autre discrétisation que si elles sont multiples l'une de l'autre. Sinon, on ne sait pas faire dans le cas général et donc une *opération* n'est pas pertinente pour une telle manipulation.

Augmenter la discrétisation est déjà moins naturel, car elle requiert de savoir comment, dans une case donnée que l'on découpe en sous-cases, distribuer la donnée de la case initiale dans ces sous-cases. Nous avons souvent rencontré le besoin d'une telle *opération* et nous nous sommes rendus compte que la plupart du temps, la meilleure solution était une distribution homogène. En d'autres termes, la valeur de chaque case est divisée en autant de sous-cases qui composent la case initiale et le résultat est alors affecté dans ces sous-cases.

Opération de réduction : elle permet d'extraire une partie de l'*artefact* et de réduire l'occupation des axes de modélisation de l'*artefact* (voir figure 4.5c).

La figure 4.6a présente un exemple de système d'*agent-modèle*. L'*agent-modèle* du bas met en œuvre un modèle de peuplements et écrit dans un *artefact* qui décrit la distribution spatiale de ce peuplements au cours du temps et qui a donc deux dimensions. L'*agent-modèle*

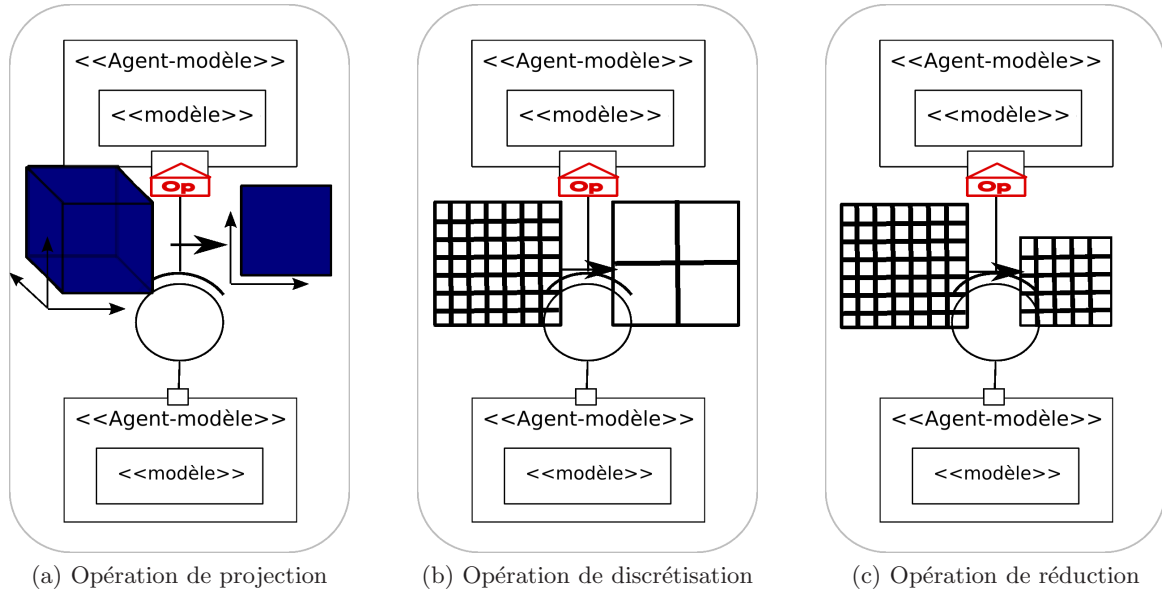


FIG. 4.5: Les trois types d'opérations qu'un agent peut manipuler.

du haut met en œuvre un modèle de pêcheries non spatialisé et qui nécessite par conséquent en entrée un *artefact* des peuplements non spatialisé, donc de dimension 0. En l'état, l'*agent-modèle* "pêcheries" ne peut lire l'*artefact* auquel il est attaché. Il utilise par conséquent une *opération* de projection afin de calculer, à partir des données de l'*artefact*, l'effectif total du peuplement. La figure 4.6b en présente la configuration.

4.2.2 - C Comportement d'un *agent-modèle* et types de modèles

L'intérêt des agents est avant tout de permettre de les considérer comme des boîtes noires. Leur comportement, et donc le modèle qu'ils mettent en œuvre, doit pouvoir décrire des phénomènes divers et être construit avec différents niveaux de description et différentes approches de modélisation. Pour le modélisateur, les *agents-modèles* représentent, dans le modèle global, des phénomènes qui doivent reproduire une certaine dynamique pour expliquer l'émergence des patterns dans les données (les *artefacts*). Mais, à l'intérieur de l'agent, le modélisateur doit pouvoir accéder aux données d'entrées et produire des données de sorties, au travers des *propriétés* qui lisent et écrivent dans des *artefacts*, et construire sa propre représentation du phénomène que doit mettre en œuvre l'agent. Le comportement de l'agent dépend alors du modèle choisi par le modélisateur et donc du paradigme de modélisation choisi. Les *agents-modèles* peuvent ainsi manipuler différents types de modèles que nous avons nommé : 1° extensionnel, 2° intensionnel, 3° émergent et 4° participatif.

Remarque Extensionnel signifie la portée d'un terme ou concept tel que mesuré par les

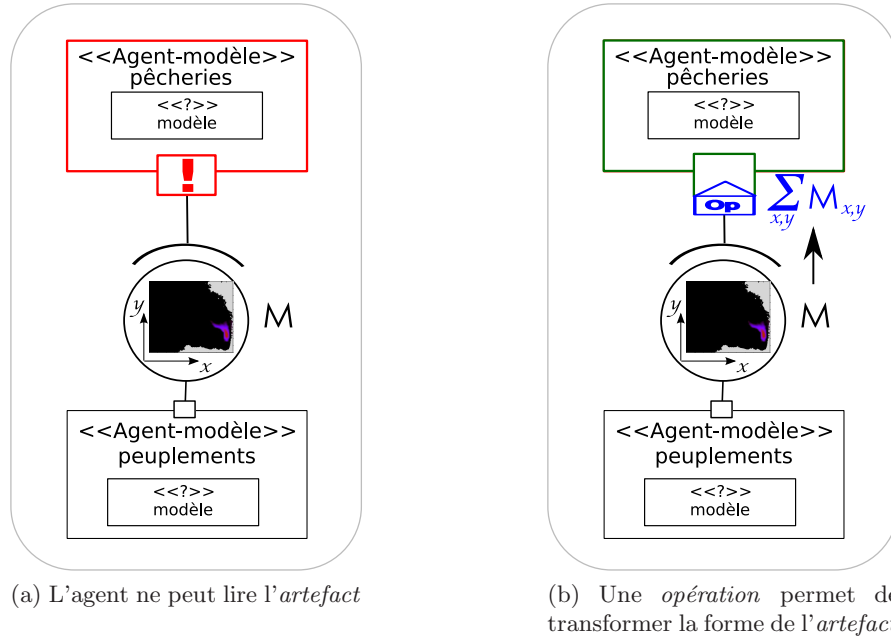


FIG. 4.6: Exemple de système d'*agents-modèles* : problème de lecture d'un *artefact* par un *agent-modèle* «pêcheries» et utilisation d'une *opération*.

objets qu'il dénote ou contient (New Oxford American Dictionary). On dit alors qu'un modèle M est décrit par extension, qu'il est de type extensionnel, si sa définition est l'exhibition de tous ses états S_i . M est alors du type : $M : \{S_1, \dots, S_n\}$

Un *agent-modèle* extensionnel met en œuvre un scénario pré-calculé d'évolution de données. Il n'est pas influencé par l'extérieur et, à chaque pas de temps, suit le scénario d'évolution prévu pour ses *propriétés*. Un *agent-modèle* extensionnel nous permet d'intégrer les forçages par des données au cours de la simulation, tout en considérant ces forçages pour ce qu'ils sont, c'est-à-dire des modèles. Les forçages sont ainsi explicités au sein du modèle global, ce qui permet d'un point de vue méthodologique de pouvoir à tout moment connaître les éléments de modélisation qui expliquent la dynamique globale. On peut penser par exemple aux données de température dans un modèle d'écosystème influencé par le réchauffement climatique. Le modélisateur peut vouloir forcer des données de température qu'il a par exemple observé dans un environnement réel, en les intégrant dans son modèle global via un *agent-modèle* représentant un phénomène de réchauffement climatique.

Remarque "Plus l'intension d'un terme (le nombre de traits) est grande, plus l'extension (la classe des objets dénotés) est restreinte" (TLF). En conséquence, un modèle M est décrit par intension et dit intensionnel s'il est du type : $M : S(t+1) = f(S(t) + E(t))$ avec $S(t)$ un état de M à l'instant t , f une fonction et $E(t)$ une fonction intégrant les entrées du système.

Un *agent-modèle* intensionnel permet de mettre en œuvre un système d'équations différentielles *ED* ou d'équations aux dérivées partielles *EDP*. Ces *propriétés*, paramètres ou variables, forment ainsi un système d'équations. Avec ces équations, l'*agent-modèle* calcule son état à chaque pas de temps et modifie les valeurs des *propriétés* de sorties de l'agent. Le modèle intensionnel permet alors une grande diversité de modèles. Si l'on veut par exemple modéliser un système proies-prédateurs classique en utilisant le système d'équations proposé par Lotka-Volterra (Volterra 31), on peut par exemple utiliser deux *agents-modèles* de type intensionnel. Chaque agent met alors en œuvre une de deux équations du système et représente ainsi soit le peuplement des proies, soit celui des prédateurs. Chaque agent a également deux *propriétés* qui lui permet de lire l'*artefact* décrivant l'état (le nombre d'individus) de l'autre population et d'écrire son propre état à chaque pas de temps dans l'*artefact* correspondant, les modèles ne percevant que les *propriétés* des agents et non les *artefacts*.

Remarque Un modèle *M* est dit émergent si l'observation de son état global n'est pas directement ou *a priori* accessible, mais émerge de l'interaction des agents-individus qui le composent.

Un *agent-modèle* émergent met en œuvre un IBM. Il est constitué d'un système multi-agents (ou toute autre solution permettant une simulation IBM) qui utilise les *propriétés* d'entrées et modifie les *propriétés* de sorties. A chaque cycle, l'*agent-modèle* fait évoluer son système multi-agents d'un ou plusieurs cycles d'une fréquence interne qui dépend des phénomènes qu'il cherche à reproduire (Redou 05). Prenons l'exemple d'un peuplement biologique et d'un *artefact* portant la distribution spatiale de ce peuplement par zones de pêche, l'*agent-modèle* a pour rôle de calculer l'évolution du peuplement et le SMA interne le représente. Il a une *propriété* qui est la représentation de la distribution spatiale et le SMA de l'*agent-modèle* ne perçoit que cette *propriété* et non l'*artefact*.

Remarque A chaque pas d'exécution, le comportement/état d'un modèle participatif est donné par un expert.

Un *agent-modèle* participatif permet à un acteur humain de s'immerger dans la simulation. Sa place dans le système est la même que celle des autres *agents-modèles* si ce n'est que sa rationalité n'est pas modélisée. L'*agent-modèle* participatif est composé de *propriétés* qui lui permettent d'interagir avec le reste du système en influençant des *artefacts* ou de les observer (avec l'aide d'outils de visualisation). A chaque pas de temps, l'humain analyse ses *propriétés* et les modifie.

Enfin, remarquons qu'il est alors facile de changer le comportement d'un agent, c'est-à-dire son modèle, sans impact sur le reste du système. Un agent peut même être remplacé par

un autre agent en cours de simulation. Pour faire cela, il suffit que le premier agent arrête d'écrire dans les *artefacts* qu'il doit réaliser à une date précise et que le second agent commence à écrire dans ces mêmes *artefacts* au pas de temps suivant. Il est alors très simple d'intégrer et d'expérimenter différents modèles et il est possible de penser avec plus de souplesse la simulation.

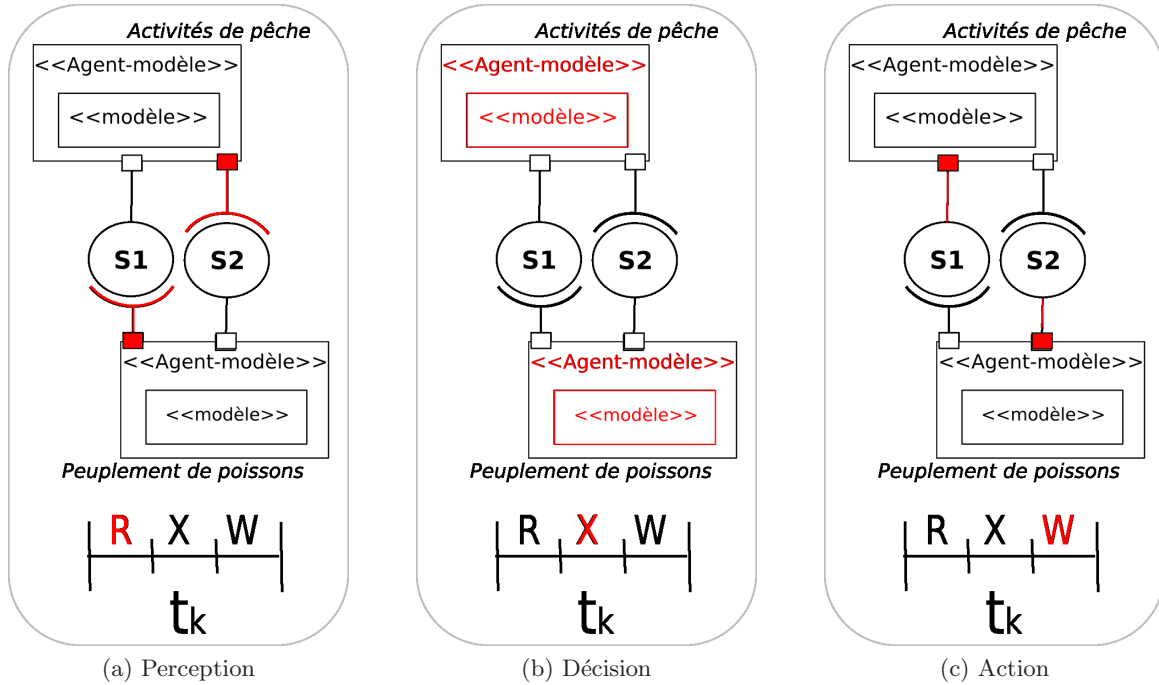
4.2.3 I–nteraction : notion de *jonction*

4.2.3 - A Synchronisation et autonomie comportementale

Dans notre société de modèles, les *agents-modèles* permettent l'émergence du modèle global *via* des interactions indirectes, médiées par l'environnement, ce qui permet l'hétérogénéité de modélisation. Les modèles des différents phénomènes, qui forment la rationalité des *agents-modèles*, doivent cependant fonctionner ensemble. En effet, le problème est de pouvoir garantir la cohérence des modèles et le respect du temps simulé et des dates de production des données. Il faut par conséquent initialiser les *artefacts* et synchroniser les agents. Pour ce faire, il est nécessaire avant tout que les agents mettent à jour l'estampille temporelle des *artefacts* qu'ils modifient. Cela se traduit concrètement par les *propriétés* des agents influençant des *artefact* qui ont accès à la date courante d'une part et qui l'écrivent dans l'*artefact* d'autre part. Concernant l'initialisation des *artefacts*, nous proposons que ce soit les agents eux-mêmes qui initialisent les *artefacts* qu'ils écrivent. En effet, pour toute activité de modélisation, il est indispensable de préciser et de maîtriser l'état initial du modèle et il n'y a par ailleurs aucune raison pour que ce ne soit pas les agents qui effectuent cette initialisation, l'état initial du modèle d'un agent étant associé au modèle lui-même.

Pour la question de la synchronisation des agents, nous proposons d'utiliser des activations désolidarisées des différents éléments d'un agent. Pour être plus précis, un agent est constitué de *propriétés* dont certaines peuvent influencer un *artefact* ou encore être influencées par un *artefact*. Cette influence est indépendante de l'activité de l'agent. On parlera donc, pour un agent, de *jonction* entrante ou de *jonction* sortante pour nommer respectivement les *propriétés* d'entrées et les *propriétés* de sorties de l'agent. Il est alors nécessaire d'ordonner l'activation de ces différentes jonctions qui ont chacune un rôle. L'ordre d'activation pour un pas de temps donné correspond à la boucle d'exécution **R-X-W** : 1° **Read** – activation des *jonctions* entrantes (mise à jour des représentations internes des agents) ; 2° **eXecute** – activation des agents ; 3° **Write** – activation des *jonctions* sortantes (mise à jour des *artefacts*). C'est grâce à cette activation des différents éléments du système que l'on peut synchroniser les agents par rapport à leur environnement.

Nous proposons de nous intéresser à un exemple de système d'agents, qui nous permet d'exhiber la question de l'initialisation du système en même temps que notre solution de synchronisation. La figure 4.7 présente une illustration de la boucle d'activation **R-X-W** avec

FIG. 4.7: Illustration de la boucle d'exécution **Read-eXecute-Write**.

un exemple de système d'*agents-modèles* qui comprend deux *agents-modèles* en interaction via deux *artefacts* S1 et S2. Remarquons alors que les deux *agents-modèles* sont dans une boucle de dépendance, c'est-à-dire qu'ils dépendent chacun d'un *artefact* écrit par l'autre. Cette boucle de dépendance temporelle pose avant tout la question de l'initialisation du système. En effet, dans l'exemple que nous présentons, si les *artefacts* ne sont pas initialisés, chaque *agent-modèle* attend que l'autre écrive dans son *artefact* (et que son estampille temporelle soit mise à jour : $t=0$) pour pouvoir calculer son prochain état. Ils sont alors en situation d'interblocage.

Comme nous pouvons le voir sur la figure 4.8, l'initialisation des *artefacts* solutionne ce problème. Les *artefacts* sont initialisés avec N_0 et M_0 et leur estampille temporelle est mise à $t = 0$. La simulation peut alors commencer et, pour chaque pas de temps, la boucle **R-X-W** est mise en œuvre. On a ainsi d'abord les *jonctions* entrantes qui sont mises à jour en lisant les *artefacts* auxquelles elles sont attachées (figure 4.8b). Ensuite, les agents s'exécutent et ils peuvent le faire car leur *propriétés* entrantes sont à jour (figure 4.8c). Sur la figure, les agents calculent respectivement N_1 et M_1 et modifient uniquement leurs *jonctions* sortantes. Et enfin, les *jonctions* sortantes sont activées et mettent à jour les *artefacts* en modifiant également leur estampille temporelle qui passe à $t = 1$ (figure 4.8d). Remarquons enfin que si les agents sont synchronisés au premier pas de temps, il est facile de voir qu'ils le seront alors à tous les pas de temps. De plus, si notre exemple fonctionne avec une boucle de dépendance

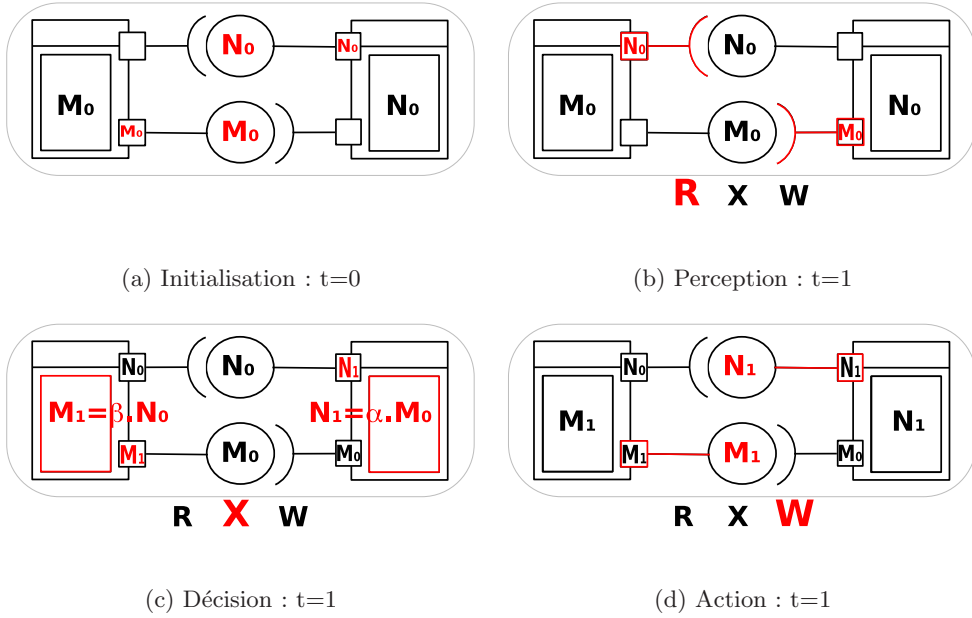


FIG. 4.8: Détails d'une boucle de dépendance entre deux *agents-modèles*— initialisation et premier pas de temps (exhibition de la boucle d'exécution **R-X-W**).

minimale, donc contenant seulement deux *agents-modèles*, le cas d'une boucle contenant plus de deux agents fonctionne.

En conclusion, nous proposons de synchroniser les agents via l'environnement. Cette synchronisation est garantie par la boucle d'exécution **R-X-W** des *jonctions* entrantes, des agents et des *jonctions* sortantes. On peut se demander par contre si un modèle asynchrone ne permettrait pas d'obtenir des résultats équivalents à un modèle synchrone. Nous constatons cependant qu'une telle solution doit être étudiée et sa convergence vers la solution synchrone prouvée. De plus, il est de notre responsabilité de garantir la cohérence du modèle. Par exemple, nous pouvons voir que pour des *agents-modèles* de type extensionnels jouent correctement à des dates précises des dynamiques observées dans un système réel et que celles-ci influencent au bon moment les autres agents, il est nécessaire de maîtriser les dates de production des données et de synchroniser ses agents.

Nous remarquons enfin que notre proposition permet aux agents d'être autonomes d'un point de vue comportemental. C'est-à-dire que l'exécution du modèle est indépendante des activités de perception et d'action de l'agent. En effet, l'activité des *jonctions* est désolidarisée de celles des agents et un agent peut à la limite évoluer sans que ses *jonctions* aient d'activité. Il n'y aurait donc pas de modification de ses *jonctions* entrantes. L'intérêt est que l'agent est fortement détaché du reste du système et que le système ne repose sur la présence d'aucun agent en particulier.

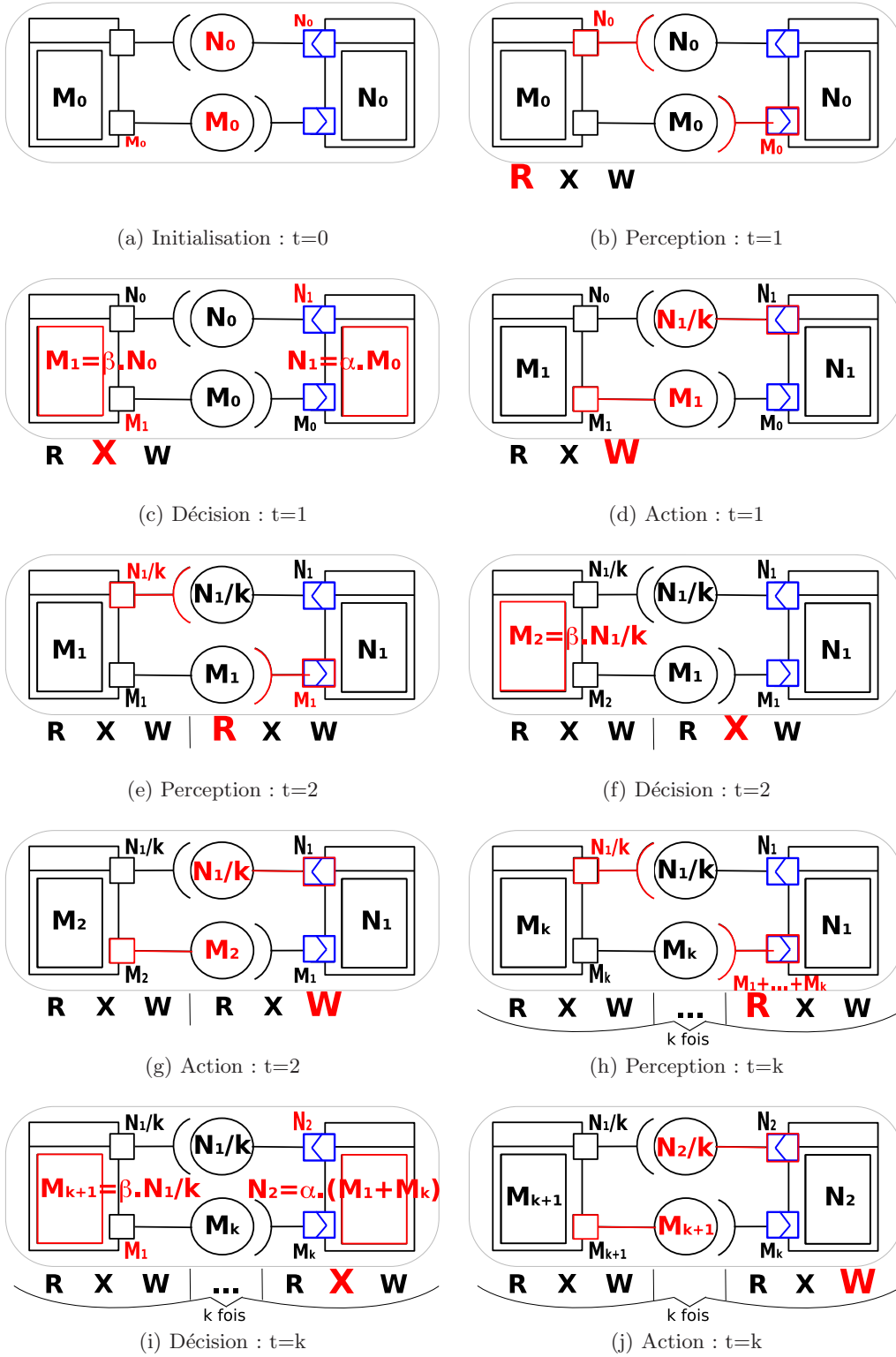
4.2.3 - B Activités des agents et rythmes d'évolution hétérogènes

Dans la section précédente, nous avons présenté un exemple simple illustrant la synchronisation entre deux *agents-modèles* évoluant au même rythme. Cet exemple nous a permis de présenter une boucle de dépendance et d'éviter une situation d'interblocage grâce à la phase d'initialisation des *artefacts* par les agents et de montrer la synchronisation de deux agents grâce à la boucle d'exécution **R-X-W**. Nous voulons montrer dans cette partie que notre solution fonctionne également quand les deux *agents-modèles* n'évoluent pas au même rythme.

Nous proposons pour cela que toute *propriété* puisse avoir son propre rythme d'exécution. Les *propriétés* peuvent de ce fait être associées à une ou plusieurs activités. Une activité est caractérisée par une date de début d'exécution, une date de fin d'exécution et un pas d'exécution. Par ailleurs, ce pas d'exécution peut être variable, mais il est au minimum égal au pas de temps minimal de l'ordonnanceur. L'agent est donc autonome au travers de ces *propriétés* qui peuvent mettre en œuvre toute sorte d'heuristiques et c'est la somme des activités des *propriétés* qui forment le comportement de l'agent. Remarquons enfin que la possibilité d'associer autant d'activités que l'on veut à un agent (via ses *propriétés*) permet également de ne lui en associer qu'une seule. Cependant, cela permet une grande souplesse dans la construction du modèle de l'agent.

Pour aborder ce problème, il manque cependant une dernière propriété à notre système d'agents, la possibilité pour tout objet actif d'avoir son propre rythme d'exécution. Comme nous l'avons vu les agents sont constitués de *propriétés* qui peuvent être des *jonctions*. Le modèle que l'*agent-modèle* met en œuvre peut également être composé d'une ou plusieurs *propriétés*. De plus, les agents peuvent également être composés de comportements qui sont très proches des *propriétés* sauf qu'ils ne portent pas de données. Ces *propriétés* et comportements peuvent alors être actifs ou non, c'est-à-dire associés à une activité ou non. De plus, toute activité est caractérisée par une date de début d'exécution, une date de fin d'exécution et un pas d'exécution. Par ailleurs, ce pas d'exécution peut être variable mais il est au minimum égal au pas de temps minimal de l'ordonnanceur. L'agent est donc autonome au travers de ces *propriétés* et comportements actifs qui peuvent mettre en œuvre toute sorte d'heuristiques et c'est la somme des comportement et *propriétés* actifs qui forment le comportement de l'agent. Remarquons tout de même avant de passer à la suite que la possibilité de mettre autant d'activités que l'on souhaite dans un agent permet également de n'en mettre qu'une (à l'exception des activités attachées aux *jonctions*), mais elle permet également une grande souplesse dans la construction du modèle de l'agent.

Nous proposons maintenant d'étudier un exemple abstrait afin de valider notre proposition de synchronisation des agents ayant des rythmes d'activation hétérogènes. Reprenons l'exemple de la section 4.2.3 - A dans lequel on considérait deux *agents-modèles* qui étaient dans une boucle de dépendance, chacun lisant l'*artefact* produit par l'autre. La figure 4.9 présente l'exécution des agents pour les pas de temps 1, 2 et k. Notons que là encore la phase d'ini-


 FIG. 4.9: Illustration du fonctionnement d'une *jonction*

tialisation permet d'éviter une situation de *dead lock* (figure 4.9a). Cette fois-ci par contre, l'*agent-modèle* de gauche évolue au rythme le plus élevé, c'est-à-dire tous les pas de temps, alors que celui de droite évolue k fois plus lentement, donc tous les k pas de temps. Ce dernier s'exécute de plus au premier pas de temps, et ensuite au $(k + 1)^{eme}$ pas de temps. Enfin, pour pouvoir interagir, les agents doivent pouvoir lire l'*artefact* produit par l'autre et il est nécessaire à l'un des deux ou les deux d'utiliser une ou des *opérations*. Nous avons choisi de fixer la forme des *artefacts* à la discrétisation temporelle de l'agent le plus rapide, celui de gauche. C'est donc l'autre agent, à droite, qui doit transformer aussi bien ce qu'il perçoit que ce qu'il écrit.

Les *jonctions* de l'*agent-modèle* de droite s'activent au rythme des *artefacts* avec lesquels ils sont liés, donc à chaque pas de temps. L'*agent-modèle* utilise par conséquent deux *opérations* de discrétisation modifiant la discrétisation de l'axe temporel. Il suffit de transformer l'unité de temps u_1 de l'*artefact* en l'unité de la *propriété* qui est $u_2 = ku_1$. Il faut donc que l'activité de la *propriété* aille au rythme temporel u_1 mais que les données de la *propriété* soient exprimées en u_2 . Tant que u_2 est multiple de u_1 , il n'y a donc aucun problème pour faire la conversion.

Dans cet exemple, à chaque pas de temps, la *propriété* entrante de l'*agent-modèle* lit l'*artefact* et s'ajoute la valeur lue à elle-même. On voit dans la figure 4.9h que la valeur de la *propriété* correspond au pas de temps $t = k$ à $(M_1 + \dots + M_k)$. Tous les k pas de temps, l'agent vient alors lire sa *propriété* entrante. Celle-ci doit ensuite également se mettre à zéro tous les k pas de temps après avoir été lue par l'agent.

C'est l'inverse pour la *propriété* sortante. A chaque pas de temps, la *propriété* sortante produit dans l'*artefact* auquel elle est liée sa valeur divisée par k comme on peut le voir au pas de temps $t = 1$ (figure 4.9d), au pas de temps $t = 2$ (figure 4.9g) et au pas de temps $t = k$ (figure 4.9j). L'agent quant à lui peut venir modifier la *propriété* tous les k pas de temps sans en troubler l'exécution.

Pour conclure, nous avons introduit d'abord une nouvelle propriété du système qui est la capacité pour toute activité d'aller à un rythme variable qui doit cependant être multiple du pas de temps minimal du système. De plus, nous avons montré sur un cas abstrait que nos agents étaient bien synchronisés par rapport à leur environnement et ce quelque soit leur rythme d'évolution. Il faut cependant bien évidemment qu'ils soient capables de fournir les services pour lesquels ils ont été construits, c'est-à-dire l'écriture des *artefacts* qu'ils réalisent et ils peuvent pour le faire utiliser des *opérations*.

4.2.4 O-rganisation d'une société de modèles

La question de l'organisation d'une société de modèles est celle des rôles des agents. L'enjeu est d'identifier les différents rôles impliqués au cours des processus de modélisation et

de simulation. Ces rôles s'expriment en termes de contrôle et d'observation locale ou globale sur le modèle, de manipulation et de construction de modèle, d'exploration de solutions, etc. Nous identifions ainsi trois rôles des agents en modélisation : 1° un *agent-modèle* manipule un modèle dont le résultat est une modification d'*artefacts* ; 2° un *agent-observateur* observe des *artefacts* et produit des observables ; 3° un *agent-contrôleur* observe des *artefacts*, les compare et produit des messages pour les modélisateurs et plus généralement, met en œuvre des heuristiques de contrôle sur les *artefacts*. Nous ne revenons pas sur le rôle de l'*agent-modèle* qui permettent de mettre en œuvre les modèles des phénomènes que l'on étudie et dont nous venons de présenter tous les aspects.

Le rôle des agents-observateurs est de construire des observables. Ajoutés au cadre de modélisation, ils permettent d'extraire des *artefacts* les dynamiques que les modélisateurs désignent comme pertinentes. Il s'agit donc pour eux de pouvoir 1° extraire ces données avec un rythme propre, 2° effectuer des manipulations sur les données et 3° construire des observables sous forme de fichiers, de graphes ou au moyen d'applications extérieures. Les agents-observateurs peuvent également utiliser des opérations afin de transformer les données des *artefacts* et observer plusieurs *artefacts* en même temps afin de mettre des données différentes en relation. Le rôle des agents-observateurs est d'abord de permettre la construction des patterns et l'observation du système. Ils traduisent explicitement dans le modèle l'activité d'observation, de construction des patterns et de production des données qui pourront être ensuite analysées. La description de l'observable est faite au moyen de la sémantique et de la forme voulue. Pour construire des observables complexes, il est possible de mettre en œuvre un comportement spécifique de l'agent.

Le rôle des agents-contrôleurs est de mettre en place des contrôles automatiques locaux sur le modèle. Pour faire cela, ils mettent en œuvre des heuristiques particulières qui définissent un contrôle sur les dynamiques d'*artefacts* et donc indirectement sur les *agents-modèles* qui les produisent. Les agents-contrôleurs représentent un méta-contrôle sur le modèle en cours de simulation. En effet, ils doivent relayer et mettre en œuvre des heuristiques qui ne font pas partie du modèle thématique et de son implémentation, mais qui le caractérise et permette de l'introspecter. Il faut donc pouvoir construire ces heuristiques. Par exemple, l'extinction d'un peuplement dans un modèle donné en cours de simulation peut représenter un comportement aberrant du système et il n'est pas alors nécessaire de continuer la simulation. Un agent-contrôleur peut mettre en œuvre ce genre de contrôle et cela permet d'invalidier un jeu de paramètres, des conditions initiales et les résultats de simulation obtenus. L'agent-contrôleur peut également comparer les *artefacts* produits par deux *agents-modèles* qui n'ont pas le même type de modèle. On peut imaginer que si l'un manipule un modèle extensionnel reproduisant des données réelles, la fonction de l'agent-contrôleur peut alors être d'arrêter la simulation en cas de divergence. Remarquons que les agents-contrôleurs prennent leur place dans la société de modèles en phase de simulation et servent alors de relais d'information et de contrôle.

Enfin, la figure 4.10 présente un exemple de système d'agents dans lequel on peut voir

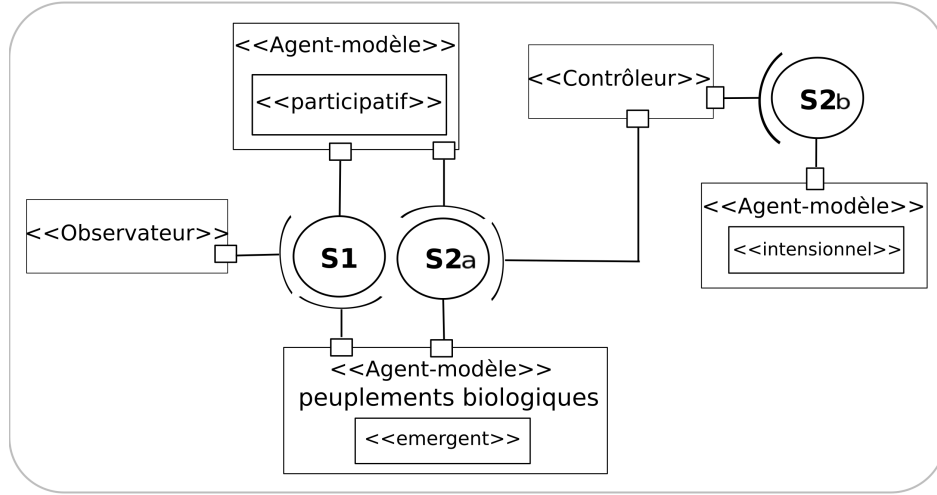


FIG. 4.10: Notation agent – exemple de système d’agents

trois *agents-modèles*, dont deux qui sont en interaction directe. Il y a également un agent-observateur et un agent-contrôleur qui compare les dynamiques des *artefacts* S2a et S2b.

4.2.5 U–tilisateur : modélisateur ou *agent-modèle* ?

Nous abordons enfin la question de l’utilisateur qui peut avoir deux rôles dans notre société d’agents. L’utilisateur peut soit être un modélisateur, soit jouer un *agent-modèle*, on parle alors de simulation participative. Quand l’utilisateur est modélisateur, il construit le modèle, éventuellement avec d’autres modélisateurs, et le modèle n’est donc pas en train d’être exécuté. Quand l’utilisateur joue un *agent-modèle*, qui est alors de type participatif, le modèle est en train d’être simulé. Donc intégrer ”l’humain dans la boucle” se fait pour nous à au moins deux niveaux. Quand il s’agit de construire le modèle, il est important que l’utilisateur soit en interaction avec le modèle, mais surtout qu’il puisse se construire une représentation du système qu’il veut modéliser. L’intérêt de l’approche est donc d’abord de permettre à l’utilisateur de centrer son étude sur les données, mais également de l’obliger à les formaliser complètement. Ensuite, les agents représentent les phénomènes qui viennent produire les données et tous les modèles sont explicités de manière indépendante soit dans un *agent-modèle*, soit dans une opération. L’intégration de l’humain dans le système en tant qu’*agent-modèle* nécessite que l’utilisateur puisse percevoir son environnement afin de décider du comportement qu’il souhaite mettre en œuvre. De plus, le système attend l’utilisateur à chaque fois que c’est au tour de l’humain de prendre la main, ce qui est rendu possible par l’ordonnanceur qui fonctionne en temps virtuel, c’est-à-dire qu’un pas de temps du simulateur n’est pas fonction du temps réel.

4.3 Architecture du simulateur et modèle émergent

Dans le cadre de ce travail de thèse, nous avons construit une plate-forme de modélisation et de simulation basée sur notre proposition conceptuelle d’agent-modèle. Les objectifs de notre travail d’implémentation ont été de : 1^o construire un simulateur pour la production de résultats dans le cadre du projet Chaloupe ; 2^o rechercher des solutions aux problématiques conceptuelles, méthodologiques et computationnelles liées à la construction et à l’exécution de modèles écosystémiques. Notre plate-forme de modélisation et de simulation est donc un outil de recherche et d’expérimentation, mais également un outil opérationnel.

L’intérêt de construire ce simulateur s’exprime de la manière suivante :

«**Apprendre en faisant**». Nous avons commencé par modéliser des écosystèmes exploités en cherchant à identifier comment abstraire les problématiques et pouvoir ensuite modéliser la modélisation d’écosystèmes exploités. Construire en parallèle le simulateur et notre proposition conceptuelle nous a permis de façonner un outil à notre main et de comprendre les problématiques intrinsèques et fondamentales d’un tel simulateur.

«**Travailler sur la paillasse de son laboratoire**». AREVI est développé au LISYC et structure la majorité des développements du laboratoire. Plus qu’une plate-forme, elle concrétise la dynamique et les questionnements du laboratoire en une paillasse virtuelle sur laquelle on expérimente la construction de modèles.

«**Revenir aux fondamentaux**». Nous nous sommes très rapidement rendu compte qu’appréhender la complexité des systèmes nécessitait une maîtrise rigoureuse du modèle et des simulations. Implémenter un modèle complexe, hétérogène et multi-échelles, requiert d’utiliser des artifices computationnels de modélisation qu’il est indispensable d’identifier et dont l’impact doit être quantifié. Ces processus ne sont pas assez souvent étudiés en tant que tel, alors qu’ils peuvent entraîner des biais qu’il est ensuite très difficile de repérer dans le modèle complet. Ils sont liés entre autre à l’asynchronisme, la stochasticité et l’ordonnancement.

Nous ne présentons ici les éléments nécessitant un éclairage particulier, en restant à un niveau abstrait de description. Le système n’est donc pas présenté dans sa globalité. Il est cependant entièrement construit sur notre proposition conceptuelle et nos résultats de simulation permettent de le valider implicitement. Ainsi, nous détaillons ici d’abord l’architecture globale, puis nous abordons notre proposition permettant de construire des modèles émergents, donc des IBM.

4.3.1 Architecture

Pour construire notre simulateur, nous avons spécialisé la plate-forme AREVI (Harrouet 06) qui fournit de nombreuses fonctionnalités pour : la construction et la manipulation d’agents (activités, ordonnancement, communication par message, introspection...),

la construction d'environnements virtuels (rendu graphique), la manipulation de fichiers, de fenêtres, et de nombreux autres mécanismes de base.

Le simulateur est construit en trois couches :

Couche abstraite : elle contient les composants de base permettant de faire des agents abstraits ayant des *propriétés*, des comportements et des rôles. Elle fournit également des éléments de construction d'un environnement, comme l'espace (et les régions de l'espace) et le temps.

Couche de modélisation : elle contient les composants pour construire des agents pour la modélisation : *agent-modèle*, agent-observateur et agent-contrôleur. Elle définit également les composants de l'environnement de modélisation comme les *artefacts*.

Couche thématique : cette dernière couche spécifie les modèles thématiques – émergent, intensionnel et extensionnel – et tous les composants qui y sont associés comme les agents thématiques ou les environnements spécifiques. Elle contient également les modèles particuliers, que nous qualifions d'experts, parce qu'ils sont liés à une problématique (écologique ou économique) particulière, en particulier les comportements IBM des agents-individus de peuplements écologiques ou économiques et les modèles d'EDP de peuplements.

Notons qu'il est possible d'utiliser des sondes pour observer localement le modèle d'un *agent-modèle*. Ces sondes peuvent créer des observables de *propriétés* et facilitent ainsi l'observation de la dynamique interne du modèle lors de sa construction et du choix des paramètres.

4.3.2 Modèle émergent

L'objectif du modèle émergent est de permettre de construire des modèles IBM. Les objectifs de ce modèle sont : 1^o la modularité du système et la possibilité d'ajouter facilement des modèles thématiques de comportements individuels; 2^o la capacité à mettre en œuvre différentes solutions IBM, basées individu ou orientées individu; 3^o la capacité à construire des modèles intégrant des processus à différentes échelles et niveaux de description.

La première problématique a été d'intégrer au mieux le modèle IBM en tant que phénomène dans l'environnement de modélisation des *agents-modèles*. Nous abordons donc dans un premier temps la question de l'environnement, que nous appelons phénoménologique, des agents-individus. Nous présentons ensuite les agents-individus, qui implémentent des individus conceptuels, et la question de leur exécution au sein du SMA. Enfin, nous exposons nos solutions pour pouvoir considérer différents niveaux d'abstraction, en particulier celui de l'espèce ou de la population et plus largement du groupe.

4.3.2 - A Un environnement phénoménologique

Pour construire un IBM, il est nécessaire de pouvoir définir un environnement. L'environnement phénoménologique contient ainsi l'ensemble des agents-individus, qui peuvent être des groupes, desquels émergent le phénomène global à modéliser. L'environnement peut être discrétisé suivant la distribution des agents-individus d'une part et celle des caractéristiques de l'environnement d'autre part. Ainsi, la distribution des agents peut être distribuée suivant une discrétisation différente de celle des caractéristiques de l'environnement.

D'abord, l'environnement peut être associé à une grille spécifique ou non qui permet de mailler l'environnement avec une précision donnée afin de maintenir des informations de distributions des individus en fonction de leur espèce ou d'un groupe donné auquel ils appartiennent si la grille est spécifique (si elle ne l'est pas, les informations ne dépendent pas du groupe/espèce). Ces distributions peuvent être en nombre, en biomasse, en densité (de nombre ou de biomasse) et peuvent prendre en compte le fait que des agents-individus représentent plusieurs individus au travers d'un facteur d'échelle (voir 4.3.2 - C). Le maillage peut également être associée à une activité qui lui permet de maintenir à jour lui-même les informations de distributions et de densité. Par contre, l'utilisation de cette activité synchronise alors les agents-individus par rapport aux autres agents-individus, car la mise à jour des informations n'est effectuée qu'une fois par pas de temps du système (ou un multiple de celui-ci).

Le deuxième type de discrétisation se fait au moyen d'habitats. Ceux-ci sont décrits ci-après, dans la section 4.3.2 - D, mais de manière générale ils permettent de décrire des caractéristiques distribuées de manière différente pour chaque caractéristique. Ces habitats permettent ainsi d'ajouter des informations dans l'environnement suivant une description qui est propre à chaque information.

4.3.2 - B Agents-individus

Description de l'agent-individu

Les agents-individus permettent d'implémenter des individus de l'IBM au sens large. Ainsi, un agent-individu est caractérisé par un facteur d'échelle qui lui permet de représenter un nombre d'individus supérieur à un (voir section 4.3.2 - C). Ces agents-individus sont constitués, tout comme les agents de l'environnement de modélisation, par des *propriétés* et des comportements. Chacun de ces comportements et *propriétés* peut être actifs et mettre en œuvre un modèle individuel particulier. Une *propriété* a un rôle au sein du modèle qui peut être celui de variable, de paramètre ou de constante et l'*agent-modèle* a toujours accès à ces éléments au travers essentiellement du groupe auquel appartient l'agent-individu.

L'agent-individu est caractérisé fondamentalement par un âge, une date de naissance et

une masse. En effet, le temps de la simulation ayant une unité, l'agent-individu peut avoir un comportement de vieillesse qui incrémente son âge afin de reproduire le vieillissement naturel. La date de naissance peut être requise suivant les modèles et afin de prendre en compte des processus de maturité ou des changements de groupes liés à l'âge. Enfin, la masse d'un agent-individu correspond conceptuellement à son "poids" dans son environnement. Pour un poisson par exemple, c'est exactement sa biomasse et elle est le plus souvent en relation avec sa taille. Pour un pêcheur ou plus largement un acteur économique, c'est son "poids" économique.

Activation des agents-individus

Abordons maintenant la question de l'activation, de la mortalité et de la synchronisation des agents-individus. Dans un agent-individu, ce qui fait sa rationalité, c'est l'ensemble de ces *propriétés* et comportements actifs. Le modèle du comportement individuel est donc mis en œuvre grâce à un ensemble de comportements et de *propriétés* actifs. Donc, exécuter le SMA, c'est activer de manière chaotique asynchrone l'ensemble des *propriétés* et des comportements de l'agent. Remarquons qu'il est évidemment possible de mettre en œuvre une seule activité par agent-individu, car les comportements et *propriétés* ne sont pas forcément actifs d'une part, et un comportement ou une *propriété* peut exécuter toute autre comportement ou *propriété*.

Se pose cependant la question de la synchronicité des comportements et *propriétés* actifs pour un agent donné. En effet, il est important de pouvoir garantir que pour un agent donné tous les comportements et *propriétés* vont s'activer autant de fois les uns que les autres s'ils ont le même pas d'activation. En d'autres termes, si un agent-individu est constitué de plusieurs activités et, qu'à un pas de temps donné, une de ces activités provoque la mort de l'agent-individu, il faut que le système garantisse que les autres activités de l'agent-individu, devant s'activer à ce pas de temps là et ne l'ayant pas fait, seront effectivement activées. Ce mécanisme permet alors de garantir la synchronicité des comportements et *propriétés* de chaque agent-individu. Ainsi, la mort de l'agent-individu entraîne l'arrêt de tous les comportements, mais pas forcément de manière instantanée (l'arrêt peut donc prendre au moins un pas de temps pour être effectif). On peut toutefois ne pas mettre en œuvre cette synchronicité.

Mécanismes d'optimisation

Enfin, nous utilisons deux mécanismes qui permettent de faciliter et d'optimiser la construction d'un modèle émergent : 1^o l'état initial et 2^o le clonage. Le premier permet de construire facilement l'état initial d'un *propriété*. Ce mécanisme est valable par ailleurs pour toutes les *propriétés* du système (donc également celles des agents de l'environnement de modélisation : *agent-modèle*, etc.). Il permet ainsi d'initialiser les données portées par une *propriété* avec toutes sortes de mécanismes (chargement de fichier, initialisation aléatoire ou

en fonction de distribution de probabilités, copie d'une autre *propriété*, etc.).

Ensuite, le clonage permet d'optimiser la création d'agents-individus. En effet, nous décrivons nos modèles dans des fichiers de configuration XML. La configuration est ensuite chargée dans un objet, mais toutes les informations restent sous un format qui ne permet pas de créer rapidement un agent-individu. Pour répondre à ce problème, un agent-individu de chaque population est créé au chargement du fichier de configuration. Il est ensuite mis de côté et cloné à chaque création d'un nouvel individu, ce qui permet d'utiliser un maximum le constructeur par copie et de limiter le temps de création.

4.3.2 - C Organisations des individus : peuplement et groupe

Il est nécessaire avec les modèles IBM de peuplements de pouvoir manipuler des échelles d'abstraction différentes. En effet, on a souvent besoin d'acquérir des informations au niveau de l'individu, d'un groupe d'individus suivant leur localisation ou une autre caractéristique individuelle et de l'espèce. Nous définissons donc deux niveaux d'abstraction : 1° le peuplement et 2° le groupe.

Le peuplement regroupe tous les agents-individus du SMA quelque soit leur espèce et centralise un certain nombre d'algorithmes statistiques qui peuvent être spécifiques (dépendant de l'espèce/groupe) ou non. Il permet également d'accéder de différentes manières aux agents-individus, en utilisant une sélection aléatoire par exemple.

Le groupe définit le type de l'agent-individu. C'est par exemple, dans le cas d'une forêt, les pins, ou les chênes âgés de plus de 50 ans, et dans le cas d'une pêcherie, les pêcheurs ou les navires de pêche qui sont dans une zone géographique donnée. Souvent cependant, le groupe est synonyme d'espèce et permet de définir toute une série de caractéristiques liées à l'espèce et auxquelles peuvent accéder les agents-individus du groupe. Ce niveau d'abstraction permet plusieurs choses. D'abord, il permet de décrire l'état initial d'un ensemble d'agents-individus au début de la simulation. En effet, autant un agent-individu peut avoir un état initial qui correspond à son état quand il naît, autant au démarrage de la population, il peut être nécessaire d'avoir des agents-individus d'âges différents ou distribués dans l'environnement de manière particulière. Le groupe permet de faire cela en redéfinissant l'état initial de l'agent-individu à sa création¹. Il permet également de définir des *propriétés* ou des comportements de groupe afin par exemple de calculer une caractéristique globale à ce type d'agents-individus. Cette *propriété* peut alors modifier directement un *artefact* afin d'externaliser l'information qu'elle porte. Enfin, le groupe définit le nombre d'agents-individus qu'il est nécessaire de construire à l'initialisation².

¹ Ceci est basé sur le mécanisme d'initialisation des *propriétés* de l'agent et le mécanisme de clonage.

² On peut préciser un nombre relatif au nombre total d'agents-individus possible et définit au niveau du peuplement, ou un nombre absolu d'agents-individus par groupe.

4.3.2 - D Habitats

Les habitats sont une extension des *propriétés* et sont associés à l'environnement phénoménologique. Les habitats peuvent être discrétisés de toutes les manières possibles dans un environnement donné. Ils représentent des lieux dans l'environnement, c'est-à-dire qu'ils permettent d'ajouter des informations dans l'environnement. L'accès à cette information est ensuite faite par les agents-individus au moyen d'une position dans l'espace donc indépendamment de la discrétisation. Les habitats peuvent également produire ou lire d'autres *propriétés* (ou habitats), et en particulier ceux de l'*agent-modèle* émergent. Ainsi, il est possible d'intégrer des informations provenant d'*artefacts* dans l'environnement des agents-individus au moyen des habitats.

4.4 Méthodologie

L'enjeu d'une méthodologie est de pouvoir organiser et rationaliser les différentes activités de modélisation et de simulation lors de la construction et de l'expérimentation d'un modèle. Nous commençons par présenter notre proposition méthodologique de manière générale, avant de rentrer dans les détails des trois phases principales que nous avons identifiées lors de l'étude d'un modèle : 1^o identification de la question de modélisation et de la construction du cadre de modélisation ; 2^o modélisation ; et 3^o simulation.

4.4.1 Méthode basée POM

Notre méthodologie de modélisation et de simulation est basée sur POM et permet de mettre en œuvre notre proposition conceptuelle. De plus, notre simulateur contraint le modélisateur à la suivre. Nous en présentons maintenant les différentes phases (voir également figure 4.11), en explicitant leur fonction :

1. *Choix et explicitation d'une question de modélisation*
2. *Construction du cadre de modélisation correspondant à la question : formalisation du problème.*

Les questions posées sont : «Quels sont les patterns identifiant mon problème?»
«Comment rend-on compte des patterns?»

- (a) Identification des patterns.
- (b) Description formelle des *artefacts* dans lesquels pourront être observés les patterns.
- (c) Définition du cadre théorique pouvant expliquer les patterns : ajout des agents-modèles au cadre de modélisation et déploiement des *jonctions* les liant aux *artefacts*.

- (d) Distribution de l'observation / contrôle afin de construire les patterns et de poser des contrôles de méta-modélisation : ajout des agents observateurs et contrôleurs dans le cadre de modélisation

3. Phase de modélisation

La question fondamentale ici est : "Comment expliquer les patterns?" Il s'agit alors de :

- (a) Modélisation des phénomènes : Construction des modèles thématiques.
- (b) Construction des modèles conceptuels issus des modèles thématiques.
- (c) Construction des modèles computationnels issus des modèles conceptuels : Implémentation des modèles computationnels dans les *agents-modèles*.

Cette étape permet la construction des mécanismes internes du modèle global et l'implémentation des modèles des phénomènes qui sont censés expliquer l'émergence des patterns.

4. Choix des paramètres

Il faut à cette étape 1° choisir les paramètres, donc explorer le modèle, et 2° expérimenter et valider au cours de cette exploration le modèle construit.

- (a) Choix d'un jeu de paramètres pour le modèle de chaque *agent-modèle*.
- (b) Expérimentation du modèle global par simulation interactive. Expérimentation locale de certains *agents-modèles* par construction incrémentale du cadre de modélisation.

Pour la deuxième étape, l'expérimentation, il s'agit de pouvoir expérimenter directement le modèle global, ou de pouvoir le construire de manière incrémentale en expérimentant et en validant les *agents-modèles* individuellement par exemple avant de les ajouter dans le système global. L'expérimentation est de plus interactive, dans le sens où l'on veut visualiser dynamiquement l'évolution du système. Il n'y a donc pas de production de résultats. Il ne s'agit pas non plus de revenir sur le cadre de modélisation.

5. Phase de simulation

Production de résultats, simulations non interactives.

4.4.2 Question et cadre de modélisation

Au préalable d'une action de modélisation, il y a un cadre théorique. Ce cadre théorique identifie un objet d'étude et des hypothèses sur cet objet d'étude. Il s'agit ensuite de concevoir un ensemble d'expérimentations pour valider (ou réfuter) les hypothèses, ce qui conduit à poser une question de modélisation. Cette question de modélisation est fondamentale parce qu'elle structure les patterns d'abord et le cadre de modélisation ensuite. Elle pose les conditions dans lesquelles sont étudiées le système, permet d'identifier le phénomène (au sens large) qui est à l'étude et détermine ce qui va être observé.

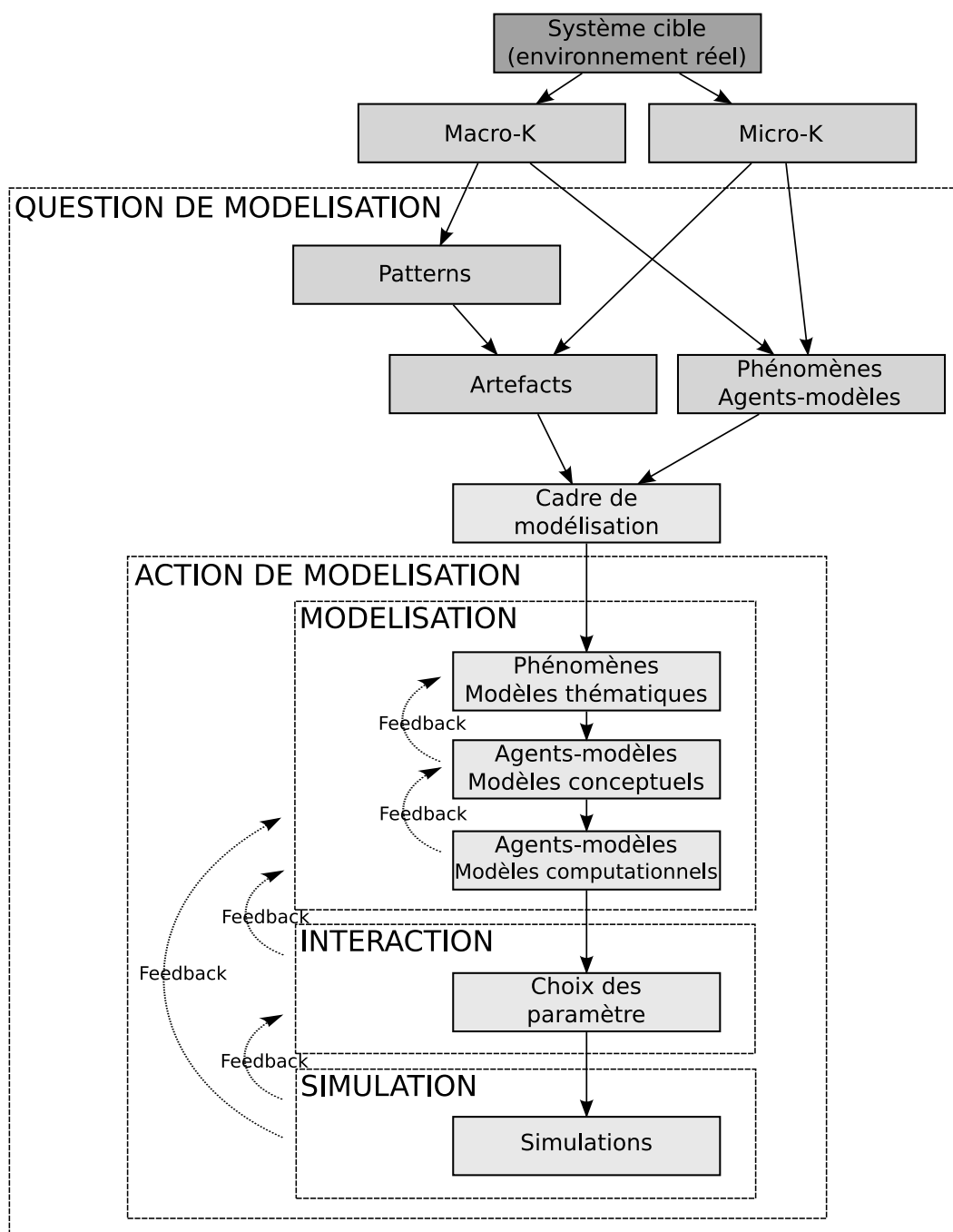


FIG. 4.11: Proposition méthodologique pour une modélisation et une simulation basées POM et *agent-modèle*.

Ensuite, c'est à partir de la question de modélisation que les thématiciens identifient les patterns qui sont extraits des données et qui vont rationaliser tout le processus de modélisation et de simulation. Les patterns identifiés, les thématiciens avec les modélisateurs déterminent 1° les artefacts dans lesquels seront observés les patterns et 2° les phénomènes qui peuvent expliquer ces patterns. Il s'agit d'identifier les phénomènes sans présupposer de leur fonctionnement ou de leur nature, mais simplement en fixant leur fonction ou rôle au sein du modèle global. On est dans une démarche systémique et pragmatique et on cherche d'abord à décrire le système dans sa globalité en exhibant les composantes et les interactions entre composantes qui peuvent expliquer les dynamiques du système global. On est de plus dans une approche centrée sur les données. On identifie donc avant tout les *artefacts* qui doivent être décrits formellement et permettre l'observation ultérieure des patterns. On exhibe ensuite les composantes encapsulant les phénomènes au moyens des *agents-modèles* qui explicitent et formalisent le rôle des phénomènes. Et on matérialise enfin les interactions entre phénomènes au moyen des *jonctions* entre *agents-modèles* et *artefacts* et qui définissent ainsi les services rendus et requis par les *agents-modèles*.

Une fois la structure thématique du cadre de modélisation posée, c'est-à-dire les *artefacts* et *agents-modèles* définis, il reste à identifier les actions d'observation, d'analyse et de contrôle que l'on veut appliquer sur le système. Ces actions se traduisent par l'ajout d'agents-observateurs et d'agents-contrôleurs. Définir les observables et les contrôles de haut niveau avant de construire les modèles thématiques participe à la démarche globale : on pose une question et on essaie de décrire le système et la manière dont on regarde le système. De plus, le cadre complet ainsi déterminé, toutes les simulations qui seront effectuées dans les phases suivantes auront les mêmes observables et contrôles.

4.4.3 Phase de modélisation

La phase de modélisation se partage en deux processus distincts : 1° l'implémentation des agents-modèles et 2° le choix des paramètres. Ce processus peut nécessiter plusieurs itérations et des expérimentations interactives du modèle global ou d'*agents-modèles* en particulier. C'est donc un processus qui se traduit par une activité progressive, voire incrémentale, de construction et de manipulation locales ou globales du modèle. Les thématiciens, modélisateurs et informaticiens doivent pouvoir expérimenter le modèle ou une partie du modèle pour le valider, le dimensionner et fixer les paramètres.

La phase de modélisation doit être une expérience durant laquelle la construction du modèle est interactive. Ce sont par exemple tel ou tel agent qui signale qu'il ne peut pas fonctionner à cause de tel ou tel problème dans la structure du modèle. Les agents doivent pouvoir se construire, c'est-à-dire construire leur modèle, et vérifier qu'ils sont capables de fournir les bons services et qu'ils peuvent accéder à tous les services dont ils ont besoin. Ce contrôle passe par une vérification syntaxique (contrôle de la sémantique et de la forme), mais

également structurelle et il s'agit pour le système de faciliter la construction du modèle.

En phase de modélisation, il est possible d'exécuter partiellement ou complètement le modèle tout en interagissant avec lui. Il s'agit de pouvoir expérimenter le modèle, de pouvoir changer le modèle d'un *agent-modèle* et observer immédiatement le résultat de ces changements afin de valider ou non tel ou tel aspect structurel du système. Il s'agit également de pouvoir tester indépendamment un *agent-modèle* en ajoutant les *agents-modèles* de type extensionnel permettant de lui fournir les *artefacts* en entrée dont il a besoin. En effet, cette construction incrémentale ne peut se faire qu'à l'intérieur du cadre de modélisation. On peut par exemple fixer le comportement de tous les *agents-modèles* avec des modèles extensionnels, pour expérimenter un à un le modèle, non extensionnel cette fois-ci, de tel ou tel *agent-modèle*. On peut alors ainsi envisager une construction incrémentale du modèle dans laquelle on ajoute petit à petit les différents modèles des *agents-modèles* que l'on aura validé préalablement. C'est également à ce moment là que l'on peut fixer ou explorer un certain nombre de comportement du modèle et choisir les paramètres.

4.4.4 Phase de simulation

La phase de simulation se distingue de la phase de modélisation. Avec notre outil, il faut même lancer un autre exécutable, donc on utilise bien un autre système. L'objectif de la phase de simulation est de lancer des séries de simulation respectant des plans de simulation. Ce qui est important, c'est que les interactions avec le système en tant qu'expérimentateur sont minimales durant cette phase, à l'exception des interactions que les modèles participatifs requièrent.

Notre point de vue est donc de distinguer les simulations expérimentales qui permettent de s'approprier le système en le regardant évoluer et qui sont effectuées durant la phase de modélisation et les simulations qui produisent des résultats et qui doivent donc être effectuées durant la phase de simulation de manière classique : 1^o simulation et production de données et 2^o traitement et analyse des données une fois la simulation terminée. Il est toutefois possible d'utiliser des agents-observateurs, pour avoir des informations sur la simulation par exemple, seulement il est préférable dans cette phase d'utiliser les agents-contrôleurs afin qu'ils contrôlent le bon fonctionnement de la simulation et qu'ils l'arrêtent éventuellement prématurément s'il y a un problème, ce qui libère des ressources système et ne produit plus de données inutiles par exemple. Enfin, toutes les simulations effectuées durant cette phase doivent être différenciées, répertoriées afin de pouvoir être retrouvées et identifiées sans aucune ambiguïté.

4.5 Conclusion

L'objectif de notre proposition est de rationaliser les processus de modélisation et de simulation pour permettre de mieux maîtriser le modèle complexe et d'envisager des solutions d'automatisation des activités de modélisation et de simulation. Cela suppose de pouvoir distinguer ce qui n'est pas formalisable, ce qui est de la responsabilité de l'expert, et ce qui est formalisable et qui peut alors être pris en charge par l'ordinateur. L'enjeu est alors de pouvoir faciliter l'activité de l'expert afin qu'il puisse se concentrer sur l'essentiel et pratiquer son expertise. La solution pour nous est alors 1° de perdre en contrôle sur le système pour gagner en maîtrise sur le modèle et 2° de distribuer ce contrôle parce qu'il ne nous est pas possible, dans l'état actuel des choses, de qualifier le modèle global (comme nous l'avons vu dans l'état de l'art). Nous nous intéressons donc pour commencer sur la question de la construction d'un modèle, activité qui conditionne tous les autres activités de simulation. Nous proposons alors de concevoir le modèle complexe comme une société de modèles en interaction.

Nous nous appuyons alors sur l'approche POM qui rationalise la modélisation autour des données. Et nous proposons également de construire notre solution à l'aide de l'approche agent et en particulier de la notion d'*agent-modèle*. Pour cela, nous reformulons la méthodologie orientée pattern en proposant de construire le modèle en trois étapes : 1° la formulation de la question de modélisation qui conditionne le cadre de modélisation ; 2° la mise en œuvre d'expérimentations du modèle global, qui peuvent être incrémentales, au sein du cadre de modélisation, et qui permettent de construire les modèles des *agents-modèles* et de les paramétrer ; 3° produire des résultats à partir d'un modèle figé du système et sans interaction avec lui. Le cadre de modélisation est construit en deux étapes : 1° identification des patterns et exhibition des *artefacts* qui permettent d'observer les patterns et 2° formulation des phénomènes qui vont pouvoir expliquer l'émergence des patterns, ce qui revient à l'identification des *agents-modèles* qui représentent ces phénomènes (on ne fait alors aucune hypothèse sur le fonctionnement et la description des phénomènes) et la formalisation de leurs interactions au moyen des *artefacts*.

Nous proposons alors d'aborder la question de l'hétérogénéité au moyen d'une solution de couplage par les données. Les *agents-modèles* nous offrent alors la possibilité d'utiliser différentes approches de modélisation, mais plus généralement de construire des modèles hétérogènes d'un point de vue de la description et des échelles utilisées. Enfin, pour garantir la cohérence du modèle global, nous synchronisons nos agents grâce aux *jonctions* qui leur permettent également d'être autonome d'un point de vue comportemental et ainsi de désolidariser le modèle d'un phénomène du modèle global tout en maîtrisant complètement son intégration. Le contrôle de la structure du modèle passe alors par la capacité des agents à percevoir et à agir sur les *artefacts* qu'ils doivent manipuler et qui concrétisent les services qu'ils utilisent et qu'ils rendent.

Enfin, nous avons proposé trois rôles d'agent dans la modélisation : 1° l'*agent-modèle*, 2° l'agent-observateur et 3° l'agent-contrôleur. Ces différents rôles nous permettent d'envisager les différents processus que nous pouvons souhaiter mettre en œuvre dans un modèle global, processus allant des modèles de phénomènes, aux mécanismes d'observation et enfin aux heuristiques de contrôle. La construction de notre société de modèles passe alors par une réflexion sur les différentes capacités des agents et sur la formalisation des connaissances que nous pouvons avoir sur le modèle global : comme le fait de pouvoir décrire ce qu'est un "bon" modèle, c'est-à-dire un modèle opérationnel, ou le fait de pouvoir identifier les différents éléments de modélisation qui vont devoir faire l'objet d'études particulières, comme les paramètres des modèles.

Chapitre 5

Biais computationnels dans les modèles complexes de peuplements

5.1 Introduction

La question de la traduction de modèles conceptuels en modèles computationnels est centrale dans le travail de modélisation d'un système complexe. En effet, l'implémentation d'un modèle dans un système informatique induit des hypothèses qu'il est nécessaire d'exhiber afin d'en étudier les influences sur les résultats. La discrétisation de l'espace et du temps fait partie de ces hypothèses. Par conséquent, nous proposons dans ce chapitre une étude expérimentale et incrémentale de modèles de peuplements d'individus, qui se traduit par un ensemble d'expériences et d'analyses de sensibilité de ces modèles.

Pour effectuer cette étude, nous analysons expérimentalement les problématiques et les limites de l'approche IBM pour des modèles de peuplements d'agents. Nous avons choisi ce type de modèles précisément, car ils sont relativement abstraits, mais requièrent de mettre en œuvre des processus de création et de destruction d'agents qui sont particulièrement sensibles aux biais que l'on veut étudier. Ainsi, le premier problème abordé est la question des comportements d'un agent et des hypothèses que l'on fait pour mettre en œuvre un agent dans une simulation. Nous nous basons pour cette étude sur des agents réactifs, qui peuvent être composés de plusieurs comportements, ce qui est plus général qu'un seul comportement par agent. Nous proposons ensuite d'étudier la sensibilité de nos modèles à la discrétisation homogène en nombre d'agents, et aux discrétisations homogènes spatiales et temporelles. Nous proposons ensuite d'étudier la sensibilité de nos modèles à la discrétisation hétérogène en nombre d'agents et aux discrétisations hétérogènes spatiales et temporelles.

Avant d'aborder ces études, nous détaillons les problématiques abordées et nous

présentons notre méthodologie. Ensuite, nous exhibons le cadre conceptuel, c'est-à-dire le modèle conceptuel de peuplements que l'on cherche à reproduire. Celui-ci est basé sur un ensemble de patterns qui permettent de décrire, sans *a priori* sur les phénomènes, les dynamiques caractéristiques que l'on cherche à reproduire avec nos modèles.

5.2 Problématique et méthode

Nous proposons d'identifier les fondamentaux computationnels et méthodologiques permettant de traduire des modèles de peuplements conceptuels en modèles computationnels. En effet, notre problématique étant la modélisation et la simulation de systèmes complexes, il est nécessaire d'identifier les biais computationnels liés à l'implémentation informatique des modèles. Il n'est donc pas question ici de statuer sur les modèles thématiques ou conceptuels, mais de réfléchir à la manière de les mettre en œuvre de manière cohérente et juste. Ce travail théorique mais fondamental s'inscrit dans la perspective de pouvoir qualifier les modèles et abstraire et formaliser ces connaissances sur les modèles. Dans notre approche, il s'agit donc de réfléchir aux limites des modèles que peuvent porter les *agents-modèles* en vue de leur possible manipulation ou contrôle automatique par les agents.

Nous avons par conséquent privilégié une étude de modèles IBM de peuplements à l'aide de SMA. En effet, nous nous intéressons aux systèmes complexes et donc aux cas d'étude qui posent des questions d'hétérogénéité de situations, de scénarios et d'échelles. Les modèles agrégés ont été largement étudiés et il existe plusieurs méthodes pour mettre en œuvre dans un simulateur de tels modèles (voir état de l'art section 2.3.4 - B). L'étude de modèles agrégés relève de plus généralement de la recherche mathématique fondamentale alors que les biais et problématiques computationnels pour l'implémentation de ces modèles sont bien connus. L'approche IBM, plus que tout autre paradigme, permet ou favorise la construction de modèles complexes (voir état de l'art section 2.3.3 - B). Nous nous sommes particulièrement intéressés aux modèles de peuplements, parce qu'ils permettent d'aborder la richesse et la complexité d'interactions entre des entités différentes, à des échelles différentes. Enfin, l'utilisation de SMA pour mettre en œuvre cette étude se justifie en tant que support naturel pour ce type de modèles IBM.

Pour chaque expérience, nous avons cherché à formuler le système minimal nous permettant de raisonner sur la problématique posée. En dehors de l'aspect analytique, nous abordons de manière incrémentale des modèles de plus en plus complexes. De plus, nous montrons comment des systèmes élémentaires peuvent avoir des comportements non attendus si l'on change légèrement les hypothèses. La maîtrise de ces systèmes élémentaires est indispensable pour envisager des systèmes plus complexes et en comprendre les dynamiques.

Nous avons bâti cette étude sur un cadre théorique spécifique issu de patterns que nous identifions et exprimé sous la forme d'un modèle conceptuel qui sera précisé au gré des

études et des besoins de chaque problématique. Ayant identifié les patterns, nous pouvons discriminer les résultats de simulation et désigner les «bons» modèles computationnels. Concrètement, à chaque problématique de modélisation, c'est-à-dire pour chaque grande section, nous précisons le cadre théorique général au sein de cette problématique. Ensuite, pour chaque expérience de simulation, nous explicitons le point particulier d'étude, le modèle computationnel utilisé, les valeurs des paramètres et les résultats obtenus. Enfin, chaque problématique de modélisation fait l'objet d'une courte analyse. La méthodologie est donc orientée pattern et pour chaque question (problématique) de modélisation, il s'agit de : 1° identifier le système minimal permettant de reproduire les patterns ; 2° déterminer les paramètres du modèle, c'est-à-dire leur valeurs ou domaine de variabilité ; 3° mettre en oeuvre le modèle et comparer ses dynamiques avec les patterns de référence qui servent alors de filtres (Wiegand 03).

Remarque Dans ce chapitre, nous parlons d'agent non en référence à nos *agents-modèles*, mais pour nommer les entités atomiques des systèmes que l'on étudie. Ces entités seront la plupart du temps, dans un modèle IBM, les individus du système, mais elles peuvent cependant représenter une entité différente d'un individu réel. Les nommer agent nous permet donc de distinguer l'individu (ou un groupe d'individus) conceptuel d'une population de son implémentation dans un agent.

5.3 Patterns

Nous nous intéressons à des systèmes de peuplements composés d'une ou deux populations en interaction. Ces peuplements ont des comportements élémentaires de reproduction, de mortalité et de migration. Comme nos populations sont théoriques, et donc non observées et non observables dans la nature, définir des patterns pour ces populations est arbitraire et peut paraître artificiel. Néanmoins, les patterns nous permettent de construire une représentation de la dynamique de nos populations sans requérir et présupposer une représentation des mécanismes internes.

Ainsi, nous attendons de nos populations qu'elles soient plus ou moins en équilibre dans un écosystème théorique. Les populations maintiennent un effectif d'individus et une distribution dans l'espace relativement stables. Nous posons donc comme dynamiques remarquables de nos populations leur stabilité en nombre, ainsi que la stabilité de leur distribution dans l'espace (dans leurs habitats). Concernant les systèmes proies-prédateurs, la dynamique des populations est caractérisée par un pattern de type Lotka-Volterra (voir état de l'art section 2.4.4), qui se traduit par des dynamiques corrélées des populations caractérisant l'inter-dépendance des espèces

5.4 Modèle conceptuel

Posons maintenant le cadre théorique de notre étude, ainsi que toutes les hypothèses structurantes comme la définition de l’habitat des populations ou des échelles de temps. Les populations que nous étudions sont élémentaires : les individus qui les composent se reproduisent, meurent et se déplacent dans leur habitat. Les systèmes proies-prédateurs se distinguent d’une population seule par des phénomènes de prédation inter-spécifique¹. Les populations sont relativement stables dans le temps, donc les phénomènes de reproduction et de mortalité, “naturelle” ou par prédation, ont tendance à s’équilibrer. La distribution spatiale des populations est stable et homogène, ce qui traduit l’existence de phénomènes de migration ou d’occupation de l’habitat qui permettent cette homogénéité et qui se basent sur l’habitat des populations.

La figure 5.1a présente l’habitat utilisé pour toutes les populations de notre étude : l’environnement est représenté ici suivant deux axes arbitraires x et y et un axe a qui représente l’affinité des espèces à l’environnement. Plus l’affinité est grande en un point (x, y) , avec $a(x, y) \in [0, 1]$, plus un individu de la population est « bien » en ce point. Nous avons choisi un habitat simple, donc un disque homogène (à l’exception de ses bords) positionné au centre de l’environnement. Les bords sont lissés, ce qui permet de décrire un gradient d’affinité allant du maximum (habitat préférentiel) au minimum (environnement non habitable). L’affinité est donc une variable de l’environnement qui le caractérise pour une espèce donnée. Enfin, l’habitat représente un espace relativement grand pouvant accueillir des millions d’individus, mais il a cependant une capacité d’accueil maximale.

Pour finir, le processus de reproduction dans une population donnée se caractérise par un accroissement de la population. En opposition, le processus de mortalité se traduit par un déclin de la population. De manière schématique, le système pourrait se modéliser à l’aide de l’équation suivante² :

$$f(t+1) = f(t) + \tau_n \times f(t) - \tau_m \times f(t) \quad (5.1)$$

avec $f(t)$ le nombre d’individus dans la population à l’instant t , τ_n le taux de natalité et τ_m le taux de mortalité de la population. La figure 5.1b nous montre l’évolution du nombre d’individus si le taux de mortalité (τ_m) est nul et le taux de croissance (τ_n) est 0,05. La figure 5.1c nous montre l’évolution du nombre d’individus si le taux de croissance est nul et celui de mortalité est 0,05. Théoriquement, la population sera donc en équilibre et le nombre d’individus constant si $\tau_n = -\tau_m$. Ce modèle est évidemment extrêmement conceptualisé et il nous permet d’appréhender une dynamique générale très simple.

¹ inter-spécifique signifie ici qu’il n’y a pas de cannibalisme au sein de la population de prédateurs.

² Remarquons que nous n’utilisons cette équation dans aucun de nos modèles. Elle nous permet simplement de décrire une loi générale qui décrit la dynamique du système considéré.

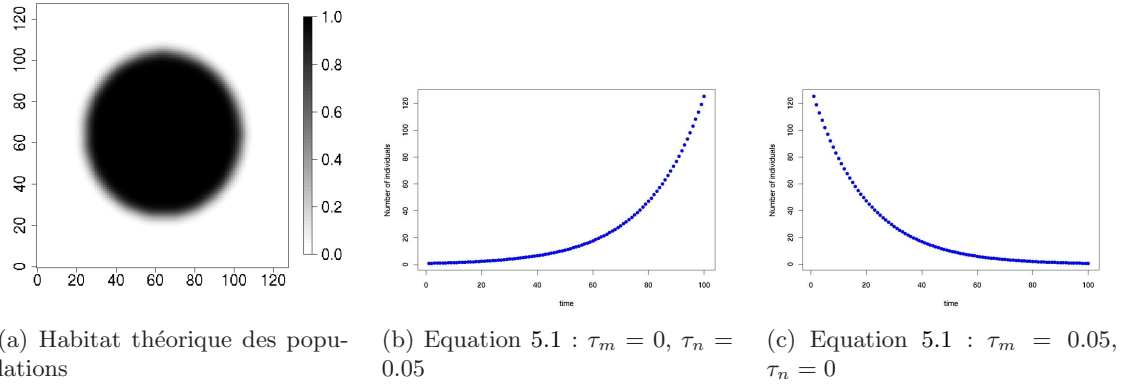


FIG. 5.1: Graphe 5.1a : habitat des populations (affinité maximale : $a = 1$) ; Graphes 5.1b et 5.1c : évolution théorique d'une population suivant l'équation 5.1.

Avant de passer à la suite, il nous faut introduire les notions de densité et de capacité de charge du milieu qui sont classiques dans le cas de modèles de peuplements.

Espèce et type d'agents : Le phénomène de la prédation implique dans notre système une différenciation des agents en deux espèces, les prédateurs et les proies, qui définissent alors le type du peuplement auquel peut appartenir un agent donné.

Notion de densité et de capacité de charge du milieu : La densité d'un peuplement peut s'exprimer en biomasse ou en effectif par unité de surface ou de volume. Ici, la densité est le nombre d'individus par unité de surface.

La capacité de charge du milieu correspond à la densité maximale théorique que le milieu peut supporter. Les agents peuvent alors utiliser cette information de l'environnement pour déclencher des actions de mort ou de migration dans des situations de compétition spatiale. Cette capacité de charge du milieu est définie par espèce ou pour toutes les espèces.

5.5 Comportements de l'agent dans un IBM

5.5.1 Problématique

Dans cette section, nous nous proposons d'étudier une population non spatialisée d'individus qui vivent, se reproduisent et meurent (que l'on peut modéliser par l'équation 5.1). L'objectif est donc uniquement de construire des modèles computationnels dont les dynamiques sont stables en nombre d'individus au cours de la simulation (voir section 5.3). Nous sommes par conséquent sur un système théorique minimal de peuplement. Cela nous

permet d'identifier avec précision les problématiques de synchronicité et de synchronisation des comportements des agents, de sensibilités des comportements aux états initiaux et de modélisation computationnelle déterministe et stochastique. Structurés par les processus élémentaires et antagonistes de création et de destruction d'agents, nos modèles développent alors, avec des hypothèses très simples, des dynamiques parfois divergentes.

Nous structurons notre travail autour des deux solutions computationnelles classiques pour la modélisation de tels processus : 1^o une modélisation déterministe, qui nécessite *de facto* des hypothèses ou des connaissances *a priori* supplémentaires sur le système, et 2^o une modélisation stochastique, qui ne requiert aucune hypothèse additionnelle. Nous abordons alors, dans le cas de la modélisation déterministe, le cas trivial, mais également les problématiques autour de la synchronicité et de la synchronisation des comportements pour un agent donné. De même, la création d'un agent entraîne la question de son initialisation et donc des impacts des choix liés à cette initialisation pour la dynamique de la population. Le cas déterministe traité, nous examinons ensuite le cas stochastique.

5.5.2 Comportement déterministe

5.5.2 - A Modèle computationnel

L'équation (5.1) peut se reformuler ainsi : une certaine proportion de la population se reproduit à chaque pas de temps, une autre meurt à chaque pas de temps. Il nous faut traduire ce modèle conceptuel global en un comportement individuel déterministe. Or, dans les populations réelles, les individus sont rarement, en naissant, matures sexuellement et il n'est pas rare d'observer chez certaines espèces des périodes de reproduction qui rythment la vie des individus. Partant de cette notion de maturité, nous proposons de construire nos comportements individuels sur la notion de seuil : un individu se reproduit quand il atteint un âge donné. Nous utilisons algorithmiquement un accumulateur qui se remplit au cours du temps et qui, une fois «chargé», permet le déclenchement d'un comportement. De la même manière, nous pouvons modéliser le processus «naturel» de mortalité. L'âge de l'individu est alors un «proxy» jouant le rôle de l'accumulateur servant à représenter la longévité de l'individu. Donc, un individu en vieillissant atteint le(s) seuil(s) qui permet(tent) à la reproduction et la mortalité de se déclencher, c'est-à-dire respectivement de créer un nouvel individu et de détruire un individu.

5.5.2 - B Cas trivial

Problématique : Nous voulons reproduire la stabilité en nombre de la population avec le modèle computationnel le plus simple possible. Notre but ici est de montrer que le modèle conceptuel peut être implémenté et reproduire alors la dynamique de stabilité attendue. Nous

Paramètre	: Description
$N_{init} = 10^3$	: Nombre initial d'agents
$\tau_\Phi = 0.03$	: Taux d'accroissement de l'accumulateur (algo. 1)
$age_{max} = 1$	: Valeur maximale de la variable âge
$age_{init} \in [0, 1[$	: Pour un agent donné, la variable <i>age</i> (algo. 1) est initialisée aléatoirement de manière uniforme
$T_{max} = 10000$	: Nombre de pas de temps

TAB. 5.1: Comportement de l'agent dans un IBM, cas trivial : valeurs des paramètres.

pourrons ainsi partir d'un cas opérationnel que nous savons juste et que nous pourrons ensuite enrichir.

Moyen : Le comportement de chaque agent exécute l'algorithme *DeterministicLive* (algo. 1) à chaque pas de temps, algorithme qui décrit les processus de reproduction et de mortalité basés sur un mécanisme de seuil (age maximum). Ce comportement est déterministe, il incrémente l'âge d'une quantité τ_Φ ($\in [0, 1]$) à chaque pas de temps et déclenche la reproduction et la mort de l'agent quand l'âge atteint sa valeur maximale. Par commodité, l'âge est un réel dont la valeur maximale est 1. Le nom des fonctions utilisées dans l'algorithme est généralement suffisamment explicite, précisons tout de même que la fonction FLOOR renvoie la partie entière du paramètre qui lui est passé. Donc, en l'occurrence le test *if* est vrai quand la variable *age* atteint la valeur 1. Concernant l'initialisation de la population, nous distribuons les valeurs des variables *age* des agents de la population de manière uniforme sur $[0, 1]^3$. Enfin, le tableau 5.5.2 - B présente les hypothèses principales que nous utilisons.

Algorithm 1: DETERMINISTICLIVE(τ_Φ)

```

external age : real, NEWAGENT(natural), DIE(), FLOOR(real) : natural
age  $\leftarrow$  age +  $\tau_\Phi$ 
if FLOOR(age)  $\geq$  agemax
  then { NEWAGENT(1)
        DIE()
  }
```

Résultats : Les résultats sont triviaux et la population est constante au cours du temps. En effet, quand un agent atteint son âge maximal, il crée d'abord un nouvel agent, puis meurt immédiatement. Notons que la population est constante quelque soit l'état de la population initiale (distribution des valeurs de la variable *age*_{init} des agents).

³ Nous utilisons pour cela un nombre pseudo-aléatoire uniforme sur $[0, 1[$ de la bibliothèque GSL (Gnu Scientific Library).

5.5.2 - C Synchronicité de comportements indépendants

Problématique : Nous voulons exhiber l’impact de la synchronicité des comportements d’un agent sur la dynamique du système. Rappelons que nos agents peuvent avoir plusieurs comportements et propriétés qui peuvent être actifs (voir section 4.2.3 - B) et qui sont exécutés dans un ordre aléatoire à chaque pas de temps. Nous considérons de plus ici que tous les comportements et propriétés d’un agent sont indépendants, c’est-à-dire que pour toute paire de comportements (et/ou propriétés) d’un agent donné qui sont exécutées au même pas de temps, ces deux comportements s’exécutent (lisent et écrivent) sur des ensembles disjoints de variables. Le comportement global du système émerge alors des activités des agents qui se traduisent par l’exécution de plusieurs comportements différents dans un ordre aléatoire. La question est donc de savoir si, pour un agent donné, il est nécessaire que ses comportements soient synchrones, c’est-à-dire que, pour un agent donné, on garantit que si un comportement s’exécute à un pas de temps donné alors tous les autres comportements de cet agent s’exécutent à ce pas de temps⁴.

Moyen : Nous utilisons cette fois-ci deux comportements disjoints *DeterministicMate* (algorithme 2) et *DeterministicDie* (algorithme 3). Ces comportements sont déterministes et basés sur un mécanisme de seuil. Ils sont indépendants, donc chacun manipule sa propre variable *age* (*age_{mate}* et *age_{die}*). Les valeurs des variables *age* des agents sont distribuées aléatoirement de manière uniforme entre 0 et 1, cependant, pour un agent donné, les deux variables *age* utilisées par *DeterministicMate* et *DeterministicDie* sont identiques. Le tableau 5.2 présente les hypothèses principales que nous utilisons.

Algorithm 2: DETERMINISTICMATE(τ_Φ)

```
external agemate : real, NEWAGENT(natural), FLOOR(real) : natural
agemate  $\leftarrow$  agemate +  $\tau_\Phi$ 
if FLOOR(agemate)  $\geq$  agemax
  then NEWAGENT(1)
```

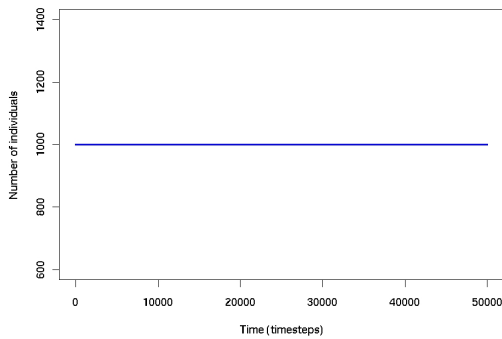
Algorithm 3: DETERMINISTICDIE(τ_Θ)

```
external agedie : real, DIE(), FLOOR(real) : natural
agedie  $\leftarrow$  agedie +  $\tau_\Theta$ 
if FLOOR(agedie)  $\geq$  agemax
  then DIE()
```

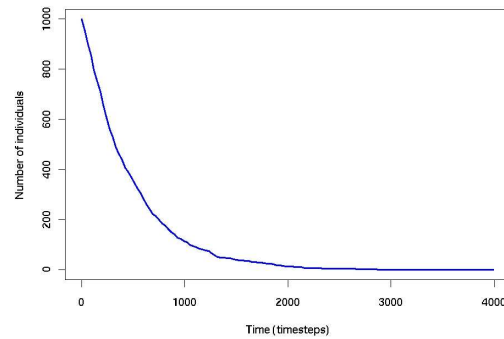
⁴ On part de l’hypothèse que tous les comportements ont le même rythme d’activation, ce qui n’est pas forcément le cas.

Paramètre	Description
$N_{init} = 10^3$	: Nombre initial d'agents
$\tau_\Phi = \tau_\Theta = 0.03$	: Taux d'accroissement des accumulateurs de la reproduction et de la mortalité (algo. 2 et 3)
$age_{max} = 1$	: Valeur maximale de l'âge
$age_{mate,init} \in [0, 1[$	: Pour un agent donné, la variable age_{mate} (algo. 2) est initialisée aléatoirement de manière uniforme
$age_{die,init} \in [0, 1[$	: Pour un agent donné, la variable age_{die} (algo. 3) est initialisée aléatoirement de manière uniforme
$age_{die} = age_{mate}$	: Pour un agent donné, et quelque soit le pas de temps, les variables age_{mate} et age_{die} sont identiques
$T_{max} = 10000$	: Nombre de pas de temps

TAB. 5.2: Comportement de l'agent dans un IBM, synchronicité de comportements indépendants : valeurs des paramètres.



(a) Activation synchronisée



(b) Activation désynchronisée

FIG. 5.2: Evolution d'une population d'agents composés des comportements 2 et 3 : pour chaque agent, les comportements sont soit synchronisés soit désynchronisés.

Résultats : La figure 5.2 montre que l’hypothèse de synchronicité est nécessaire pour que la population soit stable. C’est d’autant plus vrai dans notre exemple que les comportements ont des impacts antagonistes sur le peuplement (créations et destructions d’agents). Tous les comportements d’un agent doivent donc être activés de manière synchrone par rapport à l’état de l’agent (vie ou mort).

5.5.2 - D Synchronisation de comportements dépendants

Problématique : Nous avons jusque là fait l’hypothèse que les comportements (et propriétés) d’un agent étaient indépendants. Mais si nous reprenons les deux comportements *DeterministicMate* et *DeterministicDie*, il est plus cohérent d’un point de vue conceptuel qu’un agent n’ait qu’une seule variable *age* et que ce soit le comportement de mortalité, qui se rapporte alors davantage à un processus de vieillissement, qui incrémente l’âge de l’individu au cours du temps. Ce modèle est sûrement plus spontané pour un expert que le fait d’utiliser deux variables *age* par agent. Mais, implémenter directement un tel modèle conceptuel suppose que les comportements d’un agent ne sont plus indépendants et opèrent sur un ensemble joint de variables (ici la variable *age*). Se pose alors la question de la synchronisation des comportements par rapport à l’agent et à l’état de ses variables. En d’autres termes, quelles précautions doivent être prises concernant l’utilisation de variables par les comportements des agents ? Notons qu’en mode asynchrone, à un pas de temps donné, si un comportement s’exécute avant les autres et modifie une variable liée *v*, les comportements suivants s’exécutent et travaillent avec un état de la variable *v* qui a été modifiée pendant le pas de temps. C’est d’ailleurs vrai même si les comportements ne sont pas actifs, mais activés dans une boucle d’exécution unique et fixée.

Moyen : Les algorithmes 2 et 3 sont légèrement modifiés afin que le comportement *DeterministicMate* ne modifie plus son accumulateur (la variable *age*). Il n’y a plus qu’une variable *age* partagée par les deux comportements et seul le comportement *DeterministicDie* incrémente cette variable à chaque pas de temps. Le moment où est modifiée la variable *age* dans le comportement *DeterministicDie* par rapport au test *if* sur la variable est alors fondamental. Plus précisément, on voit dans l’algorithme *DeterministicDie* que la modification de *age* ($age \leftarrow age + \tau_\theta$) est faite avant le test (IF FLOOR(*age*) > 0) qui déclenche ou non la mort de l’agent. Nous mettons en œuvre deux expériences dans lesquelles la modification de *age* est faite 1^o avant le test ou 2^o après le test. Les comportements sont cette fois-ci synchronisés dans le temps et on garantit que si un des comportements d’un agent s’exécute, alors tous les autres s’exécutent aussi à ce pas de temps là. Le tableau 5.3 présente les hypothèses principales que nous utilisons.

Résultats La figure 5.3 présente les résultats des deux expériences. Une modification de la variable *age* avant le test *if* dans les comportements *DeterministicDie* produit une dynamique globale présentée dans la figure 5.3a. Si cette modification est faite après le *if* la dynamique du système est alors celle de la figure 5.3b. Nous décrivons dans l’annexe A une

Paramètre	Description
$N_{init} = 10^3$	: Nombre initial d'agents
$\tau_\Phi = \tau_\Theta = 0.03$	: Taux d'accroissement des accumulateurs de la reproduction et de la mortalité (algo. 2 et 3)
$age_{max} = 1$	: Valeur maximale de l'âge
$age_{init} \in [0, 1[$	: Pour un agent donné, la variable <i>age</i> (algo. 3) est initialisée aléatoirement de manière uniforme
$T_{max} = 10000$	: Nombre de pas de temps

TAB. 5.3: Comportement de l'agent dans un IBM, synchronisation de comportements dépendants : valeur des paramètres.

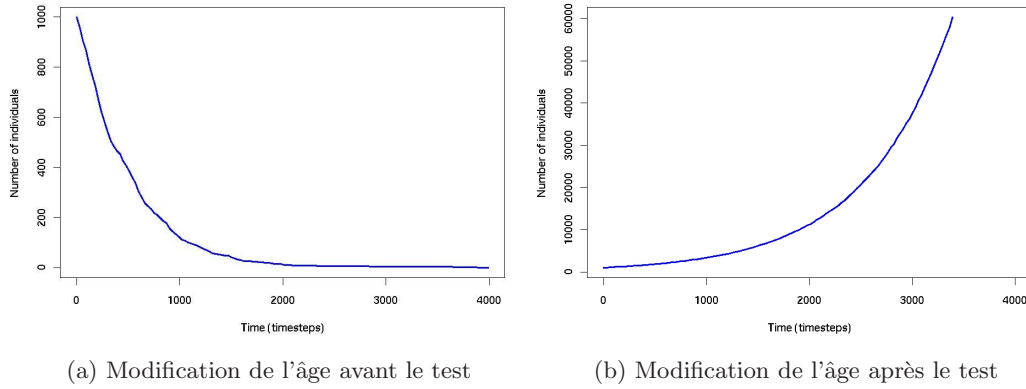


FIG. 5.3: Evolution d'une population d'agents composés des comportements *DeterministicMate* et *DeterministicDie* partageant une même variable *age* (expérience 5.5.2 - D).

preuve formelle qui montre que les résultats étaient tout à fait prévisibles. Nous constatons cependant que l'utilisation d'une variable liée pose des problèmes qui peuvent entraîner des dynamiques aussi divergentes que celles obtenues.

5.5.2 - E Sensibilité des comportements aux états initiaux

Problématique : Reprenons les deux comportements *DeterministicMate* et *DeterministicDie*. Ils sont maintenant synchronisés dans le temps et par rapport à l'agent. Pour les synchroniser par rapport à l'agent, nous les rendons indépendants, donc ils manipulent des ensembles de variables disjoints (deux variables *age* par agent). Nous nous intéressons ici à l'initialisation des agents et, en l'occurrence, dans le cas de notre modèle simple, à l'initialisation des deux

Paramètre	: Description
$N_{init} = 10^3$	: Nombre initial d'agents
$\tau_\Phi = \tau_\Theta = 0.03$	: Taux d'accroissement des accumulateurs de la reproduction et de la mortalité (algos 2 et 3)
$age_{max} = 1$	: Valeur maximale de l'âge
$age_{mate,init} \in [0, 1[$	: Pour un agent donné, la variable age_{mate} (algo. 2) est initialisée aléatoirement de manière uniforme
$age_{die,init} \in [0, 1[$	: Pour un agent donné, la variables age_{die} (algo. 3) est initialisée aléatoirement de manière uniforme
$T_{max} = 10000$	: Nombre de pas de temps

TAB. 5.4: Comportement de l'agent dans un IBM, sensibilité des comportements aux états initiaux : valeur des paramètres.

variables age (age_{mate} et age_{die}) d'un agent. C'est la question de la sensibilité des comportements aux états initiaux des variables age . Nous avons jusqu'à présent travaillé avec une distribution en âge de la population uniforme. De plus, dans la section 5.5.2 - C, les deux variables age d'un agent étaient initialisées de manière homogène. La question ici est de savoir si cette homogénéisation est nécessaire. En d'autres termes, si l'on ne peut garantir qu'un modèle utilisant des variables liées exhibe la dynamique recherchée, peut-on pour autant multiplier les variables pour rendre les comportements indépendants sans précautions concernant leur manipulation (dont l'initialisation en est un cas particulier) ?

Moyen : Nous ré-utilisons les algorithmes 2 et 3. Comme nous avons déjà expérimenté une initialisation homogène des deux variables age (age_{mate} et age_{die}) d'un agent, nous expérimentons ici uniquement une initialisation aléatoire et indépendante des variables age pour chaque agent. Donc, les variables age de la population initiale sont distribuées de manière aléatoire uniforme entre 0 et 1 ($[0, 1[$) et, pour un agent, les valeurs des deux variables ne sont pas forcément égales. Le tableau 5.4 présente les hypothèses principales que nous utilisons.

Résultats : La figure 5.4b montre que la dynamique générale de la population n'est pas stable. La figure ne montre que les 600 premiers pas de temps de l'évolution du nombre d'agents. On voit que la population décroît pendant un peu plus de 300 pas de temps avant de se stabiliser. Elle reste alors stable jusqu'à la fin de la simulation et le nombre d'agents est proche de 500, le système tendant de manière générale vers cette valeur. En effet, pour un agent donné, les valeurs des deux variables étant tirées aléatoirement entre 0 et 1, il y a une chance sur deux que l'agent meurt avant de se reproduire, et une chance sur deux qu'il se reproduise avant de mourir. En moyenne, un agent sur deux mourra ainsi avant d'avoir pu se reproduire. Pour appuyer notre propos, la figure 5.4a présente l'évolution de la population avec une initialisation homogène des variables age par agent.

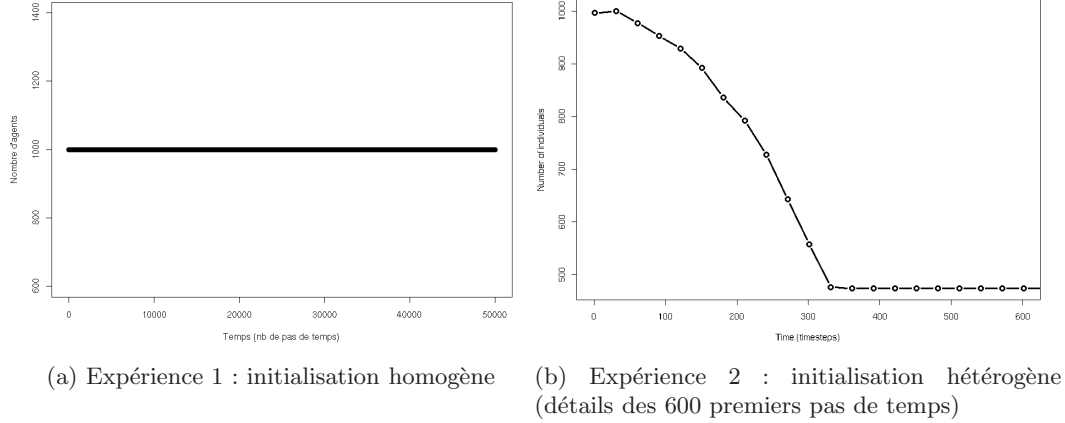


FIG. 5.4: Sensibilité des comportements aux états initiaux – Etude des initialisations homogène ou hétérogène des variables *age* des comportements d'un agent.

5.5.3 Comportement stochastique

5.5.3 - A Modèle computationnel

Il s'agit maintenant d'utiliser des processus stochastiques et indépendants pour décrire les deux comportements individuels de mortalité et de natalité. Stochastiques, ils utilisent une variable aléatoire pour savoir s'ils peuvent s'exécuter ou non. Indépendants, les variables aléatoires utilisées sont indépendantes l'une de l'autre. Les algorithmes utilisés sont l'algorithme 4 pour la reproduction et l'algorithme 5 pour la mortalité.

Algorithm 4: STOCHASTICMATE(τ_Φ)

```

external RAND() : real{ $\in [0, 1]$ }, NEWAGENT(natural)
if RAND() <  $\tau_\Phi$ 
  then NEWAGENT(1)

```

Algorithm 5: STOCHASTICDIE(τ_Θ)

```

external RAND() : real{ $\in [0, 1]$ }, DIE()
if RAND() <  $\tau_\Theta$ 
  then DIE()

```

Paramètre	: Description
$\tau_{\Phi} = \tau_{\Theta} = 0.03$	: Respectivement taux de natalité et de mortalité (algo. 4 et 5)
$T_{max} = 5.10^4$	: Durée des simulations
$N_{sim} = 30$	: Pour chaque expérience, on effectue 30 simulations

TAB. 5.5: Comportement de l’agent dans un IBM, comportement stochastique : valeur des paramètres.

5.5.3 - B Application

Problématique : Nous proposons ici d’étudier l’impact de l’utilisation de processus stochastiques comme solution computationnelle pour la modélisation. Les modèles IBM, et en particulier les modèles de peuplements d’agents, sont souvent associés à tort ou à raison à des processus stochastiques. L’utilisation de modèles stochastiques permet de ne pas formuler d’hypothèse sur le mécanisme individuel à mettre en œuvre. En effet, pour la modélisation déterministe, nous avons dû par rapport au modèle conceptuel, introduire la notion de seuil et d’accumulateur et lier la reproduction et la mortalité à un moyen computationnel pour représenter l’âge de l’individu. La modélisation stochastique permet de s’affranchir de ces hypothèses et d’exprimer au niveau individuel ce que le modèle conceptuel formule au niveau de la population : à un pas de temps donné, un individu a une certaine chance de se reproduire et il a une certaine chance de mourir. Nous distinguons bien ici l’utilisation de modèles stochastiques au niveau thématique et qui sont alors considérés en tant que tels au niveau conceptuel et l’utilisation de moyens stochastiques au niveau computationnel. Juger de la validité de la modélisation stochastique d’un point de vue conceptuel n’est pas de notre ressort. Par contre, utiliser l’artifice stochastique en tant que moyen computationnel peut entraîner des biais sur la dynamique du système et pose la question de l’analyse des résultats obtenus. Pouvons-nous évaluer la validité de moyens stochastiques au niveau computationnel pour rendre compte d’un modèle conceptuel donné qui n’en fait pas l’usage ?

Moyen : Les comportements des agents sont décrits par les algorithmes 4 et 5. Chaque comportement est associé à son propre générateur de nombres aléatoires. Nous effectuons un certain nombre d’expériences pour lesquelles nous faisons varier le nombre initial d’agents. De plus, pour chaque expérience, nous effectuons 30 simulations, nous en observons la dynamique moyenne et nous faisons un certain nombre d’analyses statistiques : régression linéaire, calcul de la moyenne globale, de la déviation standard moyenne globale, de l’erreur moyenne globale, du coefficient de variation et du biais de la moyenne globale par rapport à la valeur espérée. Le tableau 5.5 présente les hypothèses principales que nous utilisons

Résultats : Les résultats sont présentés dans le tableau 5.6. La moyenne globale correspond à la moyenne du nombre moyen d’agents au cours du temps, donc après avoir calculé à chaque pas de temps la moyenne sur toutes les simulations et établi ainsi l’évolution de la moyenne (voir figure 5.5), nous avons calculé la moyenne de cette évolution. La déviation standard

N_{agents}	Moyenne globale	Déviatiion standard moyenne globale	Erreur moyenne globale	Coefficient de variation	Biais de la moyenne globale
10	12.14	42.17	18.78	3.47	21.44%
50	30.05	53.55	47.14	1.78	39.91%
100	97.72	102.61	79.18	1.05	2.28%
500	498.60	295.56	233.94	0.59	0.28%
1000	1004.61	326.72	254.30	0.33	0.46%
5000	4980.93	777.25	607.51	0.16	0.38%

TAB. 5.6: Comportement de l'agent dans un IBM, comportement stochastique : simulations et résultats.

moyenne globale a été calculée de la même manière, c'est-à-dire que nous avons calculé à chaque pas de temps l'évolution moyenne de la déviation standard et ensuite la moyenne de cette évolution moyenne. Pour calculer l'erreur moyenne globale, nous avons d'abord calculé à chaque pas de temps l'erreur moyenne sur toutes les simulations, erreur qui est donnée par $e = \sqrt{(\bar{\Phi} - \Phi)^2}$, avec Φ la valeur observée et $\bar{\Phi}$ la valeur espérée. L'erreur moyenne globale est alors la moyenne de l'évolution de l'erreur moyenne. Enfin le coefficient de variation correspond à la division de la déviation standard par la moyenne.

La figure 5.6a donne l'évolution du coefficient de variation au cours du temps. On voit qu'à partir de 500 agents, on a un coefficient de variation qui est relativement bas, ce que vient confirmer la moyenne globale qui a un biais inférieur à 1% (de l'ordre de 0.28%). En augmentant encore le nombre d'agents, l'amélioration qualitative du résultat n'est plus aussi significative que quand on passe de 100 à 500 agents. De plus, les biais sur la moyenne globale avec 1000 et 5000 agents n'ont plus tendance à diminuer. Il semble donc, qu'à partir de 500 agents, l'augmentation du nombre d'agents n'améliore pas dramatiquement la précision du résultat.

La figure 5.6b présente les trente résultats de simulations pour 1000 agents. On peut y voir que les différentes trajectoires forment une gerbe avec certaines trajectoires qui atteignent des valeurs en nombre d'agents supérieur à 2000 et d'autres proches de 0. Il est par conséquent essentiel de travailler sur des moyennes et donc sur un nombre suffisant de simulations.

Les algorithmes utilisés dans le modèle stochastique ne se basent pas sur une variable *age*, mais nous en avons tout de même ajouté une afin d'observer la dynamique des agents plus finement. Cette variable *age* a donc été incrémentée à chaque pas de temps (sans que le système n'en soit influencé) et nous avons observé à chaque pas de temps la distribution en âge de la population. De plus, chaque agent de la population initiale se voyait attribuer un âge tiré aléatoirement entre $[0, 1[$, comme pour les modèles déterministes, et donc la distribution initiale en âge de la population était uniforme. La figure 5.6c présente la distribution en âge de la population d'agents à un pas de temps quelconque de la simulation ($t \sim 10000$). Nous constatons que la distribution en âge n'est plus du tout uniforme et qu'à la différence du

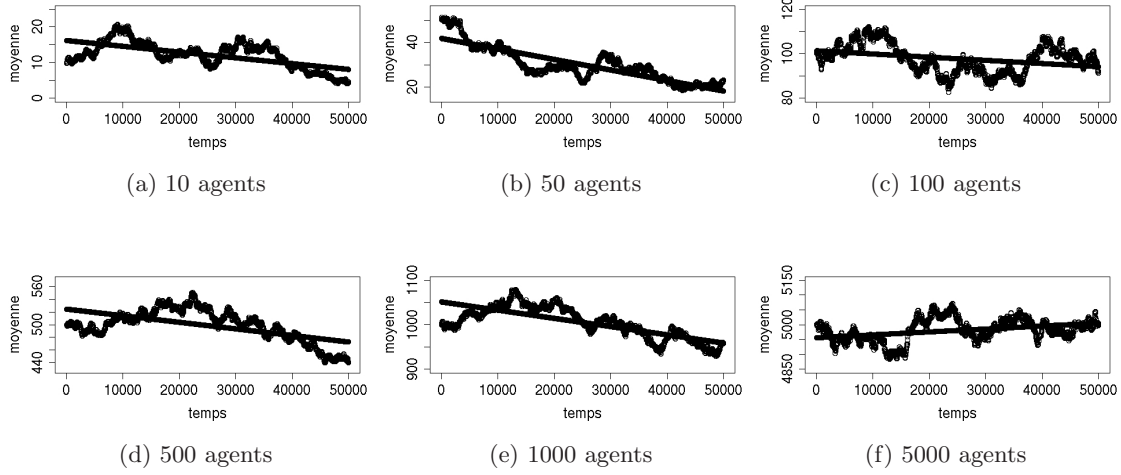


FIG. 5.5: Modèle stochastique : évolution de la moyenne dans le temps de 30 simulations et ajustement par un modèle linéaire.

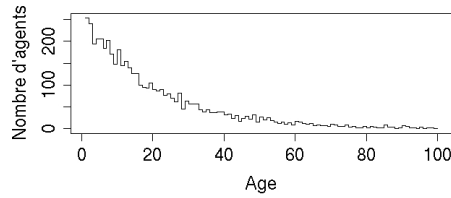
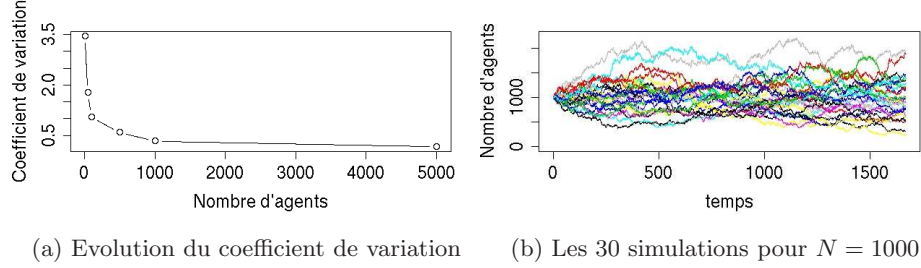


FIG. 5.6: Modèle stochastique : (a) voir tableau 5.6 ; (b) graphe des 30 résultats de simulation (1000 agents) ; (c) exemple d'une distribution en âge de la population (pas de temps quelconque, 1000 agents).

modèle déterministe, elle n'est pas fonction de l'état initial.

5.5.4 Analyse

Abordons maintenant l'analyse des résultats obtenus jusqu'à maintenant concernant la mise en œuvre de modèles de peuplements non spatialisés d'agents et de leurs comportements. Premièrement, nous avons abordé la question de la synchronicité des comportements d'un agent. A partir du moment où un agent peut avoir plusieurs comportements (et propriétés) actifs, la question se pose de rendre leurs exécutions synchrones, c'est-à-dire de garantir que, pour un agent donné, tous ses comportements s'exécutent autant de fois les uns que autres. Cette notion n'apparaît pas au niveau conceptuel dans lequel les différents comportements sont implicitement exécutés en parallèle pour un pas de temps donné, à partir du moment où le temps est d'ailleurs discrétisé (ce qui n'est pas toujours le cas au niveau conceptuel). Les résultats montrent qu'il est nécessaire pour notre système de rendre synchrone les comportements de chaque agent pour pouvoir rendre compte de la dynamique attendue et donc traduire correctement le modèle conceptuel.

Nous avons ensuite abordé la question de la synchronisation des comportements de l'agent par rapport à l'état de ses variables. En d'autres termes, on peut se demander si les comportements d'un agent, qu'ils soient actifs ou non, doivent forcément être indépendants d'un point de vue computationnel. Nous avons vu que la dépendance à une même variable de deux comportements d'un agent peut entraîner une dynamique aberrante du système. Il est donc nécessaire soit de garantir que tous les comportements d'un agent sont indépendants, soit de synchroniser l'action des comportements sur la variable et de gérer les conflits. Garantir que les comportements d'un agent sont indépendants, c'est en l'occurrence connaître les ensembles de variables que manipulent les comportements et garantir qu'ils sont tous disjoints. Synchroniser l'action des comportements pose davantage de problèmes. Si deux comportements modifient la variable liée, il s'agit alors de rendre effectives les deux modifications après que les comportements se soient tous les deux exécutés, mais il s'agit alors de savoir comment cumuler ces deux modifications. On peut plus simplement limiter la modification de la variable à un seul comportement et rendre la modification effective après que tous les autres comportements accédant en lecture à cette variable aient été exécutés. Dans tous les cas, on constate que l'utilisation de variables liées pose des problèmes computationnels qui peuvent avoir un impact dramatique sur la dynamique globale. Si les modélisateurs n'ont pas connaissance de ce biais, il faut alors que le simulateur garantisse une bonne gestion des variables liées, en les interdisant ou en proposant des solutions de gestion de conflits.

La sensibilité des comportements aux conditions initiales des agents nous a permis d'exhiber le fait qu'il ne suffit pas de rendre des comportements indépendants en dupliquant la variable liée pour que la dynamique du système ne soit pas biaisée. Si les comportements sont dépendants au niveau conceptuel, c'est qu'ils manipulent une même variable conceptuelle,

qu'ils sont liés à une même "réalité". Donc, rendre deux comportements indépendants en dupliquant la variable à laquelle ils sont liés sans aucune précaution sur la manipulation de ces deux variables ne permet pas de rendre compte du lien entre les deux comportements qui existe au niveau conceptuel. Pour un modèle donné, il est donc nécessaire que les simulateurs offrent un cadre de construction qui permette d'exhiber ces problématiques, voire de proposer des solutions automatiques qui seraient par exemple une synchronisation automatique des variables computationnelles qui représentent la même variable conceptuelle.

Enfin, utiliser au niveau conceptuel des moyens stochastiques est de la responsabilité des experts qui sont censés connaître la nature des processus stochastiques. Cependant, nous remarquons deux choses importantes. Tout d'abord, que ce soit au niveau conceptuel ou computationnel, utiliser des processus stochastiques requiert un nombre suffisant d'agents et un nombre suffisant de simulations. Cela nécessite de toute façon de travailler sur des moyennes. Ce lien entre la taille d'un échantillon et les fluctuations autour d'une moyenne est bien sûr en cohérence avec la loi probabiliste des grands nombres. Ensuite, on a vu que pour un même modèle conceptuel, utiliser une solution déterministe ou stochastique entraîne un comportement du système qui est très différent. Même si les processus stochastiques reproduisent une dynamique de stabilité que l'on pourrait qualifier de plus naturelle par rapport à l'idée que l'on pourrait se faire du système réel, parce qu'elle reproduit une variabilité chaotique locale que l'on retrouve souvent dans les systèmes complexes par exemple, la nature stochastique de ces processus, qui ne nécessitaient pas *a priori* d'hypothèses supplémentaires, induisent en réalité implicitement des hypothèses sur la dynamique. La structuration en âge de la population en est un exemple. L'artifice stochastique au niveau computationnel doit donc être utilisé en connaissance de cause et il est nécessaire d'en maîtriser les biais.

5.6 Sensibilités d'un IBM aux discrétisations homogènes

5.6.1 Problématique

Nous étudions ici la sensibilité à la discrétisation en nombre d'agents et aux discrétisations spatiales et temporelles de modèles de populations nous permettant de reproduire les dynamiques identifiées dans la première section de ce chapitre. Il s'agit de déterminer l'influence des artefacts computationnels que sont les différentes discrétisations sur la dynamique de nos systèmes et de mesurer expérimentalement les contraintes intrinsèques de l'approche IBM pour ce genre de systèmes. Nous n'étudions dans cette section que des discrétisations homogènes pour ne pas interférer avec les problématiques d'échelles multiples.

Nous avons précédemment travaillé avec des systèmes non spatialisés, ce qui nous a permis d'étudier la simulation de peuplements d'agents en nous concentrant sur l'agent (ses comportements, son initialisation, les variables qu'il manipule, etc.). Nous proposons maintenant de mesurer l'impact des choix computationnels nécessaires pour construire l'environnement et le temps de la simulation. De plus, par rapport à l'approche EDP, se pose la question de la discrétisation des populations en entités individuelles. Sur ce dernier point, l'étude de peuplements d'agents est particulièrement pertinente. En effet, nous sommes limités par les capacités de nos machines et le nombre d'agents ne peut pas toujours correspondre à la quantité d'individus des systèmes étudiés. Reproduire un phénomène ne requiert pas forcément qu'il le soit à l'échelle un, c'est-à-dire qu'il y ait une bijection entre l'ensemble des agents et celui des individus du système réel. Pour autant, cette approximation doit être mesurée. Plus généralement, l'approche IBM rend la dynamique en nombre du peuplement non continue. C'est bien cette discrétisation, justifiée par des contraintes sur le nombre d'agents, qui requiert une étude de sensibilité.

Pour chaque problématique de discrétisation, nous reformulons le modèle computationnel afin de l'adapter à la problématique qui nous intéresse tout en restant dans le cadre théorique fixé au début de ce chapitre. Nous étudions dans un premier temps la sensibilité de nos modèles à la discrétisation en nombre d'agents. Nous introduisons ensuite des processus de migration plus fins et nous nous intéressons alors à la sensibilité à la discrétisation spatiale de notre modèle, c'est-à-dire 1° dans la relation de l'agent avec son environnement et 2° dans la relation de l'agent avec les autres. Enfin, nous expérimentons la sensibilité à la discrétisation temporelle, ce qui nous oblige à définir précisément l'échelle temporelle de notre système.

5.6.2 Sensibilité au nombre d'agents

5.6.2 - A Modèle computationnel – comportements des agents

Comportement de reproduction : Il est déterministe et se base sur la notion écologique de maturité sexuelle que nous proposons de traduire algorithmiquement par un accumulateur (*potential*). Cette quantité est fonction d'un coefficient τ_Φ . Le nombre d'agents à créer est alors la plus grande valeur entière inférieure à la valeur de l'accumulateur *potential* (cf. fonction *floor* de l'algorithme 6). Notons que $\alpha_H(x, y)$ permet de connaître l'affinité à l'habitat H en (x, y) .

Algorithm 6: MATE(x, y, t, τ_Φ)

```

external potential : real, FLOOR(real) : natural,  $\alpha_H(x : real, y : real) : real$ , NEWAGENT(natural)
potential  $\leftarrow$  potential + ( $\alpha_H(x, y) \times \tau_\Phi$ )
NEWAGENT(FLOOR(potential))
potential  $\leftarrow$  potential - FLOOR(potential)
    
```

Comportement de mortalité : La mortalité est influencée par l’affinité et la densité locale. Le comportement est également basé sur la notion de capacité du milieu et dépend d’un paramètre de densité maximale K_{max} . Là encore, le processus est déterministe. On voit dans l’algorithme 7 que l’agent meurt si la densité locale en agents est supérieure à la densité maximale pondérée par l’affinité locale à l’environnement. Notons que $K(x, y, t)$ est une méthode de l’environnement permettant de connaître la densité locale en (x, y, t) .

Algorithm 7: DIE(x, y, t, K_{max})

```
external K( $x : real, y : real, t : real$ ) :  $real, \alpha_H(x : real, y : real) : real$   
if  $K(x, y, t) > \alpha_H(x, y) \times K_{max}$   
  then DIE()
```

Comportement de recrutement : Pour implémenter au niveau individuel la distribution spatiale décrit dans le modèle conceptuel, nous nous inspirons du phénomène biologique du recrutement. Dans les populations réelles, l’intégration des nouvelles recrues n’est pas forcément corrélée avec la zone de reproduction des géniteurs. Nous proposons donc ici de modéliser la distribution spatiale de la population en positionnant les nouveaux individus en fonction de leur affinité à l’environnement. Nous utilisons pour cela une distribution de probabilité non paramétrique à laquelle l’agent peut accéder à l’aide de la méthode *getProbabilityDistribution* et qui est initialisée par la distribution de l’affinité à l’environnement de la population. Ce comportement n’est donc activé qu’une seule fois par l’agent, au moment de sa création.

Algorithm 8: RECRUIT()

```
external GETPROBABILITYDISTRIBUTION() :  $Distribution, SETLOCATION(l : Location)$   
 $d \leftarrow GETPROBABILITYDISTRIBUTION()$   
 $loc \leftarrow d.GETLOCATION()$   
SETLOCATION( $loc$ )
```

5.6.2 - B Cas d’une population

Problématique : Partant du modèle computationnel identifié ci-dessus (section 5.6.2 - A), l’étude porte sur la discrétisation en nombre d’agents et son impact sur la dynamique globale d’un modèle de peuplement composé d’une seule population, donc d’un seul type d’agents. Nous cherchons à établir des connaissances en termes de précision sur les dynamiques d’une population en fonction du nombre d’agents.

Moyen : Chaque agent est composé des comportements *mate*, *die* et *recruit*. Le tableau 5.7 présente l'initialisation et les valeurs des paramètres utilisés. Concernant la discrétisation spatiale, celle-ci a un impact sur les comportements des agents, il s'agit donc de la préciser. Nous choisissons la même discrétisation spatiale pour décrire l'affinité et la densité qui sont des caractéristiques locales de l'environnement. La première dépend uniquement de l'habitat, alors que la deuxième doit être calculée à chaque fois qu'un individu perçoit son environnement⁵. Nous utilisons l'habitat du modèle conceptuel et nous discrétisons l'environnement en 16×16 cases pour toutes les variables de l'environnement (i.e. affinité et densité). L'habitat, qui est en forme de disque, a un rayon d'à peu près cinq cases (même si, de par la discrétisation, le disque est déformé). Nous considérons donc une zone peuplée de ~ 80 cases (l'environnement total fait 256 cases) ayant chacune une affinité donnée et une densité observée de manière asynchrone par tous les agents.

Le comportement *die* dépend de l'affinité et de la densité locale, mais surtout d'un paramètre de capacité du milieu ou densité maximale. Donc, ce paramètre nous permet de contrôler le nombre d'agents que le milieu peut accueillir. En effet, le comportement de mortalité est le seul phénomène qui limite la population dans l'environnement et le nombre total d'agents dépend de la capacité de charge du milieu. Par conséquent, nous proposons de faire varier ce paramètre afin de jouer sur le nombre total d'agents de la population. Seulement, pour pouvoir faire cela, il faut pouvoir calculer le nombre d'agents qui est global en fonction de cette densité maximale qui est locale.

Pour se faire, nous calculons une estimation du nombre d'agents, que l'on nommera N_e pour nombre d'agents espérés, en fonction de la densité maximale locale en chaque case de l'environnement, donc du nombre d'agents que cette case peut accueillir. Ainsi :

$$N_e = K_{max} \sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \alpha_H(x, y) \quad (5.2)$$

L'équation ci-dessus donne N_e en fonction de K_{max} qui est la densité maximale locale et $\alpha_H(x, y)$ qui donne la valeur de l'affinité en (x, y) . Or, comme nous pouvons le voir dans l'algorithme 7, la densité maximale en une case est pondérée par la valeur de l'affinité α_H dans cette case (avec $\alpha_H \in [0, 1]$). Donc, pour estimer le nombre total d'agents que l'environnement est capable d'accueillir, il faut estimer la quantité possible dans chaque case et sommer toutes ces quantités.

Comme l'habitat n'est pas variable au cours du temps, nous pouvons calculer le terme de l'affinité de l'équation (5.2) ci-dessus à partir de notre carte d'affinité : $\sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \alpha_H(x, y) = 77.9686$. Et donc : $N_e = K_{max} \times 77.9686$. Notons enfin que si nous voulons avoir au moins un agent par case, il faut que $K_{max} = 1$ et alors $N_e = 77.9686$ (soit 77 car N_e est un entier). Pour toute valeur inférieure de K_{max} , un agent percevra une densité supérieure ou égale à 1 (et donc $> K_{max}$) même s'il est seul dans cette

⁵ Les agents ne sont pas synchronisés par rapport à leur environnement.

Paramètre	: Description
$potential_{init} = 0$	: Pour tous les agents, l'accumulateur de <i>mate</i> est mis à zéro au départ de la simulation.
$\alpha(x, y)$	: Affinité dans l'espace (distribution)
$K(x, y, t)$	: Densité dans l'espace (discrétisation)
K_{max}	: Densité locale maximale
$N_{init} = N_e$	: Nombre initial d'agents égal au nombre total d'agents espéré
$t_{max} = 3600$	: Nombre total de pas de temps

TAB. 5.7: Sensibilité en nombre d'agents, cas d'une population : valeurs des paramètres.

K_{max}	N_e	Moyenne	Déviati on standard	Coefficient de variati on
0.64	50	-	-	-
1	77	86.89	4.89	0.05624
6	467	537.1	10.74	0.02
13	1013	1109.78	11.19	0.01
65	5067	5210.74	18.22	0.0035
129	10057	10226.55	19.80	0.0019

TAB. 5.8: Sensibilité en nombre d'agents, cas d'une population : simulations et résultats.

case et donc il mourra. Toute initialisation du système avec $K_{max} < 1$ induira une disparition de la population.

Enfin, le nombre initial d'agents est fixé au nombre espéré d'agents : $N_{init} = N_e$.

Résultats : Le tableau 5.8 présente les différentes simulations effectuées suivant la valeur de K_{max} et celle de N_e qui en découle. Le tableau est à mettre en relation avec la figure 5.7 qui montre les évolutions au cours de la simulation de la population pour chaque valeur de K_{max} . Avant toute chose, notre modèle nous permet de reproduire un peuplement qui est stable en nombre au cours de la simulation, sauf pour la première simulation qui a été faite avec $K_{max} < 1$ et qui nous permet de vérifier que la population s'éteint dès les premiers pas de temps (figure 5.7a). Pour les autres simulations, nous présentons dans le tableau la moyenne en nombre d'agents au cours de la simulation, la déviation standard qui lui est associée et le coefficient de variation qui est calculé en divisant la déviation standard par la moyenne. Nous présentons alors dans la figure 5.8a l'évolution du coefficient de variation en fonction de la valeur de K_{max} . Nous pouvons ainsi constater que l'augmentation du nombre d'agents entraîne une diminution du coefficient de variation. En particulier, le coefficient de variation est diminué d'un facteur supérieur à 5 quand on passe de 1 agent par case à 13 agents par case. D'après la courbe 5.8a, on peut penser qu'à partir de 13 agents par case, le coefficient de variation ne diminue plus autant. Seulement, on remarquera que si l'on multiplie encore par dix le nombre d'agents, pour atteindre 129 agents par case, on obtient une amélioration de la précision au travers d'une diminution du coefficient de variation d'un facteur 10.

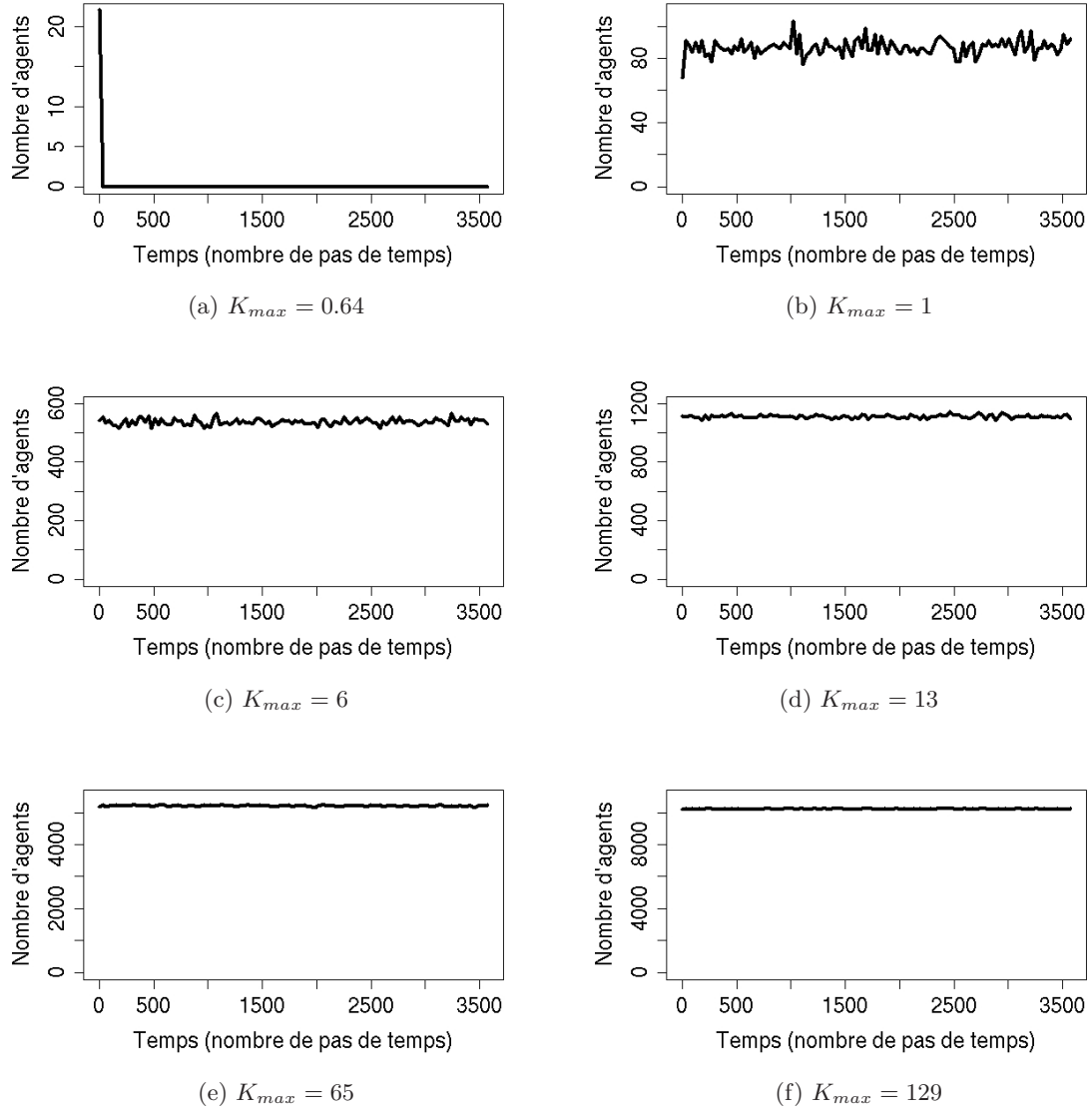
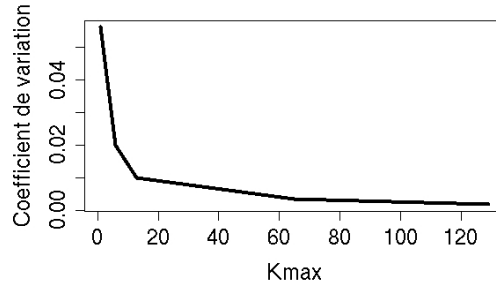


FIG. 5.7: Discrétisation homogène en nombre d'agents avec K_{max} la densité maximale par case.



(a) Coef vs Kmax

FIG. 5.8: Discrétisation homogène en nombres d’agents – Evolution du coefficient de variation en fonction de K_{max} .

Enfin, la figure 5.9 présente des exemples de distributions du peuplement pris à un pas de temps quelconque (mais avancé dans la simulation) pour K_{max} valant 1 (figures 5.9a et 5.9b), 13 (figures 5.9c et 5.9d) et 129 (figures 5.9e et 5.9f). On peut constater que plus le nombre d’agents est important par case, plus la distribution dans l’espace est uniforme. Notons qu’elle est également stable au cours de la simulation et que ces deux résultats valident le modèle computationnel au regard des dynamiques attendues.

5.6.3 Sensibilité à la discrétisation spatiale

5.6.3 - A Modèle computationnel – comportements des agents

Nous avons vu dans l’étude précédente que la question de la discrétisation spatiale était intimement liée à la discrétisation en nombre d’agents. En effet, pour faire varier le nombre d’agents, nous faisons varier le terme de densité maximale par case, ce qui dépend évidemment de la discrétisation spatiale. Par conséquent, pour l’étude de la sensibilité de modèles de peuplements à la discrétisation spatiale, nous allons fixer les dynamiques d’évolution en nombre d’agents. Nous nous concentrons ainsi uniquement sur les dynamiques spatiales permettant de reproduire la stabilité de la distribution du peuplement dans son habitat en fonction de l’affinité. Nous proposons par conséquent de reformuler un modèle computationnel, basé sur le modèle conceptuel, dans lequel les phénomènes de création et de destruction d’agents sont fixés. La stabilité de nos peuplements dans cette section en termes de nombre d’agents va donc de soi, de par le modèle, et nous ne revenons pas sur cet aspect. Il nous faut par contre proposer des comportements individuels de migration, pour les proies et pour les prédateurs, qui permettent d’étudier les problématiques de relation d’un agent avec son environnement et de relation d’un agent avec les autres agents. Notons enfin que la

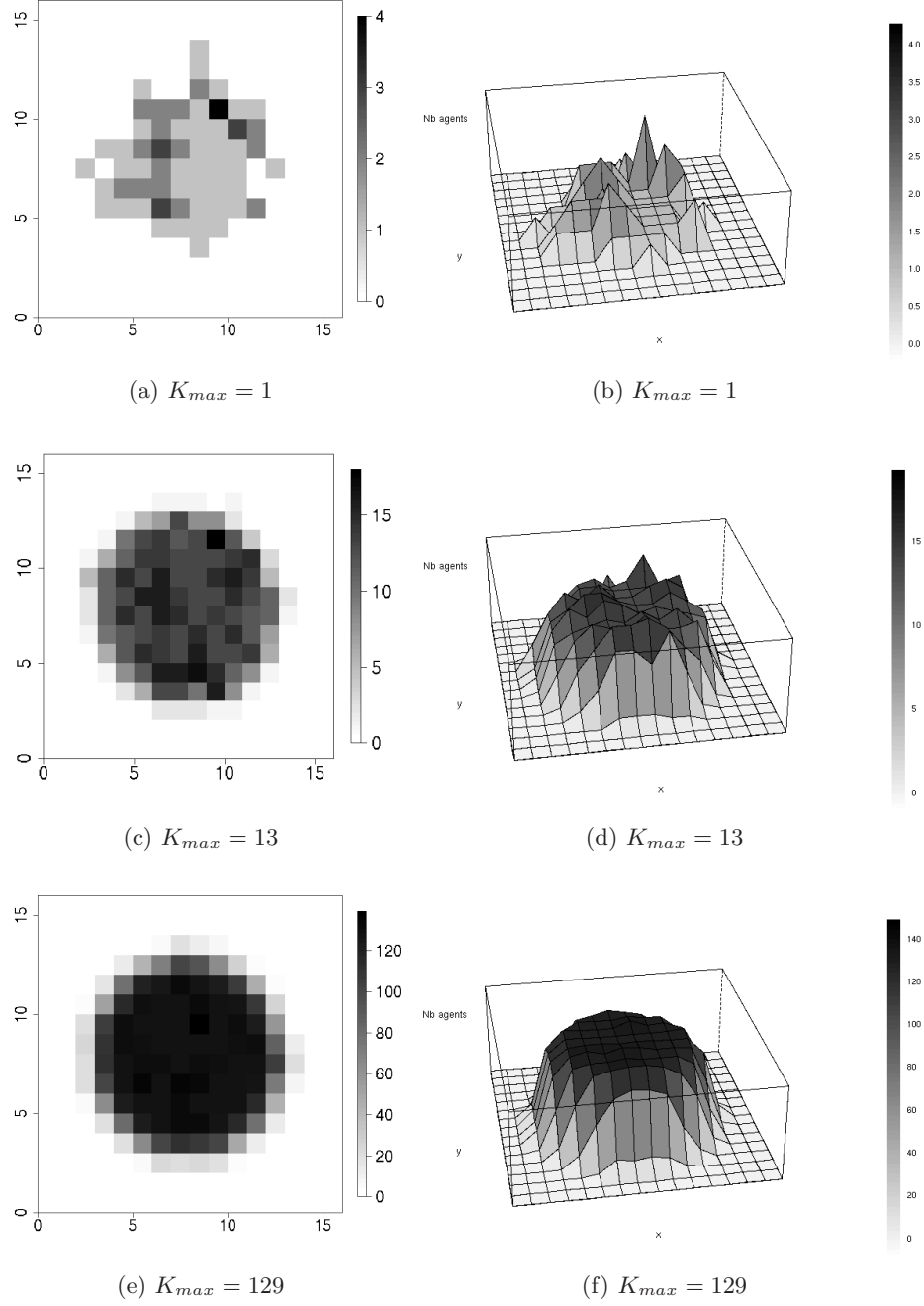


FIG. 5.9: Discrétisation homogène en nombre d'agents – Distributions spatiales du peuplement suivant K_{max} ($t=2000$).

perception se base sur un voisinage de type Von Neumann, donc un agent ne perçoit que les cases qui l'entourent directement, ainsi que la case où il se trouve.

Comportement de migration des proies : La distribution des proies dépend de l'habitat, de la densité intraspécifique et de la densité en prédateurs. L'habitat «idéal» d'une proie est caractérisé localement par une affinité maximale et une densité minimale en proies et en prédateurs. La proie perçoit son environnement et calcule un vecteur de déplacement qui est la somme de trois vecteurs : 1° le vecteur de déplacement vers la case de meilleure affinité, 2° le vecteur de déplacement opposé à la case de densité en proies la plus forte et 3° le vecteur de déplacement opposé à la case de densité en prédateurs la plus forte.

Notons que nous avons intégré un processus dépendant de la densité intraspécifique et qui, même dans le cas d'une seule population (cas étudié pour la problématique de la relation de l'agent à son environnement), introduit un processus lié aux autres agents. Il nous faut pourtant introduire un tel processus, car suivant la distribution initiale de la population, des phénomènes de concentration d'agents sur les bords de l'habitat maximal apparaissent sans lui. Prenons un exemple, si nous distribuons aléatoirement en fonction de l'habitat les agents à l'initialisation de la simulation, cela implique que certains agents se trouvent dans des zones où l'affinité à l'habitat n'est pas maximale (mais n'est pas nulle non plus). Ces agents vont alors forcément se déplacer en cours de simulation vers un habitat avec une affinité supérieure et s'arrêter aussitôt atteinte la zone d'affinité maximale. Il y a alors une accumulation mécanique des agents sur les bords de l'habitat maximal et plus aucun agent dans les zones où l'affinité est non nulle et non maximale. Pour traduire, dans un comportement individuel, le fait que certains agents ont parfois tendance à rester dans des zones qui ont une affinité plus faible, il est nécessaire d'introduire un processus lié à la densité en agents.

Algorithm 9: MIGRATEANDFLEE($x, y, t, r_{migration}$)

```

 $p_0 \leftarrow \text{GETCURRENTLOCATION}()$ 
/* Process vector towards highest affinity*/
 $p_1 \leftarrow \text{HIGHESTAFFINITYINNEIGHBORHOOD}(x, y, t)$ 
 $v_1 \leftarrow p_1 - p_0$ 
 $v_1 \leftarrow \frac{v_1}{\|v_1\|}$ 
/* Process vector opposite to highest density in preys*/
 $p_2 \leftarrow \text{HIGHESTDENSITYINNEIGHBORHOOD}(\text{"preys"}, x, y, t)$ 
if GETINTRASPECIFICDENSITY("preys",  $p_0$ )  $\leq$  GETINTRASPECIFICDENSITY("preys",  $p_2$ )
  then  $v_2 \leftarrow 0$ 
  else  $\begin{cases} v_2 \leftarrow p_0 - p_2 \\ v_2 \leftarrow \frac{v_2}{\|v_2\|} \end{cases}$ 
/* Process vector opposite to highest density in predators*/
 $p_3 \leftarrow \text{HIGHESTDENSITYINNEIGHBORHOOD}(\text{"predators"}, x, y, t)$ 
if GETINTRASPECIFICDENSITY("predators",  $p_0$ )  $\leq$  GETINTRASPECIFICDENSITY("predators",  $p_3$ )
  then  $v_3 \leftarrow 0$ 
  else  $\begin{cases} v_3 \leftarrow p_0 - p_3 \\ v_3 \leftarrow \frac{v_3}{\|v_3\|} \end{cases}$ 
MOVE_TOWARDS( $(v_1 + v_2 + v_3) \times r_{migration}$ )

```

Comportement de migration des prédateurs : Le comportement de migration des prédateurs dépend de la densité intraspécifique, donc de la densité en proies. Le prédateur perçoit son voisinage et calcule un vecteur de déplacement qui est la somme du vecteur de déplacement vers la case de densité intraspécifique la plus faible et du vecteur de déplacement vers la case de densité en proies la plus élevée. Nous avons également intégré une attirance des prédateurs à leur habitat préférentiel dans le cas où le prédateur ne perçoit pas de proie autour de lui, ce qui permet de le maintenir «en tendance» dans son habitat, ce qui est cohérent avec le modèle conceptuel. Donc, l'influence de l'habitat n'intervient que ponctuellement pour éviter que certains agents se «perdent» hors de leur habitat et, concrètement, un vecteur de déplacement vers la case d'affinité la plus élevée est alors calculé et ajouté au vecteur de déplacement.

Algorithm 10: SEEKPREYS($x, y, t, r_{migration}$)

```

 $p_0 \leftarrow \text{GETCURRENTLOCATION}()$ 
/* Process vector opposite to highest intraspecific density */
 $p_1 \leftarrow \text{HIGHESTDENSITYINNEIGHBORHOOD}(\text{"predators"}, x, y, t)$ 
 $v_1 \leftarrow p_0 - p_1$ 
 $v_1 \leftarrow \frac{v_1}{||v_1||}$ 
/* Process vector towards highest density in preys */
 $p_2 \leftarrow \text{HIGHESTDENSITYINNEIGHBORHOOD}(\text{"preys"}, x, y, t)$ 
 $d_0 \leftarrow \text{GETINTRASPECIFICDENSITY}(\text{"preys"}, p_0)$ 
 $d_2 \leftarrow \text{GETINTRASPECIFICDENSITY}(\text{"preys"}, p_2)$ 
if  $d_0 == 0$  and  $d_2 == 0$ 
    then  $\begin{cases} p_2 \leftarrow \text{HIGESTAFFINITYINNEIGHBORHOOD}(x, y, t) \\ v_2 \leftarrow p_2 - p_0 \end{cases}$ 
    else if  $d_0 \leq d_2$ 
        then  $v_2 \leftarrow 0$ 
    else  $v_2 \leftarrow p_0 - p_2$ 
 $v_2 \leftarrow \frac{v_2}{||v_2||}$ 
MOVE TOWARDS( $(v_1 + v_2) \times r_{migration}$ )
    
```

Paramètre de migration $r_{migration}$: Notons que les deux comportements *migrateAndFlee* et *seekPreys* sont fonction d'un paramètre $r_{migration}$ qui décrit la distance qu'un agent va effectivement parcourir lors de l'activation d'un des deux comportements de migration. Ce paramètre a une unité qui dépend de l'échelle spatiale de l'environnement. Nous proposons un environnement de 128 km de longueur par 128 km de largeur ⁶. Sa superficie est donc de 16384 km².

⁶ Ce choix est motivé par le fait que ce sont les dimensions d'un environnement relativement important et c'est ce type d'échelles qui nous intéresse en premier lieu. De plus, nous préférons des discrétisations qui sont facteur de 2 pour des raisons de performances graphiques. Ces dimensions ne servent toutefois qu'à être cohérent dans la démarche suivie et n'ont aucun impact particulier sur les dynamiques tant que le choix des autres paramètres est fait en fonction de ces dimensions.

Nous proposons de distinguer dans nos comportements de migration le déplacement effectif d'un agent (qui dépend du paramètre $r_{migration}$) du calcul de la direction du déplacement (qui dépend de la discrétisation spatiale). Ainsi, nous limitons l'impact de la discrétisation spatiale sur la dynamique du système au seul processus de perception ou presque. Ce choix se justifie à plusieurs égards. Il est plus facile de ramener le paramètre $r_{migration}$ à un paramètre métier, dont le choix est alors à la discrétion des experts au niveau conceptuel. Il est plus difficile dans le cas du processus de perception d'avoir la même démarche. Enfin, disjoindre la migration de la perception permet de décrire un déplacement «continu» de l'agent dans son environnement.

Le paramètre $r_{migration}$ ne peut toutefois pas être totalement étranger à la discrétisation spatiale. Dans le cas du modèle computationnel que l'on propose, $r_{migration}$ doit être inférieur à la largeur d'une case de l'environnement. Si c'est le cas, l'individu se déplace au plus loin dans une case voisine et la migration est modélisée comme un tropisme de l'agent vers une zone où les conditions sont meilleures pour lui. Sinon, l'agent pourra se déplacer dans des cases au-delà des cases qui sont voisines à la sienne, alors même que sa perception s'est basée sur ces cases voisines. C'est alors la question de la validité du modèle qui est posée et l'on ne peut garantir dans une telle situation des oscillations artificielles du système, voir des comportements aberrants d'agents (sortie de l'environnement, de l'habitat, etc.). Donc, le paramètre de migration conditionne le modèle de la perception. Et dans notre cas, il conditionne le choix de la discrétisation spatiale. En d'autres termes, nous ne pouvons avoir une largeur de case qui soit inférieure au paramètre $r_{migration}$ choisi. Nous proposons d'utiliser $r_{migration} = 1km$ et donc à chaque pas de temps, un individu se déplace d'au plus un kilomètre dans son environnement.

5.6.3 - B Interaction avec l'environnement

Problématique : Nous proposons de nous intéresser à l'influence de la discrétisation spatiale sur le phénomène de distribution spatiale du peuplement. En effet, la discrétisation de l'information de l'environnement conditionne évidemment l'acquisition de cette information par les agents et donc leur comportement de migration.

Moyen : Nous ne nous intéressons qu'à une seule population d'agents, ayant le comportement *migrateAndFlee*. Ce comportement dépend de l'affinité des cellules voisines, de la densité intraspécifique (locale et des cellules voisines) et de la densité de l'autre espèce (locale et des cellules voisines). Nous utilisons l'habitat défini dans le modèle conceptuel et nous faisons varier la discrétisation de cet habitat, ainsi que la description des densités dans l'environnement. L'affinité et les densités interspécifique et intraspécifique ont ainsi la même maille pour une simulation donnée. Notons toutefois que la densité en prédateurs et le processus qui lui est lié n'ont pas d'impact sur la dynamique, car il n'y a pas d'agents de type prédateur. Le nombre d'agents est forcément constant de par notre modèle computationnel et le paramètre de densité maximale n'a plus qu'un intérêt descriptif. Il est cependant important

Paramètre	: Description
$N_a = 10^4$	: Nombre d'agents
Taille de l'environnement	: $128 \times 128 \text{ km}^2$
$t_{max} = 3600$	: Nombre de pas de temps

TAB. 5.9: Sensibilité à la discrétisation spatiale, interaction avec l'environnement : valeurs des paramètres

de le connaître, parce que comme nous l'avons vu précédemment, il peut conditionner la qualité des résultats obtenus. Enfin, le tableau 5.9 présente les principales hypothèses que nous utilisons.

Résultats : Le tableau 5.10 présente les différentes discrétisations utilisées, ainsi que les valeurs de K_{max} et les résultats qui y sont associés. Nous y précisons la superficie d'une case S_{case} ainsi que le rapport du rayon de migration $r_{migration}$, fixé à 1, par la largeur d'une case, ce qui nous permet de nous rendre compte du déplacement effectif possible d'un agent à un pas de temps donné par rapport à la taille de la case. La figure 5.10 présente différentes distributions au cours du temps suivant les différentes discrétisations étudiées. Sur les 3600 pas de temps de la simulation, nous présentons les pas de temps 1, 1800 et 3600 pour chaque discrétisation.

Nous constatons d'abord que notre modèle reproduit la dynamique de stabilité de la distribution spatiale de la population. Nous constatons ensuite que plus la maille est fine, plus la distribution semble homogène. Il faut cependant mettre en perspective le fait qu'une maille fine implique un nombre d'agents plus faible par case. Nous remarquons également une tendance de la population à s'agréger sur les bords ou à certains endroits de l'habitat. C'est d'autant plus vrai que la discrétisation est faible (16×16). Remarquons que pour un modèle donné, la distribution initiale du peuplement étant aléatoire, chaque simulation est différente. Les points de densité élevée semblent être des points d'équilibre qui ne sont pas forcément à la même place pour deux simulations. Remarquons également que de par le modèle même de migration des agents, de telles dynamiques ne sont pas étonnantes. Il faudrait pouvoir étudier davantage la modélisation de processus de migration sur des systèmes aussi minimaux pour être catégorique sur la validité de telles dynamiques. On constate donc qu'un même modèle, dans les mêmes conditions, à part pour la discrétisation spatiale, donne des résultats aussi différents que ceux de la figure 5.10c et 5.10l. Si nous avions été confrontés à un modèle complexe, nous aurions pu voir dans ces pics de densité l'expression de processus dûs au modèle conceptuel, d'interactions particulières entre différents peuplements par exemple, là où seule la discrétisation spatiale peut expliquer de telles hétérogénéités dans la distribution.

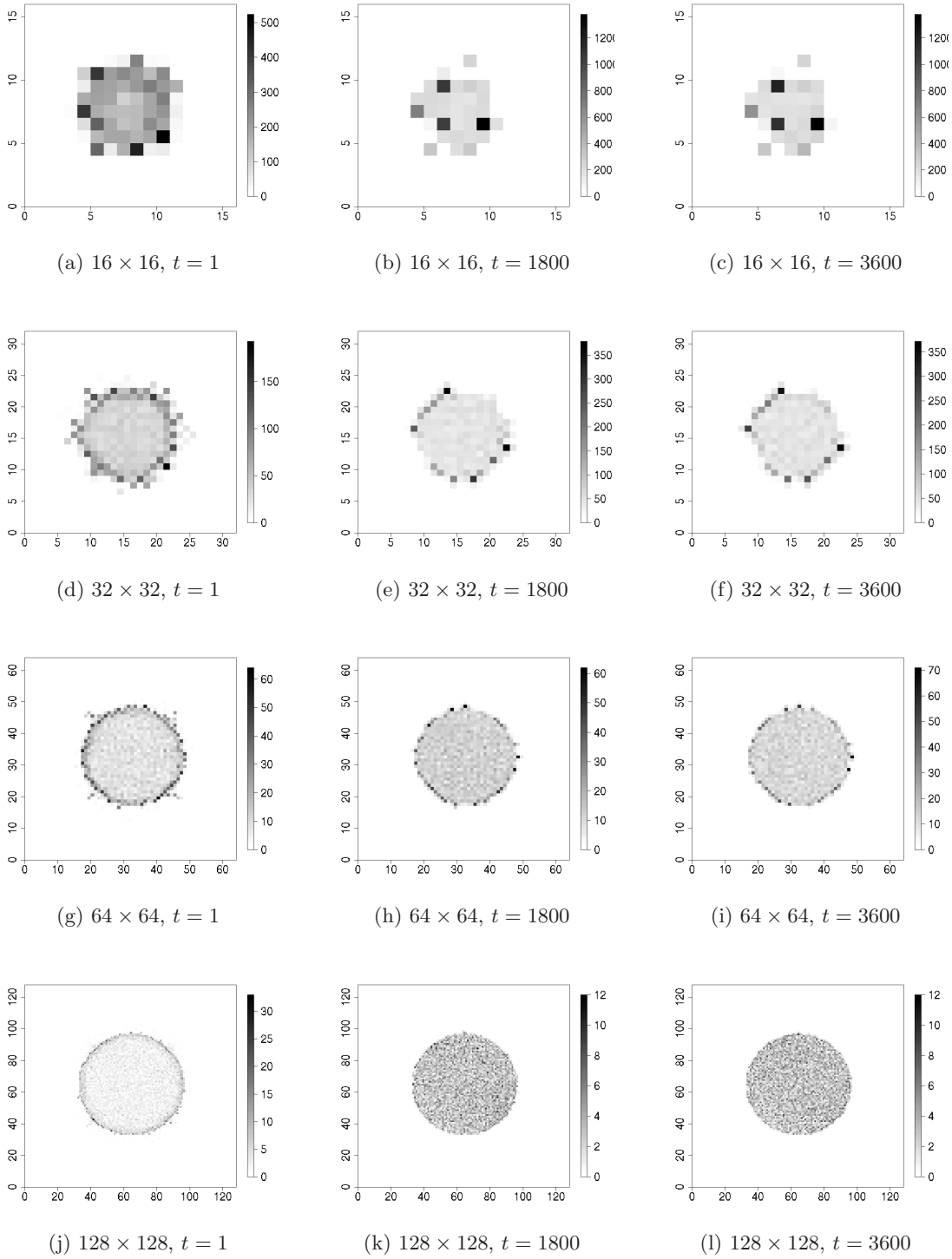


FIG. 5.10: Discretisation spatiale homogène – Distributions spatiales du peuplement en fonction de la discretisation spatiale.

Discrétisation	S_{case}	K_{max}	$r_{migration}/l_{case}$
16×16	$16 \times 16 \text{ km}^2$	90.93	0.0625
32×32	$8 \times 8 \text{ km}^2$	26.61	0.125
64×64	$4 \times 4 \text{ km}^2$	7.23	0.25
128×128	$2 \times 2 \text{ km}^2$	1.89	0.5

TAB. 5.10: Sensibilité à la discrétisation spatiale, interaction avec l'environnement : simulations et résultats.

Paramètre	Description
$N_{a,proies} = N_{a,predateurs} = 5.10^3$	Nombre d'agents de type proies ($N_{a,proies}$) et de type prédateurs ($N_{a,predateurs}$).
K_{max}	Densité locale maximale, calculée en fonction du nombre total d'agents espéré
$t_{max} = 3600$	Nombre total de pas de temps

TAB. 5.11: Sensibilité à la discrétisation spatiale, interaction entre les agents : valeurs des paramètres.

5.6.3 - C Interaction entre les agents

Problématique : Nous proposons maintenant de nous intéresser à l'influence de la discrétisation spatiale sur le phénomène de perception par un agent des autres agents. Nous nous plaçons cette fois-ci dans le cadre d'un système proies-prédateurs.

Moyen : Déjà dans l'expérience précédente, la migration des agents (type proie) était fonction de la densité intraspécifique. Nous avons vu qu'il nous fallait introduire un tel processus si nous voulions que certains agents restent dans les zones de faible affinité et donc occupent «mieux» leur habitat (et également que l'agglomération des agents en périphérie de l'habitat à affinité maximale soit limitée). En introduisant des agents de type prédateurs, nous permettons aux proies de s'éloigner de la zone la plus densément peuplée en prédateurs. Pour les prédateurs, le comportement de migration est pratiquement uniquement fonction de processus liés à la densité. En effet, un prédateur cherche à s'éloigner de la zone la plus densément peuplée en prédateurs et à se rapprocher de la zone la plus densément peuplée en proies⁷. Nous étudions alors ce système en faisant à nouveau varier la discrétisation des informations prises dans l'environnement (affinité et densités intraspécifique et interspécifique). Enfin, le tableau 5.11 présente les principales hypothèses que nous utilisons.

Résultats : Le tableau 5.12 présente les résultats des différentes simulations effectuées

⁷ Comme nous l'avons dit, l'affinité à son environnement n'est prise en compte par le prédateur que quand celui-ci se retrouve isolé des proies, ce qui évite que des agents prédateurs sortent de leur habitat pour se perdre dans l'environnement.

Discrétisation	S_{case}	K_{max}	$r_{migration}/l_{case}$
16×16	$16 \times 16 km^2$	90.93	0.0625
32×32	$8 \times 8 km^2$	26.61	0.125
64×64	$4 \times 4 km^2$	7.23	0.25
128×128	$2 \times 2 km^2$	1.89	0.5

TAB. 5.12: Sensibilité à la discrétisation spatiale, interaction entre les agents : simulations et résultats.

en fonction de la discrétisation spatiale. Nous y présentons également la surface d'une case suivant la discrétisation de l'environnement, la densité maximale par case K_{max} qui en découle et le rapport du rayon de migration et de la largeur d'une case. Les figures 5.11 et 5.12 présentent différentes distributions au cours du temps suivant les différentes discrétisations étudiées. A nouveau, nous présentons les pas de temps 1, 1800 et 3600 pour chaque discrétisation et pour les deux types d'agents (proies et prédateurs). Il apparaît avant toute chose que le modèle computationnel reproduit la dynamique de stabilité de la distribution spatiale pour les deux populations. Concernant la qualité de la distribution dans l'habitat, elles sont plus ou moins homogènes. Pour la population de prédateurs, l'homogénéité est moins sûre et les agents prédateurs semblent s'organiser en bandes. Plus la discrétisation est grande (maille fine), plus la distribution semble être homogène. Seulement, ce résultat est là encore à mettre en perspective avec le fait qu'une grande discrétisation entraîne un nombre d'agents par case plus faible. On notera simplement qu'une maille très fine permet d'obtenir une distribution des agents qui semble être conforme au modèle conceptuel.

5.6.4 Sensibilité à la discrétisation temporelle

5.6.4 - A Modèle computationnel – comportements des agents

Le dernier facteur de discrétisation est lié à la représentation du temps. Nous proposons pour cette étude de re-formuler à nouveau le modèle conceptuel et de définir une échelle temporelle

Pour souligner l'impact de la discrétisation temporelle sur les résultats, nous proposons de revenir sur la formulation du modèle conceptuel. En particulier, nous proposons d'étudier à nouveau les phénomènes de création et de destruction d'agents. Nous réutilisons le modèle computationnel de la section 5.6.2 - A. Nous considérons donc les comportements agents *mate* (algorithme 6), *die* (algorithme 7) et *recruit* (algorithme 8). La différence avec l'étude de la sensibilité à la discrétisation en nombre d'agents (section 5.6.2) est la mise du système dans un contexte temporel défini. Notons que le comportement de recrutement *recruit*, n'étant activé qu'une seule fois par l'agent à sa création, n'est pas influencé par la discrétisation temporelle.

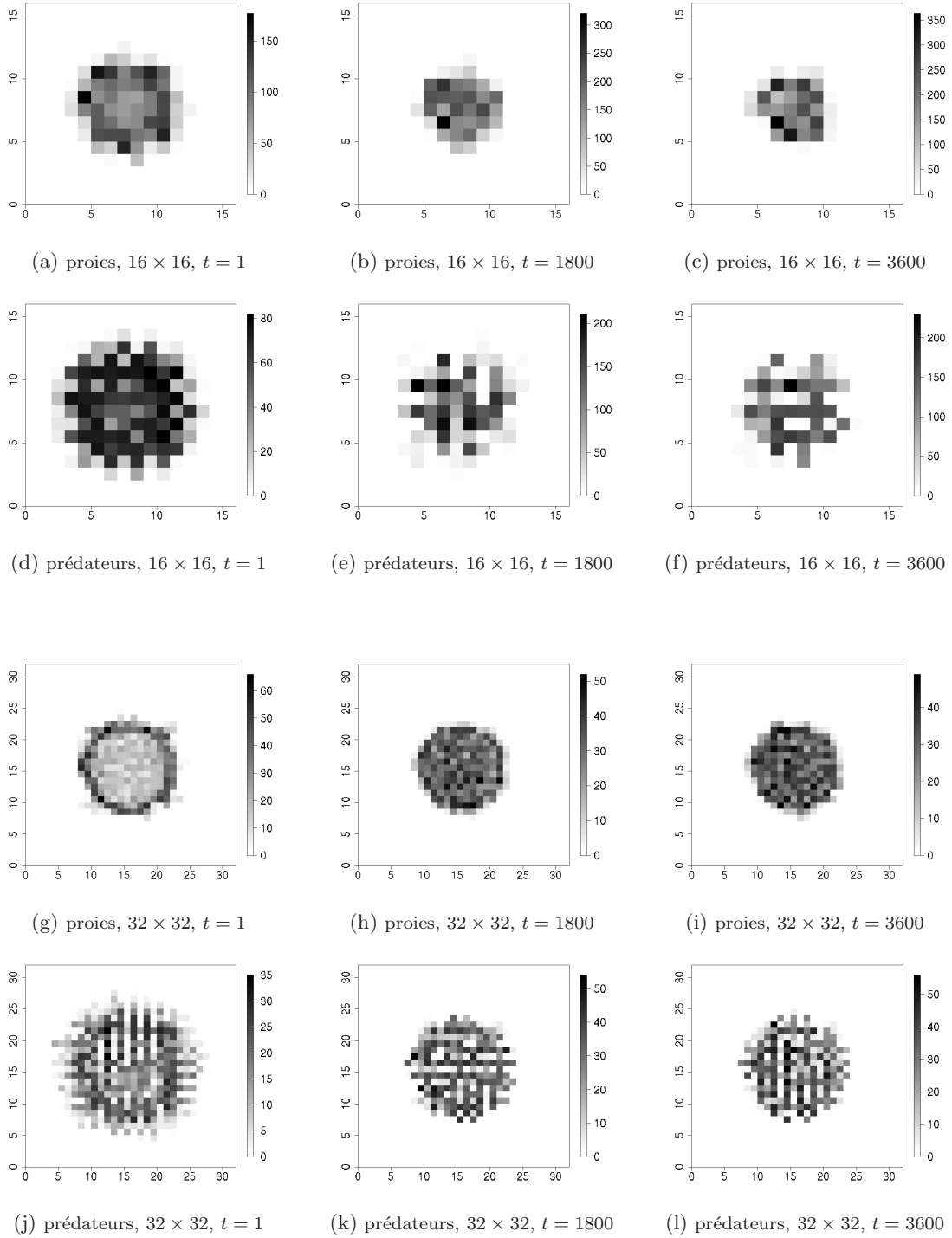


FIG. 5.11: Influence de la discrétisation spatiale, interaction entre agents – Evolution du nombre d'agents en fonction du temps (proies et prédateurs).

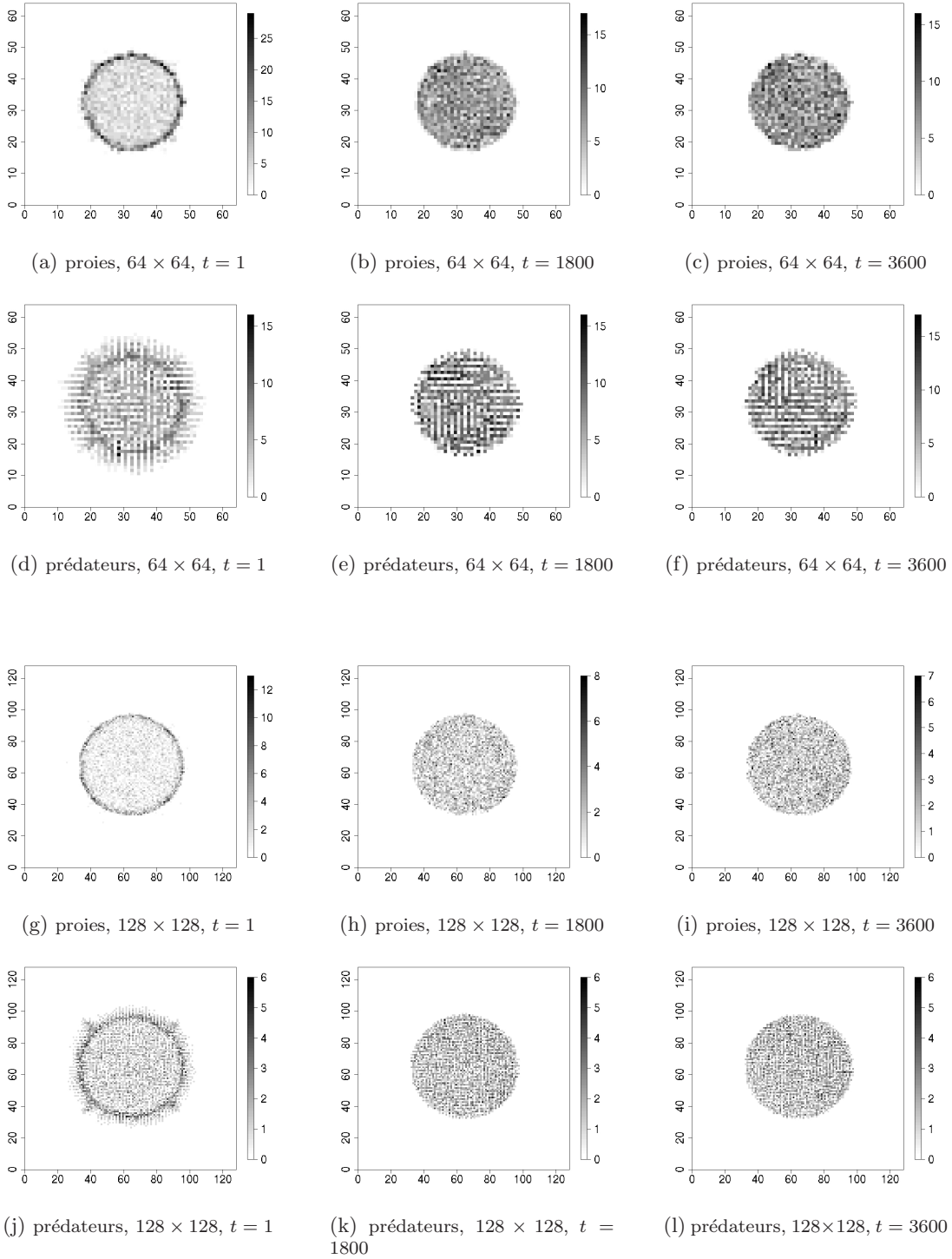


FIG. 5.12: Influence de la discrétisation spatiale, interaction entre agents – Evolution du nombre d'agents en fonction du temps (proies et prédateurs).

De même, le comportement de mortalité ne dépend que de l'affinité et de la densité et n'est donc pas non plus fonction du pas de temps. Seul le comportement de reproduction est fonction du pas de temps.

Comportement de reproduction : Le comportement *mate*, décrit dans 5.6.2 - A, dépend de l'affinité et surtout du paramètre τ_Φ . C'est ce paramètre qui dépend implicitement du temps et qui a donc une unité. Pour qu'un changement du pas de temps influence le système, il suffit donc de pouvoir convertir le paramètre τ_Φ afin que l'accumulateur *potential* soit incrémenté à chaque pas de temps de manière cohérente par rapport au pas de temps. Par exemple, si un agent active *mate* tous les jours et que celui-ci incrémente de 0.003 son accumulateur, alors si l'agent n'active *mate* que tous les mois, pour avoir le même comportement, il devra incrémenter son accumulateur de 30×0.003 . τ_Φ s'exprime donc ainsi : $\tau_\Phi^u = \tau_\Phi^{jour} \times c_{jour,u}$, avec c_{u_1,u_2} le facteur d'échelle ou de conversion de l'unité u_1 en u_2 .

Algorithm 11: MATE(x, y, t, τ_Φ^u)

```

external potential : real, FLOOR(real) : natural,  $\alpha_H(x : real, y : real) : real$ , NEWAGENT(natural)
potential  $\leftarrow$  potential + ( $\alpha_H(x, y) \times \tau_\Phi^u$ )
NEWAGENT(FLOOR(potential))
potential  $\leftarrow$  (potential - FLOOR(potential))
    
```

5.6.4 - B Cas d'une population

Problématique : Nous étudions l'impact de la discrétisation temporelle sur la dynamique d'un peuplement d'agents.

Moyen : Nous considérons l'évolution d'un peuplement d'agents sur une période de dix ans. Nous utilisons comme hypothèse de départ $\tau_\Phi^{jour} = 0.003$. Nous mettons alors en œuvre le système avec différents pas de temps en convertissant, suivant le pas de temps, τ_Φ à la bonne unité (voir tableau 5.13). De plus, l'environnement (affinité et distribution des agents) est discrétisé en 16×16 et l'habitat du modèle conceptuel est utilisé. Le système d'unités du temps est approché pour simplifier le problème et un mois vaut 30 jours, une année 360 jours. Enfin, le tableau 5.13 présente les principales hypothèses que nous utilisons.

Résultats : Le tableau 5.14 présente les différents pas de temps utilisés, et pour chacun d'eux, le paramètre τ_Φ correspondant, la moyenne en nombre d'agents obtenues au cours de la simulation, la déviation standard associée à cette moyenne, ainsi que le coefficient de variation. La figure 5.13 présente l'évolution du nombre d'agents au cours de la simulation en fonction du temps. On constate que les allures des courbes sont avant toute chose très similaires (à part pour la fréquence de changement d'état du système, ce qui est évident) et que l'on reproduit la dynamique de stabilité en nombre d'agents recherchée. Enfin, la figure

Paramètre	: Description
$potential_{init} = 0$	: Initialisation du paramètre <i>potential</i> (algo. 11)
$K_{max} = 129$	: Densité locale maximale, calculée en fonction du nombre total d'agents espéré
$t_{max} = 3600$	: Nombre total de pas de temps
$N_{init} = 10^4$	: Nombre initial d'agents

TAB. 5.13: Sensibilité à la discrétisation temporelle, valeurs des paramètres.

Unité du pas de temps	τ_Φ	Moyenne	Déviati�n standard	Coefficient de variation
jour	0.003	10229	20.6	0.002
mois	0.09	10226	29.95	0.0029
6 mois	0.54	10216	52.15	0.0051
année	1.08	10198	70.71	0.0069

TAB. 5.14: Sensibilité à la discrétisation temporelle : simulations et résultats.

5.14 présente l'évolution du coefficient de variation en fonction du pas de temps utilisé. On constate alors que plus le pas de temps est petit, plus le coefficient de variation est faible et il semblerait que cette tendance soit plutôt linéaire. Remarquons que cela ne pas être dû à un τ_Φ plus élevé quand le pas de temps est grand. En effet, on pourrait penser que si le τ_Φ des comportements de reproduction des agents est plus fort, il pourrait induire une dynamique globale plus violente. Nous avons mesuré l'écart d entre les valeurs (en nombre d'agents) minimale et maximale obtenues au cours des différentes simulations. Les résultats sont présentés dans le tableau 5.15. Pour la simulation avec un pas de temps de 360, l'écart entre la quantité d'agents minimale et la quantité maximale a été de 73, alors que pour celle avec un pas de temps de 1, l'écart a été de 106 (ce qui n'est pas significativement différent). Par conséquent, la diminution du coefficient de variation est bien due à l'augmentation de la fréquence d'activation du comportement de reproduction. Remarquons cependant que cette amélioration n'est que d'un facteur 3 pour une fréquence 360 fois plus élevée.

unité du pas de temps	jour	mois	6 mois	année
d	106	107	53	73

TAB. 5.15: Sensibilité à la discrétisation temporelle : écart d entre les valeurs (nombre d'agents) minimale et maximale suivant le pas de d'activation.

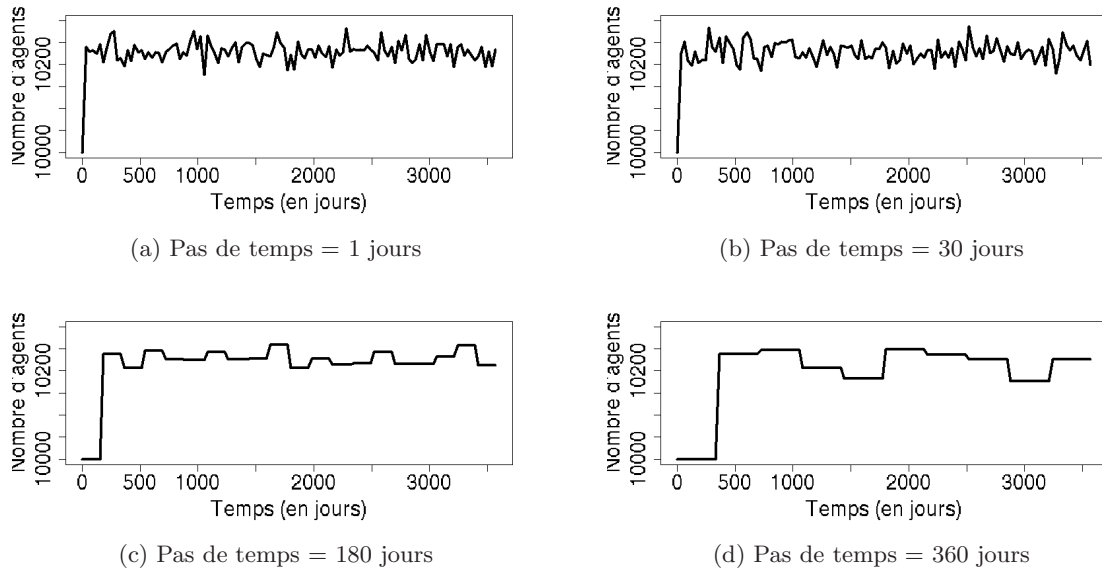


FIG. 5.13: Discrétisation homogène temporelle : évolution du nombre d'agents en fonction du temps.

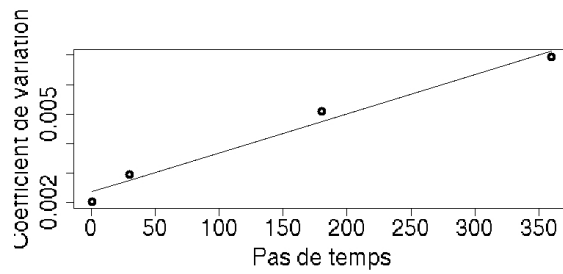


FIG. 5.14: Discrétisation homogène temporelle : coefficient de variation en fonction du pas de temps.

5.6.5 Analyse

Dans cette section, nous avons étudié les biais computationnels liés à la discrétisation en nombre d'agents et aux discrétisations spatiale et temporelle dans des modèles de peuplements d'agents. Nous nous sommes tout d'abord intéressé à la sensibilité d'un modèle de peuplement à la discrétisation en nombre d'agents. Le modèle, structuré par les processus de création et de destruction d'agents et de distribution dans l'habitat, nous a tout d'abord permis de confirmer qu'augmenter le nombre d'agents améliore la précision des résultats. Il nous a permis de constater que cette amélioration ne semble pas linéaire. Dans tous les cas, on voit bien qu'asymptotiquement, le gain est nul. Si l'on regarde les résultats obtenus sur la dynamique de la distribution spatiale du peuplement, ceux-ci vont dans le même sens. Il n'est toutefois pas possible d'interpréter les résultats obtenus comme on l'a fait en considérant les moyennes en nombre d'agents. Il faudrait pour cela travailler sur des distributions moyennes également. On peut cependant tout de même voir qu'un nombre d'agents élevé nous a permis d'obtenir des distributions très proches de la distribution théorique du modèle conceptuel. Enfin, ces résultats permettent de penser que si le nombre d'agents par case est trop faible, on risque d'obtenir des résultats qui ne sont absolument pas liés au modèle conceptuel, mais simplement à une discrétisation en agents non adaptée. Quand on considère des modèles complexes, de tels biais peuvent alors mener à de mauvaises interprétations des résultats.

Nous avons ensuite étudié les biais entraînés par la discrétisation spatiale sur un modèle de peuplements qui n'était gouverné que par des comportements de migration. Nous avons vu tout d'abord que les résultats obtenus dans le cas des expérimentations sur une seule population sont visuellement très différents suivant la discrétisation spatiale utilisée. Il est clair que le modèle étant minimal, il porte en lui-même des biais qui expliquent certains pics de densité qui sont alors des points d'équilibre (temporaires) ou des sortes de centre de gravité pour les agents. Seulement le modèle se comporte globalement très différemment suivant la discrétisation. Il faut cependant mettre en perspective le fait que plus la discrétisation spatiale est grande, plus le nombre d'agents par case est faible. Or, nous avons vu dans l'étude sur la sensibilité au nombre d'agents que cela pouvait poser des problèmes. Pour l'étude du système proies-prédateurs, là encore les résultats confirment que le système est sensible à la discrétisation spatiale. Les distributions sont cependant plus homogènes et régulières et globalement le système se comporte conformément au modèle conceptuel. Ceci nous pousse à penser que les processus d'interaction entre populations peuvent masquer l'impact des biais que peut entraîner la discrétisation spatiale.

Enfin, nous nous sommes intéressés à la sensibilité d'un modèle de peuplement à la discrétisation temporelle. Les résultats obtenus sont plus étonnants que pour la sensibilité à la discrétisation spatiale. En effet, nous montrons que la précision augmente plus la fréquence d'activation des comportements (dans notre cas, le comportement de reproduction) est grande alors même que les écarts entre les valeurs maximale et minimale en nombre d'agents augmentent eux aussi. L'amélioration de la précision semble cette fois-ci linéaire. De plus,

et c'est un résultat plus attendu, entre un pas d'activation d'un an et un pas d'activation d'un jour, la précision n'a pas été dramatiquement améliorée. Enfin, remarquons que la sensibilité du système à la discrétisation temporelle pourrait être bien plus critique avec d'autres modèles, comme un modèle de peuplements dont les individus se déplaceraient dans l'environnement par exemple. Plus la fréquence d'activation des comportements serait faible, plus l'amplitude des déplacements seraient grande. La discrétisation temporelle aurait alors sûrement un impact bien plus fort sur les résultats.

5.7 Sensibilités d'un IBM aux discrétisations hétérogènes

5.7.1 Problématique

Dans cette section, nous considérons la question des discrétisations hétérogènes et étudions la sensibilité de modèles de peuplements à ces biais computationnels qui introduisent une hétérogénéité d'échelle. Nous entendons par discrétisations hétérogènes, des discrétisations qui se traduisent par des échelles de description différentes au sein d'un modèle donné et des problématiques d'interaction entre éléments (de ce modèle) décrits différemment en termes d'espace, de temps et de nombre d'agents.

Si l'on reprend l'étude de la section précédente, nous nous positionnons dans la suite de la démarche en nous posant les questions suivantes : 1° Nous avons étudié l'impact de la discrétisation en nombre d'agents sur les résultats. Mais si la discrétisation en nombre d'agents est un artifice computationnel qui permet de simuler avec une certaine précision un peuplement d'agents, quel est alors l'impact de cet artifice computationnel si l'on met en œuvre deux populations d'agents en interaction décrites chacune avec une discrétisation en nombre d'agents qui lui est propre. Et quelles sont les hypothèses supplémentaires à faire pour pouvoir mettre en œuvre une telle étude ? 2° Nous avons vu qu'un modèle de deux populations en interaction n'était que peu sensible à la discrétisation spatiale. Mais est-ce toujours le cas quand ces deux populations évoluent dans des environnements dont la discrétisation est différente. En d'autres termes, pourquoi les deux peuplements devraient-ils évoluer dans un environnement avec une discrétisation spatiale unique ? 3° Nous avons pu constater que la fréquence d'activation des comportements, tant qu'elle est la même pour tous les comportements, n'a pas une influence conséquente sur le système, même si une fréquence plus élevée améliore légèrement les résultats. Dans ce cas, qu'en est-il quand les comportements d'un agent n'ont pas la même fréquence ? Plus encore, quel est l'impact sur les résultats que deux populations interagissent alors même qu'elles évoluent à des rythmes temporels différents ?

Remarquons que nous nous concentrons ici sur la question de l'hétérogénéité d'échelle en tant que solution computationnelle permettant l'implémentation d'un modèle conceptuel qui requiert des précisions multiples. On peut dans ce cadre là se demander où commencent les problématiques strictement multi-échelles, en d'autres termes où est la limite entre artifice computationnel et solution multi-échelle conceptuelle. De manière pragmatique, nous proposons de continuer dans notre démarche d'identification de systèmes minimaux permettant d'appréhender chaque problématique de manière singulière. Nous pourrions alors identifier petit à petit les éléments de description d'échelle permettant l'interaction entre les différents phénomènes.

5.7.2 Sensibilité à la discrétisation en nombre d'agents

5.7.2 - A Modèle computationnel – facteur d'échelle et comportements des agents

Jusqu'à présent, un agent représentait un individu du modèle conceptuel. Mais envisager une interaction entre deux populations n'ayant pas la même discrétisation en nombre d'agents nous oblige à poser la question de la place de l'agent dans le modèle computationnel par rapport aux individus conceptuels. Introduire un facteur d'échelle est alors la solution la plus simple pour exhiber dans le modèle computationnel la discrétisation en nombre d'agents.

Facteur d'échelle : Il décrit la granularité en nombre d'agents du modèle computationnel. Il correspond ainsi au nombre d'individus qu'un agent représente dans le système. En d'autres termes, si l'on modélise une population de 10^6 individus avec 10^3 agents, alors le facteur d'échelle est de 10^3 .

Exhiber ainsi un facteur d'échelle nous permet de manipuler directement les valeurs des paramètres conceptuels sans avoir à les convertir suivant la discrétisation en nombre d'agents. Par exemple, si le modèle conceptuel décrit une densité maximale locale (exprimée en nombre d'individus conceptuels), elle peut être directement utilisée dans le modèle computationnel. Quand précédemment nous avons implémenté des IBM dans des SMA, nous avons implicitement utilisé ce facteur d'échelle en choisissant une granularité en nombre d'agents. Le facteur d'échelle est donc un artifice computationnel que l'on explicite.

Comportements de reproduction et de recrutement : Les modèles computationnels sont ceux présentés dans la section 5.6.2 - A. Ces comportements ne dépendant pas d'interactions entre agents, il n'est en effet pas nécessaire d'y introduire un facteur d'échelle. Nous utilisons donc l'algorithme 6 pour le comportement de reproduction et l'algorithme 8 pour celui du recrutement.

Notons tout de même que le comportement de recrutement révèle la limite du facteur d'échelle. En effet, un agent peut représenter plusieurs individus conceptuels, il ne reproduit

pour autant qu'une seule histoire particulière, qu'un seul trait de vie. Cette approximation prend du sens dans la relation de l'agent avec son environnement, mais est peut-être plus critique encore dans les interactions avec d'autres agents, et c'est là que porte notre étude.

Comportement de mortalité : Nous ré-utilisons l'algorithme 7. Mais, comme il dépend de la capacité de charge du milieu, ce comportement est fonction du nombre d'individus conceptuels que le milieu peut accueillir, il est donc fonction du nombre d'agents et il est nécessaire d'y intégrer un facteur d'échelle. Le comportement vérifie que la densité intraspécifique locale à l'agent est inférieure à la capacité locale du milieu (densité intraspécifique maximale en nombre d'individus conceptuels par case multipliée par l'affinité locale). Intégrer le facteur d'échelle oblige alors à convertir la densité locale perçue par l'agent et exprimée en nombre d'agents en une densité locale en nombre d'individus conceptuels. Mais, en réalité, comme nous comparons ici une densité locale d'agents de type proies à une capacité locale du milieu exprimée en nombre d'individus conceptuels de types proies, et comme le facteur d'échelle est le même pour tous les agents de même type, il est tout aussi simple de calculer pour chaque simulation la densité maximale par case en termes de nombre d'agents (densité qui est alors calculée en fonction du facteur d'échelle choisi).

Comportement de prédation : Ce comportement est seulement exécuté par les agents de type prédateur. Il permet de reproduire l'interaction trophique entre les populations : les proies et les prédateurs. Il dépend donc fortement des autres agents, il est donc nécessaire d'y intégrer un facteur d'échelle. Le comportement *hunt* dépend de deux paramètres : Q_{max} et Q_{min} qui sont respectivement la quantité de proies qu'un prédateur cherche à manger à chaque activation de *hunt* et la quantité minimum qu'il doit manger pour survivre. C'est donc en d'autres termes, respectivement l'impact de la prédation sur la population de proies et l'impact de la prédation sur la population de prédateurs. Q_{max} et Q_{min} sont exprimés en nombre d'individus qu'un prédateur mange, sachant qu'un agent prédateur représente lui-même un certain nombre d'individus.

Algorithm 12: HUNT()

```

external  $Q_{min} : natural, Q_{max} : natural$ 
external LOOKFORPREYSAROUND() : boolean, GETRANDOMPREYAROUND() : Agent, DIE()
 $Q \leftarrow 0$ 
 $Q_{agent,max} = Q_{max} * a_P / a_p$ 
 $Q_{agent,min} = Q_{min} * a_P / a_p$ 
while  $Q < Q_{agent,max}$ 
    if LOOKFORPREYSAROUND() == false
        then BREAK()
    do {
         $p \leftarrow$  GETRANDOMPREYAROUND()
         $p.DIE()$ 
         $Q \leftarrow Q + 1$ 
    }
if  $Q < Q_{agent,min}$ 
    then DIE()
    
```

Valeur des paramètres	
Paramètre	: Description
$N_{i,proies} = 10^6$	: Nombre d'individus de type proie espérés
$N_{i,predateurs} = 10^4$	: Nombre d'individus de type prédateur espérés
$K_{max,proies} = 129$	: Densité maximale par case pour la population de proies (exprimée en nombre d'individus)
$K_{max,predateurs} = 12827$	: Densité maximale par case pour la population de prédateurs (exprimée en nombre d'individus)
$t_{max} = 3600$	: Nombre de pas de temps

TAB. 5.16: Hétérogénéité du nombre d'agents : valeurs des paramètres.

5.7.2 - B Hétérogénéité du nombre d'agents

Problématique : Nous proposons d'étudier la mise en oeuvre du modèle computationnel précédent avec deux peuplements qui ont chacun leur propre discrétisation en nombre d'agents. Il s'agit donc d'observer l'impact sur les résultats de ces discrétisations hétérogènes et d'étudier l'interaction multi-échelles intervenant entre ces deux populations d'agents.

Moyen : Le tableau 5.16 présente les hypothèses, ainsi que les valeurs des paramètres utilisés pour les simulations. L'habitat des populations est toujours celui décrit dans le modèle conceptuel et la discrétisation spatiale de l'environnement est celle utilisée pour l'étude de la discrétisation homogène en nombre d'agents (16×16 cases). Nous utilisons donc pour reproduire ces populations le modèle computationnel décrit précédemment (section 5.7.2 - A). Notons que nous savons de plus que ce modèle a permis de reproduire les dynamiques recherchées dans le cas d'une population (section 5.6.2 - B). Nous étudions ici la discrétisation en nombre d'agents, donc le facteur étudié est le nombre d'agents que nous souhaitons avoir pour représenter chacune des populations considérées. Nous pouvons alors à partir du nombre d'agents pour une population donnée connaître le facteur d'échelle de cette population.

Nous considérons deux populations telles que : 1° la population de proies regroupe en moyenne 10^6 individus et 2° celle des prédateurs en compte en moyenne 10^4 . Notons que nous savons qu'à partir de 500 agents et pour le cas d'une seule population, les résultats ont révélés un coefficient de variation inférieur ou égal à 3.10^{-3} . Nous faisons donc varier le nombre d'agents autour de cette valeur et jusqu'à atteindre 5000 agents pour une population donnée.

Résultats : Le tableau 5.17 présente les résultats obtenus. Nous y faisons figurer de gauche à droite : le nombre d'agents de la population de proies $N_{a,proies}$, son facteur d'échelle a_p , la moyenne puis la déviation standard pour la population de proies, le nombre d'agents de la population de prédateurs $N_{a,predateurs}$, son facteur d'échelle a_P , la moyenne puis la déviation standard pour la population de prédateurs. La moyenne pour chaque simulation correspond à la moyenne des individus au cours du temps.

Simulations					
$N_{a,proies}, a_p$	moyenne	déviations standard	$N_{a,prédateurs}, a_P$	moyenne	déviations standard
5000, 200	978343	3271	5000, 2	10003	13
1000, 1000	836317	24590	1000, 10	9562	55
500, 2000	777650	32292	500, 20	9484	104
100, 10000	631250	46010	100, 100	7129	399
5000, 200	989015	90353	1000, 10	0	0
5000, 200	990970	70540	500, 20	0	0
5000, 200	824072	50308	100, 100	2120	500
1000, 1000	957425	5506	5000, 2	10005	11
500, 2000	954283	9935	5000, 2	10007	13
100, 10000	817583	33632	5000, 2	10004	12

TAB. 5.17: Hétérogénéité du nombre d'agents : simulations et résultats.

Les figures 5.15, 5.16 et 5.17 présentent les évolutions au cours du temps des populations de proies (à gauche) et de prédateurs (à droite) suivant les différents facteurs d'échelle utilisés. Rappelons que nous nous attendions à avoir 10^6 individus de type proie et 10^4 individus de type prédateur pour chaque simulation. Nous remarquons alors que pour les quatre premières simulations (figure 5.15 ou les quatre premières rangées du tableau 5.17) plus le facteur d'échelle est grand, et donc moins il y a d'agents, plus l'écart par rapport à la valeur espérée est grande. Il y a une perte de précision qui est fonction du nombre d'agents et qui plus spectaculaire que dans l'étude sur la discrétisation homogène à cause du facteur d'échelle.

Nous remarquons ensuite que les trois simulations, pour lesquelles le facteur d'échelle des proies était minimal ($= 200$) et que le facteur d'échelle des prédateurs était supérieur à 2, ont donné des résultats dramatiques pour la population de prédateurs qui s'est soit éteinte, soit est restée aux alentours de 2000 individus (voir figure 5.16). Enfin les trois dernières simulations (figures 5.16 et 5.17), pour lesquelles le facteur d'échelle des prédateurs étaient minimal ($= 2$) et celui des proies supérieur à 100, ont donné de bons résultats.

Pour conclure, ce que l'on peut voir, c'est que toutes les simulations dont le nombre d'agents pour la populations de prédateurs étaient à 5000 ont donné de très bons résultats et ce, quelque fut le nombre d'agents de la population de proies.

5.7.3 Sensibilité à la discrétisation spatiale

Problématique : Cette étude porte sur la sensibilité d'un modèle de peuplements à la discrétisation spatiale hétérogène dans le cas d'un système de deux populations d'agents. Ces deux populations évoluent dans des environnements ayant des discrétisations différentes. La

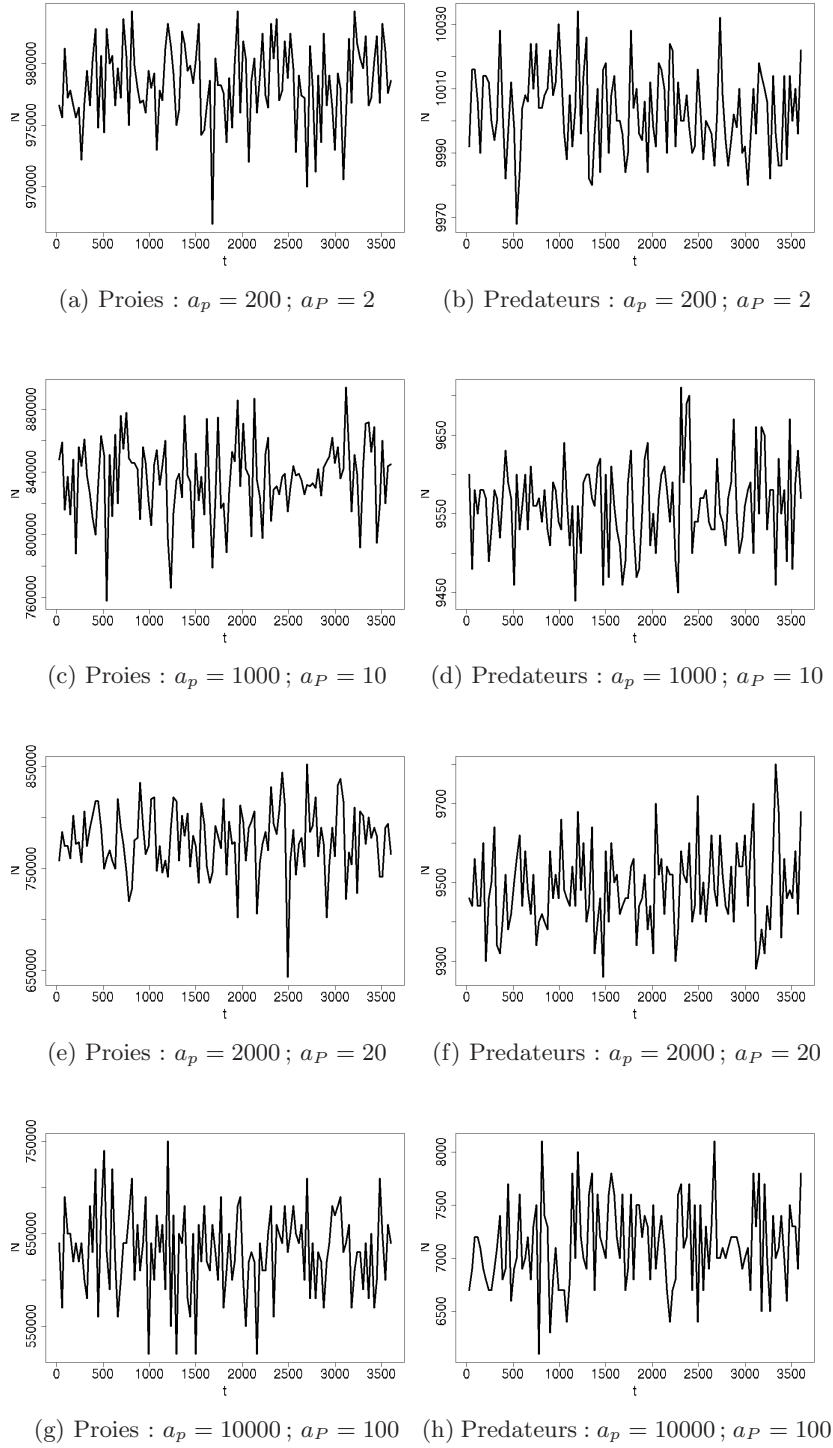


FIG. 5.15: Hétérogénéité du nombre d'agents : évolution du nombre d'agents en fonction du temps (a_p : facteur d'échelle des proies, a_P : facteur d'échelles des prédateurs).

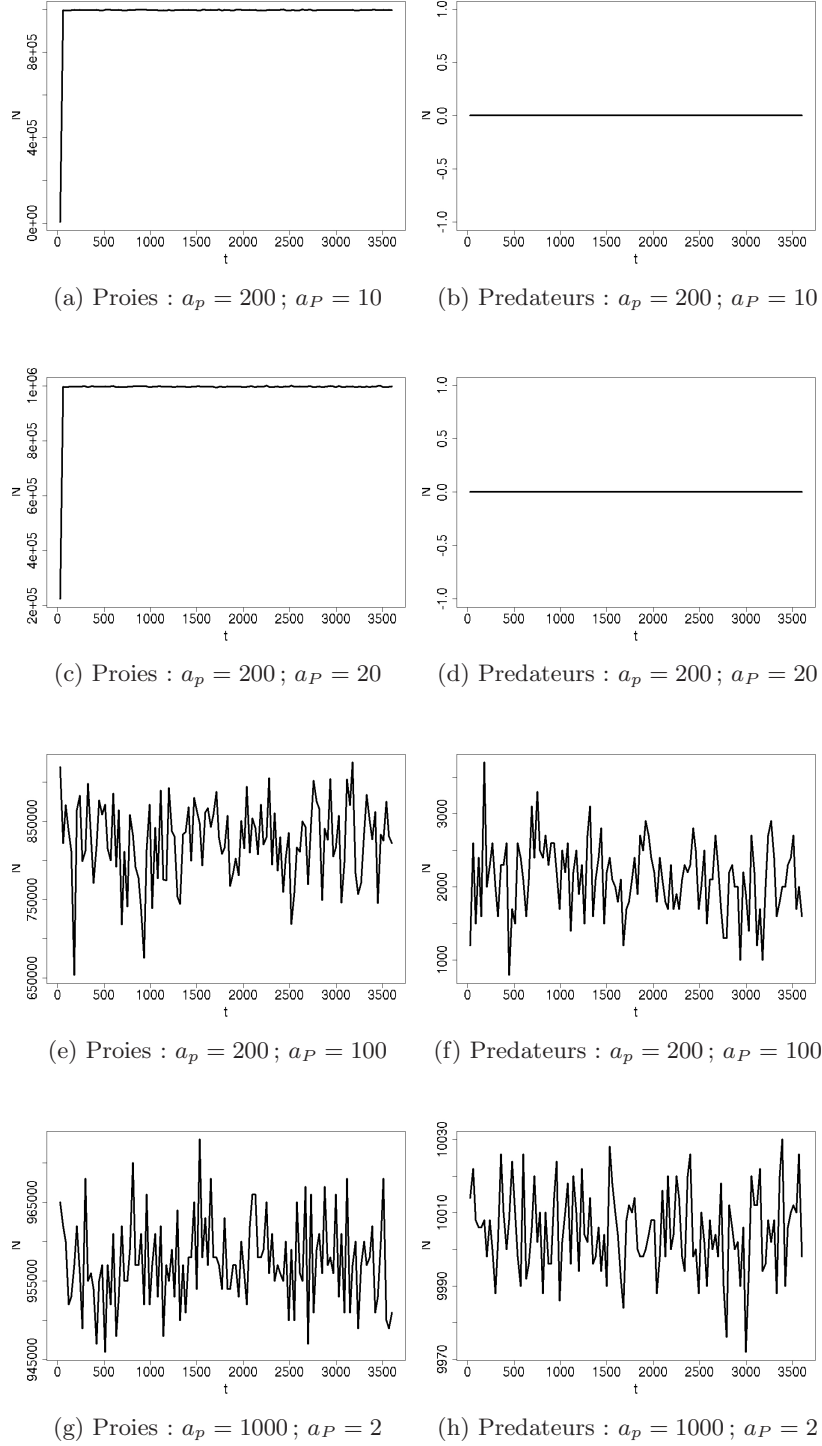


FIG. 5.16: Hétérogénéité du nombre d'agents : évolution du nombre d'agents en fonction du temps (a_p : facteur d'échelle des proies, a_P : facteur d'échelles des prédateurs).

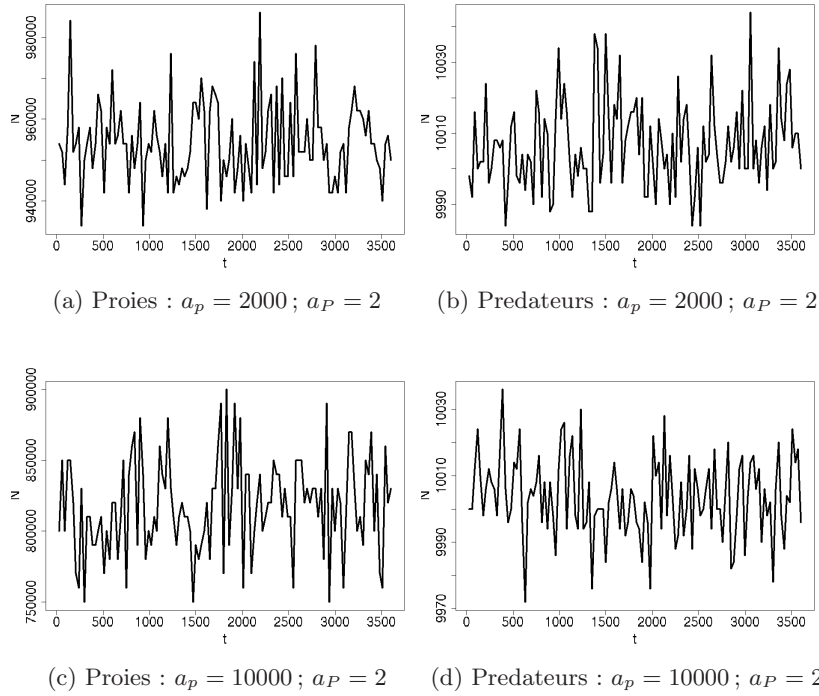


FIG. 5.17: Hétérogénéité du nombre d'agents : évolution du nombre d'agents en fonction du temps (a_p : facteur d'échelle des proies, a_P : facteur d'échelles des prédateurs).

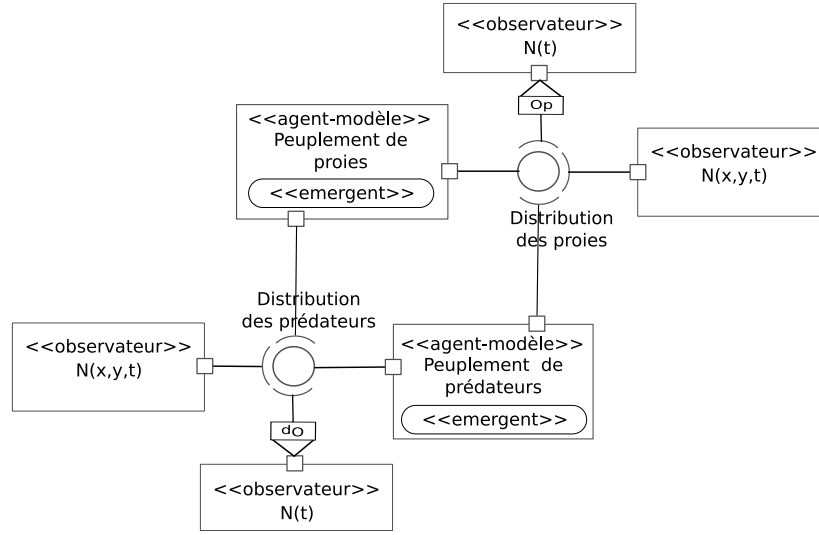


FIG. 5.18: Diagramme d'agents-modèles – cas de deux agents-modèles en interaction trophique.

question est donc d'étudier l'impact sur les résultats de cette hétérogénéité de la discrétisation spatiale et de l'interaction multi-échelles entre les deux type d'agents.

Moyen : Dans le prolongement de l'étude faite dans la section 5.6.3 sur la discrétisation spatiale homogène, nous ré-utilisons le modèle computationnel de cette étude (voir la section 5.6.3 - A). Nous nous intéressons par contre ici uniquement au cas de deux populations de proies et de prédateurs en interaction.

Nous utilisons pour effectuer cette étude deux *agents-modèles*. En effet, pour pouvoir discrétiser différemment le modèle des proies et celui des prédateurs, il nous faut deux environnements. La solution des *agents-modèles* nous permet de facilement décrire chaque modèle séparément et de les coupler au moyen d'*artefacts*. Chaque *agent-modèle* reconstruit alors l'évolution d'une des deux populations. La figure 5.18 présente le système d'*agents-modèles* que nous manipulons. On y voit l'*agent-modèle* décrivant la population de proies qui produit un artefact contenant la distribution spatiale des proies. Cet agent lit également l'*artefact* produit par l'*agent-modèle* des prédateurs, *artefact* qui décrit la distribution spatiale des prédateurs. De même, l'*agent-modèle* décrivant la population de prédateurs produit un *artefact* contenant la distribution spatiale des prédateurs et lit l'*artefact* de la distribution des proies. Remarquons que les *propriétés* lisant les *artefacts* ont la même distribution que les données qu'ils lisent. De plus, les *agents-modèles* produisent des *artefacts* dont la discrétisation spatiale est celle de leur modèle. Donc, par exemple, l'*artefact* de la distribution des proies a la même discrétisation spatiale que celle du modèle des proies et donc l'attribut qui lit cet *artefact* a également cette discrétisation. Donc, si l'on modifie le modèle des proies, ici la discrétisation spatiale, il est nécessaire de modifier l'attribut dans l'*agent-modèle* des

Nom	Description
$t_{max} = 3600$	Nombre de pas de temps
$N_{proies} = N_{predateurs} = 5.10^3$	Nombre d'agents par espèce

TAB. 5.18: Hétérogénéité spatiale : valeurs des paramètres

prédateurs qui lit l'*artefact* des proies. Nous avons intentionnellement choisi un tel couplage qui viole l'encapsulation des hypothèses de chaque modèle pour cette étude, car nous voulons justement que le modèle des prédateurs interagissent avec une distribution spatiale des proies qui n'est pas la sienne. Mais, rappelons que nous proposons des solutions de couplage qui permettent de respecter l'encapsulation des modèles.

Remarquons également qu'il est nécessaire de modifier légèrement le système de perception des agents afin que la perception interspécifique soit faite via les attributs liés aux *artefacts* et non via l'environnement du modèle émergent directement. En effet, les agents de type proie par exemple ne perçoivent plus les prédateurs directement dans leur environnement, mais via un attribut du modèle qui lit l'*artefact* de la distribution des prédateurs. C'est cependant là une modification technique minime et qui ne modifie pas directement les comportements de migration.

Le tableau 5.18 présente les hypothèses et la valeur des paramètres utilisés. Les deux modèles partagent les mêmes hypothèses, à part évidemment pour la discrétisation de leur environnement qui est le seul facteur que l'on fait varier.

Résultats : Le tableau 5.19 présente les différentes simulations effectuées en fonction de la discrétisation spatiale du modèle des proies et de celle du modèle des prédateurs. Nous présentons également dans ce tableau le déplacement effectif des agents relativement à la taille de la case, ce qui correspond au rapport de $r_{migration}$ par la largeur d'une case, ce qui permet là encore de se faire une idée de la vitesse de déplacement des agents par rapport à la taille des cases.

Les figures 5.19, 5.20 et 5.21 présentent les résultats obtenus en termes de distribution spatiale. Sur les 3600 pas de temps, nous présentons pour chaque type d'agents (proies et prédateurs) et pour chaque simulation, la distribution spatiale du peuplement aux pas de temps 1, 1800 et 3600. Nous constatons immédiatement que les résultats sont bien moins convaincants que dans le cas homogène. En effet, les résultats sont ici encore plus fortement influencés par les différences de discrétisations que dans le cas homogène. En particulier, nous pouvons voir qu'une discrétisation faible de l'environnement des proies induit des distributions qui ne sont pas fonction de l'habitat et encore moins homogènes. Remarquons en particulier que la distribution des proies de la figure 5.20c et celle des prédateurs de la figure 5.20f sont loin de la distribution théorique de l'habitat. A l'opposé, nous remarquons qu'une discrétisation fine de l'environnement des proies induit des résultats très corrects au regard de la distribution théorique attendue et des résultats que nous avons obtenus dans le cas

Simulations			
Discrétisation proies	Discrétisation prédateurs	Proies : $r_{migration}/l_{case}$	Prédateurs : $r_{migration}/l_{case}$
16×16	32×32	0.0625	0.125
16×16	64×64	0.0625	0.25
16×16	128×128	0.0625	0.5
128×128	16×16	0.5	0.0625
128×128	32×32	0.5	0.125
128×128	64×64	0.5	0.25

TAB. 5.19: Hétérogénéité spatiale : simulations et résultats.

homogène, et ce quelque soit la discrétisation spatiale de l'environnement des prédateurs.

5.7.4 Sensibilité à la discrétisation temporelle

Problématique : Nous nous intéressons maintenant au cas de deux populations d'agents en interaction évoluant à des rythmes temporels différents et nous étudions l'impact sur les résultats d'une telle hétérogénéité de la discrétisation temporelle.

Moyen : Nous utilisons un système d'*agents-modèles* qui est décrit dans la figure 5.22. Un des *agents-modèles* représente la population de prédateurs. Il est construit avec un modèle extensionnel qui reproduira des données de mortalité par prédation pré-établies. L'autre *agent-modèle* représente la population d'agents de type proies et manipule un modèle de type émergent. L'avantage du modèle extensionnel est qu'il nous permet de contrôler complètement la population de prédateurs et de nous concentrer sur celle des proies.

Nous proposons deux scénarios d'évolution du modèle extensionnel de l'*agent-modèle* des prédateurs : 1° une évolution constante et qui correspond au cinquième de l'effectif initial des proies ; 2° une évolution décrite par l'équation suivante : $y = \frac{N_{max}}{t_{max}} \times x$, avec N_{max} le nombre total espéré d'agents et t_{max} le nombre de pas de temps de la simulation. Cette dernière évolution est donc linéaire en commençant avec une mortalité nulle pour finir avec une mortalité correspond à la quantité totale espérée de proies. L'*agent-modèle* des prédateurs utilise ces scénarios pour construire à chaque pas d'exécution une distribution possible⁸ de la mortalité par prédation⁹. Remarquons que cette mortalité est exprimée à l'aide de nombres réels.

Le modèle utilisé pour construire la population de proies est celui de la section 5.6.2 - A. Les agents sont donc constitués du comportement *mate* (algorithme 6 section 5.6.2 - A), du

⁸ Il utilise pour cela une distribution de probabilités basées sur l'habitat des proies.

⁹ La mortalité correspond à un nombre d'agents tués par la prédation.

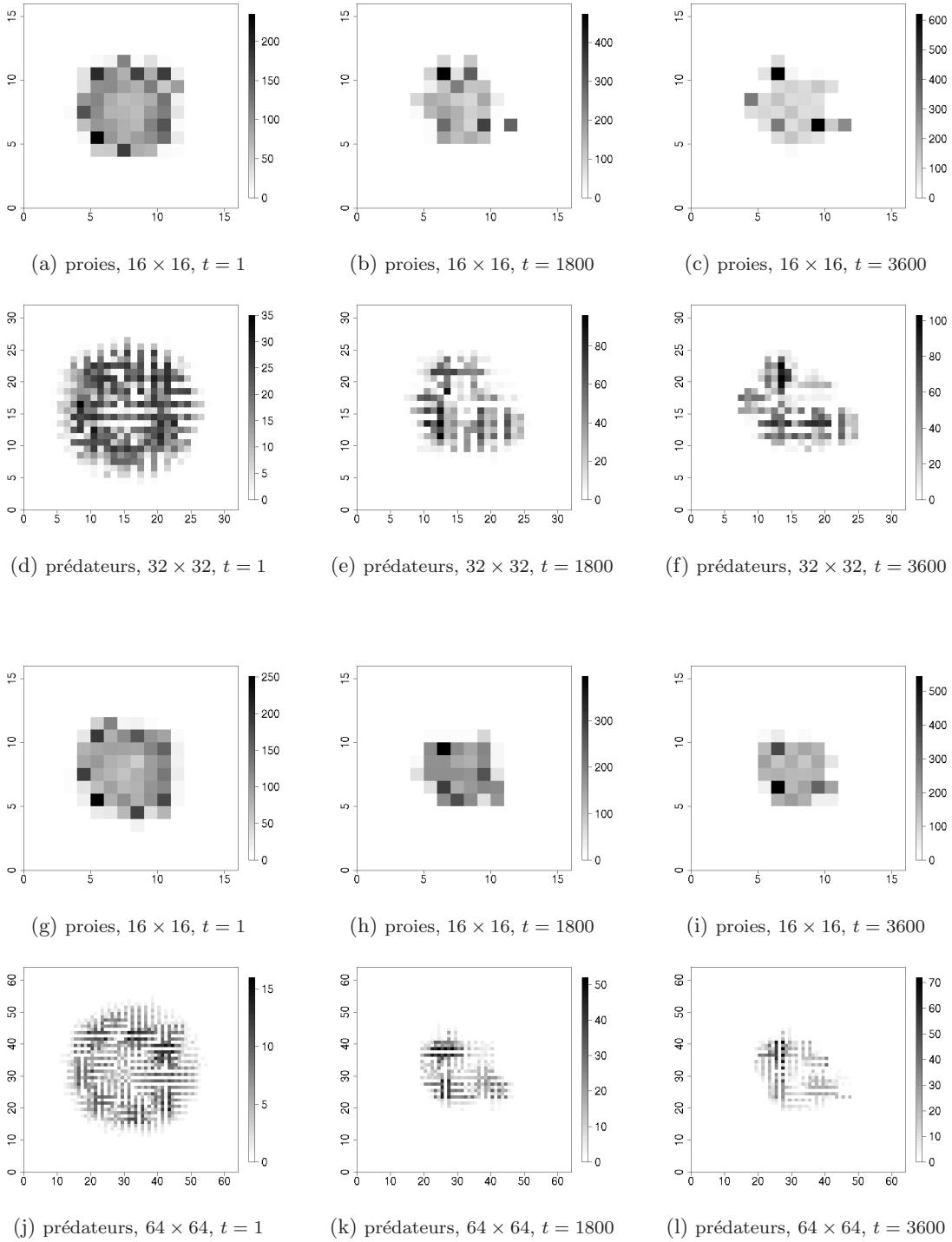


FIG. 5.19: Hétérogénéité spatiale : distributions spatiales des proies et des prédateurs aux pas de temps 1, 1800 et 3600 en fonction de la discrétisation.

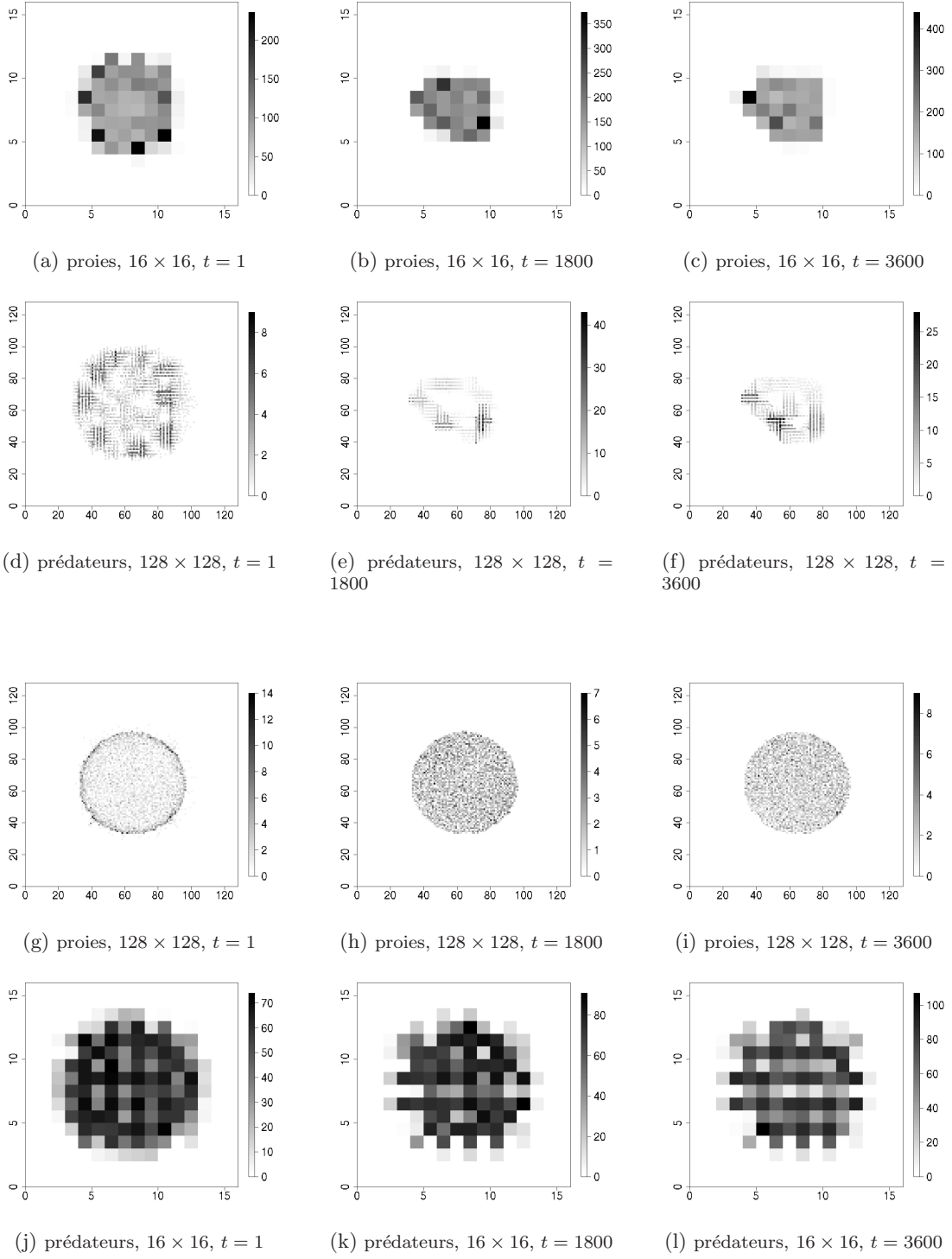


FIG. 5.20: Hétérogénéité spatiale : distributions spatiales des proies et des prédateurs aux pas de temps 1, 1800 et 3600 en fonction de la discrétisation.

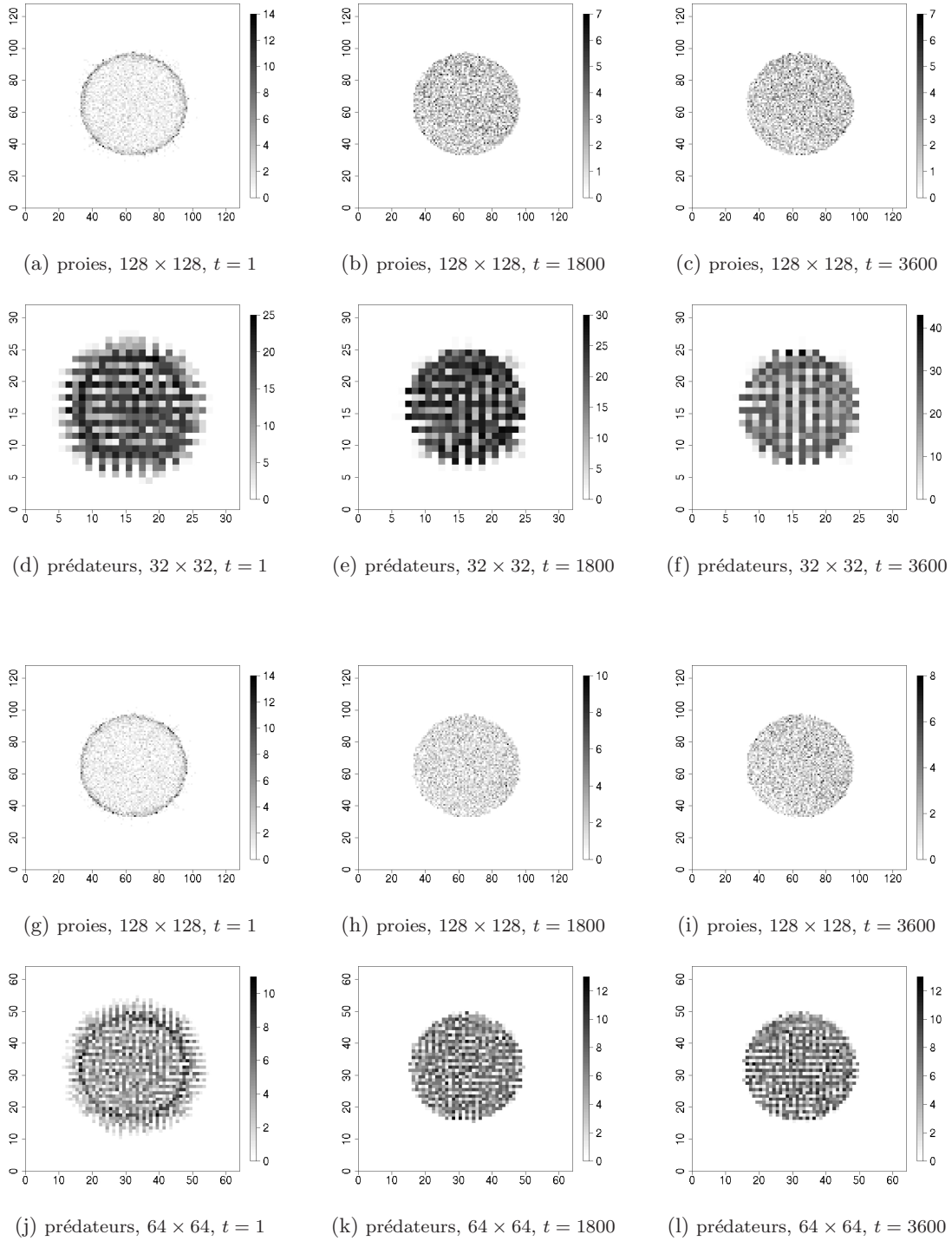


FIG. 5.21: Hétérogénéité spatiale : distributions spatiales des proies et des prédateurs aux pas de temps 1, 1800 et 3600 en fonction de la discrétisation.

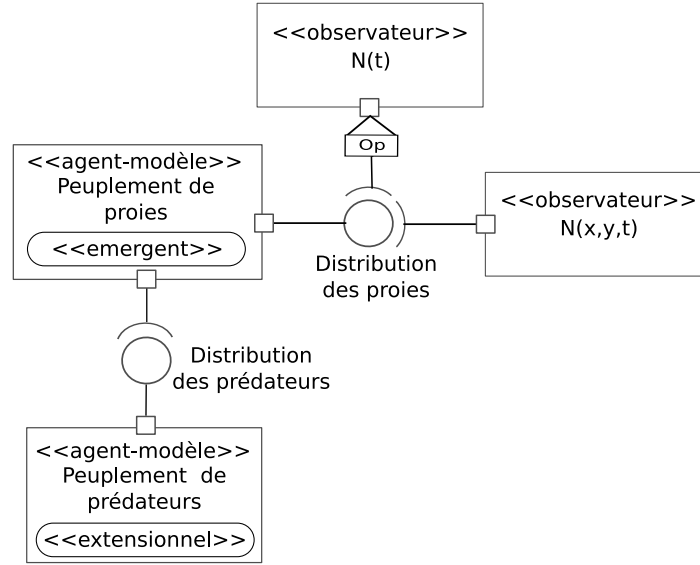


FIG. 5.22: Diagramme d'agents-modèles – cas de deux agents-modèles en interaction trophique.

Nom	Description
Discrétisation habitat	16×16
$t_{max} = 3600$	Nombre de pas de temps
$N_{init} = 10^3$	Nombre d'agents (proies)
$dt_{predateurs}$	Pas d'activation de l'agent-modèle des prédateurs
$M(t)$	Mortalité totale générée par les prédateurs à l'instant t .

TAB. 5.20: Discrétisation temporelle hétérogène : valeurs des paramètres

comportement *die* (algorithme 7 section 5.6.2 - A) et du comportement *recruit* (algorithme 8 section 5.6.2 - A). De plus, il s'agit également d'intégrer au modèle des proies l'influence de la prédation exercée par les prédateurs. Pour cela, nous utilisons un comportement actif qui est associée à la population des proies et qui est donc unique.

Nous envisageons deux scénarios du modèle extensionnel, chacun impliquant cinq simulations qui correspondent chacune à un pas de temps donné de l'agent-modèle des prédateurs. L'agent-modèle des prédateurs produit ses données soit au jour, soit tous les 15 jours, soit tous les mois, soit tous les six mois, soit chaque année. Les données sont évidemment cohérentes quelque soit le pas de temps et l'agent-modèle produit par exemple au jour une mortalité correspondant à $1/360$ de la mortalité qu'il produit à l'année. Le tableau 5.20 présente les hypothèses et la valeur des paramètres utilisés.

Résultats : Pour le scénario 1, la mortalité au cours de la simulation est constante. Le tableau

$dt_{predateurs}$ $M(t)$	jour	15 jours	mois	6 mois	Année
	2.78	41.67	83.33	5.10^2	10^3

TAB. 5.21: Hétérogénéité temporelle – scénario 1 : simulations

$dt_{predateurs}$ $M(t)$	jour	15 jours	mois	6 mois	Année
	$y = 1.39 \times x$	$y = 20.84 \times x$	$y = 41.67 \times x$	$y = 250 \times x$	$y = 500 \times x$

TAB. 5.22: Hétérogénéité temporelle – scénario 2 : simulations

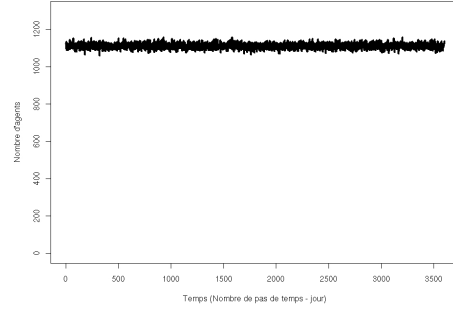
5.21 présente le pas d’activation de l’*agent-modèle* des prédateurs et la mortalité induite sur les proies correspondante à ce pas d’activation. La figure 5.23 présente la dynamique obtenue pour la population de proies en fonction du pas d’activation de l’*agent-modèle* des prédateurs. On voit que la dynamique est différente selon le pas de temps. On constate cependant, que même si le système est sensible à la discrétisation temporelle, le modèle ne permet pas d’exhiber au mieux le biais induit par cette discrétisation. En effet, la population revient très vite à son état d’équilibre.

Pour le scénario 2, l’évolution de la mortalité au cours de la simulation est linéaire et s’accroît au cours du temps. Le tableau 5.22 présente le pas d’activation de l’*agent-modèle* des prédateurs et la mortalité induite sur les proies correspondante à ce pas d’activation. La figure 5.24 présente les dynamiques obtenues pour la population de proies en fonction du pas d’activation de l’*agent-modèle* des prédateurs. Les résultats sont très différents suivant la discrétisation temporelle choisie et ainsi plus intéressant que pour le scénario 2.

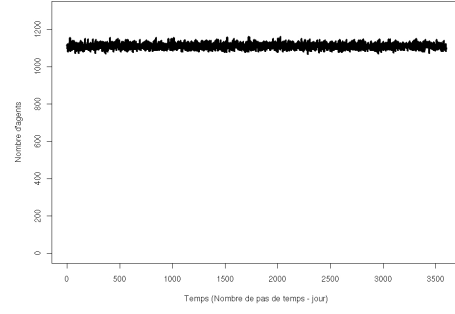
5.7.5 Analyse

Dans cette section, nous avons étudié les biais computationnels liés à la discrétisation hétérogène en nombre d’agents et aux discrétisations hétérogènes spatiales et temporelles dans des modèles de peuplements. Concernant la discrétisation en nombre d’agents, nous avons vu que le système était très sensible à cette discrétisation et à son hétérogénéité. Les résultats sont alors parfois très mauvais. Une discrétisation faible en nombre de prédateurs entraîne ainsi les pires résultats. Par contre, une discrétisation faible en nombre de proies semble moins influencer les résultats. On comprend que diminuer le nombre d’agents sans diminuer le nombre d’individus fait perdre d’autant le nombre de traits particuliers individuels qui évoluent au cours de la simulation. Mais, il est toutefois impossible en l’état d’abstraire ces résultats et la discrétisation hétérogène en nombre d’agents est un facteur sensible dans la modélisation de modèles de peuplements.

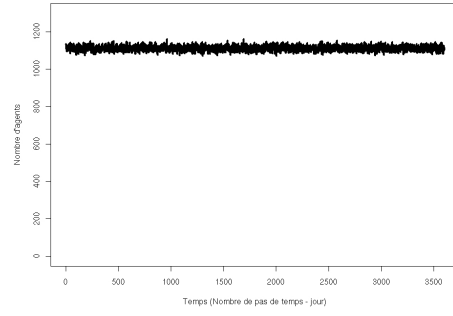
Les résultats sur la sensibilité du système à la discrétisation spatiale hétérogène montre



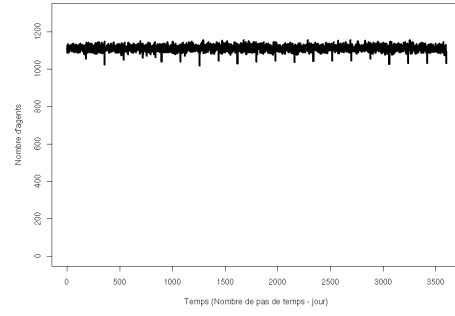
(a) Pas d'activation : jour



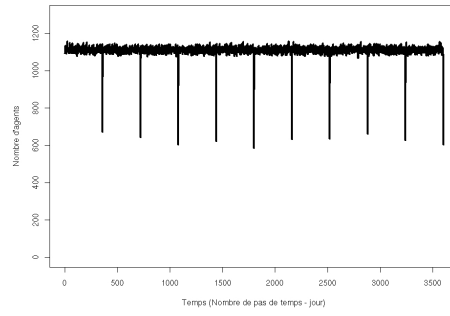
(b) Pas d'activation : 15 jours



(c) Pas d'activation : mois

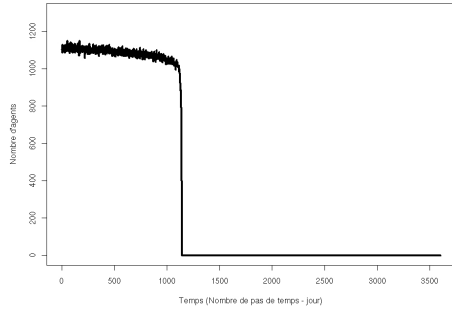


(d) Pas d'activation : 6 mois

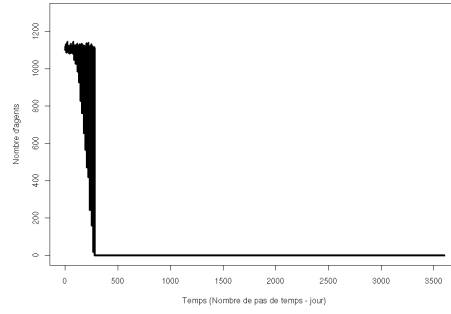


(e) Pas de temps : année

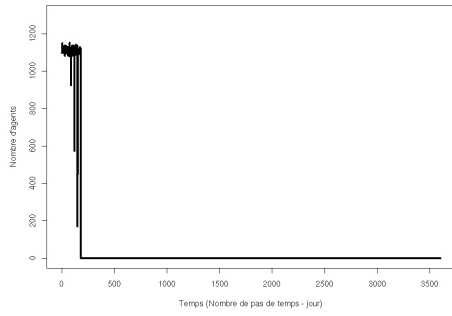
FIG. 5.23: Discrétisation temporelle hétérogène – cas de deux populations – scénario 1.



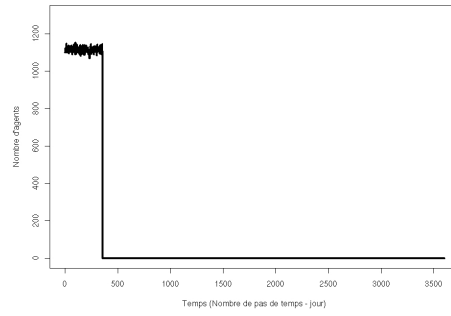
(a) Pas d'activation : jour



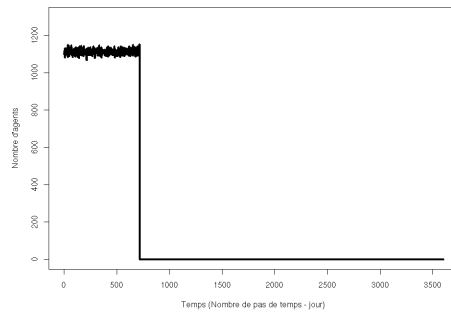
(b) Pas d'activation : 15 jours



(c) Pas d'activation : mois



(d) Pas d'activation : 6 mois



(e) Pas d'activation : année

FIG. 5.24: Discretisation temporelle hétérogène – cas de deux populations – scénario 2.

eux-aussi une forte sensibilité du système à ce facteur. Là encore, les résultats sont plus éloignés des dynamiques attendues que dans le cas homogène. En particulier, nous avons constaté qu'une faible discrétisation donnait des résultats non conformes au modèle conceptuel, alors que les résultats obtenus avec une discrétisation forte étaient conformes.

Enfin, les résultats concernant la sensibilité du système à la discrétisation temporelle hétérogène vont dans le même sens. Les dynamiques observées sont différentes suivant la discrétisation, et c'est particulièrement marqué dans le deuxième scénario étudié. En effet, les résultats du premier scénario montrent bien évidemment des dynamiques différentes suivant le pas d'activation de la population de prédateurs, mais le système revient vite à son état d'équilibre, ce qui tend alors à masquer le biais introduit par la discrétisation temporelle. Le scénario 2 nous a permis de mieux exhiber le biais induit par la discrétisation.

5.8 Conclusion

L'objectif de ce chapitre était d'exhiber un certain nombre de propriétés qu'un outil de simulation doit garantir pour mettre en œuvre des agents dans une simulation et d'identifier les hypothèses computationnelles introduites dans le modèle au moment de son implémentation et qui peuvent entraîner des biais critiques sur les résultats. Nous avons mis en œuvre des modèles issus d'un cadre conceptuel unique afin d'identifier de manière systématique les hypothèses introduites dans chaque modèle computationnel. Cette étude a porté sur les modèles IBM de peuplements, parce qu'ils permettent de mettre en œuvre des processus particulièrement sensibles aux biais computationnels : création, destruction et déplacement d'agents.

Nous avons tout d'abord étudié les propriétés qu'un outil de simulation doit garantir pour pouvoir mettre en œuvre des modèles agents. La synchronicité des comportements, leur synchronisation et le choix des états initiaux des agents sont des points critiques pour le système. En particulier, nous avons vu que les variables liées, donc manipulées par différents comportements, pouvaient poser des problèmes majeurs de cohérence du modèle, entraînant des résultats aberrants. Nous avons également montré qu'une solution stochastique pour la mise en œuvre d'un modèle conceptuel qui ne l'est pas doit être utilisée en connaissance de cause, le modèle stochastique entraînant des hypothèses implicites qui n'existent pas forcément au niveau conceptuel. Il est par conséquent absolument nécessaire d'écrire les algorithmes avec beaucoup de précaution et d'en vérifier le bon comportement. Ceci est un point critique pour les modèles IBM dont le code informatique est la seule représentation.

Ensuite, nous avons étudié la sensibilité des modèles aux discrétisations homogène et hétérogène en nombre d'agents et aux discrétisations homogènes et hétérogènes spatiales et temporelles. Nous avons constaté que les modèles sont très sensibles à ces facteurs et, suivant la discrétisation utilisée, les résultats peuvent ne plus être conformes au modèle

conceptuel. Nous remarquons, que même si certains résultats peuvent être intuitifs, c'est-à-dire que l'origine des biais est connue, il n'est pas possible d'en connaître de manière générale les conséquences sur les résultats. Selon les modèles, les biais sont plus ou moins marqués et certains processus modélisés peuvent même en masquer en partie l'influence. Par conséquent, l'étude de ces biais, introduits au niveau computationnel, doit être systématique pour tout modèle étudié. De plus, la construction d'un modèle et le choix des niveaux de granularité de description est de fait un compromis entre la précision des résultats requise et les ressources disponibles pour les produire. Ce compromis dépend alors du système, des ressources et du modèle mis en œuvre.

Il est donc nécessaire de disposer d'outils spécialisés de simulation. Les thématiciens doivent également connaître les biais computationnels et doivent ainsi complètement expliciter leur modèle (variable liée, discrétisations, etc.). Enfin, un outil de simulation doit permettre d'explicitement le modèle, garantir les propriétés que nous avons identifiées et permettre des analyses systématiques des biais computationnels. Par conséquent, c'est ce que nous avons fait dans notre simulateur.

Chapitre 6

Etude de modèles écosystémiques des pêches

6.1 Introduction

Le travail de cette thèse s'inscrit en partie dans le cadre du projet ANR Chaloupe¹. Ce projet part du constat que l'analyse des séries temporelles de la biodiversité du golfe de Gascogne (*GdG*) présente une évolution significative en termes de qualité et de quantité de la composition faunistique au cours des trente dernières années. Deux facteurs exogènes ont été identifiés comme pouvant expliquer ces modifications : 1^o le réchauffement climatique mesuré dans le *GdG* (Blanchard 05b; Poulard 05; Blanchard 01) et 2^o la pression anthropique, en particulier le prélèvement plus ou moins sélectif opéré par les flottilles de pêche du *GdG*. Le point de départ de ce projet est donc de rassembler des compétences multiples, écologues, halieutes, informaticiens et modélisateurs, pour essayer d'appréhender cette réalité afin d'en comprendre les dynamiques et d'étudier le rôle des différents processus qui les structurent. Les objectifs de ce projet sont donc : 1^o identifier l'évolution de la biodiversité en lien avec les tendances environnementales locales liées au changement climatique et à l'évolution de la pression de pêche ; 2^o déterminer l'adaptation des exploitations aux évolutions du contexte environnemental, économique et institutionnel ; 3^o établir les diagnostics sur le statut de différents types de systèmes régionaux², les risques de crises et leurs conditions de viabilité ; et enfin 4^o développer des modèles de simulation des systèmes bio-économiques.

¹ CHAngement gLObal, dynamiqUe de la biodiversité marine exploitée et viabilité des PEcheries – Projet Biodiversité ANR-IFB 2005. Coordination : F. Blanchard et O. Thébaud (Ifremer) – <http://www.projet-chaloupe.fr/>

² plateau continental tempéré du Golfe de Gascogne GdG, plateau tropical amazonien de Guyane française, et écosystème d'upwelling du Maroc.

Notre rôle dans ce projet est de : 1° produire des modèles IBM ou mixtes du peuplement de poissons exploité du Golfe de Gascogne (GDG) dans un environnement variable (influence du réchauffement climatique) en collaboration avec des écologues et des halieutes de l'IFREMER et de l'IRD ; 2° construire une plate-forme de simulation opérationnelle basée scénarios et permettant d'étudier la biodiversité du GDG dans un environnement variable avec notre modèle écosystémique ; 3° retrouver les évolutions de séries historiques du peuplement de poissons du GDG afin de distinguer l'influence du réchauffement climatique de celui des activités de pêche. Ainsi, l'objectif de ce chapitre est de s'appuyer sur les travaux effectués dans le cadre du projet Chaloupe afin d'exhiber les problématiques que nous avons rencontrées et qui justifient notre proposition. Nous cherchons à mettre en évidence la pertinence de notre proposition en termes de modèle, d'outil et de méthodologie, tout en faisant transparaître le travail de modélisation effectué (présenté pour l'essentiel en annexes) et le simulateur qui nous a permis de produire tous nos résultats.

L'objet d'étude premier de ce chapitre est le système peuplement-pêcheries évoluant dans un environnement variable. Mais, à cela viennent s'agréger les problématiques de modélisation et de multi-modélisation, soit des questions de couplage, de construction incrémentale de modèles et de manipulation de modèles complexes. Ces questions, nous proposons de les aborder de manière incrémentale, afin d'appréhender une à une les problématiques principales, en enrichissant petit à petit les systèmes modélisés, avant de mettre en commun l'ensemble de ces efforts.

Nous avons construit ce chapitre sur des actions de modélisation qui s'appuient chacune sur un système d'*agents-modèles* qui définit le cadre de modélisation. La première action de modélisation illustre des travaux effectués sur la modélisation de peuplements marins. Ces travaux nous ont permis d'étudier et de valider un certain nombre d'éléments fondamentaux de modélisation, tels que le couplage de modèles IBM et EDP de peuplements, et ils nous permettent ici d'exhiber les mécanismes principaux de construction de systèmes d'*agents-modèles*. Ensuite, nous changeons d'échelle et abordons deux actions de modélisation du peuplement écologique du GDG. Celles-ci résument des travaux effectués sur le dimensionnement de notre modèle EDP de peuplements et du modèle IBM de peuplements du GDG. La quatrième action de modélisation nous a permis de valider le couplage de modèles de peuplements et de modèles de pêcheries et donc plus largement de couplage de modèles hétérogènes. C'est l'occasion ici d'exhiber les problématiques d'hétérogénéité de modélisation et d'échelles. La cinquième action se focalise sur la modélisation économique du marché des navires de pêche. Enfin, nous revenons sur un certain nombre de problématiques que nous avons rencontré. Nous présentons alors l'outillage de notre simulateur et la manière dont nous avons géré la complexité des processus de modélisation et de simulation. Cet outillage permet d'ouvrir des pistes pour l'automatisation plus poussée de certaines actions de modélisation et de simulation ; en cela, il est essentiellement un prolongement concret de notre proposition méthodologique. Avant d'aborder les actions de modélisation, nous introduisons le contexte et les problématiques de ce travail. Puis nous présentons les patterns structurants du projet Chaloupe.

6.2 Contexte et problématiques

L'étude de la co-viabilité des systèmes écologiques et des systèmes économiques d'exploitation des pêcheries s'articule autour de la nécessité d'évaluer les processus et interactions en jeu, donc de comprendre les mécanismes du système biodiversité-pêcheries ; on parle alors d'approche écosystémique. Cette analyse, sous-tendue par la construction de modèles de ces systèmes, doit nous permettre d'apprécier l'évolution de ces modèles selon différents scénarios, afin d'envisager l'effet de différentes mesures de gestion sur la survie des systèmes.

L'approche écosystémique s'intéresse aux interactions réciproques dans lesquelles sont impliqués les différents systèmes et que le réchauffement climatique vient également bousculer. La modification de la biodiversité n'est pas sans conséquence sur l'économie qui l'exploite. Les variations d'abondances d'espèces d'intérêt commercial, de même que la condition sociale des pêcheurs, peuvent directement peser sur les pêcheries et l'économie de la pêche dans son ensemble. De même, les modifications des équilibres entre espèces peuvent influencer positivement ou négativement les espèces marines, qui voient leurs prédateurs ou leurs proies plus ou moins nombreux, alors même que le réchauffement climatique peut influencer à son tour les différentes espèces de différentes manières. Le couple écosystème-secteur économique constitue ainsi un système complexe : il est composé de nombreuses entités autonomes en interaction aux comportements non linéaires et localisées dans un environnement hétérogène.

Comprendre, d'une part, l'influence d'une modification climatique sur la biodiversité de l'écosystème marin, et d'autre part, appréhender les devenir possibles du système en fonction de diverses hypothèses et scénarios, requiert alors un outil de simulation pouvant prendre en compte les hétérogénéités du système et la variabilité de l'environnement. Plus encore, envisager des politiques de gestion des pêcheries et pouvoir les expérimenter par simulation dans la perspective d'interactions fortes entre le système économique et l'écosystème nécessite de pouvoir modéliser et considérer une grande richesse de comportements, divers scénarios et hypothèses, différents modèles avec différentes échelles (de modélisation et d'analyse) et donc différentes granularités des modèles.

Enfin, les différents modèles, EDP et IBM, permettant de modéliser les peuplements écologiques, doivent être validés structurellement, expérimentés et leurs paramètres définis. Il en est de même pour les modèles IBM de pêcheries. Ainsi, il est nécessaire d'expérimenter un modèle après l'autre, en complexifiant petit à petit les problèmes abordés. Ces différentes expérimentations et études doivent trouver leur place au sein de la démarche globale de modélisation qui doit permettre *in fine* de répondre à la question posée.

Remarque Nous avons, dans le chapitre précédent, fait une légère digression en nommant agent un individu d'un IBM. Nous distinguons maintenant à nouveau les agents de notre proposition – les *agents-modèles*, agents-contrôleurs ou agents-observateurs – et les individus

qui peuplent nos IBM. En particulier, nous appelons agents-individus les individus computationnels des IBM.

6.3 Objet d'étude et patterns

L'objectif de cette section est de présenter les patterns majeurs caractérisant les dynamiques du système peuplements-pêcheries du GDG. Nous avons choisi de présenter ces patterns hors du cadre d'une action de modélisation, car ils ont structuré notre démarche de modélisation et ils justifient par conséquent, au moins indirectement, l'organisation de ce chapitre. De plus, la présentation de ces patterns nous oblige à mieux définir le système peuplements-pêcheries du GDG. Ils décrivent le système dans sa globalité, fixant ainsi les échelles auxquelles nous nous intéressons et les données sur lesquelles nous devons nous appuyer et que nous devons observer en simulation. Nous avons donc choisi d'esquisser dans cette section, en même temps que les patterns, les informations qui structurent le système peuplements-pêcheries du GDG.

6.3.1 Problématique écologique

Le GDG est peuplé de nombreuses espèces de poissons³, mais 80% de la biomasse est constituée par neuf espèces qui sont parmi les plus importantes du point de vue économique (pour la pêche). Ces espèces sont : les anchois, les baudroies communes, les chinchards, les merlans, les merlans bleus, les merlus européens, les sardines, les tacauds et les petits tacauds. Elles s'organisent en un réseau trophique, c'est-à-dire que chaque espèce occupe une place particulière dans un réseau de relations de prédation.

Le GDG représente une superficie d'environ 225000 km^2 . Schématiquement, il commence au Nord à la pointe de la Bretagne et se prolonge au Sud jusqu'aux côtes de l'Espagne. Les écologistes peuvent décrire, avec plus ou moins de précision, les habitats des différents peuplements à l'intérieur du GDG en fonction de la latitude et de la profondeur. De nombreux peuplements ont cependant des aires de répartition plus étendues. Les habitats des espèces étudiées ne coïncident pas et la coexistence spatiale (voire temporelle) des espèces doit être prise en compte dans les interactions trophiques. Ainsi, la distribution spatiale reste une composante importante de la dynamique des espèces.

Le réchauffement des eaux du GDG a été observé au cours des trente dernières années. En réalité, il semble avoir eu lieu de manière brutale en trois ans, de 1987 à 1991, et se serait traduit par une augmentation de la température moyenne des eaux de surface de 1,5 degrés.

³ Poisson est pris ici au sens large, c'est-à-dire essentiellement poissons au sens strict, crustacés et mollusques

Ce réchauffement peut être considéré comme relativement homogène dans l'environnement de par la géographie du GdG.

Maintenant, concernant les patterns qui caractérisent l'écosystème du GdG sur les trente dernières années, il nous faut d'abord décrire les indicateurs sur lesquels ils se basent. En effet, les écologistes ont observé cet écosystème au travers de trois indicateurs globaux (Blanchard 05b; Poulard 05; Blanchard 01) : le poids moyen global, la latitude préférentielle moyenne et le niveau trophique moyen.

(Poids moyen global) Chaque espèce biologique est caractérisée par le poids moyen d'un individu de cette espèce. Ainsi, si un système n'est composé que d'une espèce donnée, le poids moyen global de ce système est le poids moyen de l'espèce qui le compose. Prenons un autre exemple, si l'on considère un écosystème composé de deux espèces A et B, le poids moyen global de cet écosystème est la moyenne des poids moyens des espèces A et B, chacun pondéré par le nombre d'individus de l'espèce correspondante. De cette manière, dire que le poids moyen global diminue pour un écosystème donné, c'est dire qu'il y a en proportion une augmentation des espèces dont les individus ont un poids moyen faible. Remarquons par ailleurs que généralement plus le poids moyen d'une espèce est élevé, plus cette espèce est intéressante commercialement et donc recherchée par les pêcheurs.

(Latitude préférentielle moyenne) Chaque espèce biologique est caractérisée par une répartition géographique qui lui est propre. On peut ainsi caractériser une espèce donnée par la latitude du centre de gravité de sa répartition géographique qui est sa latitude préférentielle. L'hypothèse à l'origine de cet indicateur est que l'on peut penser que la répartition géographique des espèces est fonction, entre autres choses, de la température des eaux. Le centre de gravité de la répartition géographique d'une espèce donnée représente alors la température «idéale» de cette espèce. Et la répartition est d'autant plus large (du Nord au Sud) que les individus de l'espèce supportent des écarts par rapport à cette température «idéale». Revenons alors à un écosystème comme celui du GdG. Une espèce dont la latitude préférentielle est en-dessous des latitudes du GdG est dite à affinité chaude. Une espèce dont la latitude préférentielle est au-dessus des latitudes du GdG est dite à affinité froide. Et enfin, celles qui ont pour latitude préférentielle une latitude qui passe par le GdG sont dites indigènes. Par conséquent, si l'on prend l'exemple d'un réchauffement des eaux de la planète, on peut s'attendre à voir les espèces d'affinité chaude augmenter en proportion dans le GdG par rapport aux espèces d'affinité froide. Cet exemple de dynamique se traduit alors par une diminution de la latitude préférentielle moyenne dans le GdG. Celle-ci est donc calculée en faisant la moyenne des latitudes préférentielles de chaque espèce présente dans l'écosystème pondérées par le nombre d'individus par espèce. Remarquons par ailleurs que les espèces d'affinité froide par rapport au GdG sont en général des espèces caractérisées par un poids moyen élevé et les espèces d'affinité chaude par un poids moyen faible.

(Niveau trophique moyen) Un écosystème est composé de divers peuplements qui sont en interaction trophique, c'est-à-dire qu'il y a des relations de prédateurs entre les espèces. Ces relations trophiques peuvent être décrites au moyen d'un réseau trophique dans lequel

chaque espèce prend sa place. Au «sommet» du réseau se trouve les top-prédateurs qui ne sont mangés par aucune autre espèce de l'écosystème, mais qui en chassent certaines. «Tout en bas» du réseau se trouve les proies qui ne mangent aucune autre espèce de poissons, mais sont mangées par certains prédateurs. Enfin au milieu se trouve un certain nombre de prédateurs intermédiaires. Cette hiérarchisation des espèces se traduit par un niveau trophique qui caractérise chaque espèce et qui correspond à la place de l'espèce dans le réseau trophique. Ainsi une espèce avec un niveau trophique élevé est plutôt de type grand prédateur, alors qu'un niveau trophique faible correspond à des proies qui sont peu prédatrices. Le niveau trophique moyen pour un écosystème donné est alors la moyenne des niveaux trophiques de chaque espèce présente dans l'écosystème pondérés par le nombre d'individus de chaque espèce. On peut ainsi caractériser l'évolution en proportion des prédateurs et des proies pour un écosystème donné.

Nous pouvons alors enfin décrire les patterns principaux de notre étude. Ceux-ci se traduisent par une diminution au cours de la période considérée du poids moyen global, de la latitude préférentielle moyenne et du niveau trophique moyen pour l'écosystème marin du GdG. Remarquons que les indicateurs sont globaux et que leur dynamique décrit une réalité qui peut être expliquée par différents processus structurants du système.

6.3.2 Problématique économique

Nous nous intéressons ici aux aspects sociaux et économiques de la pêche. Nous considérons deux types d'entités : les pêcheurs et les unités de pêche. Dans le GdG, il y a environ 2000 patrons-pêcheurs qui font la plupart du temps travailler chacun un navire de pêche. Une unité de pêche correspond donc généralement à un navire, mais elle peut aussi représenter une flottille de navires. Au niveau phénoménologique, il y a les activités de pêche en tant que telles et le marché des navires de pêche. Ce sont là les deux grands phénomènes du système. Il y a cependant d'autres processus à considérer : la formation du prix des poissons, les politiques de gestion, l'évolution des conditions sociales des pêcheurs, etc.

Le système des pêcheries est caractérisé par l'hétérogénéité et la dynamique de ses composantes. En effet, différents métiers structurent les activités de pêche et les pêcheurs peuvent en changer pour se porter sur des espèces plus intéressantes économiquement. C'est le même principe avec les zones de pêche qui divisent le GdG en six zones et vers lesquelles peuvent se déplacer les pêcheurs suivant leur métier et les abondances des peuplements de poissons. Plus généralement, il y a différents processus responsables des dynamiques d'allocation de l'effort de pêche entre les différentes composantes de la pêcherie (métiers, zones, etc.).

La question de l'hétérogénéité est également incontournable concernant le marché des navires de pêches, neufs et d'occasion, qui est, pour les économistes des pêches, un phénomène

essentiel pour comprendre et expliquer les dynamiques des pêcheries. Les mécanismes de formation des prix des navires sont complexes et traduisent l'attractivité du secteur de la pêche entre autre. De plus, comme les flottilles sont très différentes, de par leur âge ou leur capacités techniques, le marché des navires de pêche est un marché de biens fortement hétérogènes.

Les aspects sociaux s'étudient au travers du rapport de l'activité des pêcheurs avec le reste de la société : départ en retraite, reconversion. Leurs comportements sont conditionnés par leur capital qui dépend de la ressource, mais également des politiques de gestion des pêches et de l'état de l'économie en général.

Maintenant que nous avons les informations nous permettant d'appréhender quelque peu la réalité du système économique des pêcheries, abordons la question des patterns. Tout d'abord, la dépréciation d'une unité de pêche suit une évolution décrivant la perte d'efficacité régulière due au vieillissement et la compensation de ce vieillissement par des réinvestissements réguliers effectués par le pêcheur. Il n'est pas évident que cette évolution soit un pattern, mais c'est une dynamique structurante d'une unité de pêche.

L'évolution du prix moyen d'un navire de pêche au cours de la période considérée est un pattern fondamental du système, la formation de ce prix fait d'ailleurs l'objet de nombreuses recherches en économie des pêches. Le prix réel des unités de pêche (c'est-à-dire le prix objectif du matériel), l'état de l'économie et l'attractivité des métiers de la pêche conditionnent le prix du navire. Notons donc que le prix d'un navire correspond au prix réel du matériel (qui est fonction de son efficacité technique) plus un prix résiduel (positif ou négatif) qui caractérise la fonction d'une rente.

Enfin, l'évolution du nombre d'unités de pêche dans le GDG est due à un certain nombre de processus : l'attractivité du métier, l'état des ressources, le prix des navires de pêche, les phénomènes sociaux (traditions, héritages, culture) et les politiques de gestion des pêches. Cette évolution est donc aussi un pattern du système.

6.4 Action 1 : Modélisation de peuplements marins

6.4.1 Question de modélisation et patterns

Question de modélisation : Pour commencer, nous modélisons deux populations de poissons en interaction. Nous voulons ainsi valider un système d'*agents-modèles* de proies-prédateurs avec différentes approches de modélisation (EDP et IBM). En effet, dans une démarche incrémentale de modélisation, nous voulons valider les mécanismes d'interaction entre populations (dont le couplage de modèles) et de construction des habitats des peuplements afin de pouvoir s'appuyer, dans les études suivantes, sur un système d'*agents-modèles*

opérationnel. Le modèle EDP en particulier est un modèle original qu'il est nécessaire d'étudier sur des exemples simples. Enfin, d'un point de vue plus pédagogique, cette étude est l'occasion d'exhiber et vérifier les mécanismes élémentaires de nos systèmes d'*agents-modèles*, c'est-à-dire l'encapsulation des modèles de phénomènes, la construction d'un cadre de modélisation, la modification du comportement d'un *agent-modèle* sans impact sur le reste du système et enfin l'ajout d'un nouvel agent sans impact sur le reste du système.

Patterns et *artefacts* : Nous voulons reproduire les dynamiques de peuplements marins théoriques. Isolé, un peuplement occupe l'environnement en fonction de son habitat qui décrit ainsi la manière dont les individus s'y répartissent. De plus, l'effectif a tendance à se stabiliser, car l'environnement ne peut pas accueillir un nombre infini d'individus. Donc, il y a un point d'équilibre de la population qui est fonction de l'environnement. Dans le cas d'un système proies-prédateurs, nous nous basons sur le pattern proies-prédateurs classique (voir section 2.4.4). Les dynamiques en nombre des populations exhibent alors des cycles qui décrivent une inter-dépendance des peuplements. Par conséquent, nous observons dans cette étude l'évolution des effectifs pour chaque espèce au cours du temps, ainsi que leur distribution spatiale.

6.4.2 Cadre de modélisation

Hypothèses et phénomènes : Les habitats des poissons sont forcés dans le système, mais pourraient éventuellement être modifiés. Il est donc nécessaire pour nous de pouvoir les décrire indépendamment du reste du système comme un phénomène à part.

Ensuite, les deux populations font l'objet d'études séparées, il est donc nécessaire pour nous de les considérer comme des phénomènes distincts. Remarquons cependant que nous développons ici un modèle de prédation dans lequel chaque espèce calcule l'impact de la prédation (positif pour les prédateurs et négatif pour les proies) sur ses individus, comme c'est le cas souvent dans les modèles agrégés et en particulier dans notre modèle EDP (voir annexe C). Dans une démarche IBM classique, on aurait davantage envie que les individus prédateurs suppriment eux-mêmes les proies qu'ils mangent. C'est alors un autre modèle qu'il faudrait aborder dans une autre action de modélisation.

Cadre de modélisation : Les phénomènes que l'on veut modéliser sont une population de proies, une population de prédateurs, l'habitat des proies et celui des prédateurs. Il y a donc quatre *agents-modèles* comme le montre la figure 6.1. Nous souhaitons observer les distributions spatiales des peuplements, ainsi que le nombre total d'individus au cours du temps. Les *agents-modèles* des peuplements doivent donc écrire ces *artefacts*. De plus, l'*agent-modèle* des prédateurs doit avoir des informations sur les proies, pour calculer un impact de la prédation, et il lit par conséquent l'*artefact* décrivant la distribution spatiale des proies. De la même manière, l'*agent-modèle* des proies doit avoir des informations sur l'effectif des

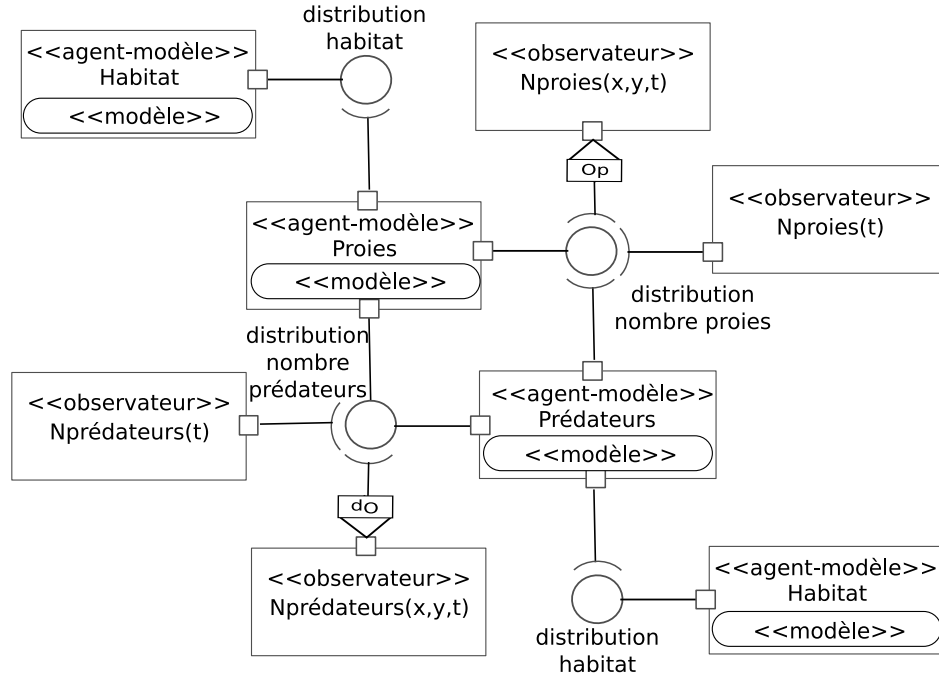


FIG. 6.1: Diagramme d'agents-modèles – Action de modélisation 1 – Cadre de modélisation d'un système prédateurs-proies.

prédateurs afin de calculer une mortalité par prédation. Enfin, chaque espèce doit pouvoir connaître son habitat et donc les *agents-modèles* des peuplements lisent chacun un *artefact* le décrivant.

Il y a également dans le cadre de modélisation quatre agents-observateurs qui produisent des observables des distributions spatiales des peuplements et de leur effectif total.

Éléments de modélisation : Remarquons que pour connaître l'effectif total de chaque espèce, il nous suffit d'utiliser un agent-observateur qui à l'aide d'une *opération* peut calculer cette quantité au rythme temporel voulu. Ainsi, il n'est pas nécessaire de faire prendre en charge ce calcul par les *agents-modèles* des peuplements qui n'utilisent pas directement cette valeur.

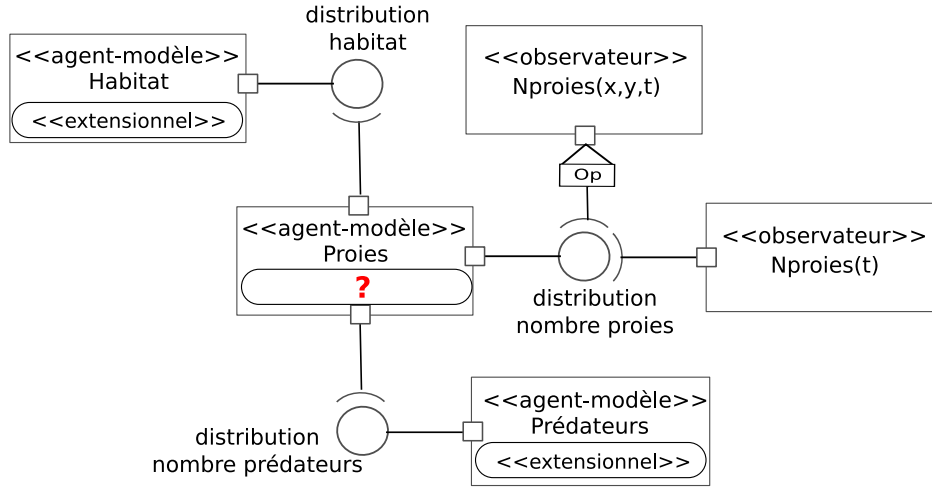


FIG. 6.2: Diagramme d'agents-modèles – Cadre de modélisation pour expérimenter différentes approches de modélisation pour un peuplement de poissons avec un habitat fixé.

6.4.3 Expérimentation unitaire : choix de l'approche de modélisation

Problématique : Pour pouvoir valider le système proies-prédateurs, nous voulons dans un premier temps valider le cas d'un peuplement. Nous voulons de plus pouvoir évaluer les dynamiques obtenues avec un modèle IBM et avec un modèle EDP de ce peuplement.

Moyens : Pour nous concentrer sur une seule population, nous ne mettons en œuvre qu'une partie du cadre de modélisation. Ceci est rendu facile par la possibilité d'utiliser des modèles extensionnels qui n'ont pas besoin de données d'entrées pour fonctionner et qui permettent surtout de forcer le comportement de n'importe quel *agent-modèle*. De plus, les données partagées étant complètement explicitées et formalisées au moyen des *artefacts*, il suffit que ces derniers soient initialisés pour que le système de modèles puissent être mis en œuvre. En l'occurrence ici, nous proposons de nous concentrer sur le cadre simplifié de la figure 6.2, mais remarquons que tous les éléments permettant à la population d'interagir avec les autres éléments du cadre initial sont là. Ainsi, les *agents-modèles* «Habitat» et «Prédateurs» sont composés de modèles extensionnels. Nous maîtrisons alors complètement l'environnement perçu (les *artefacts*) par l'agent «Proies». Enfin, pour n'étudier que le modèle des proies, l'*agent-modèle* des «Prédateurs» écrit une distribution en nombre de prédateurs nulle.

L'agent-modèle «Habitat» construit quant à lui une distribution de l'affinité positionnée dans un environnement quelconque. Nous utilisons le même habitat que dans la section 5.4, habitat qui est en forme de disque, centré dans l'environnement et avec une affinité qui est

Paramètres (principaux) : Informations	
$t_{max} = 3600$	: durée de la simulation (nombre de pas de temps)
$N_e = 10^4$	: nombre initial d'individus de la population dans la modèle EDP
$N_{agent} = 10^4$	: nombre initial d'agents dans le modèle IBM
discrétisation spatiale	: 16×16

TAB. 6.1: Action 1 – expérience 1 : valeurs des paramètres.

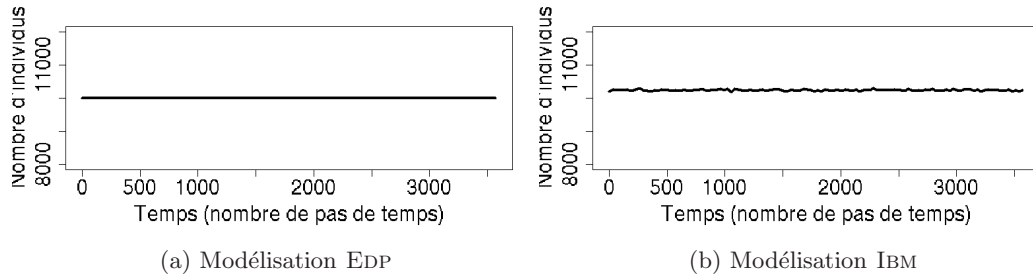


FIG. 6.3: Action 1 – expérience 1 – section 6.4.3 : évolution en nombre du peuplement au cours de la simulation.

maximale au centre du disque, nulle à l'extérieur, et graduée sur les bords du disque.

Nous utilisons deux modèles pour mettre en œuvre cette simulation, le modèle IBM utilisé dans la section 5.6.2 - A du chapitre précédent et le modèle EDP présenté dans l'annexe C. Enfin, le tableau 6.1 présente les hypothèses principales que nous avons utilisées pour effectuer la simulation.

Résultats : Les résultats sont présentés dans les figures 6.3 et 6.4. La figure 6.3a présente l'évolution en nombre du peuplement obtenue avec le modèle EDP. La figure 6.3b présente la même évolution mais obtenue avec le modèle IBM. Ensuite, la figure 6.4 présente l'état de la distribution spatiale du peuplement pour le modèle EDP (6.4a) et IBM (6.4b) pris à un pas de temps quelconque de la simulation.

On constate ainsi que le peuplement est stable quelle que soit l'approche de modélisation, mais que le modèle IBM est plus irrégulier, ce qui n'est pas surprenant.

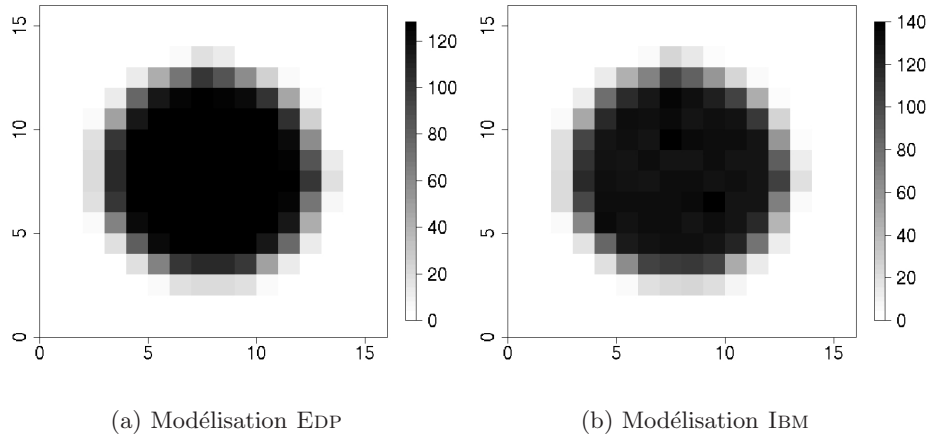


FIG. 6.4: Action 1 – expérience 1 – section 6.4.3 : distribution spatiale du peuplement suivant la modélisation utilisée (pas de temps quelconque).

6.4.4 Ajout d'un agent sans impact sur le reste du système

Problématique : Nous proposons maintenant d'étudier l'impact sur la dynamique des proies de l'ajout d'un peuplement de prédateurs. Nous souhaitons pouvoir avant tout retrouver un pattern de type proies-prédateurs. Ce pattern est décrit dans la section 2.4.4 de l'état de l'art et se traduit dans les données par des cycles dans les dynamiques en nombre des deux peuplements. L'intérêt ici est également de montrer comment l'ajout d'un nouvel agent dans le système n'influence pas le reste du système, à part au niveau des résultats bien évidemment.

Moyens : Nous reprenons le système d'*agents-modèles* utilisé dans l'expérience précédente et remplaçons le modèle extensionnel de l'agent-modèle «Prédateurs» par un modèle intentionnel (modèle EDP utilisé pour les proies). L'*agent-modèle* «Proies» reprend quant à lui son modèle EDP. Il nous faut également rajouter l'agent-modèle construisant l'habitat des prédateurs et les agents-observateurs nous permettant d'observer les évolutions en termes de quantité et de distribution spatiale de l'*artefact* produit par l'agent «Prédateurs». Nous utilisons le même habitat pour les proies et pour les prédateurs. Enfin, le tableau 6.2 présente les hypothèses principales que nous avons utilisées pour effectuer la simulation.

Résultats : La figure 6.5 présente les résultats obtenus. On y présente l'évolution dans l'espace des phases du nombre de prédateurs en fonction du nombre de proies. On peut constater que l'on retrouve le pattern proies-prédateurs recherché. De manière stricte, le pattern proies-prédateurs est un cycle stable dans le temps qui tourne autour d'un point fixe, formant ainsi une ellipse (cercle), alors qu'ici la dynamique converge vers ce point fixe.

Paramètres (principaux) : Informations	
$t_{max} = 3600$	: durée de la simulation (nombre de pas de temps)
$N_{individu,proies} = 10^4$	: nombre initial d'individus de la population de proies
$N_{individu,predateurs} = 10^2$	: nombre initial d'individus de la population de prédateurs
discrétisation spatiale	: 16×16

TAB. 6.2: Action 1 – expérience 2 : valeurs des paramètres

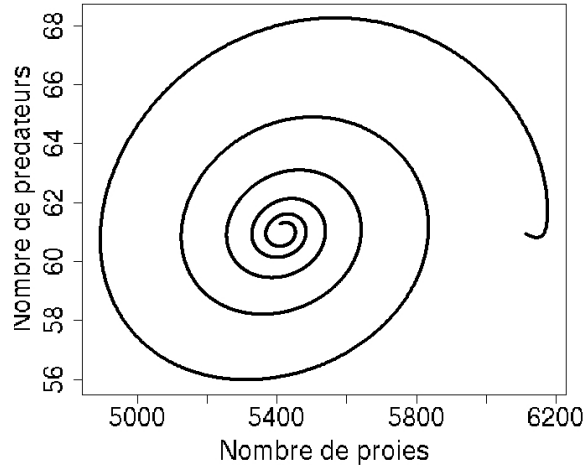


FIG. 6.5: Action 1 – expérience 2 – section 6.4.4 : évolution dans l'espace des phases du nombre de prédateurs en fonction du nombre de proies.

Mais, ce n'est ici qu'une question de valeurs de paramètres et d'étude du modèle EDP.

6.4.5 Analyse

Cette action de modélisation nous permet de valider les mécanismes élémentaires de notre système d'*agents-modèles* et notre méthodologie. Tout d'abord, le cadre de modélisation nous permet de fixer les différentes composantes du système, c'est-à-dire les données partagées et les données observées au moyen des *artefacts* et les phénomènes au moyen des *agents-modèles*. Ensuite, nous voyons que dans la mise en œuvre du cadre de modélisation, il est possible de construire les différents *agents-modèles* de manière incrémentale. Le cadre de modélisation nous guide dans cette construction incrémentale en rationalisant les services que les *agents-modèles* doivent rendre. Dans la première expérience, nous avons ainsi pu

produire des résultats avec un peuplement modélisé à l’aide d’une EDP, puis nous avons pu facilement changer le modèle de l’*agent-modèle* et le remplacer par un modèle IBM avant de produire des résultats très comparables par ailleurs. Enfin, dans la seconde expérience, nous avons pu montrer qu’il était ensuite possible d’intégrer un nouvel *agent-modèle* sans impact sur le reste du système. Les interactions entre les *agents-modèles* sont en effet décrites dans le cadre. Par ailleurs, nous avons par là-même pu retrouver le pattern proies–prédateurs que nous cherchions.

6.5 Action 2 : Modélisation EDP de l’écosystème du Golfe de Gascogne

6.5.1 Question de modélisation et patterns

Question de modélisation : L’objectif de cette étude est le passage à l’échelle et le dimensionnement du modèle EDP au peuplement écologique du GDG. Il s’agit donc de régler les paramètres biologiques identifiés par les experts afin de reproduire un certain nombre de dynamiques recherchées et, si cela est possible, de retrouver au moins dans les grandes tendances des séries historiques de ces peuplements. D’un point de vue thématique, il s’agit également d’étudier l’influence des processus de prédation et de co-occurrence spatiale sur la dynamique des peuplements. En effet, le réseau trophique dans lequel sont impliqués les peuplements et les habitats qui induisent la co-occurrence spatiale de certains prédateurs avec leurs proies conditionnent les dynamiques des différentes espèces en termes de nombre et de distribution spatiale. Or notre modèle EDP permet tout particulièrement de modéliser ces processus de prédation et de distribution spatiale (voir annexe C). Par conséquent, ce qui est intéressant dans cette étude, c’est de voir dans quelle mesure notre modèle est capable de reproduire les dynamiques observées dans le système réel, alors même que le réchauffement climatique n’est pas pris en compte et que l’impact de la pêche est considéré comme constant.

Patterns et artefacts : Nous souhaitons reproduire en tendance à l’aide de notre modèle EDP les séries historiques des différentes espèces que nous étudions. Nous nous basons pour cela sur les données issues des campagnes de mesure EVHOE⁴. Ces données ont été observées dans le système réel par échantillonnage et sont donc en réalité issues d’un modèle statistique. La question est d’étudier dans quelle mesure il est possible de reproduire les tendances observées.

⁴ Les campagnes EVHOE sont des campagnes scientifiques de chalutages conduites par l’IFREMER qui permettent d’obtenir les variations interannuelles des abondances des espèces du GDG.

6.5.2 Cadre de modélisation

Hypothèses et phénomènes : Il nous faut modéliser neuf espèces de poissons, c'est l'hypothèse que nous avons présentée dans la section 6.3.1. De plus nous ne voulons pas modéliser le phénomène du réchauffement climatique et le phénomène de la pêche est considéré comme constant. Remarquons que dans le système réel, les peuplements de poissons sont influencés par une pêche non constante et par le réchauffement climatique, entres autres choses, et les données EVHOE qui vont nous servir de dynamiques de référence sont donc influencées par ces processus exogènes.

Cadre de modélisation : L'objectif ici est de pouvoir mettre en œuvre nos modèles EDP sur un cas réel et avec neuf espèces différentes toutes en interaction. Nous proposons de mettre en œuvre chaque population de poissons dans un *agent-modèle* différent. Nous ne présentons pas ici le système d'*agents-modèles* que nous utilisons de par sa taille, mais la figure 6.6, qui est le cadre réduit utilisé dans la suite, donne une idée de ce qu'il peut être.

Eléments de modélisation : Utiliser un *agent-modèle* par espèce est indispensable pour pouvoir travailler sur un ou quelques *agents-modèles* intensionnels à la fois, en fixant le comportement des autres *agents-modèles* à l'aide de modèles extensionnels. De plus, comme nous avons les données EVHOE, et que c'est d'ailleurs ces données que nous voulons reproduire, nous proposons de baser les modèles extensionnels que nous pourrions utiliser sur ces données là directement. Ainsi, si tous les *agents-modèles* ont des modèles extensionnels basés sur les données EVHOE, le système reproduira exactement les dynamiques recherchées. Il est alors simple et méthodologiquement efficace de configurer le modèle EDP sur une espèce à la fois. Il ne s'agit alors plus de chercher la valeur des paramètres du modèle EDP pour toutes les espèces à la fois, mais une espèce après l'autre. On maîtrise ainsi complètement le contexte de l'*agent-modèle* que l'on configure et le contexte reproduit alors au plus proche la dynamique recherchée.

Nous utilisons donc ici une solution de forçage un peu particulière. En effet, nous utilisons les séries historiques, qui décrivent des quantités totales d'individus des différentes espèces au cours de la période étudiée, pour reconstruire les évolutions des peuplements en termes d'effectif total mais également de distribution spatiale. Cette reconstruction d'une distribution spatiale en fonction de l'effectif total d'une espèce est faite en fonction de l'habitat de cette espèce. L'habitat nous permet de construire une distribution de probabilités dans l'environnement du GDG. Il est alors facile de redistribuer en fonction de cette distribution une population contenant le nombre d'individus décrit par les données.

6.5.3 Dimensionnement du modèle EDP pour deux espèces

Problématique : Avant de chercher à dimensionner le modèle EDP pour toutes les populations étudiées, nous proposons de nous intéresser au cas de deux espèces : les merlus européens et les sardines. De plus, l’approche basée sur le forçage explicite par les séries historiques EVHOE en cours de simulation pour dimensionner les modèles EDP d’un ou deux peuplements est très facile avec notre système d’*agents-modèles*.

Moyens : La figure 6.6 présente le cadre réduit dans lequel nous allons effectivement travailler. En effet, nous n’avons pas besoin de tous les *agents-modèles* du cadre général de modélisation présenté précédemment, mais seulement des *agents-modèles* «Merlus» et «Sardines» et de tous les *artefacts* avec lesquels ils sont en interaction. Il faut également alors tous les *agents-modèles* qui produisent ces *artefacts*.

Pour effectuer le dimensionnement des deux populations, nous réalisons en réalité dans un premier temps le dimensionnement d’une des deux populations, en forçant la deuxième à l’aide d’un modèle extensionnel. Dans un deuxième temps, nous faisons de même avec la deuxième population en forçant la première. Une fois les deux peuplements dimensionnés individuellement, nous pouvons remettre le modèle intensionnel dans les *agents-modèles* des peuplements de merlus et de sardines avec les valeurs des paramètres précédemment identifiés. Il s’agit alors éventuellement de ré-ajuster certains de ces paramètres.

Résultats : Les résultats sont présentés dans les figures 6.7 et 6.8. Les figures 6.7 présente les dynamiques en nombre obtenues – les courbes représentées par des bulles – et les dynamiques en nombre observées (données EVHOE) – les courbes représentées par des lignes. La figure 6.7a présente les résultats pour la population de merlus et la figure 6.7b pour celle des sardines. On peut voir ensuite les distributions spatiales à un pas de temps quelconque de la simulation dans la figure 6.8a pour la population de merlus et la figure 6.8b pour la population de sardines.

Ces résultats ont été obtenus dans une simulation avec les deux *agents-modèles*, de merlus et de sardines, mettant en œuvre le modèle EDP. Les résultats intermédiaires, produits quand un seul des deux peuplements est intensionnel, ne sont pas présentés ici.

Nous arrivons au final à reproduire la dynamique générale en termes de quantité totale d’individus, en tendance en tout cas, même si nous avons de grandes difficultés à reproduire les dynamiques précisément, ce à quoi nous nous attendions. Il était prévisible d’avoir ces difficultés, car dans le modèle EDP la pêche n’est modélisée qu’au travers d’un impact constant (de mortalité) et le réchauffement climatique n’est pas pris en compte. Remarquons enfin que les distributions spatiales correspondent tout à fait aux habitats. Nous ne présentons qu’un exemple de distribution pris à un pas de temps quelconque, mais les dynamiques spatiales sont stables dans le temps.

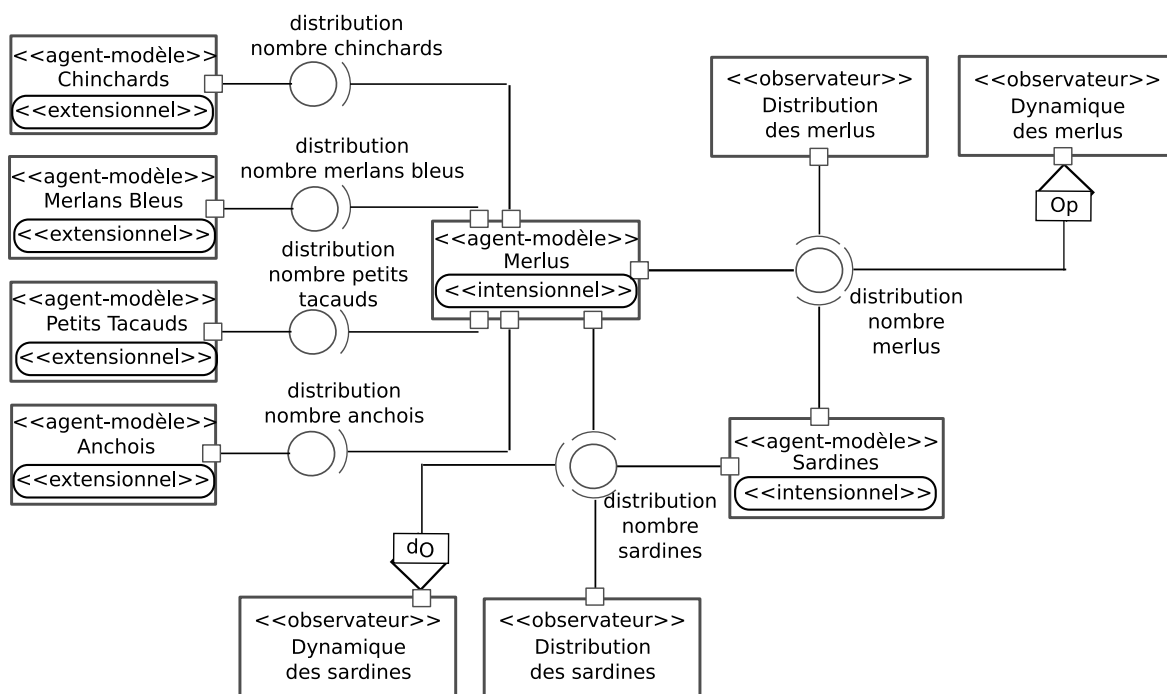


FIG. 6.6: Diagramme d'agents-modèles – Cadre de modélisation pour des scénarios d'étude des merlus et des sardines.

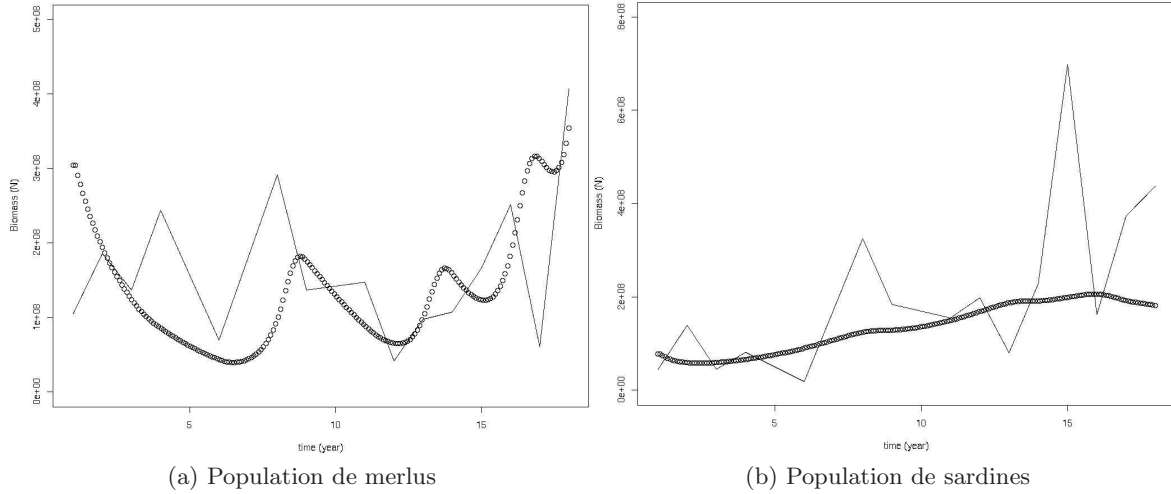


FIG. 6.7: Expérience dimensionnement du modèle EDP au GDG— résultats obtenus en simulation (courbe avec les bulles) et données EVHOE observées (courbe en traits continus) pour les peuplements de merlus et de sardines.

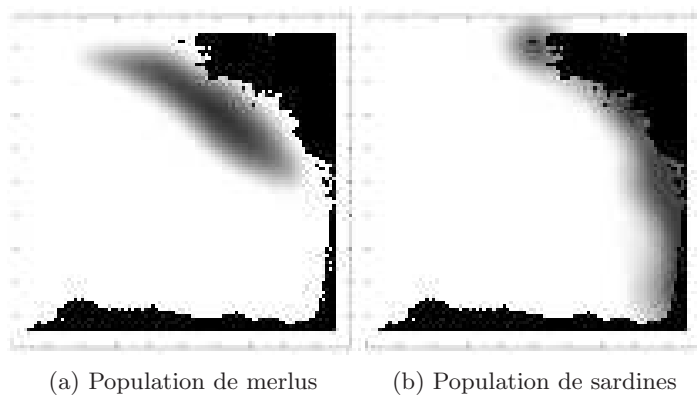


FIG. 6.8: Expérience dimensionnement du modèle EDP au GDG— résultats obtenus en simulation : exemples de distribution spatiale des deux espèces étudiées.

6.5.4 Analyse

De manière générale, nous pouvons voir que notre approche est pertinente pour la pratique de la simulation. Elle permet en effet de modifier sans impact sur le reste du système le modèle des différents *agents-modèles*, c'est-à-dire la manière de modéliser les phénomènes structurant le modèle global. Ainsi, nous avons pu utiliser une approche à base de forçages de séries historiques pour dimensionner nos modèles, approche qui est originale. Et, nous pouvons alors chercher les valeurs des paramètres du modèle EDP pour chaque espèce indépendamment des autres espèces en forçant les dynamiques de celles-ci avec les séries historiques.

D'un point de vue méthodologique, il est intéressant de pouvoir considérer tous les éléments qui ont une influence sur la dynamique du système en tant que telle. Considérer les éléments, c'est pouvoir repérer ce qu'ils représentent et les données qu'ils influencent. C'est en particulier vrai pour les forçages qui ne sont pas toujours explicites dans les modèles. Or, dans une approche de manipulation d'un modèle complexe, il n'est pas question de simplement analyser des peuplements virtuels en interaction, mais l'évolution d'une représentation computationnelle complexe de peuplements qui contient des modèles de peuplements, mais également des artifices conceptuels ou computationnels. Or, les forçages sont les limites d'un système et, en cela, ils doivent être clairement identifiés comme des modèles. Remarquons enfin pour aller dans le même sens, que nous avons utilisé des données issues d'un modèle statistique et qui nous permettent de reconstruire à chaque pas de temps les distributions des populations forcées à l'aide des habitats qui sont eux aussi des approximations des habitats réels, soit des modèles.

Remarquons également que nous n'avons dimensionné notre système pour deux populations. En effet, ce travail de dimensionnement pour les neuf espèces est très long et celui effectué sur ces deux espèces est déjà fructueux d'un point de vue scientifique. Notons cependant qu'il est maintenant simple d'engager de nouvelles tâches de modélisation sur le dimensionnement d'autres espèces, ce qui est le but justement du cadre de modélisation.

Enfin, d'un point de vue thématique, le modèle EDP que nous utilisons nous permet de reproduire les tendances en nombre des séries historiques. Seuls les impacts des co-occurrences spatiales entre les espèces en interaction trophique et de la prédation elles-mêmes sont modélisés de manière explicite et variable. L'impact de la pêche sur le système est modélisé dans les équations au travers d'un paramètre de mortalité qui est constant tout au long de la simulation. Il est alors extrêmement intéressant de voir que ces processus expliquent déjà en partie les tendances dans les dynamiques des peuplements. On peut alors en déduire de cette étude, qui nécessite cependant des études complémentaires, que les processus de prédation et de co-occurrence spatiale entre peuplements pourraient être décisifs par rapport à d'autres processus dans la compréhension des dynamiques des peuplements, et en tout cas de ceux des merlus et des sardines.

6.6 Action 3 : Modélisation IBM de l'écosystème du Golfe de Gascogne

6.6.1 Question de modélisation et patterns

Question de modélisation : Nous proposons ici de présenter le travail effectué sur le dimensionnement du modèle IBM au peuplement écologique du GDG. D'un point de vue thématique, outre l'exécution d'un modèle IBM de cette taille et de ce cas d'étude, nous nous intéressons à l'influence des processus de prédation et de co-occurrences spatiales sur la dynamique des peuplements, comme nous l'avons fait dans l'étude de la section précédente. Donc, nous sommes sur une question de modélisation qui est relativement proche de la précédente. La différence tient essentiellement à l'approche de modélisation, mais aussi au modèle lui-même qui ne contient pas de composante liée à la pêche. De plus, dans le cas du modèle EDP, les interactions entre populations ne sont pas influencées par l'utilisation d'*agents-modèles*. L'approche IBM permet par contre des interactions plus fines entre espèces, interactions qui ne sont alors possibles qu'au sein d'un même modèle et donc d'un même *agent-modèle*. Donc, la problématique se pose davantage en termes de problématiques de modélisation IBM avec un très grand nombre d'agents, ce qui peut entraîner des questions de discrétisations hétérogènes au sein du modèle.

Patterns et *artefacts* : Les phénomènes qui sont modélisés dans le modèle IBM sont strictement ceux du peuplement de poissons (voir annexe B.1). Il y a donc moins de sens ici à utiliser comme dynamiques de référence les séries historiques issues des campagnes de mesure EVHOE qui intègrent l'influence de la pêche entre autres. Hors contexte, il est difficile pour les écologues d'exhiber *a priori* des patterns dans les dynamiques alors même que les données observées n'existent pas. Nous proposons par conséquent de partir de l'hypothèse faite par les écologues que les peuplements sont relativement stables dans le temps, aussi bien en termes de nombre que de distribution spatiale. Le problème est quelque peu délicat ici parce qu'il n'est pas évident de qualifier clairement cette stabilité. En effet, certains peuplements réels exhibent des dynamiques extrêmement variables dans le temps, qui peuvent s'expliquer par différents processus : une forte reproduction et une forte mortalité avec des périodes de reproduction courtes, un recrutement qui ne dépend pas des stocks, c'est-à-dire qu'il n'y a pas de lien clair entre le nombre de nouveaux poissons intégrant la population et le nombre de géniteurs, etc. Cependant, nous avons identifié un certain nombre d'indicateurs pour le peuplement dans la section 6.3.1 : le poids moyen global, la latitude préférentielle moyenne et le niveau trophique moyen. Or, la stabilité du peuplement pourrait s'exprimer au travers d'une stabilité au cours du temps de ces indicateurs. Nous pouvons ainsi nous baser sur les dynamiques de ces indicateurs. Par conséquent, nous avons besoin des *artefacts* décrivant le poids moyen global, la latitude préférentielle moyenne et le niveau trophique moyen.

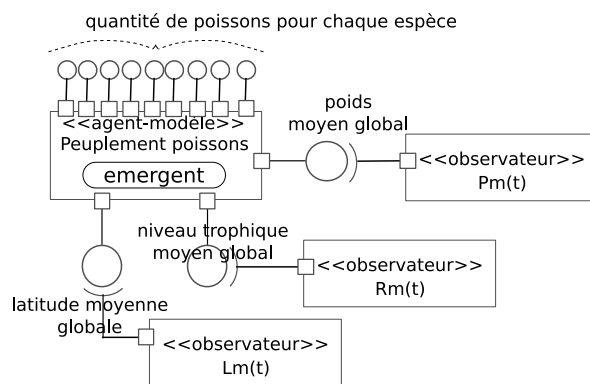


FIG. 6.9: Diagramme d'agents-modèles – Cadre de modélisation pour des scénarios d'étude d'écosystème marin (Pm : poids moyen global, Rm : Niveau trophique moyen, Lm : latitude préférentielle moyenne).

6.6.2 Cadre de modélisation

Hypothèses et phénomènes : Les peuplements sont considérés comme un seul phénomène et intègre donc un *agent-modèle*. Les habitats n'étant pas variables, ils sont directement construits dans l'*agent-modèle* des peuplements (voir section 4.3.2 - D).

Cadre de modélisation : Le cadre de modélisation est celui de la figure 6.9. On peut voir l'*agent-modèle* représentant le peuplement de poissons qui réalise neuf *artefacts* décrivant les effectifs pour chaque espèce. De même, l'*agent-modèle* réalise trois *artefacts* décrivant le poids moyen global, la latitude préférentielles globale et le niveau trophique moyen global à chaque pas de temps. Il y a également des agents-observateurs, dont la plupart ne sont pas présentés dans la figure pour des raisons de clarté de présentation.

6.6.3 Dimensionnement du modèle IBM sans hétérogénéités de discrétisation

Problématique : L'objectif de cette étude est de dimensionner notre modèle IBM (voir annexe B.1) en exhibant les problèmes de précision des résultats. En effet, le peuplement du GDG que nous modélisons est composé de neuf espèces qui ont des proportions respectives très différentes. Par exemple, et comme nous le détaillerons ensuite, la population de chinchards représente environ 74% du nombre total d'individus du peuplement du GDG. En comparaison, la population de baudroies n'en représente que 0.02%. Il est donc nécessaire d'étudier ce système au regard de ces problèmes de précision. Nous proposons donc ici d'étudier les

Données réelles									
Espèce	Anchois	Baudroie	Chinchard	Merlan	Merlan Bleu	Merlu	Petit Tcaud	Sardine	Tcaud
$Q_{individus}$	114.10^6	4.10^6	15.10^9	44.10^6	3.10^9	407.10^6	1.10^9	438.10^6	237.10^6
Proportion (%)	0.5631	0.0198	74.0960	0.2173	14.8192	2.0104	4.9397	2.1636	1.1707
Q_{agents}	563.1	19.8	74096	217.3	14819.2	2010.4	4939.7	2163.6	1170.7

TAB. 6.3: $Q_{individus}$: nombre d'individus par espèce (données EVHOE-2004) – Q_{agents} : nombre d'agents espéré par espèce pour un nombre total de 10^5 .

dynamiques globales du système sans solution particulière permettant d'améliorer la précision pour telle ou telle espèce.

Enfin, nous ne nous intéressons pas particulièrement ici à la dynamique en termes de distributions spatiales, même si celle-ci est mise en œuvre dans le modèle et que les processus spatiaux restent structurants. Seulement, nous ne chercherons pas à valider la qualité des distributions spatiales, nous ne ferons que valider le fait que les distributions ne sont pas aberrantes.

Moyens : Pour mettre en œuvre cette étude, nous achoppons immédiatement sur un premier problème. Tout d'abord, il s'agit de modéliser les neuf espèces du GdG qui représentent pour l'année 2004 de l'ordre de 20.10^9 individus⁵. Pour nous, il s'agit de modéliser ces peuplements en utilisant un nombre d'agents computationnels qui soit au maximum de l'ordre de 10^5 . Le problème est que si nous voulons que les agents computationnels puissent interagir à la manière des individus réels, il faut que l'on puisse conserver les proportions entre espèces. Le tableau 6.3 présente, pour chaque espèce, le nombre total d'individus de la population estimée en 2004 par les campagnes EVHOE, la proportion de chaque population par rapport au nombre total des individus de toutes les espèces, et enfin, le nombre d'agents⁶ utilisé par espèce si l'on utilise au total 10^5 agents. Ainsi, nous constatons que pour deux peuplements, les baudroies et les merlans, le nombre d'agents est inférieur à 500. On peut estimer avant d'effectuer la simulation que, d'après ce que l'on a vu dans le chapitre précédent, la précision des résultats pour ces peuplements sera faible. En particulier, la moindre variation du nombre de baudroies est extrêmement critique quand on manipule une vingtaine d'agents computationnels pour reproduire une population de 4.10^6 individus. Nous pouvons donc déjà en déduire qu'il sera difficile d'interpréter et d'utiliser ces résultats.

Résultats : La figure 6.10a présente l'évolution en nombre des différentes espèces étudiées. La figure 6.10b présente l'évolution de la latitude préférentielle moyenne au cours de la simulation. La figure 6.10c présente l'évolution du niveau trophique moyen au cours de la simulation. Nous pouvons constater que les indicateurs sont relativement stables au cours du temps. Ils semblent avoir une légère tendance à baisser qui semble s'expliquer par une augmentation au cours du temps du nombre de chinchards (courbe turquoise) qui est une

⁵ Estimation faite à partir des données estimées par les campagnes EVHOE. Cette estimation est bien évidemment un ordre de grandeur.

⁶ es nombres d'agents doivent bien sûr être arrondis

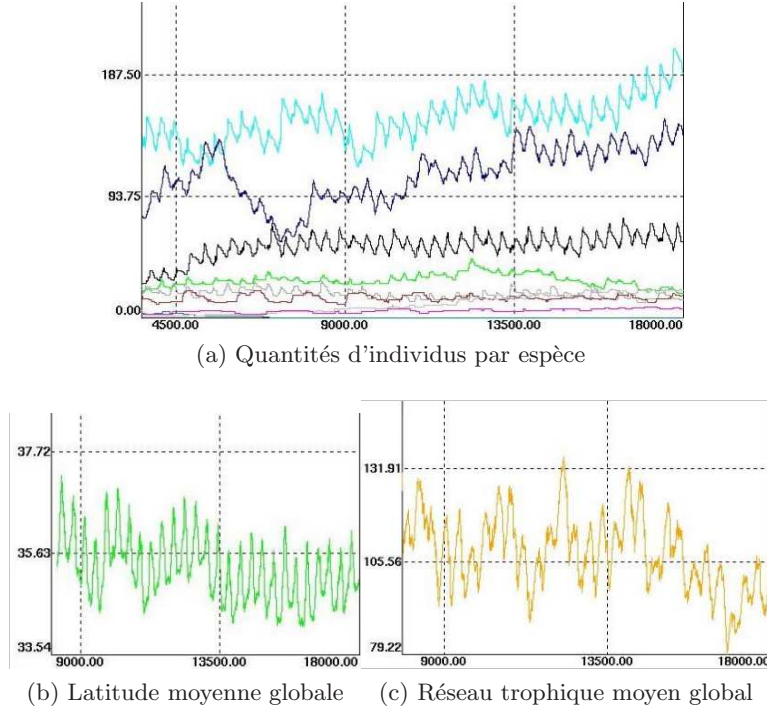


FIG. 6.10: Action 3 – expérience 1 : résultats de simulation.

espèce d'affinité chaude et de niveau trophique moyen, ce qui est conforme au cadre théorique.

6.6.4 Analyse

Cette étude nous a permis de dimensionner le modèle IBM au moins en partie pour le cas des neuf espèces étudiées du GDG. Les résultats montrent que les indicateurs sont plutôt stables au cours du temps, même si l'augmentation régulière du nombre de chinchards provoque une légère baisse des indicateurs. Cependant, les résultats sont difficilement utilisables en l'état et il nous reste encore du travail de simulation sur cette question de modélisation.

Le travail de dimensionnement sur un modèle de ce type et sur un cas d'étude de cette taille est très long et les résultats présentés ne sont que préliminaires. Nous remarquons que les effectifs des populations réelles, et surtout les proportions entre espèces qui doivent être respectées dans le modèle IBM font que l'expérimentation du modèle avec un facteur d'échelle de 1 est critique. C'est cette expérience qui nous a conduit à ajouter un facteur d'échelle dans notre système.

6.7 Action 4 : Couplage avec une pêcherie et hétérogénéités

6.7.1 Question de modélisation et patterns

Question de modélisation : L’objectif de cette action de modélisation est d’aborder les problèmes d’hétérogénéité d’échelles. En effet, la première perspective ouverte par l’utilisation d’*agents-modèles* est de pouvoir construire des modèles hétérogènes. La question de modélisation est ainsi de pouvoir mettre en œuvre un modèle mixte représentant un peuplement de poissons exploité par une pêcherie dans un environnement variable. La question porte sur le couplage de modèles hétérogènes en termes de description et d’échelles spatiales et temporelles.

Patterns et *artefacts* : Nous voulons observer un peuplement théorique exploité dans un environnement variable. Le peuplement se caractérise par une stabilité *a priori* du nombre d’individus. De plus, la distribution spatiale du peuplement évolue dans l’environnement et on observe une dérive du peuplement vers le Nord du GdG.

6.7.2 Cadre de modélisation

Hypothèses : Nous faisons l’hypothèse que le réchauffement climatique explique la migration du peuplement vers le Nord du GdG et que l’habitat du peuplement est fonction de la profondeur et de la température.

Les activités de pêche sont effectuées par un ensemble d’unités de pêche et ne sont pas spatialisées. Et les peuplements sont influencés par la pêche sur une base annuelle. Par contre, les peuplements de poissons évoluent au jour et les processus de densité dépendance requièrent des descriptions fines de la distribution spatiale du peuplement.

Cadre de modélisation : Il est donné par la figure 6.11. On y voit quatre *agents-modèles* représentant le peuplement de poisson, les activités de pêche, la description de l’habitat du peuplement et enfin la description du réchauffement des eaux du GdG. L’*agent-modèle* décrivant les scénarios de réchauffement nous permet de modéliser le réchauffement des eaux indépendamment du reste du système.

Le point important ici est celui des *agents-modèles* représentant le peuplement et les activités de pêche. Le phénomène des activités de pêche évolue à l’année et est global, c’est-à-dire que les pêcheurs pêchent sur un stock de poissons qui est la quantité totale de poissons du GdG et produisent une mortalité par pêche globale. Le phénomène du peuplement évolue

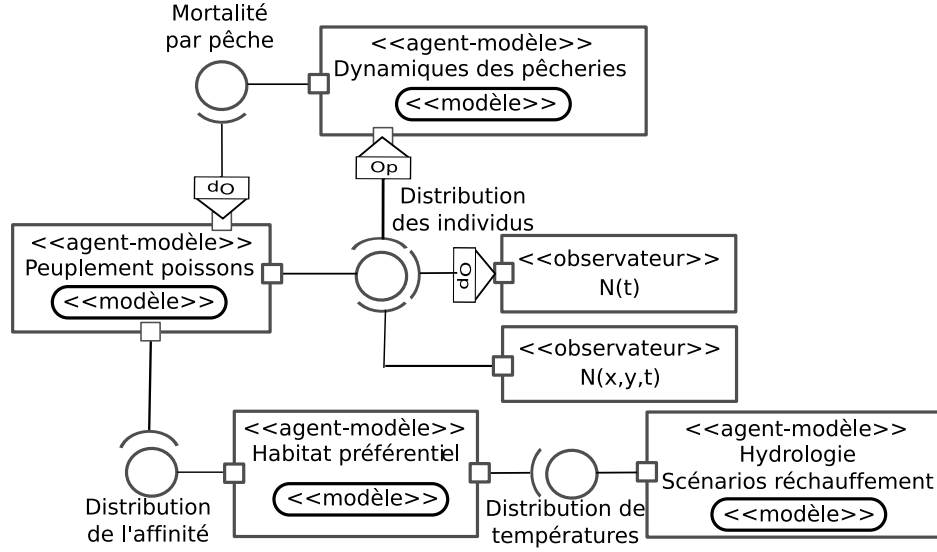


FIG. 6.11: Diagramme d'agents-modèles – Cadre de modélisation pour des scénarios d'étude du peuplement marin du Golfe de Gascogne couplé à une pêche

quant à lui au jour et l'environnement y est discrétisé de manière fine. Ainsi, les *artefacts* qu'il écrit, respectivement la mortalité par pêche (le nombre de poissons morts par pêche) et la distribution du peuplement en nombre ne peuvent qu'être décrits de manière différente. Ainsi, pour permettre aux deux *agents-modèles* d'interagir, il est nécessaire qu'ils utilisent des *opérations* pour s'adapter aux *artefacts*.

Eléments de modélisation : Regardons plus en détails les *opérations* qui sont utilisées par les *agents-modèles* du peuplement et des activités de pêche. Concernant l'*agent-modèle* du peuplement d'abord, la mortalité par pêche doit être prise en compte par le peuplement de manière quotidienne. Cependant, remarquons que la mortalité reste globale. En effet, pour pouvoir distribuer dans la population des poissons la mortalité par pêche, il faut avoir des connaissances sur l'habitat des poissons. Nous aurions pu alors soit construire un autre *agent-modèle* pour effectuer cette manipulation, soit directement intégrer la mortalité par pêche globale dans le modèle de peuplement. Nous choisissons la deuxième solution qui semble plus pertinente, car c'est à l'*agent-modèle* des peuplements qui manipulent un modèle de peuplement de traduire la mortalité globale en un impact sur le peuplement. Reste alors la question de la discrétisation temporelle. Ainsi l'*artefact* écrit par l'*agent-modèle* décrivant les activités de pêche est décrit comme suit : $\langle M, id, \langle t, t_{min}, t_{max}, 360, year \rangle \rangle$ avec M la sémantique correspondant à la mortalité par pêche. Nous constatons que l'*artefact* est bien global et n'occupe que l'axe temporel, avec un pas de temps de mise à jour qui est de 360 (le pas de temps minimum du système est le jour) et l'unité de mise à jour est alors bien l'année.

Or la *jonction* qui permet à l'*agent-modèle* de récupérer la mortalité par pêche requiert une information dont la description est : $\langle M, id, \langle t, t_{min}, t_{max}, 1, day \rangle \rangle$. L'*opération* a alors le fonctionnement suivant : à chaque pas de temps, la *propriété* s'active est récupère la valeur de l'*artefact*. Cette valeur est transformée par l'*opération* qui retourne la valeur d'entrée divisée par 360 (le facteur de conversion de l'année en jour).

Remarquons que pour distribuer la mortalité par pêche dans l'environnement, nous ajoutons un comportement à l'*agent-modèle* des peuplements qui, basé sur la mortalité globale (journalière), modifie une autre *propriété* qui décrit la distribution dans l'habitat de la mortalité par pêche (dont se sert le modèle de peuplement). Ce comportement utilise une distribution de probabilités qui est fonction de l'habitat et qui permet alors de répartir la mortalité dans l'environnement. Ensuite, un comportement de mortalité peut récupérer la distribution de mortalité et la reporter dans l'environnement des poissons en récupérant case après case le nombre de morts et en détruisant le nombre d'individus de la case correspondante dans l'environnement du peuplement. Notons que le comportement peut alors se retrouver avec un nombre de morts dans une case supérieur au nombre d'individus existant à ce pas de temps dans la case correspondante de l'environnement du peuplement. Le comportement mémorise alors le nombre d'individus qu'il aurait dû détruire et après être passé sur toutes les cases, il tire aléatoirement dans la population d'individus le nombre d'individus qu'il est nécessaire de détruire encore.

Enfin, il reste à voir comment l'*agent-modèle* des pêcheries récupère l'information sur la quantité totale d'individus. La *propriété* de l'*agent-modèle* des pêcheries qui lit l'*artefact* de distribution du peuplement utilise ainsi une *opération* qui transforme l'information de l'*artefact* décrivant la distribution du peuplement. Cette *propriété* est caractérisée par deux activités, une donnée (une *propriété* porte une information) et une variable temporaire. Ainsi, l'objectif de cette *propriété* est de calculer la moyenne au cours de l'année du nombre total d'individus du peuplement. Pour ce faire, elle a une première activité qui, chaque année met sa donnée et sa variable temporaire à zéro. Ensuite, la deuxième activité s'exécute quotidiennement et utilise l'*opération* pour récupérer le nombre total d'individus chaque jour. L'*opération* fonctionne de la manière suivante : la *propriété* récupère la valeur de l'*artefact*, la passe à l'*opération* qui additionne le nombre d'individus de chaque case et renvoie la valeur totale. La *propriété* peut alors faire la moyenne entre la donnée qu'elle porte et le nombre total d'individus qu'elle récupère. Pour faire cette moyenne, elle utilise sa variable temporaire pour connaître le nombre de valeurs que représente sa donnée. Il faut alors que pour chaque nouvelle valeur récupérée, elle incrémente sa variable temporaire.

6.7.3 Comparaison des modèles EDP et IBM du peuplement

Problématique : L'objectif est de modéliser le peuplement à l'aide d'un modèle EDP dans un premier temps, puis d'un modèle IBM ensuite.

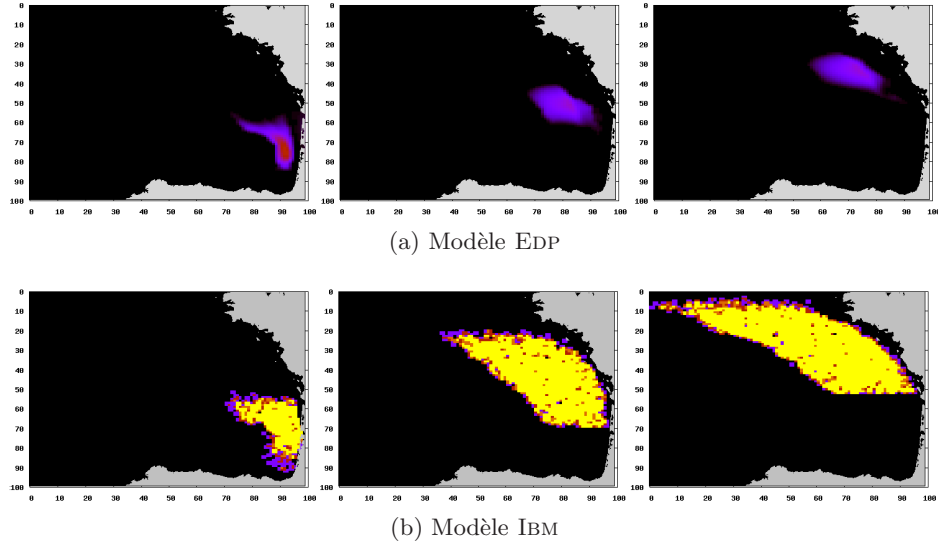


FIG. 6.12: Action 4 – expérience 1 : évolution de la distribution du peuplement suivant le modèle utilisé : 3 pas de temps sont présentés.

Moyens : Nous utilisons pour cela le modèle EDP décrit dans l'annexe C et le modèle IBM décrit dans l'annexe B.1. De plus, les activités de pêche sont modélisées à l'aide d'un modèle extensionnel.

Résultats : Les résultats sont présentés dans la figure 6.12. On y voit trois pas de temps pris au début, au milieu et à la fin de la simulation. On constate alors que le peuplement migre correctement vers le Nord du GDG quel que soit le modèle utilisé. On remarque également que la distribution spatiale du peuplement avec le modèle IBM est davantage conforme au cadre théorique qu'avec le modèle EDP.

6.7.4 Analyse

Nous avons montré un modèle global composé d'*agents-modèles* mettant en œuvre à des rythmes d'évolution différents des modèles hétérogènes, de par leur description spatiale et l'approche de modélisation utilisée. Nous avons pu alors reproduire le pattern recherché, expliqué essentiellement par un réchauffement des eaux du GDG. La solution d'*agents-modèles* a donc permis l'expérimentation de modèles hétérogènes, mais également de deux modèles différents pour le peuplement (IBM et EDP) ce qui nous permet d'en comparer les dynamiques facilement.

6.8 Action 5 : Modélisation économique du marché des navires de pêche

6.8.1 Question de modélisation et patterns

Question de modélisation : Nous avons jusqu'ici concentré nos efforts sur les problématiques d'écosystème et sur le couplage de système peuplements-pêcheries avec des pêcheries modélisées comme des super-prédateurs. Nous souhaitons ici valider hors contexte les processus structurants du modèle économique que nous avons construit en collaboration avec des économistes des pêches de l'IFREMER (voir l'annexe B.1).

Il y a deux aspects qui nous intéressent ici. D'un point de vue thématique, l'étude nous permet de dimensionner notre modèle IBM économique et de valider les processus structurants que sont 1° la dépréciation de la valeur d'un navire de pêche et 2° les mécanismes de réponse du marché de la pêche à des scénarios théoriques de régulation de ce marché, régulations qui se traduisent par la limitation des entrants potentiels dans l'économie, par des incitations à la sortie des pêcheurs de l'économie ou encore par le contrôle des prix des navires neufs. D'un point de vue plus général, cette action nous permet de renforcer l'idée qu'il est facile avec notre proposition de système d'*agents-modèles* d'isoler un agent précis afin d'expérimenter localement son comportement en forçant le comportements des autres agents du système. De même, nous montrons qu'il est tout à fait possible de construire des modèles IBM économiques dans notre simulateur.

Patterns et *artefacts* : Nous cherchons à reproduire la dynamique de la dépréciation du prix d'un navire de pêche (voir section 6.3.2). Nous voulons également observer le nombre de pêcheurs en fonction de scénarios de régulation du marché. Ces scénarios peuvent entraîner des dynamiques différentes en termes de nombre de pêcheurs. Nous nous attendons par exemple à ce que des incitations à la sortie des pêcheurs de l'économie et un contrôle des nouveaux entrants impliquent une diminution des acteurs de l'économie de la pêche, avec éventuellement une reconversion de tous les pêcheurs vers d'autres économies (scénario théorique). Les *artefacts* dont nous avons besoin sont donc celui décrivant le prix moyen d'un navire de pêche, celui représentant le nombre de pêcheurs de l'économie et enfin un certain nombre d'*artefacts* permettant le contrôle du marché et la mise en œuvre de scénarios de régulation : la valeur de la prime de sortie de l'économie, le prix d'un navire neuf, le nombre d'entrants possibles.

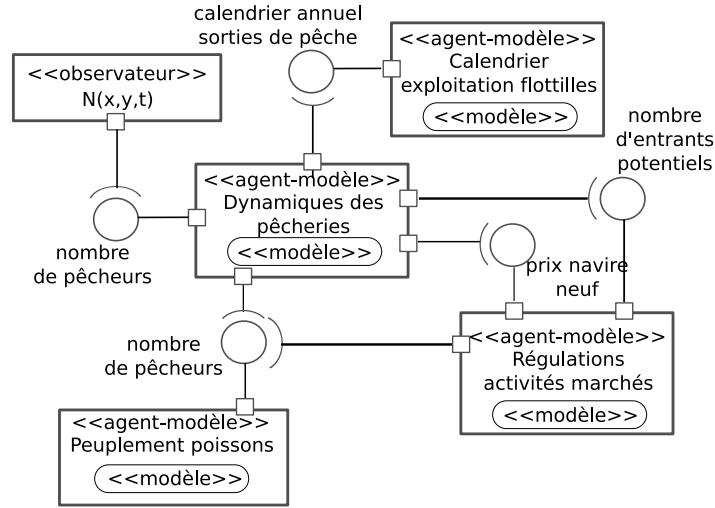


FIG. 6.13: Diagramme d'agents-modèles – Cadre de modélisation pour des scénarios d'étude du marché des navires de pêches

6.8.2 Cadre de modélisation

Hypothèses et phénomènes : Le phénomène que nous voulons étudier est celui du marché des navires de pêche. Dans l'étude précédente (couplage du peuplement avec une pêcherie, section 6.7), nous avons considéré les activités de pêche comme un super-prédateur. Ici, nous voulons enrichir la modélisation de ce phénomène en permettant l'interaction entre lui et le phénomène reproduisant les politiques de régulations en particulier. De plus, même si nous voulons isoler le phénomène du marché de la pêche, il nous faut intégrer le phénomène reproduisant le peuplement écologique (les stocks) afin que les pêcheurs puissent exercer leur activité de pêche et calculer un profit.

Nous pouvons également remarquer que comme la question de modélisation est l'étude de la réaction du marché à des scénarios pré-établis de régulation, nous proposons que ce phénomène de régulation du marché soit forcé, c'est-à-dire construit à partir d'un scénario pré-calculé.

Cadre de modélisation : Le cadre de modélisation que nous mettons en œuvre est celui de la figure 6.13. Nous pouvons y voir un agent-modèle "Régulation activités marchés" qui décrit les politiques de gestion du nombre d'entrants potentiels dans l'économie des pêches, des prix de navires de pêche neufs et d'éventuelles restrictions sur les quantités de ressource pouvant être prélevées par les pêcheries. Son modèle est donc extensionnel.

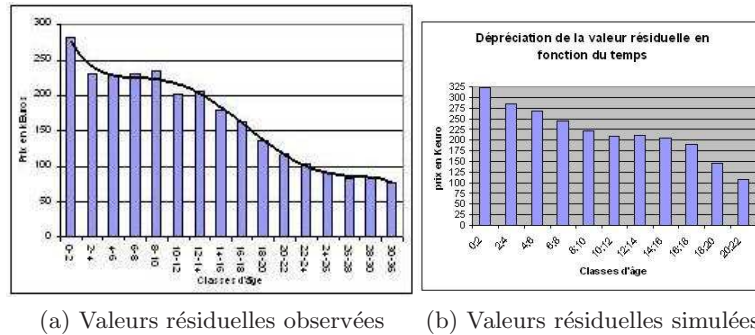


FIG. 6.14: Diagramme d'agents-modèles – Cadre de modélisation pour des scénarios d'étude du marché des navires de pêches

6.8.3 Processus de dépréciation de la valeur d'un navire de pêche

Problématique : Nous voulons observer l'évolution de la valeur d'un navire de pêche au cours du temps. Pour cette étude, nous voulons donc que les différents processus soient fixés pour permettre d'évaluer le seul processus de dépréciation de la valeur d'un navire.

Moyens : Pour effectuer cette étude, nous mettons en œuvre un seul pêcheur dans le marché. Les stocks ne sont pas une limitation et le pêcheur prélève de la ressource suivant un calendrier d'activités (exprimé en nombre de jours de sorties par mois pour une année d'activité). De plus, il n'y a qu'une régulation qui est mise en œuvre, celle qui limite le nombre d'entrants possibles dans le marché, afin qu'aucun autre pêcheur ne puisse y entrer. La seule observation est celle de l'évolution de la dépréciation de la valeur (prix réel) d'un navire de pêche qui reflète les réinvestissements réguliers du pêcheur. Ces réinvestissements se font grâce au profit que le pêcheur fait en pratiquant son activité. Il décide ainsi en fonction de la valeur de son navire de réinvestir à partir du moment où celle-ci diminue suffisamment et qu'il a les moyens économiques de réinvestir.

Résultats : La figure 6.14a présente la valeur résiduelle moyenne observée dans l'économie réelle et la figure 6.14b montre la valeur résiduelle obtenue par simulation. Nous pouvons constater que les deux dynamiques sont très proches pour les douze premières classes d'âge et que nous arrivons à reproduire la dynamique recherchée.

6.8.4 Scénarios d'incitation à la sortie de l'économie

Problématique : L'objectif de cette étude est d'observer la réaction du marché à des scénarios de régulation. En particulier, nous proposons d'étudier deux scénarios d'incitation à la sortie de l'économie des pêches. Ces incitations se font au travers des primes qui visent à encourager les pêcheurs à se reconvertir et donc à quitter la pêche. Ce type de régulation n'a pas toujours dans le système réel les effets attendus. On a pu ainsi constater que la diminution du nombre de pêcheurs entraînée par des mesures d'incitation s'est accompagnée d'une augmentation de l'effort de pêche, donc de la biomasse pêchée. Alors que certains pêcheurs quittaient l'économie, d'autres ré-armaient leurs navires de pêche, ce qui en augmentait la capacité de production. Il est ainsi très difficile d'envisager les équilibres possibles d'un tel marché et il est nécessaire de pouvoir jouer un certain nombre de scénarios, en commençant avec des scénarios relativement simples, mais qui peuvent déjà en eux-mêmes entraîner des dynamiques inattendues du système.

Moyens : Nous proposons de mettre en œuvre deux scénarios de gestion du marché. Le premier ne limite en aucune manière le marché, sauf que là encore, aucun nouveau pêcheur ne peut entrer dans l'économie. Il n'y a donc pas d'incitation à la sortie. Le deuxième scénario propose une incitation financière à la sortie de l'économie.

Pour cette étude, nous avons initialisé le système avec 70 pêcheurs. Nous modélisons l'activité des chalutiers 12–16 mètres et 16–24 mètres. L'exploitation se fait dans la zone Sud Bretagne et le calendrier d'exploitation est le même que pour l'expérience précédente (nombre de jours de sorties par mois pour l'année d'activité). La biomasse est considérée comme un stock que les différents pêcheurs se partagent. Nous mettons en œuvre les deux scénarios suivants :

Scénario 1 : pas d'incitation à la sortie (figure 6.15).

Scénario 2 : prime de sortie (fonction de la jauge du bateau) (figure 6.16).

Résultats : Les résultats consistent en deux courbes, qui représentent : 1° le nombre de pêcheurs au cours du temps (figure du haut) et 2° le profit moyen d'un pêcheur au cours du temps. Nous constatons que sans incitation à la sortie, il n'y a plus de pêcheurs après 6 ans d'exploitation (figure 6.15). Alors, qu'avec des incitations à la sortie, il y a des pêcheurs pendant 13 ans (figure 6.16). Ainsi, sans incitation, les pêcheurs profitent de leur outil de production tant que le profit moyen est positif. Dès qu'ils commencent à perdre de l'argent, ils quittent la pêche. Les incitations permettent ici de prolonger l'exploitation des stocks en poussant certains pêcheurs à sortir plus vite de l'économie des pêches, ce qui permet aux autres d'augmenter leur production et leur gain associé.

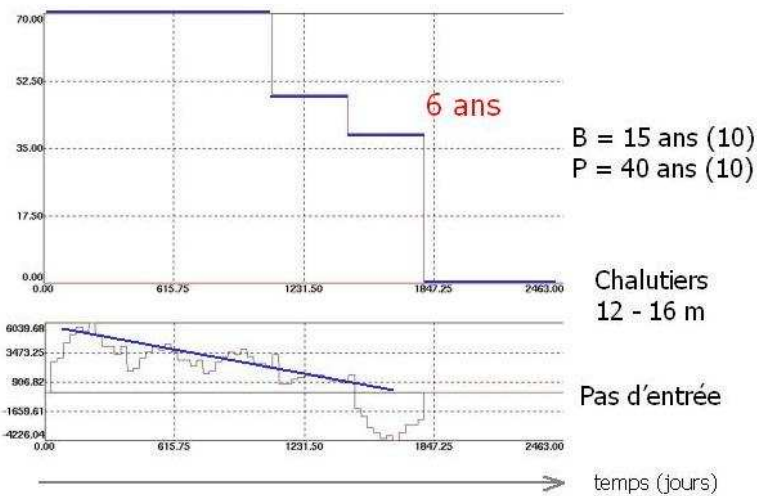


FIG. 6.15: Action de modélisation du marché des navires de pêche – résultats scénario 1.

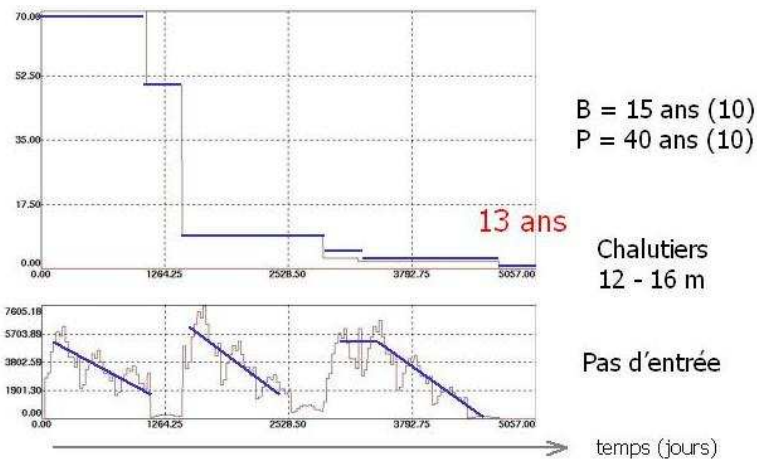


FIG. 6.16: Action de modélisation du marché des navires de pêche – résultats scénario 2.

6.9 Complexité, outillage et rationalisation

Un projet de la taille de Chaloupe induit un processus de simulation long et fastidieux qui s'articule autour de plusieurs actions de modélisation qu'il faut pouvoir mettre en cohérence et capitaliser. Tout au long du projet, nous avons dû gérer la complexité des différents processus de simulation. Du modèle complexe à construire, associant différentes approches de modélisation et différentes échelles ou discrétisations, en passant par l'expérimentation de ce modèle, qui entraîne le plus souvent une multiplication des jeux de paramètres, mais surtout une multiplication des versions du modèle, à la production de résultats, qu'il faut pouvoir associer au modèle exact qui les a produit et au jeu de paramètres utilisé, l'étude d'un seul modèle requiert déjà un outil et une méthodologie qui offre souplesse et en même temps rigueur.

De plus, l'étude d'un modèle s'intègre avant tout dans un processus de simulation plus complet qui aborde différentes questions et points de vue et qui se traduit par une multiplication des actions de modélisation, des modèles et des simulations. Il existe un certain nombre de méthodologies qui visent à rationaliser de manière locale tel ou tel processus de la simulation, les plans d'expérience en sont un exemple. Classiquement cependant, la méthodologie de simulation propose un enchaînement simple : 1° construction du modèle conceptuel, 2° construction du modèle computationnel, 3° expérimentation et production de résultats et enfin 4° analyse. On peut ensuite reprendre cette «boucle» méthodologique et enrichir le modèle de départ avec les résultats obtenus. Seulement, concrètement, dans un projet aussi important que celui de Chaloupe, avec des modèles aussi peu commodes que ceux que nous avons manipulés, cette méthodologie ne peut pas s'appliquer telle quelle. Si l'on s'intéresse à un modèle en particulier, il s'agit d'un côté de pouvoir construire et expérimenter un modèle complexe en interagissant avec lui librement, en testant certains processus localement ou en explorant les jeux de paramètres possibles, et en même temps, de l'autre, il faut pouvoir figer la structure du modèle pour produire des résultats. De plus, la collaboration entre différents experts nécessite de pouvoir effectuer des contributions locales indépendamment du reste du modèle et de pouvoir s'entendre sur un modèle global rigoureux. Enfin, la méthodologie classique ne prend pas en compte tout le cycle de simulation qui requiert l'étude de plusieurs modèles. En effet, une action de modélisation coïncide le plus souvent avec plusieurs autres actions de modélisation qui s'enrichissent les unes les autres et que l'on doit pouvoir mettre en rapport afin de rationaliser tout le processus de simulation.

Pour répondre à ces problèmes, nous avons cherché des solutions en termes de modèle, d'outils et de méthodologie. En termes de modèle d'abord, nous avons mis au point notre proposition d'*agents-modèles*. Cette proposition permet de rationaliser la construction d'un modèle complexe et ainsi d'envisager le couplage de modèles hétérogènes (hétérogénéité de modélisation et d'échelles). Elle permet également d'exhiber un cadre de modélisation qui facilite autant les processus collaboratifs de construction de modèles que les modélisations incrémentales. Enfin, la proposition d'*agents-modèles* doit permettre de concevoir le modèle

complexe comme une société de modèles en interaction qui doivent pouvoir être en partie autonomes.

En termes d'outils, nous avons construit une plate-forme de simulation qui permet de mettre en œuvre des systèmes d'*agents-modèles* décrit au moyen de fichiers XML. Nous avons également mis au point un certain nombre de scripts permettant d'automatiser la production et l'analyse des résultats. Ceci est rendu possible avant tout par le fait que la phase de construction (dite de modélisation) d'un modèle et la phase de simulation de ce modèle sont séparées (voir section 4.4). En effet, en phase de simulation, le système part de l'hypothèse que le fichier de configuration qui lui est donné est bien formé. Le script permet alors de lancer autant de simulations que l'on veut sur un fichier de configuration précis ou sur un ensemble de fichiers contenus dans un répertoire. De plus, nous avons construit un certain nombre de scripts R⁷ qui permettent de construire automatiquement un fichier PDF rassemblant les données de sorties des simulations en construisant automatiquement les graphes associés à ces sorties et en y détaillant certaines informations : la durée de la simulation, la date, la version du simulateur utilisée, les *artefacts* observés, les messages d'alertes ou d'erreurs éventuels, etc. Remarquons que pour que les graphes automatiques soient construits, il faut simplement définir dans le fichier de configuration les sorties que l'on veut voir traiter et le type des données produites afin que le bon type de graphe soit utilisé. Notons également que pour construire le fichier PDF, nous utilisons un fichier latex qui est associé à du code R, ce qui permet de garder ensemble le fichier source latex et les données avec lesquelles sont produits les graphes. Ainsi, le fichier PDF peut toujours être réutilisé et les données associées manipulées pour d'éventuelles analyses.

Enfin, en termes de méthodologie, nous avons cherché à faciliter les tâches de simulation. Pour chaque exécution du modèle d'*agents-modèles* en phase de simulation, un répertoire est créé pour cette simulation. Le nom de ce répertoire est construit à partir du nom du fichier de configuration et de la date (année, mois, jour, heure, minute et seconde). Si ce nom existe, la simulation est arrêtée et un message d'erreur est affiché. Dans ce répertoire sont automatiquement copiés tous les fichiers de configuration utilisés pour la simulation, un lien vers le code du simulateur y est créé, et toutes les données de sorties de la simulation y sont produites. Chaque sortie produite par le simulateur porte le nom que l'agent-observateur a choisi. Par défaut, ce nom est le nom de l'*artefact* duquel sont issues les données. Des sondes⁸ peuvent également produire des données, c'est alors le nom de la sonde qui est utilisé. Enfin, les éventuels fichiers d'analyse automatiques sont également copiés dans ce répertoire. Pour finir, toute simulation a une phase d'initialisation pendant laquelle, en particulier, les *agents-modèles* initialisent les *artefacts* qu'ils produisent. A la fin de l'initialisation, l'état de tous les *artefacts* est copié dans un fichier qui porte le nom de l'*artefact* précédé d'une mention

⁷ R est un langage et environnement donnant accès à des bibliothèques statistiques et graphiques (<http://www.r-project.org/>).

⁸ Une sonde permet des observations locales de toutes sortes d'objets de notre simulateur. Ainsi, certaines informations n'ont pas à être écrites dans des *artefacts* et ensuite observées par des agents-observateurs, mais on peut toutefois vouloir les observer pour différentes raisons de vérification et d'expérimentation.

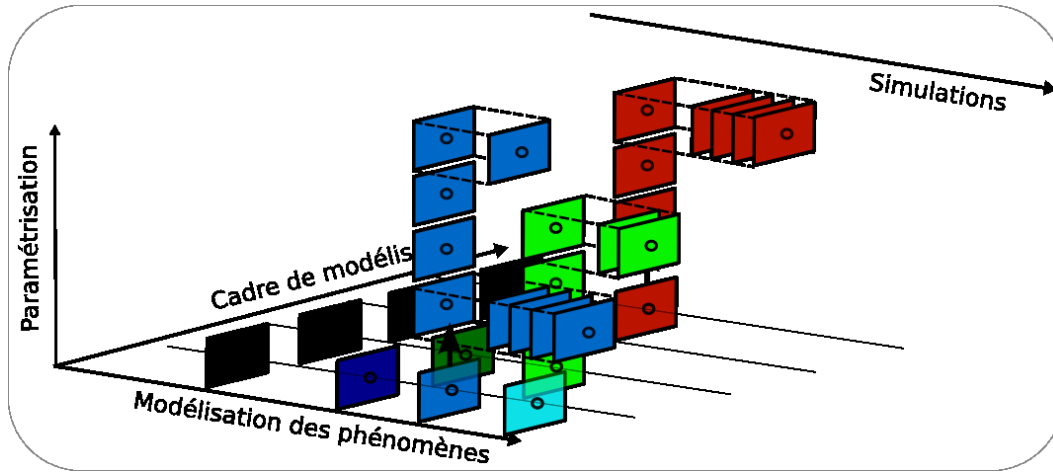


FIG. 6.17: Illustration d'une organisation graphique des actions de modélisation, des différents modèles, des jeux de paramètres et des simulations.

permettant de le reconnaître comme l'état initial de cet *artefact*. Il est bien sûr possible d'empêcher que l'état initial d'un *artefact* soit sauvegardé, mais il faut le spécifier. Enfin, les fichiers contenant les états initiaux des *artefacts* du système sont tous copiés dans le répertoire associé à la simulation.

Avant de conclure, notons que nous avons commencé à entreprendre une rationalisation des différentes actions de modélisation. L'objectif est de pouvoir visualiser chaque expérience de simulation, que l'on souhaite par ailleurs mémoriser. Ces simulations sont organisées d'abord par actions de modélisation qui sont caractérisées chacune par un cadre de modélisation précis. Ensuite, dans chaque action de modélisation se trouve une ou plusieurs «instances» du cadre de modélisation, instances qui peuvent par ailleurs être partielles (ne mettre en œuvre qu'une partie du cadre de modélisation). Ces différentes instances se distinguent l'une de l'autre par leur manière de modéliser les phénomènes. Donc, elles s'organisent en fonction de cela. Ensuite, chaque instance du cadre de modélisation peut être associée à plusieurs jeux de paramètres. On organise ainsi, pour une instance donnée, les différents jeux de paramètres qui ont permis de produire des résultats. Enfin, pour chaque jeu de paramètres, un certain nombre de simulations ont été faites, simulations qui ont produit des résultats auxquels on peut alors accéder. La figure 6.17 présente une illustration d'une telle organisation. L'objectif de cette approche est alors de permettre aux experts et aux modélisateurs de rationaliser leur activité de simulation dans un espace de simulation dont les différents axes correspondent aux différents critères d'organisation. Ce travail est en cours d'élaboration et l'intérêt serait que les différentes simulations puissent être automatiquement agencées dans l'espace. Il serait également important de pouvoir visualiser cet espace et de pouvoir s'y déplacer afin de zoomer sur une zone (sur une problématique) ou

de faire apparaître toutes les simulations selon certains critères. Il reste donc beaucoup de travail à faire sur cette question.

6.10 Conclusion

Nous avons présenté dans ce chapitre différentes actions de modélisation qui prennent cohérence au sein du projet Chaloupe et en jalonnent un certain nombre d'étapes. Notre objectif ici était bien évidemment de présenter les travaux de simulation effectués, mais également de montrer la pertinence de notre proposition en termes d'*agents-modèles*.

Tout d'abord, d'un point de vue thématique, nous avons validé au travers des actions de modélisation (2) et (3) nos modèles IBM et EDP de peuplements. Nous sommes même allé plus loin avec l'action (2) qui nous permet d'aborder la question du rôle des processus de prédation et de co-occurrences spatiales des espèces dans la dynamique des peuplements. L'action (3) nous a permis de dimensionner le modèle IBM et d'identifier un certain nombre de problème dans la mise en œuvre de cette approche sur un système de cette taille. Nous avons ainsi introduit dans notre système un facteur d'échelle qui nous permet de répondre à ces problèmes. Nous cherchons par conséquent à améliorer la précision des résultats pour certaines espèces sans augmenter le nombre d'agents. Ensuite, avec l'action de modélisation (5), nous avons validé les processus structurants de notre modèle IBM économique. Même s'il reste un certain nombre de points à voir, tous les éléments de modélisation, ainsi que tout le paramétrage et les données, sont rassemblés en un modèle qui se comporte de manière cohérente. Enfin, nous avons étudié la question de l'hétérogénéité de modélisation dans l'action de modélisation (4) et en particulier la problématique du couplage de modèles hétérogènes. A l'aide de notre proposition d'*agents-modèles*, nous avons pu mettre en œuvre des modèles de peuplements et des modèles de pêcheries dans un environnement variable, ce qui nous a en particulier permis d'aborder les problématiques d'hétérogénéité d'échelles.

Ces travaux de simulation nous ont ensuite permis de valider notre proposition d'*agents-modèles* en termes de méthodologie, de modèle et d'outils. En termes de méthodologie, nous constatons avant toute chose que la rationalisation de la simulation par l'approche POM nous permet d'identifier, sans *a priori* sur la structure du modèle, les dynamiques remarquables que l'on veut reproduire. De manière plus générale, centrer la modélisation sur les données en identifiant les dynamiques dans ces données nous oblige pour chaque simulation à nous construire une représentation globale du système à la lumière de la question de modélisation et ce avant toute activité de modélisation. C'est à partir de ce questionnement sur les données qu'il est possible de construire un cadre de modélisation rigoureux, c'est-à-dire une représentation en termes de système de phénomènes. Il est alors possible, avec notre proposition de structuration de la simulation au moyen d'actions de modélisation qui sont caractérisées par un cadre unique de modélisation, d'étudier un modèle ou une série de modèles de manière rationnelle, parce que le modèle global, qui a été construit

indépendamment des modèles des phénomènes et qui répond à une question de modélisation, est fixe. Cette dimension méthodologique est de plus mise en œuvre par notre proposition en termes de modèle et d'outil. Ensuite, nous avons vu au cours des différentes actions de modélisation que les mécanismes fondamentaux de notre proposition conceptuelle d'*agents-modèles* fonctionnaient. Nous avons pu valider l'encapsulation des modèles de phénomènes et l'autonomie comportementale des agents, la formalisation des *artefacts* et leur contrôle par les agents, l'ajout d'agents sans impact sur le reste du système dans une démarche de modélisation incrémentale, ainsi que la possibilité d'effectuer des expérimentations unitaires, c'est-à-dire de se focaliser sur la modélisation d'un phénomène en particulier tout en restant dans le cadre de modélisation. Ceci nous a permis par ailleurs de présenter et de valider les *agents-modèles* de types émergent, intensionnel et extensionnel. Enfin, nous avons pu présenter des agents-observateurs et par là-même la capacité des agents en général à manipuler des opérations afin de modifier leur perception ou leurs actions sur l'environnement (c'est-à-dire les *artefacts*).

Soulignons également que l'action de modélisation (2) a fait l'objet d'une communication internationale dans une conférence d'écologie (Bonneaud 07c) et un article pour le journal *Ecological Modelling* sur ce travail est en cours d'écriture. De même, l'action de modélisation (3) a fait l'objet d'un poster au colloque du défi Golfe de Gascogne 2005 (Bonneaud 05a). Des rapports internes et des présentations orales au cours de séminaires internes ont également été régulièrement présentés depuis sur cette action de modélisation. Enfin, le travail de couplage de modèles hétérogènes, portant essentiellement sur notre proposition d'*agents-modèles*, a fait l'objet de deux articles (Bonneaud 07b), dont (Bonneaud 07a) qui portait en partie sur l'action de modélisation (4).

Les différentes actions de modélisation ouvrent cependant sur un certain nombre de questions et de perspective. Nous présentons dans la suite les travaux en cours qui poursuivent des actions de modélisation (3) et (4) et qui doivent nous permettre de produire les derniers résultats thématiques pour le projet Chaloupe, c'est-à-dire une étude à base de scénarios sur les influences respectives des activités de pêche et du réchauffement climatique sur l'écosystème du GDG. Ensuite, nous avons abordé un certain nombre de questions sur l'approche IBM et plus généralement sur les modèles de peuplements et les problématiques computationnelles qu'il est nécessaire de maîtriser pour produire des résultats cohérents (voir chapitre 5). Ainsi, l'action de modélisation (1) nous a permis d'envisager des comparaisons expérimentales des approches EDP et IBM sur des systèmes simples. Cette question est extrêmement intéressante si nous voulons être capable de mieux maîtriser l'outil théorique IBM, ses limites et ses caractéristiques. Enfin, l'action de modélisation (4) est fondamentale parce qu'elle fait partie d'une étude plus large des questions d'hétérogénéités des modèles en termes d'approches de modélisation, de discrétisations et surtout d'échelles de description. Les problématiques liées à ces questions sont stratégiques si nous voulons pouvoir aborder avec plus de maîtrise et de pertinence la construction et l'étude de modèles complexes.

6.11 Perspectives à court terme

6.11.1 Influences sur le peuplement du réchauffement climatique et des activités de pêche

Question et actions de modélisation : Dans la suite des actions de modélisation 3 et 4, nous continuons actuellement l'étude du modèle IBM de peuplements dans le cadre de Chaloupe. L'objectif est d'arriver à distinguer l'impact du réchauffement climatique de celui des activités de pêche sur la dynamique du peuplement écologique du GdG. Nous développons ainsi deux actions de modélisation qui devraient nous permettre de mieux répondre au questionnement posé par Chaloupe dans la mesure du modèle conceptuel IBM de peuplements.

Les deux premières actions de modélisation visent à produire des résultats solides sur le peuplement seul afin de confirmer le travail effectué dans l'action de modélisation 3 et dont les résultats ne sont pas valorisables en l'état. Nous avons en effet rencontré plusieurs problèmes au cours de l'action de modélisation 3, comme la précision des résultats obtenus pour certaines espèces et plus généralement sur la qualité globale des résultats. Il nous faut par conséquent dans un premier temps valider (ou non) l'approche IBM sans solutions multi-échelles pour ce cas d'étude. C'est l'objet de la première action de modélisation, qui vise donc à reproduire les résultats obtenus dans l'action 3, en améliorant les observations et en augmentant le nombre global d'agents⁹. Il nous faut ensuite aborder la question de solutions utilisant des facteurs d'échelles pour mieux reproduire les dynamiques de tous les peuplements. Cette dernière action de modélisation n'est pas triviale, car elle s'inscrit dans une étude plus globale de l'approche IBM que certains travaux du chapitre 5 sont venus préparer. Il est en effet nécessaire ici d'établir des connaissances plus approfondies sur les biais entraînés par des solutions permettant l'hétérogénéité des discrétisations dans les modèles IBM si nous voulons complètement maîtriser l'implémentation du modèle conceptuel IBM, ce que nous avons par ailleurs commencé à faire. De plus, nous abordons ici des problématiques d'hétérogénéités d'échelles qui perturbent l'étude d'un modèle IBM déjà difficile à mettre en œuvre rigoureusement sur un cas d'étude qui requiert des solutions computationnelles originales et un travail de recherche des valeurs des paramètres très important.

Une fois la question de la validation et de la valorisation du modèle IBM de peuplement traitée, nous voulons pouvoir étudier des scénarios nous permettant de mieux distinguer l'impact des activités de pêche de celui du réchauffement climatique sur le peuplement écologique du GdG. Nous avons déjà tous les éléments nous permettant d'effectuer rapidement cette action de modélisation. En effet, nous avons validé les mécanismes de couplage et

⁹ Notons que le simulateur ayant été modifié et amélioré depuis la production des résultats obtenus, nous pouvons effectivement penser que nous avons les moyens computationnels d'augmenter le nombre d'agents et ainsi mécaniquement la qualité des résultats.

expérimenté un certain nombre de modèles mixtes avec un phénomène de pêcheries simplifiées, un phénomène de réchauffement climatique et un phénomène décrivant des peuplements. L'action de modélisation (4) en est un exemple. De plus, la question ici se porte sur les dynamiques du peuplement du GdG, donc nous envisageons de forcer les activités de pêche par des séries historiques de captures sur la période étudiée. Les scénarios que nous envisageons sont les suivants : 1° Etude de l'impact du réchauffement climatique – expérimentations sans pêche et avec plusieurs scénarios de réchauffement (réchauffement observé, réchauffement plus fort que celui observé, etc.) ; 2° Etude de l'impact des activités de pêche – expérimentations sans réchauffement climatique et avec plusieurs scénarios de pêche qu'il reste à définir ; 3° Etude du système – expérimentations avec réchauffement et pêche suivant des scénarios qu'il reste à définir, mais selon toute vraisemblance, seul le scénario réaliste sera mis en œuvre et les résultats comparés aux séries historiques.

6.11.2 Couplage avec le *Virtual Laboratory Environment* (VLE) : système biodiversité–pêcheries

Dans le cadre de Chaloupe, un autre modèle économique a été élaboré et est actuellement mis en œuvre sur le *Virtual Laboratory Environment* (VLE) (Ramat 03) (<http://vle.univ-littoral.fr/>). Ce modèle, basé sur une approche à base d'EDP, est couplé dans VLE à un modèle forcé de peuplements. La question de modélisation que nous avons donc commencé à mettre en œuvre vise à coupler VLE et notre plate-forme de simulation. L'objectif est d'intégrer le modèle économique mis en œuvre par VLE dans notre simulateur au moyen d'un *agent-modèle*. D'un point de vue technique, le travail a été fait et nous sommes capables de piloter VLE et donc le modèle économique au travers d'un *agent-modèle*. D'un point de vue thématique, les expérimentations doivent être menées et l'établissement de scénarios de simulation reste à faire. Enfin, cette action de modélisation nous permet d'aborder à nouveau des questions d'hétérogénéité d'échelles.

Chapitre 7

Conclusion générale

7.1 Bilan

Modéliser, c'est se construire une représentation, la manipuler et apprendre sur le réel par son intermédiaire. Simuler, c'est mettre cette représentation en mouvement, se construire une représentation d'un mouvement, la manipuler et apprendre quelque chose du réel par son intermédiaire. Pourtant, quand la réalité que l'on cherche à se représenter est trop riche, trop complexe, toutes les activités de modélisation et de simulation sont moins accessibles, plus riches, plus complexes elles aussi. Les thématiciens ne peuvent plus alors mettre en œuvre seuls leur expertise, mais ont besoin que des informaticiens prennent en charge les activités de modélisation et d'étude de leurs représentations. Il n'est cependant pas de la responsabilité de l'informaticien d'avoir un regard sur la réalité à modéliser ou sur la qualité des modèles conceptuels, et dans notre cas sur la chose écologique et économique. Il nous a donc fallu, dans ce travail, certes acquérir quelques connaissances dans ces domaines, mais avant tout identifier les éléments de modélisation dont nous sommes responsables et proposer un outil pour aborder la modélisation de systèmes complexes.

Ce travail a pris forme dans le cadre du projet ANR Chaloupe pour lequel nous avons construit et étudié des modèles écosystémiques des pêches. L'étude de systèmes peuplements-pêcheries nous a confronté à un certain nombre de problématiques en termes de système complexe, mais également de modèle complexe et plus largement de processus de modélisation et de simulation complexes. Il nous a fallu étudier la modélisation IBM et construire des modèles de peuplements, mais également mettre en œuvre des modèles EDP, des modèles pré-calculés (forçages) et surtout aborder la question difficile du couplage de modèles. A ce travail de modélisation se sont associées les activités d'exploration d'espace de paramètres, de simulation et d'analyse, activités qui devaient *in fine* entrer en cohérence pour répondre

à la question posée. C'est en nous confrontant à Chaloupe, que nous avons pu «apprendre en faisant» et ainsi proposer des solutions aux problématiques posées, en particulier, avoir un outil qui permette de construire des modélisations hétérogènes, mais également de rationaliser le processus de simulation.

Les systèmes que nous avons dû modéliser se caractérisent par leur hétérogénéité et leurs nombreuses interactions. Ces propriétés structurent leur dynamique globale, constituent des objets d'étude en soi et doivent, par conséquent, être prises en compte dans les modèles de ces systèmes. Pour répondre à ces problèmes de complexité et de couplage de modèles hétérogènes, le modèle complexe est conçu comme une société d'*agents-modèles* en interaction et la simulation comme l'activité de cette société d'agents. La solution permet alors de distribuer le contrôle dans le système *via* les agents. La méthodologie orientée *pattern*, qui centre la modélisation sur les données, permet de rationaliser cette société d'*agents-modèles* autour des patterns, de cette manière, la dynamique du modèle global, qui exhibe les patterns recherchés, émerge des comportements et des interactions des agents du système.

Nous avons proposé les notions d'*artefacts* dans l'environnement de modélisation et d'*agents-modèles* qui constituent le cadre de modélisation. Les *artefacts* sont le support des données partagées, le support d'interaction des agents et le support de l'observation et de l'analyse du système. Les *agents-modèles* permettent de mettre en œuvre les modèles de phénomènes qui doivent expliquer les patterns. Les *agents-modèles* sont ainsi responsables du lien entre le modèle et l'environnement, de son «bon» comportement et servent de médiateurs entre les contraintes du modèle global et celles de leur modèle interne. La solution de société d'*agents-modèles* tient en particulier au fait de pouvoir coupler des modèles hétérogènes dans le respect de la cohérence du modèle global. L'hétérogénéité de modélisation est assurée par un couplage des modèles *via* les données (les *artefacts*) et une synchronisation des agents par rapport à l'environnement. Ceci permet aux agents d'être autonomes et fortement détachés du modèle global. Pour réaliser l'hétérogénéité d'échelles, nous avons proposé que les agents puissent s'adapter à leur environnement en transformant leur perception et leur action au moyen d'*opérations*. L'avantage est de dissocier les problématiques du couplage des modèles eux-mêmes.

Nous avons identifié deux autres types d'agents qui se sont révélés pertinents au cours des processus de modélisation et de simulation : les agents-observateurs et les agents-contrôleurs. Ceux-ci permettent respectivement de construire des observables, en particulier des patterns, et de distribuer un contrôle sur les comportements de plusieurs *agents-modèles* et sur la dynamique du système. Le cadre de modélisation est ainsi structuré et formalisé afin de pouvoir envisager diverses actions de modélisation qui peuvent être incrémentales.

Pour compléter notre démarche de rationalisation de la modélisation et gagner en maîtrise sur ce processus, nous avons étudié, à l'aide de notre simulateur, les propriétés qu'un outil de simulation doit pouvoir garantir sur le modèle et les biais computationnels dans les modèles IBM de peuplements. En effet, la traduction d'un modèle conceptuel en modèle computationnel introduit un certain nombre d'hypothèses qu'il est alors nécessaire

d'expliciter. De ce fait, il est essentiel de prêter beaucoup d'attention à l'écriture des algorithmes, point pour lequel la compétence d'un informaticien est indispensable. Nous avons également identifié les sources de biais computationnels, liés à la mise en œuvre d'agents et aux discrétisations de l'environnement. La sensibilité des modèles à ces biais est variable et les processus modélisés peuvent parfois masquer ces biais. Cependant, ils peuvent influencer de manière critique la dynamique du système et, quand le modèle est complexe, la distinction de l'influence des différents processus modélisés n'est pas toujours possible. Cela nous a amené à étudier des systèmes minimaux issus d'un même modèle conceptuel. Les lois sur l'impact de ces biais ne sont cependant pas accessibles et il est, selon nous, indispensable qu'un outil de modélisation permette d'explorer de manière systématique les différents biais. Cette étude nous a donc permis d'identifier ce qu'un outil spécialisé de modélisation doit garantir comme hypothèses sur le modèle et fournir comme moyens d'analyse. C'est ce que nous avons mis en œuvre dans notre simulateur.

Enfin, nous avons développé un certain nombre d'actions de modélisation dans le cadre du projet Chaloupe. La société d'*agents-modèles* a permis d'aborder les problématiques de manière rationnelle et incrémentale. Nous avons ainsi pu explorer différentes hypothèses dans divers cadres de modélisation. Lors des échanges avec les thématiciens, que ce soit la construction incrémentale de modèle complexe, l'ajout d'un agent sans impact sur le reste du système ou l'étude de différents modèles pour un même *agent-modèle*, la société d'*agents-modèles* permet une souplesse de modélisation dans la maîtrise du modèle. En particulier, nous avons pu mettre en œuvre des modèles fortement hétérogènes du point de vue de la description (modélisation et échelles) mais également de l'observation, ce qui nous a permis d'étudier des modèles écosystémiques des pêches influencés par le réchauffement climatique. Nous avons pu par conséquent établir des éléments de réponse au questionnement de Chaloupe et envisager des actions de modélisation plus complexes encore.

7.2 Perspectives

L'outil de modélisation et de simulation doit permettre de faciliter et de guider les thématiciens et modélisateurs dans la construction et la manipulation du modèle. La première perspective de ce travail est de développer cet outil en tirant parti des potentialités de la réalité virtuelle. Concernant la construction de la société d'*agents-modèles*, l'utilisation de métaphores visuelles pour faciliter l'accès aux informations pertinentes et assurer l'observabilité de l'activité des agents permet d'immerger les thématiciens dans l'environnement de modélisation. En effet, déléguer une partie du contrôle aux agents nécessite de renforcer l'observabilité du système et un environnement immersif permettrait d'augmenter les capacités d'interaction et d'expérimentation du modèle. Concernant la rationalisation du processus de simulation, l'utilisation d'outils de navigation dans les espaces informationnels composés des différents résultats de simulation, des jeux de paramètres utilisés et des actions de modélisation pourrait largement faciliter la mise en cohérence des différentes activités de

simulation. Nous avons présenté dans la section 6.9 des éléments de cette perspective. Le développement d'un tel environnement renvoie à des problématiques scientifiques à la fois en informatique graphique, en ingénierie des connaissances et en ergonomie cognitive.

Nous avons entrepris une étude systématique des biais computationnels dans les modèles complexes de peuplements, en nous attachant notamment à décrire l'impact de la désynchronisation des comportements au sein d'un agent, ou encore la sensibilité de notre modèle, tant au nombre d'agents qu'aux discrétisations -homogènes comme hétérogènes- temporelle et spatiale. A l'instar de la preuve formelle présentée dans l'annexe A pour la désynchronisation des comportements de mort et reproduction, nous souhaitons à l'avenir développer une analyse mathématique systématique des biais computationnels observés. Ce travail apportera une contribution importante au domaine émergent des modèles mathématiques pour les systèmes multi-agents.

La société de modèles permet de mettre les agents au service de la modélisation au travers de la co-construction et de la co-manipulation du modèle, ceci ouvre de nombreuses pistes de recherche. En effet, nous avons envisagé jusqu'ici l'apport des agents à la construction du modèle, mais il y a d'autres activités dans la simulation qui pourraient profiter de la solution d'*agents-modèles*. En particulier, le problème du choix des paramètres de ces modèles qui sont souvent sur-paramétrés est critique. L'espace des paramètres est trop grand pour que les techniques de recherche systématique trouvent un bon jeu de paramètres en un temps raisonnable. Il est par conséquent nécessaire de proposer des heuristiques d'identification des paramètres. Ces heuristiques peuvent être complétées par des systèmes de contraintes sur les *agents-modèles*, donc des contraintes locales, et des contraintes sur le modèle global. L'enjeu est alors de permettre aux agents de collaborer pour la résolution distribuée de ces contraintes. On peut ainsi également envisager que les agents puissent construire des plans de simulation automatiquement, explorer des plans d'expérience ou encore construire des analyses automatiques de résultats de simulation.

Enfin, la co-construction du modèle pose la question de la modélisation. Nous avons proposé une rationalisation de la modélisation qui passe par la formalisation de l'environnement de modélisation et des interactions des agents. L'enjeu est alors de réfléchir à la possibilité pour les agents de créer de nouveaux *artefacts* ou de proposer leur service par rapport à une question de modélisation. Il s'agit cependant de savoir comment les agents peuvent faciliter la modélisation en proposant eux-mêmes des éléments de modélisation. C'est une problématique d'Intelligence Artificielle passionnante qui requiert de formaliser l'activité même de modélisation et ce qui fait l'expertise des thématiciens.

Références bibliographiques

- [Amblard 01] Frédéric Amblard, Nils Ferrand & David R.C. Hill. *How a conceptual framework can help to design models following decreasing abstraction*. In Proceedings of the 13th SCS-European Simulation Symposium, pages 843–847, Marseille (France), Octobre 2001.
- [Amblard 03] Frédéric Amblard. *Comprendre le fonctionnement de simulations sociales individus-centrées : Application à des modèles de dynamiques d'opinions*. PhD thesis, Université Blaise Pascal - Clermont II - Ecole Doctorale Sciences pour l'Ingénieur de Clermont-Ferrand, 15 décembre 2003.
- [Andrewartha 54] H. Andrewartha & L. Birch. The distribution and abundance of animals. University of Chicago Press, 1954.
- [Anneville 98] O. Anneville, P. Cury, C. Le Page & J.P. Treuil. *Modelling fish spatial dynamics and local density-dependence relationships : detection of patterns at a global scale*. Aquatic Living Resources, vol. 11, pages 306–314, 1998.
- [Antona 98] Martine Antona, François Bousquet, Christophe Le Page, J. Weber, Alain Karsenty & P. Guizol. *Economic Theory of Renewable Resource Management : A Multi-Agent System Approach*. In MABS, pages 61–78, 1998.
- [Arthur 95] W. Brian Arthur. *Complexity in Economic and Financial Markets*. Complexity, vol. 1, no. 1, pages 20–25, April 1995.

- [Ashby 04] W. Ross Ashby. *Principles of the self-organizing system*. E :CO Special Double Issue, vol. 6, no. 1-2, pages 102–126, 2004.
- [Axtell 99] Robert Axtell. *Why Agents ? On the Varied Motivations for Agent Computing in the Social Sciences*. In *Proceedings of Agent Simulation : Applications, Models, and Tools*, 1999.
- [Banks 70] Edwin Roger Banks. *Universality in cellular automata*. In *11th Annual Symposium on Switching and Automata Theory (swat 1970)*, 1970.
- [Beer 66] Stafford Beer. *Decision and control : The meaning of operational research and management cybernetics*. Wiley, 1966.
- [Benkirane 02] Réda Benkirane. *Le complexe, ce qui est tissé ensemble : entretien avec Edgar Morin*. In *La complexité, vertiges et promesses*, pages 19–34. Le Pommier, 2002.
- [Berger 99] Uta Berger, Gerd Wagner & Wilfried F. Wolff. *Virtual biologists observe virtual grasshoppers : an assessment of different mobility parameters for the analysis of movement patterns*. *Ecological Modelling*, vol. 115, pages 119–127, 1999.
- [Blanchard 01] Fabian Blanchard & Jean Boucher. *Présence d’espèces tropicales de poissons dans l’Atlantique Nord et réchauffement climatique*. Lettre n°12 du Programme International Géosphère Biosphère-Programme Mondial de Recherches sur le Climat (PIGB-PMRC), 2001.
- [Blanchard 05b] Fabian Blanchard & Frédéric Vandermeirsch. *Warming and exponential abundance increase of the subtropical fish *Capros aper* in the Bay of Biscay (1973-2002)*. *C. R. Biologies*, vol. 328, pages 505–509, 2005.
- [Bonneaud 05a] Stephane Bonneaud, Fabian Blanchard & Pierre Chevaillier. *virtuOcean : modèle individu-centré d’un peuplement de poissons dans un environnement variable*. poster in conference ”Défi Golfe de Gascogne 2005”, 2005. poster, <http://www.ifremer.fr/gascogne/colloque2005>.
- [Bonneaud 07a] Stephane Bonneaud, Pascal Redou & Pierre Chevaillier. *Oriented pattern agent-based multi-modeling of exploited ecosystems*. In *EuroSim07*, 2007.

- [Bonneaud 07b] Stephane Bonneaud, Pascal Redou, Damien Thebault & Pierre Chevaillier. *Multi-modélisation agent orientée patterns : Application aux écosystèmes exploités*. In JFSMA07, 2007.
- [Bonneaud 07c] Stéphane Bonneaud, Fabian Blanchard, Pierre Chevaillier, Damien Thébault & Pascal Redou. *A model of fish population dynamics based on spatially explicit trophic relationships*. International communication, abstract in proceedings of the European Conference on Ecological Modelling ECEM'07, November 27-30 2007.
- [Bousquet 98a] F. Bousquet, I. Bakam, H. Proton & C. Le Page. *Cormas : common-pool resources and multi-agent Systems*. Lecture Notes in Artificial Intelligence, vol. 1416, pages 826–838, 1998.
- [Bousquet 99] F. Bousquet, O. Barreteau, C. Le Page, C. Mullon & J. Weber. *An environmental modelling approach : the use of multi-agent simulations*. In Advances in environmental and ecological modelling, 1999. Blasco F. (ed), Elsevier, Paris, p. 113-122.
- [Bousquet 01] Bousquet. *Modélisation d'accompagnement : Simulations multi-agents et gestion des ressources naturelles et renouvelables*. HDR - Mémoire pour l'obtention de l'Habilitation à Diriger des Recherches de l'Université de Lyon 1, 22 mars 2001.
- [Box 05] George E. P. Box, J. Stuart Hunter & William Gordon Hunter. *Statistics for experimenters : Design, innovation, and discovery*. Wiley-Interscience, 2005.
- [Breckling 02] B. Breckling. *Individual-based modelling : potentials and limitations*. Scientific World Journal, vol. 2, pages 1044–1062, 2002.
- [Breckling 05] Broder Breckling, Felix Muller, Hauke Reuter, Franz Holker & Otto Franzle. *Emergent properties in individual-based ecological models – introducing case studies in an ecosystem research context*. Ecological Modelling, vol. 186, pages 376–388, 2005.
- [Brooks 91] Rodney A. Brooks. *Intelligence without representation*. Artificial Intelligence, vol. 47, pages 139–159, 1991.

- [Brown 91] James Robert Brown. *The laboratory of the mind : Thought experiments in the natural sciences*. Routledge, 1991.
- [Chicoisne 02] Guillaume Chicoisne. *Dialogue entre agents naturels et agents artificiels : une application aux communautés virtuelles*. PhD thesis, Institut national polytechnique de Grenoble, 2002.
- [Christensen 04] Villy Christensen & Carl J. Walters. *Ecopath with Ecosim : methods, capabilities and limitations*. Ecological Modelling, vol. 172, pages 109–139, 2004.
- [Conte 92] R. Conte & C. Castelfranchi. *Mind is not enough : pre-cognitive bases of social interaction*. In Proceedings of 1992 Symposium on Simulating Societies, pages 93–110, 1992.
- [Costanza 93] R. Costanza, L. Wainger, C. Folke & K. Maler. *Modelling complex ecological economic systems : toward an evolutionary, dynamic understanding of humans and nature*. BioScience, vol. 43, pages 545–555, 1993.
- [Dagnelie 03] Pierre Dagnelie. *Principes d’expérimentation : Planification des expériences et analyse de leurs résultats*. Presses Agronomiques Gembloux, 2003.
- [DeAngelis 94] D.L. DeAngelis, K.A. Rose & M.A. Huston. *Individual-oriented approaches to modeling ecological populations and communities*. In Levin S.A., éditeur, *Frontiers in mathematical biology*, pages 390–410. Springer Verlag, 1994.
- [Deffuant 03] Guillaume Deffuant, Frederic Amblard, Raphael Duboz & Eric Ramat. *Une demarche experimentale pour la simulation individus-centrée*. In ENST Editions, éditeur, 10eme Journees de Rochebrune : Rencontres interdisciplinaires sur les systemes complexes naturels et artificiels, Epistemologie de la Simulation, pages 45–64, Rochebrune, Megeve, France, du 27 au 31 janvier 2003.
- [Demazeau 95] Y. Demazeau. *From interactions to collective behaviour in agent-based systems*. In Proceedings European Conference on Cognitive Science, volume 95, pages 117–132, 1995.
- [Demazeau 01] Yves Demazeau. *Voyelles*. Rapport d’Habilitation à Diriger des Recherches, Avril 2001. Institut National Polytechnique de Grenoble, Laboratoire Leibniz, Grenoble.

- [Desmeulles 06a] G. Desmeulles, G. Querrec, P. Redou, S. Kerdélo, L. Misery, V. Rodin & J. Tisseau. *The virtual reality applied to biology understanding : The in virtuo experimentation*. Expert Systems with Applications, vol. 30, pages 82–92, 2006.
- [Desmeulles 06b] Gireg Desmeulles. *Réification des interactions pour l'expérience in virtuo de systèmes biologiques multi-modèles*. PhD thesis, Université de Bretagne Occidentale, 2006.
- [Drogoul 92] A. Drogoul & J. Ferber. *Multi-Agent Simulation as a Tool for Modeling Societies : Application to Social Differentiation in Ant Colonies*. In Decentralized A.I., North-Holland, 1992. MAAMAW'92, Elsevier.
- [Drogoul 03b] Alexis Drogoul, Diane Vanbergue & Thomas Meurisse. *Multi-agent Based Simulation : Where Are the Agents ?* In Multi-Agent-Based Simulation II : Third International Workshop, MABS 2002, Bologna, Italy, July 15-16, 2002. Revised Papers, pages 43–49, 2003.
- [Duboz 01] R. Duboz, E. Ramat & P. Preux. *Towards a coupling of continuous and discrete formalisms in ecological modelling - influences of the choice of algorithms and results*. In N. Giambiasi & C. Frydman, editeurs, Proc. 13th European Simulation Symposium, pages 481–487, October 2001.
- [Duboz 03] Raphael Duboz, Frederic Amblard, Eric Ramat, Guillaume Deffuant & Philippe Preux. *Individual-based model to enrich an aggregate model*. In Model to model (M2M) a workshop to explore the relations between Multi-Agent Based Simulation models, Marseille, France, 31st March and 1st April 2003.
- [Duboz 04b] Raphael Duboz. *Intégration de modèles hétérogènes pour la modélisation et la simulation de systèmes complexes : Application à la modélisation multi-échelles en écologie marine*. PhD thesis, Ecole doctorale de l'Université du Littoral - Côte d'Opale, 30 mars 2004.
- [Edwards 04] Margaret Edwards. *Proposition d'outils pour évaluer la pertinence de descendre au niveau individu-centré pour des modèles d'utilisation de la ressource couplés à des modèles de ressource*. PhD thesis, OASIS, LIP6, Paris VI, 2004.
- [Endy 01] Drew Endy & Roger Brent. *Modelling cellular behaviour*. Nature, vol. 409, pages 391–395, 2001.

- [Epstein 96] Joshua M. Epstein & Robert L. Axtell. Growing artificial societies, social science from the bottom up. MIT Press, 1996.
- [Fahse 98] Lorenz Fahse, Christian Wissel & Volker Grimm. *Reconciling classical and individual-based approaches in theoretical population ecology : a protocol for extracting population parameters from individual-based models*. the american naturalist, vol. 152, no. 6, pages 838–852, 1998.
- [Ferber 99] Jacques Ferber. Multi-agent system : An introduction to distributed artificial intelligence. Harlow : Addison Wesley Longman, 1999.
- [Fishwick 94] Paul A. Fishwick. *Computer simulation : Growth through extension*. In Society for Computer Simulation, pages 3–20, 1994.
- [Forrester 69] Jay Wright Forrester. Principles of systems : Text and workbook chapters 1 through 10. Wright-Allen Press, 1969.
- [Frigg 06] R. Frigg & S. Hartmann. *Models in Science*. Metaphysics Research Lab, 2006.
- [Fromentin 07] Jean-Marc Fromentin, Benjamin Planque & Olivier Thébaud. *L’approche écosystémique des pêches : quelles priorités pour la recherche ?* Rapport technique, IFREMER, Institut Français de Recherche pour l’Exploitation de la MER, 2007.
- [Fulton 03] Elizabeth A. Fulton, Anthony D.M. Smith & Craig R. Johnson. *Effect of complexity on marine ecosystem models*. Marine Ecology Progress Series, vol. 253, pages 1–16, 2003.
- [Gardner 70] Martin Gardner. *The fantastic combinations of John Conway’s new solitaire game "life"*. Scientific American, vol. 223, pages 120–123, 1970.
- [Georgeff 87] Michael P. Georgeff & Amy L. Lansky. *Reactive reasoning and planning*. In AAAI-87 Proceedings, 1987.
- [Gershenson 03] Carlos Gershenson & Francis Heylighen. *When Can we Call a System Self-organizing ?* In Advances in Artificial Life, 7th European Conference, ECAL 2003 LNAI 2801, pages 606–614. Springer-Verlag, 2003.

- [Gilbert 99] Nigel Gilbert & Klaus G. Troitzsch. *Simulation for the social scientist*. Taylor & Francis, Inc., 2nd edition, 1999.
- [Ginot 98] V. Ginot & C. Le Page. *Mobidyc, a generic multi-agents simulator for modeling communities dynamics*. Congrès IEA-98-AIE, Lecture Notes in Artificial Intelligence, vol. 1416, pages 805–814, 1998.
- [Ginot 02] Vincent Ginot, Christophe Le Page & Sami Souissi. *A multi-agents architecture to enhance end-user individual-based modelling*. Ecological Modelling, vol. 157, pages 23–41, 2002.
- [Ginot 06b] Vincent Ginot, Sabrina Gaba, Rémy Beaudouin, Franck Aries & Hervé Monod. *Combined use of local and ANOVA-based global sensitivity analyses for the investigation of a stochastic dynamic model : Application to the case study of an individual-based model of a fish population*. Ecological Modelling, vol. 193, pages 479–491, 2006.
- [Ginot 06c] Vincent Ginot & Hervé Monod. *Explorer les modèles par simulation : application aux analyses de sensibilité*. In *Modélisation et simulation multi-agents : applications pour les Sciences de l’Homme et de la Société.*, chapitre 3, pages 75–101. Lavoisier, Hermes Science, 2006.
- [Green 05a] David G. Green & Suzanne Sadedin. *Interactions matter - complexity in landscapes and ecosystems*. Ecological Complexity, vol. 2, pages 117–130, 2005.
- [Grimm 96] Volker Grimm, Karin Frank, Florian Jeltsch, Roland Brandl, Janusz Uchmanski & Christian Wissel. *Pattern-oriented modelling in population ecology*. The Science of the Total Environment, vol. 183, pages 151–166, 1996.
- [Grimm 99a] Volker Grimm. *Ten years of individual-based modelling in ecology : what have we learned and what could we learn in the future ?* Ecological Modelling, vol. 115, pages 129–148, 1999.
- [Grimm 99b] Volker Grimm, Tomasz Wyszomirski, David Aikman & Janusz Uchmanski. *Individual-based modelling and ecological theory : synthesis of a workshop*. Ecological Modelling, vol. 115, pages 275–282, 1999.

- [Grimm 02] Volker Grimm & Janusz Uchmanski. *Individual variability and population regulation : a model of the significance of within-generation density dependence*. Oecologia, vol. 131, pages 196–202, 2002.
- [Grimm 05a] Volker Grimm & Steven F. Railsback. *Individual-based modeling and ecology*. Princeton University Press, 2005.
- [Grimm 05b] Volker Grimm, Eloy Revilla, Uta Berger, Florian Jeltsch, Wolf M. Mooij, Steven F. Railsback, Hans-Hermann Thulke, Jacob Weiner, Thorsten Wiegand & Donald L. DeAngelis. *Pattern-oriented modeling of agent-based complex systems : lessons from ecology*. Science, vol. 310, pages 987–991, 2005.
- [Gross 03] Louis J. Gross. Current and future challenges in mathematical and computational biology. Oral presentation Hudson River 2003, 2003.
- [Harrouet 06] Fabrice Harrouet, Eric Cazeaux & Thomas Jourdan. *AReVi*. In Philippe Fuchs, Guillaume Moreau & Jacques Tisseau, editeurs, *Le traité de la Réalité Virtuelle*, volume 3, pages 369–392. Les Presses de l'Ecole des Mines, third edition, 2006.
- [Hiebeler 94] David Hiebeler. *The Swarm Simulation System and Individual-Based Modeling*. proceedings of Decision Support 2001's conference (Toronto), 1994.
- [Holling 73] C.S. Holling. *Resilience and Stability of Ecological Systems*. Annual Review of Ecology and Systematics, vol. 4, pages 1–23, 1973.
- [Huse 02] Geir Huse, Jarl Giske & Anne Gro Veia Salvanes. *Fish and fisheries handbook*, volume 2. J.D., Blackwell Science, Oxford., 2002. chapter 11 : Individual-based modelling.
- [Huston 88] Michael Huston, Donald DeAngelis & Wilfred Post. *New Computer Models Unify Ecological Theory*. BioScience, vol. 38, no. 10, pages 682–691, 1988.
- [Kari 92] Jarko Kari. *The nilpotency problem of one-dimensional cellular automata*. SIAM Journal on Computing, vol. 21, pages 571–586, 1992.
- [Kirby 04] David S. Kirby, Gwenhael Allain, Patrick Lehodey & Adam Langley. *Individual/Agent-based Modelling of Fishes, Fishers, and Turtles*. Rapport technique, Oceanic Fisheries

- Programme, Secretariat of the Pacific Community, Noumea, New Caledonia, 2004.
- [Kleijnen 95] Jack P.C. Kleijnen. *Verification and validation of simulation models*. European Journal of Operational Research, vol. 82, pages 145–162, 1995.
- [Kreft 98] Jan-Ulrich Kreft, Ginger Booth & Julian W. T. Wimpenny. *BacSim, a simulator for individual-based modelling of bacterial colony growth*. Microbiology, vol. 144, pages 3275–3287, 1998.
- [Lamport 78] Leslie Lamport. *Time, clocks, and the ordering of events in a distributed system*. Communications of the ACM, vol. 21, pages 558–565, 1978.
- [Laval 01] Catherine Laval & Michel Lochot. *Management par la valeur et développement durable : un apport pour les collectivités locales*. Revue française de gestion industrielle, vol. 2, pages 51–60, 2001.
- [Li 75] T.Y. Li & J.A. Yorke. *Period Three Implies Chaos*. Am. Math Monthly, vol. 82, pages 985–992, 1975.
- [Lindenschmidt 06] Karl-Erich Lindenschmidt. *The effect of complexity on parameter sensitivity and model uncertainty in river water quality modelling*. Ecological Modelling, vol. 190, pages 72–86, 10 January 2006.
- [Lorek 99] Helmut Lorek & Michael Sonnenschein. *Modelling and simulation software to support individual-based ecological modelling*. Ecological Modelling, vol. 115, pages 199–216, 15 February 1999.
- [McDonald 05] A.D. McDonald, L.R. Little, R. Gray, E. Fulton, K.J. Sainsbury & V.D. Lyne. *Multiple-use management strategy evaluation for coastal marine ecosystems using InVitro*. In Proceedings of MODSIM05, 2005.
- [Merz 91] Joachim Merz. *Microsimulation – A survey of principles, developments and applications*. International Journal of Forecasting, vol. 7, pages 77–104, 1991.
- [Michel 00] Fabien Michel. *Une approche méthodologique pour la conception et l’analyse de simulateur multi-agents*. In RJCIA-2000, 2000.

- [Michel 04] Fabien Michel. *Formalisme, outils et éléments méthodologiques pour la modélisation et la simulation multi-agents*. PhD thesis, Université Montpellier II, 2004.
- [Minar 96] Nelson Minar, Roger Burkhart, Chris Langton & Manor Askenazi. *The Swarm simulation system : a toolkit for building multi-agent simulations*. Santa Fe Institute Working Papers, 1996. available at <http://www.santafe.edu/projects/swarm/swarmdoc/swarmdoc.html>.
- [Molesini 06] A. Molesini, A. Omicini, E. Denti & A. Ricci. *SODA : A Roadmap to Artefacts*. Lecture Notes in Computer Science, vol. 3963, pages 49–62, 2006.
- [Morgan 99] Mary S. Morgan & Margaret Morrison. *Models as mediators*. Cambridge University Press, 1999.
- [Morin 77] Edgar Morin. *La méthode (1) la nature de la nature*. Seuil, 1977.
- [Muller 02] Jean-Pierre Muller. *Des systèmes autonomes aux systèmes multi-agents : Interaction, émergence et systèmes complexes*. Rapport présenté pour l’obtention de l’Habilitation à Diriger les Recherches en Informatique, 8 novembre 2002. Université Montpellier II.
- [Muller 04] J.P. Muller. *The mimosa generic modelling and simulation platform : The case of multi-agent systems*. In *Proceedings on the 5th Workshop on Agent-Based Simulation*, 2004.
- [Muller 05] J.P. Muller, C. Ratzé, F. Gillet & K. Stoffel. *Modeling and simulating hierarchies using an agent-based approach*. In *Proceedings of the MODSIM 2005 International Congress on Modelling and Simulation*, 12-15 December 2005.
- [Murray 01] A.G. Murray. *The use of simple models in the design and calibration of a dynamic 2D model of a semi-enclosed Australian bay*. *Ecological Modelling*, vol. 136, no. 1, pages 15–30, January 2001.
- [Nes 05] E.H.V. Nes & M. Scheffer. *A strategy to improve the contribution of complex simulation models to ecological theory*. *Ecological Modelling*, vol. 185, no. 2-4, pages 153–164, July 2005.
- [Neumann 66] John Von Neumann & Arthur Walter Burks. *Theory of self-reproducing automata*. University of Illinois Press, 1966.

- [Nielsen 00] S.N. Nielsen & F. Muller. *Emergent Properties of Ecosystems*. In Felix Muller Sven Erik Jørgensen, editeur, *Handbook of Ecosystem Theories and Management : Theories and Management*, chapitre II.2.2, pages 195–216. CRC Press, 2000.
- [Omicini 06] Andrea Omicini, Alessandro Ricci & Mirko Viroli. *Agens Faber : Toward a Theory of Artefacts for MAS*. *Electronic Notes in Theoretical Computer Science*, vol. 150, pages 21–36, 2006.
- [Orcutt 07] Guy H. Orcutt. *A new type of socio-economic system*. *International Journal of Microsimulation*, vol. 1, pages 3–9, 2007.
- [Parrott 06] Lael Parrott & Robert Kok. *Use of an object-based model to represent complex features of ecosystems*. In Ali A. Minai & Yaneer Bar-Yam, éditeurs, *Unifying Themes in Complex Systems : Proceedings from the Third International Conference on Complex Systems*, volume IIIB of *New Research*, pages 169–179. Springer Berlin Heidelberg, 2006.
- [Parunak 98] Parunak, Savit & Riolo. *Agent-Based Modeling vs. Equation-Based Modeling : A case study and users' guide*. In *Workshop on Modeling Agent Based Systems MASBS*, Paris, 1998.
- [Pauly 00] Daniel Pauly, Villy Christensen & Carl Walters. *Ecopath, Ecosim, and Ecospace as tools for evaluating ecosystem impact of fisheries*. *ICES Journal of Marine Science*, vol. 57, pages 697–706, 2000.
- [Pereira 04a] Antonio Pereira, Pedro Duarte & Luis Paulo Reis. *Agent-based ecological model calibration - on the edge of a new approach*. In *Proceedings of the International Conference on Knowledge Engineering and Decision Support*, 2004.
- [Pereira 04b] Antonio Pereira & Luis Paulo Reis. *Agent-based simulation of ecological models*. In eds H. Coelho B. Espinasse, editeur, *Proceedings 5th Workshop on agent-based simulation*, 2004.
- [Pereira 05] António Pereira, Pedro Duarte & Luís Paulo Reis. *ECO-LANG - A Communication Language for Simulations of Complex Ecological Systems*. In E. Kerckhoffs Eds. Y. Merkuriev R. Zobel, editeur, *Proceedings of the 19th European Conference on Modelling and Simulation*, pages 493–500, 2005. ISBN : 1-84233-112-4.

- [Pitchford 01] Jonathan William Pitchford & John Brindley. *Prey Patchiness, Predator Survival and Fish Recruitment*. Bulletin of Mathematical Biology, vol. 63, pages 527–546, 2001.
- [Poulard 05] Jean-Charles Poulard & Fabian Blanchard. *The impact of climate change on the fish community structure of the eastern continental shelf of the Bay of Biscay*. ICES Journal of Marine Science, vol. 62, pages 1436–1443, 2005.
- [Quesnel 05] G. Quesnel, R. Duboz & E. Ramat. *Coupling of physical models and socials models : multi- modeling and simulation with VLE*. In Joint Conference on Multi-Agent Modelling for Environmental Management, CABM-HEMA- SMAGET’05, 2005.
- [Railsback 01b] Steven F. Railsback & Bret C. Harvey. *Analysis of habitat-selection rules using an individual-based models*. Ecology, vol. 83, pages 1817–1830, 2001.
- [Railsback 02] Steven F. Railsback, Bret C. Harvey, Roland H. Lamber-son, Derek E. Lee, Nathan J. Claasen & Shuzo Yoshihara. *Population-level analysis and validation of an individual-based cutthroat trout model*. Natural Resource Modeling, vol. 15, pages 83–110, 2002.
- [Railsback 06] Steven F. Railsback, Steven L. Lytinen & Stephen K. Jack-son. *Agent-based Simulation Platforms : Review and Deve-lopment Recommendations*. Simulation, vol. 82, pages 609–623, 2006.
- [Ramade 03] François Ramade. *Eléments d’écologie - ecologie fondamen-tale*. DUNOD, Paris, 3rd edition, 2003.
- [Ramat 03] E. Ramat & P. Preux. *Virtual Laboratory Environment (VLE) : A Software Environment oriented Agent and Object for Modeling and Simulation of Complex Systems*. Journal of Simulation Practice and Theory, vol. 11, pages 45–55, mars 2003.
- [Redou 05] P. Redou, S. Kerdelo, C. Le Gal, G. Querrec, V. Rodin, J.-F. Abgrall & J. Tisseau. *Reaction-Agents : First Mathe-matical Validation of a Multi- agent System for Dynamical Biochemical Kinetics*. Lecture Notes in Computer Science (Progress in Artificial Intelligence), vol. 3808, pages 156–166, 2005.

- [Rodin 04] Vincent Rodin. *Contribution à l'utilisation de l'informatique en biologie*. HDR - Mémoire pour l'obtention de l'Habilitation à Diriger des Recherches de l'Université de Rennes I, 6 décembre 2004.
- [Rose 93] K.A. Rose, S.W. Christensen & D.L. DeAngelis. *Individual-based modeling of populations with high mortality : a new method based on following a fixed number of model individuals*. Ecological modelling, vol. 68, pages 273–292, 1993.
- [Rumbaugh 99] James Rumbaugh, Ivar Jacobson & Grady Booch. The unified modeling language reference manual. Addison-Wesley Professional, 1999.
- [Sacks 89] Jerome Sacks, William J. Welch, Toby J. Mitchell & Henry P. Wynn. *Design and Analysis of Computer Experiments*. Statistical Science, vol. 4, pages 409–423, 1989.
- [Salvanes 01] Anne Gro Veia Salvanes. *Review of ecosystem models of fjords ; new insights of relevance to fisheries management*. In Sarsia. International scientific publisher Taylor & Francis, in collaboration with Department of Fisheries and Marine Biology, University of Bergen, and Institute of Marine Research, Bergen, Norway, 2001.
- [Santner 03] Thomas J. Santner, Brian J. Williams & William Notz. The design and analysis of computer experiments. Springer, 2003.
- [Savage 99] M. Savage & M. Askenazi. *Arborscapes : A Swarm-Based Multi-Agent Ecological Disturbance Model*, 1999.
- [Scheffer 95] M. Scheffer, JM. Baveco, DL. DeAngelis, KA. Rose & EH. Van Nes. *Super individuals a simple solution for modelling large populations on an individual basis*. Ecological Modelling, vol. 80, pages 161–170, 1995.
- [Schuler 93] Douglas Schuler & Aki Namioka. Participatory design : Principles and practices. Lawrence Erlbaum Associates, 1993.
- [Servat 98b] David Servat, Edith Perrier, Jean-Pierre Treuil & Alexis Drogoul. *When agents emerge from agents : introducing multi-scale viewpoints in multi-agent simulations*. In MABS'98, 1998.

- [Shin 00] Yunne-Jai Shin. *Interactions trophiques et dynamiques des populations dans les écosystèmes marins exploités. Approche par modélisation individus-centrée*. Thèse de doctorat spécialité biomathématiques, Université Paris 7 – Denis Diderot, 2000.
- [Shin 01] Yunne-Jai Shin & Philippe Cury. *Exploring fish community dynamics through size-dependent trophic interactions using a spatialized individual-based model*. Aquatic Living Resources, vol. 14, pages 65–80, 2001.
- [Stockler 91] Manfred Stockler. *A Short History of Emergence and Reductionism*. In Agazzi, éditeur, The Problem of Reductionism in Science, pages 71–90. Dordrecht, 1991.
- [Susi 01] Tarja Susi & Tom Ziemke. *Social cognition, artefacts, and stigmergy : A comparative analysis of theoretical frameworks for the understanding of artefact-mediated collaborative activity*. Journal of Cognitive Systems Research, vol. 2, pages 273–290, 2001.
- [Swoyer 91] C. Swoyer. *Structural Representation and Surrogate Reasoning*. Synthese, vol. 87, pages 449–508, 1991.
- [Tisseau 01a] Jacques Tisseau. *Réalité virtuelle : autonomie in virtuo*. Habilitation à diriger des recherches, Université de Rennes I, 6 dec 2001.
- [Tyler 94] Jeffrey A. Tyler & Kenneth A. Rose. *Individual variability and spatial heterogeneity in fish population models*. Reviews in Fish Biology and Fisheries (Historical Archive), vol. 4, pages 91 – 123, March 1994.
- [Uchmanski 96] J Uchmanski & V Grimm. *Individual-based modelling in ecology : What makes the difference ?* Trends in Ecology & Evolution, vol. 11, pages 437–440, 1996.
- [Uchmanski 99] Janusz Uchmanski. *What promotes persistence of a single population : an individual-based model*. Ecological Modelling, vol. 115, pages 227–241, 1999.
- [Villa 01] Ferdinando Villa. *Integrating modelling architecture : a declarative framework for multi-paradigm, multi-scale ecological modelling*. Ecological Modelling, vol. 137, pages 23 – 42, 2001.

- [Volterra 31] V. Volterra. *Variations and fluctuations of the number of individuals in animal species living together*, 1931. Translated from 1928 edition by R. N. Chapman.
- [von Foerster 60] H. von Foerster. *On Self-Organizing Systems and Their Environments*. In *Self-organizing Systems*, 1960.
- [Watzlawick 84] Paul Watzlawick. *La réalité de la réalité*. Seuil, 1984.
- [Weber 95] Jacques Weber. *Gestion des ressources renouvelables : fondements théoriques d'un programme de recherche*. Cirad Green, juin 1995. 21 p.
- [Wiegand 03] Thorsten Wiegand, Florian Jeltsch, Ilkka Hanski & Volker Grimm. *Using pattern-oriented modeling for revealing hidden information : a key for reconciling ecological theory and application*. OIKOS, vol. 100, pages 209–222, 2003.
- [Wiener 65] Norbert Wiener. *Cybernetics or control and communication in the animal and the machine*. MIT Press, 1965.
- [Wolfram 83] Stephen Wolfram. *Statistical Mechanics of Cellular Automata*. *Reviews of Modern Physics*, vol. 55, pages 601–644, 1983.
- [Wolfram 84] Stephen Wolfram. *Universality and Complexity in Cellular Automata*. *Physica D*, vol. 10, pages 1–35, 1984.
- [Wooldridge 95] M. Wooldridge & N.R. Jennings. *Intelligent agent : theory and practice*. *Knowledge Engineering Review*, vol. 10, pages 115–152, 1995.
- [Wyszomirski 99] Tomasz Wyszomirski, Izabela Wyszomirska & Ingeborga Jarzyna. *Simple mechanisms of size distribution dynamics in crowded and uncrowded virtual monocultures*. *Ecological Modelling*, vol. 115, pages 253–273, 1999.
- [Zeigler 76] B. Zeigler. *Theory of modeling and simulations*. Wiley Interscience, New York, 1976.
- [Zeigler 89] Bernard P. Zeigler. *DEVS Representation of dynamical systems : event-based intelligent control*. In *Proceedings of the IEEE*, volume 77, January 1989.

Annexe A

Synchronisation de comportements dépendants – Preuve mathématique

Dans la suite de l'étude sur les biais computationnels dans les modèles de peuplements et en particulier sur la synchronisation de comportements dépendants (voir section 5.5.2 - D), nous présentons ici la preuve mathématique démontrant que les résultats obtenus en simulation étaient prévisibles et que le système est bien régi par une loi exponentielle.

A.1 Notations et hypothèses

On note :

- ▷ T le temps de la simulation, incrémenté de manière discrète
- ▷ t le temps de vie d'un individu, compris entre 0 et 1.
- ▷ $f^T(t)$ le nombre d'individus d'âge t au temps T
- ▷ $N(T)$ désigne la population totale à l'instant T
- ▷ À chaque pas de temps de simulation $T \rightarrow T+1$ les individus vieillissent de θ ($\theta = 0.03$ dans les simulations)

Hypothèses :

A l'instant initial on a une population de N_0 individus, d'âges uniformément répartis entre 0 et 1.

Lorsqu'un individu atteint un âge supérieur ou égal à 1, il peut aléatoirement créer un individu (d'âge 0) puis mourir, ou mourir sans procréer, chacune de ces hypothèses se réalisant avec la probabilité 1/2.

A.2 Équations caractérisant $f^T(t)$

On a clairement :

$$\int_0^1 f^{T+1}(t)dt = \int_0^1 f^T(t)dt - \frac{1}{2} \int_{1-\theta}^1 f^T(t)dt \quad (\text{A.1})$$

En effet, tous les individus restent en vie sur un pas de temps de simulation, exceptée la moitié (en moyenne) de ceux dont l'âge était à l'instant précédent compris entre $1 - \theta$ et 1.

On a donc plus généralement

$$\int_0^1 f^{T+k}(t)dt = \int_0^1 f^T(t)dt - \frac{1}{2} \int_{1-k\theta}^1 f^T(t)dt \quad (\text{A.2})$$

si bien que pour $1 - k\theta = \frac{1}{2}$, i.e. pour $k = \frac{1}{2\theta}$, on obtient

$$\int_0^1 f^{T+\frac{1}{2\theta}}(t)dt = \frac{1}{2} \int_0^1 f^T(t)dt \quad (\text{A.3})$$

Remarquons que cette expression n'a de sens que si $\frac{1}{2\theta}$ est un entier. Si tel n'est pas le cas, elle nous servira d'approximation.

A.3 Évolution de la population globale

Rappelons que $N(T)$ désigne la population totale à l'instant T . D'après l'équation (A.3), nous avons

$$N(T + \frac{1}{2\theta}) = \frac{1}{2}N(T)$$

Nous supposons que l'évolution de la population entre les instants T et $T + \frac{1}{2\theta}$ est linéaire. Cette hypothèse est tout à fait cohérente, au regard de l'évolution sur un pas de temps donnée par (A.1). Si l'on suppose que la courbe sur un pas de temps est $y = ax + b$, où a et b doivent être déterminés, on a alors

$$aT + b = N(T) \quad , \quad a(T + \frac{1}{2\theta}) + b = \frac{1}{2}N(T)$$

ce qui donne

$$a = -\theta N(T), \quad b = (1 + T\theta)N(T)$$

L'évolution linéaire sur $\frac{1}{2}$ pas de temps de simulation de T à $T + \frac{1}{2}$ est donc donnée par la droite

$$y(s) = N(T) (-\theta s + 1 + T\theta)$$

On a donc

$$y(T+1) = (1-\theta)y(T)$$

si bien que l'on obtient l'équation différentielle

$$N'(s) = -\theta N(s)$$

et l'on peut considérer, avec la condition initiale $N(0) = N_0$, que la population est régie par l'expression

$$N(s) = N_0 e^{-s\theta}$$

Annexe B

Projet Chaloupe

Nous présentons dans cette section les modèles IBM utilisés pour Chaloupe, donc aussi bien pour la problématique écologique, modélisation du peuplements de poissons du GdG, que pour la problématique économique, modélisation du système économiques des pêcheries du GdG.

B.1 Problématique écologique : hypothèses et modèles

Le modèle suivant a été construit en collaboration avec un écologue de l'IFREMER pour représenter les peuplements de poissons du Golfe de Gascogne (GdG). Nous présentons ici les éléments structurants du modèle en identifiant les hypothèses pour chacun de ces éléments et éventuellement le modèle conceptuel associé.

B.1.1 Environnement

Cadre théorique – hypothèses

- ▷ L'environnement est fermé et aucun processus externe est suffisamment important pour être pris en compte.
- ▷ L'environnement peut être considéré comme ayant deux dimensions. Cette hypothèse est à mettre en relation avec l'hypothèse du réseau trophique (processus de prédation) qui permet de re-crée artificiellement, au moins partiellement, la profondeur de

l'environnement et le fait que tous les poissons ne peuvent pas se rencontrer ou très rarement.

- ▷ L'environnement est continu.

B.1.2 Vieillesse

Cadre théorique – hypothèses

- ▷ L'âge limite ${}_s a_{\max}$ correspond à l'espérance de vie moyenne (longévité). Les individus trop vieux meurent.
- ▷ Les processus biologiques et écologiques sont décrits pour les adultes, et par conséquent les individus présents dans le système ont au moins l'âge de recrutement ${}_s a_{\min}$.

Modèle

$${}_s a_{\min} \leq {}^i a \leq {}_s a_{\max}$$

Valeurs des paramètres – Golfe de Gascogne

Espèce	${}_s a_{\min}$	${}_s a_{\max}$
Anchois	.09	5
Baudroie	.024	22
Chinchard	.059	15
Merlan	.021	10
Merlan bleu	3.53	10
Merlu	0.5	15
Petit tacaud	1.7	5
Sardine	1.6	10
Tacaud	0.4	6

B.1.3 Croissance

Cadre théorique – hypothèses

- ▷ Il n'y a sur ce processus aucune influence de l'environnement, de l'habitat ou de la température.
- ▷ Le processus n'est pas lié à l'alimentation.

Modèle

- ▷ Utilisation de la relation longueur à l'âge de Von Bertalanffy.

- ▷ Relation longueur-poid

$$w(t) = a \cdot \ell(t)^b$$

- ▷ Aucune variabilité individuelle, et par conséquent aucun individu a une longueur supérieur à $L\infty$.

Valeurs des paramètres – Golfe de Gascogne

Espèce	$L\infty$	k	t_0	a	b
Anchois	22.1	.54	-.59	.002954	3.2579
Baudroie	150.0	.088	-.024	.02457	2.8561
Chinchard	44.9	.2	-.59	.007325	3.0254
Merlan	52.2	.45	-.21	.0004549	3.1669
Merlan bleu	18.0	.35	-.09	.0023	3.499
Merlu	1.10	.12	-.5	.00438	3.113
Petit tacaud	22.0	.39	-1.7	.0086	2.98
Sardine	25.6	.42	-1.6	.005936	3.0767
Tacaud	22.0	.5	-.4	0.00738	3.1561

B.1.4 Recrutement

Cadre théorique – hypothèses

- ▷ La fécondité individuelle est fonction du poid de l'individu. Il n'y a pas de distinction entre mâles et femelles.
- ▷ Les individus sont matures à une longueur donnée ${}_s\ell_{mat}$
- ▷ La température affecte la fécondité : la fécondité est maximale pour une température donnée ${}_s\theta^*$ et elle n'est pas possible (null) au-delà d'un certain domaine de valeurs $[{}_s\theta_l, {}_s\theta_u]$. Pour l'espèce s , ${}_s\theta^*$ est mis à la valeur de la température moyenne de la surface de l'eau au milieu de l'aire de distribution.
- ▷ La fécondité individuelle est fonction de l'affinité.
- ▷ Le nombre de progénitures dépend du nombre d'adultes matures.
- ▷ Le recrutement a lieu toute l'année (il n'y a pas de saison de reproduction).
- ▷ Les recrues ont le même habitat que les adultes.

Modèle

$$\begin{aligned}
 {}^i_s\phi(a) &= {}^i_s\alpha_H \times {}_s\rho({}^i\theta) \times f(\ell_i) \\
 \rho(\theta) &= \max(0, 1 + \frac{\theta - \theta^*}{\theta^* - \theta_l}) \quad (\theta \leq \theta^*) \\
 \rho(\theta) &= \max(0, 1 + \frac{\theta^* - \theta}{\theta_u - \theta^*}) \quad \text{sinon}
 \end{aligned}$$

$$f(\ell_i) = \frac{1}{2} \times \gamma_\phi \times w(\ell_i) \quad \ell > s\ell_{mat}$$

$$f(\ell_i) = 0 \quad \text{sinon}$$

Valeurs des paramètres – Golfe de Gascogne

Espèce	$s\theta_l$	$s\theta^*$	$s\theta_u$	$s\ell_{mat}$	γ_ϕ
Anchois	10.6	14.6	18.6	10	.0001
Baudroie	5.7	9.1	13.7	62	.001
Chinchard	10.1	14.1	19.1	23	.0003
Merlan	4.9	8.9	12.9	33	.001
Merlan bleu	5.2	9.2	12.2	25	.0002
Merlu	7.7	11.7	15.7	53	.001
Petit tacaud	6.6	10.6	14.6	14	.0001
Sardine	8.2	12.2	16.2	20	.0003
Tacaud	7.1	11.1	15.1	18	.00001

B.1.5 Mortalité

Cadre théorique – hypothèses

- ▷ Les habitats ont une capacité de charge maximum qui a un impact sur la survie. Cette capacité de charge est une densité.
- ▷ Une forte densité de l'espèce s entraîne une forte mortalité de cette espèce.
- ▷ Pour les prédateurs, l'absence de proies entraîne la mort.
- ▷ Pour les proies, dont le niveau trophique est le plus faible, la mortalité est fonction de la disponibilité de nourriture ; cette disponibilité est constante et n'est pas un facteur limitant.
- ▷ La capacité de charge peut être :
 - indépendante de l'affinité et donc est uniforme.
 - dépendante de l'affinité et donc proportionnelle à celle-ci.

Par conséquent,

- ▷ Dans tous les cas, un individu de l'espèce s meurt à cause de
 - son âge (voir le processus de vieillissement)
 - la capacité de charge K_H
- ▷ de plus, un prédateur meurt à cause de
 - rien (ou du manque de proies).
- ▷ une proie meurt à cause de
 - la prédation, quand elle est mangée par un prédateur.
- ▷ une espèce intermédiaire, aussi bien prédateur que proie, meurt à cause de
 - de la prédation, quand elle est mangée par un prédateur.

Pour le mécanisme de famine, voir le processus de prédation.

Modèle

$$\gamma_M = \min(1, \max(0, {}_sB - {}_sK))$$

$${}_sB = \sum_{x=x_{min}, y=y_{min}}^{x_{max}, y_{max}} w_i(x, y)$$

Si la capacité de charge est proportionnelle à l'affinité, alors on a :

$$K = \alpha_H \times {}_sK_\infty$$

Valeurs des paramètres – Golfe de Gascogne

Espèce	${}_sK_\infty$
Anchois	10428.46
Baudroie	385.41
Chinchard	322299.56
Merlan	2609.09
Merlan bleu	55182.37
Merlu	11300.75
Petit tacaud	23084.74
Sardine	12894.21
Tacaud	9233.28

${}_sK_\infty$ est calculée en utilisant les données EVHOE. Cela correspond à la densité la plus forte qui peut être trouvée dans les strates sur toutes les années.

B.1.6 Prédation

Cadre théorique – hypothèses

- ▷ Les prédateurs ont une probabilité spécifique de manger un individu de l'espèce s .
- ▷ La taille des proies consommées est limitée par la taille des prédateurs. Par conséquent, le processus de prédation est basé sur un ratio ${}_s\tau$ de la taille du prédateur sur la taille de la proie.
- ▷ Les individus jeunes peuvent avoir un régime alimentaire différent de celui des adultes. Le régime d'un poisson devient celui d'un adulte quand le poissons attend la taille ${}_s\ell_P$.
- ▷ Un individu a besoin de manger une quantité de nourriture qui est fonction de son propre poid et de η .

- ▷ L'alimentation individuelle est limitée.
- ▷ La prédation est organisée par un réseau trophique. Chaque espèce a un niveau trophique $s\tau_{trophic}$.
- ▷ La disponibilité en nourriture pour le niveau le plus faible du réseau trophique est constante et n'est pas un facteur limitant.
- ▷ La prédation a un impact positif sur les prédateurs, mais le manque de nourriture entraîne la mort. Le niveau de famine est basé sur une ration de survie $\eta_{\min}$.

Modèle

$$\begin{aligned}\eta_{\min} &\leq \eta \\ w_f &= \eta \times i_w\end{aligned}$$

Valeurs des paramètres – Golfe de Gascogne

$$\eta = 3.5$$

$$\eta_{\min} = 0$$

Espèce	$s\tau$	$s\ell_P$	$s\tau_{trophic}$
Anchois	3.57	10	2.9
Baudroie	3.57	33	4.5
Chinchard	3.57	22	3.7
Merlan	3.57	28	4.4
Merlan bleu	3.57	7	3.7
Merlu	3.57	26	4.4
Petit tacaud	3.57	33	3.7
Sardine	3.57	9	2.7
Tacaud	3.57	13	3.7

B.1.7 Occupation spatiale

Cadre théorique – hypothèses

- ▷ Les poissons sont influencés par leur habitat et tendent à aller vers leur habitat préférentiel.
- ▷ Les poissons sont influencés par la densité intraspécifique de leur environnement local.

Modèle

$$\eta$$

Valeurs des paramètres – Golfe de Gascogne

Espèce	$s\tau$	$s\ell_P$	$s\tau_{trophic}$
Anchois	3.57	10	2.9
Baudroie	3.57	33	4.5
Chinchard	3.57	22	3.7
Merlan	3.57	28	4.4
Merlan bleu	3.57	7	3.7
Merlu	3.57	26	4.4
Petit tacaud	3.57	33	3.7
Sardine	3.57	9	2.7
Tacaud	3.57	13	3.7

B.2 Problématique économique : modèles**B.2.1 Cas d’un agent “acteur économique”**

Les individus sont les pêcheurs. Ces agents sont également appelés *EconomicAgent* ou *eAgent* pour leur fonction dans le marché des bateaux de pêche que nous considérons ici. La vitalité de ces agents est leur capital qui leur permet d’exister dans le marché et de disposer d’un moyen de production pour pratiquer leur activité de pêcheur. La perception de l’environnement est la faculté d’entrevoir le marché des bateaux (neufs et d’occasion). La perception des autres consiste à avoir une visibilité sur l’activité de pêche des autres *eAgents*. Les agents sont également capables de percevoir l’écosystème sur lequel ils prélèvent la ressource. Enfin, les actions que ces agents peuvent effectuer sont le changement d’activité (départ en retraite, reconversion, changement d’activité de pêche), le prélèvement sur la ressource ou le réinvestissement sur leur unité de pêche (assimilé à des modifications de l’état interne de l’agent). Ces actions sont contrôlées par une rationalité qui leur permet de considérer leur état économique et les autres états possibles. Evoluant dans un environnement économique fortement hétérogène, ils sont capables de considérer la situation économique d’autres agents et d’en évaluer l’intérêt financier.

B.2.2 Bateau : unité de pêche

Un *eAgent* a une unité de pêche. Ce bateau est d’abord un outil permettant l’exercice d’un métier dans une zone de pêche donnée. Mais, un bateau représente également le gain futur qu’il est possible d’obtenir avec lui. Il a un coût d’acquisition, un coût d’entretien et une efficacité technique qui décroît au cours du temps (figure B.1). Cette efficacité implique une certaine capture de la ressource par effort de pêche (sortie en mer), donc un gain donné. Enfin, l’efficacité technique peut être augmentée par réinvestissement (réparations, remotorisation ou progrès techniques).

B.2.3 La pêche : agent “pêcheur”

Le premier environnement de l’*eAgent* est l’écosystème du *GdG* dans lequel est prélevée la ressource. La ressource étant partagée et renouvelable, elle est caractérisée par un coefficient de capture par unité d’effort *CPUE* qui est ici constant (C_m) pour un effort de pêche global (nombre de pêcheurs dans le *GdG*) inférieur à un certain seuil (N_l) et qui décroît ensuite au-delà de ce seuil pour atteindre zéro quand l’effort de pêche atteint un deuxième seuil (N_u) qui est théoriquement inaccessible (figure B.1 et détails de la figure tableau B.1). Il est important de noter que cette modélisation de l’accès à la ressource est d’abord rudimentaire, voire imparfaite, en tout cas irréaliste, mais elle est suffisante actuellement pour nous permettre d’atteindre nos objectifs. Surtout elle correspond à une limite du modèle qui, dans la perspective d’un couplage avec le modèle écologique, disparaîtra.

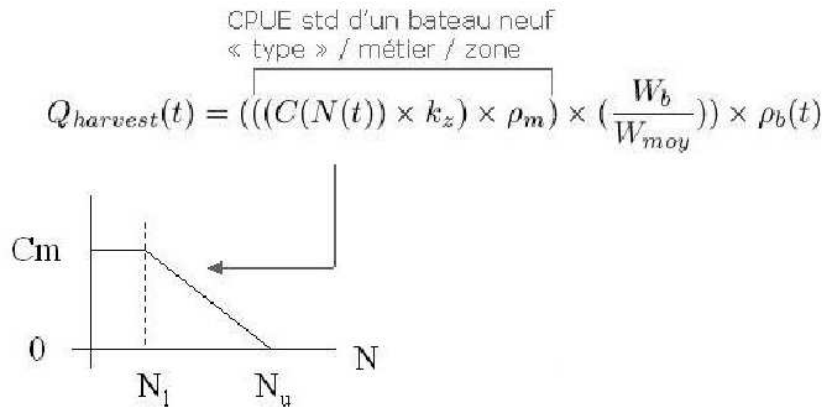


FIG. B.1: La quantité pêchée par un pêcheur en fonction de la *CPUE* (voir table B.1 pour les détails)

TAB. B.1: Détails de la figure B.1

$Q_{harvest}(t)$	La quantité pêchée à un instant t par le bateau b exerçant le métier m dans la zone z .
$C(N(t))$	La <i>CPUE</i> (<i>Capture Par Unité d'Effort</i>) d'un navire de pêche considérant le nombre de navires en activité à un instant t (cf. figure B.1).
ρ_m	L'efficacité relative d'un métier.
W_b/W_{moy}	La puissance d'un navire donné normalisée par rapport à la puissance moyenne de tous les navires.
$\rho_b(t)$	L'efficacité d'un navire de pêche donné (voir figure B.1).
k_z	Coefficient de zone (compris entre 0 et 1) permettant de moduler la <i>CPUE</i> selon la zone.

B.2.4 Patron pêcheur : agent “acteur économique”

Un patron pêcheur doit gérer son activité de pêche. Pour cela, l'*eAgent* a d'abord un capital (sa vitalité) qu'il va pouvoir utiliser pour investir dans un nouveau bateau ou dans des améliorations technologiques. Cette dernière forme d'investissement est une réponse du pêcheur à la perte régulière d'efficacité de son outil de production. L'efficacité a une évolution en dents de scie entre une enveloppe basse et une enveloppe haute comme décrit dans la figure B.2 (p. 218). Elle se voit soustrait une utilisation journalière au cours du temps et décroît jusqu'à croiser l'enveloppe basse. Le réinvestissement (maintenance, réparation, ...) effectué par le pêcheur est alors modélisé par la remontée de l'efficacité jusqu'à l'enveloppe haute. Le pêcheur peut réinvestir quelque soit l'état de son capital, mais celui-ci est mise à jour en fonction du réinvestissement. L'algorithme 1 (p. 218) précise les valeurs utilisées pour calculer les enveloppes et l'efficacité.

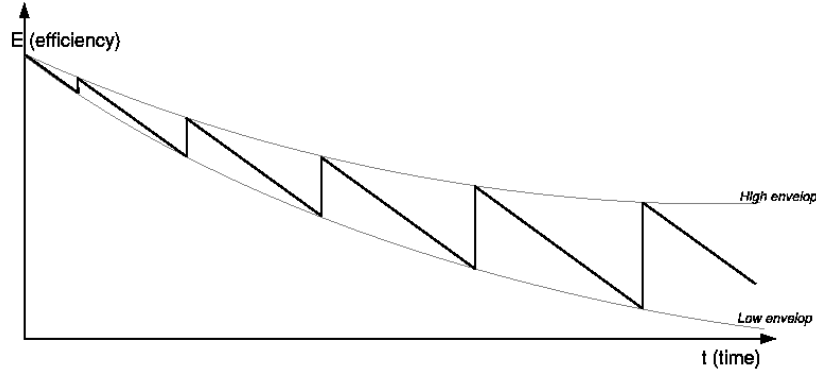


FIG. B.2: L'évolution de l'efficacité d'une unité de pêche avec réinvestissements réguliers du pêcheur.

Algorithm 1: `CALCEFFICIENCY(age t)`

```
procedure HIGHENV(t)
{ /* Describes the high of the efficiency.*/
   $H = ((0.010 \times t) + 1)^{-0.06}$ 
  return ( $H$ )

procedure LOWENV(t)
{ /* Describes the low envelop of the efficiency.*/
   $L = ((0.015 \times t) + 1)^{-0.13}$ 
  return ( $L$ )

main CALCEFFICIENCY(age t)
/* DU corresponds to the cost of daily usage*/
for  $i \leftarrow 0$  to  $t$ 
  do {  $E \leftarrow E \times (1 - DU)$ 
    if  $E < \text{LOWENV}(i)$ 
      then  $E \leftarrow \text{HIGHENV}(i)$ 
  }
return ( $E$ )
```

La capacité d'anticipation d'un agent est également modélisée et prend en compte la stratégie d'investissement et donc la perte d'efficacité de l'unité de pêche. Nous présentons l'algorithme d'anticipation (algo. 3 p. 219) et l'algorithme permettant de calculer la valeur

matérielle du navire (ou valeur résiduelle) et qui est utilisée pour l'anticipation.

Algorithm 2: GETRESIDUALVALUE(*Vessel* V_k)

```

/* E is the efficiency – Set to 1 when  $V_k$  is new*/
 $E_k \leftarrow 1$ 
for  $i \leftarrow 0$  to  $V_k$ .GETAGE()
  do  $E_k \leftarrow E_k \times (1 - \text{dailyUse})$ 

/* getFAAcquisitionCost gets acquisition cost of  $V_k$ 's Fishing Activities*/
 $P_k \leftarrow V_k$ .GETACQUISITIONCOST() +  $V_k$ .GETFAACQUISITIONCOST()
return  $(P)_k \times E_k$ 

```

Algorithm 3: ANTICIPATE(*Vessel* V_k)

```

/* Returns the estimated price of a given vessel*/
/* RDA stands for Residual Duration of Activity*/
/*  $Ag^1$  calculates the anticipated price based on his RDA*/
/*  $RDA = RETIREMENT\_AGE - \text{age of } Ag^1$ */
/*  $R$  is the individual psychological discount rate*/

local temp is a temporary variable
 $temp \leftarrow 0$ 
for  $i \leftarrow 1$  to  $t$ 
  do  $temp = temp + \frac{1}{(1+R)^i}$ 

/*  $G_k^1 = \text{residual value} + \text{profit with } V_k \text{ for RDA period}*/$ 
/* In math terms :  $G_k^1 = \sum i \frac{V_{k,i}}{(1+R)^i}$ */
 $G_k^1 \leftarrow \frac{\text{GETRESIDUALVALUE}(V_k)}{(1+R)^t} + \text{GETANNUALPROFITS}(V_k) \times temp$ 

return  $(G_k^1)$ 

```

Enfin, tout agent choisit son activité et est capable d'en changer. Il s'appuie pour ce faire sur l'anticipation qu'il fait de ses gains futurs avec son unité de pêche ou tout autre unité de pêche qu'il pourrait acquérir. Il peut ainsi décider de ne pas changer son état ou au contraire de changer d'activité. Dans cette prise de décision (en particulier pour le changement d'activité), l'agent prend en considération les aides extérieures (prime de sortie

des bateaux), le revenu moyen qu'il peut espérer avoir s'il change d'activité, ou tout autre facteur extérieur pouvant être pris en considération. Dans notre modèle, nous réduisons le pêcheur à un agent entièrement rationnel dont l'objectif est la recherche de gain. Nous adoptons donc une approche simplificatrice en ne considérant que le seul critère économique pour apprécier l'environnement de l'agent, son état ou les changements d'états possibles pour lui. Nous traduisons la rationalité des pêcheurs par des algorithmes appelés *Delta* (Δ *formulas* : algorithme 4). Le pêcheur peut calculer le prix d'un navire de pêche donné (prix du matériel strictement) et anticiper son profit futur, il peut donc comparer différentes situations économiques avec différents navires ou entre une activité dans la pêche et une activité extérieure à l'économie de la pêche. Les *formules* Δ qualifient l'intérêt (positif ou négatif) que l'agent peut avoir à garder son activité ou à en changer pour une autre. Cet intérêt au changement est appelé *coefficient d'intérêt* (δ) entre deux situations économiques. L'algorithme 4 est la *formule* Δ générale qui compare deux activités économiques quelles qu'elles soient. Il existe des *formules* Δ plus spécialisées : la *formule* Δ_C permet de comparer en pratique deux navires de pêche ou l'échange du premier navire de pêche contre le deuxième ; la *formule* Δ_X permet de comparer une activité de pêche avec une activité extérieure à l'économie de la pêche. Le développement des algorithmes Δ_C et Δ_X , leur formulation est une spécialisation de l'algorithme général Δ . Considérons un *eAgent* Ag^a . Pour que Ag^a soit intéressé par l'échange de son activité A_1 contre une activité A_2 , δ , calculé par l'algorithme 4, doit être supérieur à zéro.

Algorithm 4: $\Delta(A_1, A_2)$

```

/*  $A_1$  is  $Ag^1$ 's activity*/
/*  $A_2$  is the activity  $Ag^1$ 's thinking of changing to*/
/* Each activity  $A_i$  has a price  $P_i$  and a gain  $G_i^1$ */
/* The changing of activity has a price  $P_C$  and a gain  $G_C^1$ */
/*  $\forall A_i$ , the gain  $G_i^1$  depends on  $Ag^1$ 's capacities*/

/*  $\delta$  describes if the change of activity is profitable for  $Ag^1$ */

 $\delta \leftarrow \Delta_A(A_2) - \Delta_A(A_1) + \Delta_C()$ 
 $\Leftrightarrow \delta \leftarrow (G_2^1 - P_2) - (G_1^1 - P_1) + (G_C^1 - P_C^1)$ 
 $\Leftrightarrow \delta \leftarrow G_2^1 - P_2 - G_1^1 + P_1 + G_C^1 - P_C^1$ 
return ( $\delta$ )

```

Algorithm 5: $\Delta_C(V_1, V_2)$

```

/*  $\delta_C$  : coef that characterises an activities exchange */
/*  $\forall i, P_i =$  the price of the market of vessel  $V_i$  */
 $\delta_C \in \mathbb{R}$ 
for  $i \leftarrow 1$  to 2
  do  $\begin{cases} P_i \leftarrow V_i.\text{GETMARKETPRICE}() \\ G_i \leftarrow \text{ANTICIPATE}(V_i) \end{cases}$ 

 $\delta_C \leftarrow P_1 - P_2 + G_2 - G_1$ 
return ( $\delta_C$ )

```

Algorithm 6: $\Delta_X(V_1)$

```

/*  $\delta_X$  : coef that characterises the exiting of the market */
 $\delta_X, G_1, P_1, G_0, P_0, G_C^{1,1}, P_C^{1,1} \in \mathbb{R}$ 

 $P_1 \leftarrow V_1.\text{GETPRICE}()$ 

/* Bounties for exiting the fishing market. */
 $P_C^1 \leftarrow \text{GETBOUNTIES}()$ 
/*  $P_{C,y}^x$  : bounty depends on  $Ag^x$ 's condition and vessel  $V_y$ . */

/* Average wage of an external activity. */
 $G_0 \leftarrow \text{GETAVGEXTERNALWAGE}()$ 

/* the average starting cost of an external activity. */
 $P_0 \leftarrow \text{GETAVGOPPORTUNITYCOST}()$ 

 $\delta_X \leftarrow P_1 - G_1^1 + G_0 - P_0 - P_{C,1}^1$ 
return ( $\delta_X$ )

```

B.2.5 Environnement

L'environnement de la problématique économique correspond ici à deux environnements. Le premier est l'écosystème du GdG dans lequel est prélevée la ressource du pêcheur (voir ci-dessus – B.2.3). Le deuxième dans lequel évolue également l'*eAgent* est le marché des

bateaux. Le marché est modélisé comme une vente aux enchères. Chaque année à une date donnée, chaque *eAgent* peut décider d’être vendeur (et du coup éventuellement acheteur) ou non¹. Un commissaire priseur virtuel (*Virtual Auctioneer* ou *VA*) organise les ventes et achats des bateaux (les bateaux des agents qui ont décidé de vendre) en posant un prix de départ et en le faisant évoluer en fonction de l’offre et de la demande. Chaque bateau en vente est proposé à tous les acheteurs potentiels qui calculent en fonction des caractéristiques du bateau (ex. l’efficacité) le gain possible avec celui-ci. Un *eAgent* peut alors décider d’acheter un bateau qui est plus intéressant économiquement pour lui et vendre son bateau alors même qu’il n’était pas vendeur au début. Nous présentons l’algorithme simplifié (et “textualisé”) utilisé pour décrire les enchères ci-dessous :

▷ Les enchères se déroulent par tour (un bateau à la fois)

▷ *VA* (/tour)

Algorithme :

- Construction de la table des unités de pêche à vendre
- Mise à jour des prix tant qu’il n’y a pas qu’un seul pêcheur intéressé
- Transaction
- Si à la fin il y a conflit entre deux pêcheurs pour un bateau ou deux bateaux pour un seul pêcheur, alors sortir pêcheur et/ou sortir bateau

▷ Pêcheur (/tour)

Algorithme :

- Récupération de la table des unités de pêche
- Choix de la meilleure unité de pêche
- Attente négociation éventuelle
- Si négociation, respecter l’équilibre des fonctions Δ

B.3 Paramètres des modèles

Chaque espèce a des caractéristiques propres (variabilité interspécifique). Bien évidemment, pour deux individus de la même espèce, ces caractéristiques peuvent être légèrement différentes, créant ainsi de l’hétérogénéité (variabilité intraspécifique). La distribution des valeurs de ces caractéristiques individuelles se fait autour de la valeur moyenne caractéristique de l’espèce². Toutes les variables du modèle ont été définies en collaboration avec un écologiste et des économistes de l’IFREMER. Les paramètres et variables du modèle sont listés dans les tableaux B.2 (p. 223) et B.3 (p. 223) pour l’écologie et B.4 (p. 224) et B.5 (p. 224) pour l’économie. Dans les tableaux des paramètres, nous présentons les paramètres observables, issus d’autres modèles et intrinsèques reliés aux processus concernés. Les paramètres observables sont calculés à partir d’observations *in situ*. Ceux issus d’autres modèles

¹ Cette prise de décision dépend de la stratégie choisie. On peut par exemple vouloir forcer tous les pêcheurs à participer aux enchères, à mettre leur navire en vente et à comparer leur activité à celles des autres pêcheurs.

² Pour les agents poissons, cet artefact simulateur modélise la plasticité écologique des êtres vivants et l’impossibilité de décrire une variable universelle liée à l’espèce (Ramade 03)

sont estimés par ajustements statistiques de variables. Enfin, les paramètres intrinsèques sont inhérents à notre modélisation *IBM*. Les tableaux des variables indiquent les variables individuelles utilisées (exception faite pour la variable *nombre d'agents* qui est globale). Il y a dans la première colonne les variables libres, dans la colonne du milieu les variables liées aux variables directement à leur gauche dans la première colonne. Enfin, la dernière colonne présente l'initialisation des variables libres. Remarquons que certaines variables liées peuvent apparaître plusieurs fois si elles sont liées à plusieurs variables libres.

TAB. B.2: Problématique écologique : les paramètres

Processus \ Paramètres	Observables	Issus d'autres modèles	Intrinsèques
Croissance		K, L_{∞}, t_0 ¹ α, β ²	espérance de vie vitalité _{origine}
Reproduction	taux de reproduction périodes de reproduction taille min de reproduction	température optimale	
Prédation / Alimentation	$\frac{taille_{predateur}}{taille_{proie}}$ liste de proies	taux de conversion (ρ) énergétique	rayon de perception
Densité dépendance	extension habitat (latitude, profondeur)		rayon de vitalité k voisins
Localisation			rayon de déplacement

¹ $L(t) = L_{\infty} \times (1 - e^{-K \times (age - t_0)})$

² $poid(t) = \alpha \times L(t)^{\beta}$

TAB. B.3: Problématique écologique : les variables, les variables liées et leur initialisation

Variables	Variables liées	Initialisation de la simulation
δ (vie ou mort)	Nombre d'agents	0 ou 1 (abondances mesurées)
(x, y)		répartition spatiale des agents
t		discrétisation
âge	taille, poids	loi de distribution
vitalité		loi de distribution
capacité de reproduction		NON INITIALISEE

TAB. B.4: Problématique économique : les paramètres

Processus \ Paramètres	Observables	Issus d'autres modèles	Intrinsèques
Activité de pêche	$effort$ de pêche caractéristiques techniques du bateau ($efficacité_{m\acute{e}tier}$, $puissance_{bateau}$, $longueur_{bateau}$) liste de métiers	$CPUE$: N_l , N_u et $C(N)$ courbes englobant l'efficacité (α_i , β_i)	
Gestion de l'état interne	prix du poisson $coût_{entretien}^{bateau}$ $coût_{effort}$ $coût_{acquisition}^{bateau}$ opportunité externe : G_C , P_C revenu $_{moyen}$ (activité externe)		
Marché			prix $_{depart}$ pas de mise à jour du prix

TAB. B.5: Problématique économique : les variables, les variables liées et leur initialisation

Variables	variables liées	Initialisation de la simulation
δ (actif ou inactif)	Nombre d'agents	0 ou 1
port / région		répartition spatiale des agents
t		discrétisation
$\hat{age}_{pecheur}$		loi de distribution
$\hat{age}_{bateau}$	$efficacité_{bateau}$ $valeur_{bateau}$ (matériel) capital	loi de distribution
capital de travail		NON INITIALISE
liste de métiers	$valeur_{bateau}$ (matériel) capital	random
métier pratiqué ($\delta(z, m)$)	k_z ρ_m	random

Annexe C

Modélisation écologique à base d'équations aux dérivées partielles

Le système d'équations aux dérivées partielles (EDP) a été développé pour modéliser la dynamique de populations de poissons, en particulier les processus de distribution de la population, de croissance naturelle et d'interactions trophiques (Bonneaud 07c). Ce modèle nous permet donc d'étudier des populations d'individus sous l'influence de la prédation et des interactions de co-occurrences spatiales. Chaque équation de ce système formule l'évolution d'une espèce avec une distribution f_i , avec $f_i(x, y, t)$ le nombre de poissons en (x, y, t) de l'espèce i à l'instant t , et est donnée par :

$$\begin{aligned} \forall 1 \leq i \leq p, \\ \frac{\partial f_i}{\partial t}(x, y, t) = & (\lambda_i \nabla \alpha_i(x, y, t) - \eta_i \nabla f_i(x, y, t)) \cdot \nabla f_i(x, y, t) \\ & + (\lambda_i \Delta \alpha_i(x, y, t) - \eta_i \Delta f_i(x, y, t)) \times f_i(x, y, t) \\ & + \left(\chi_i + \frac{\sum_{j=1}^p \delta_{ij} f_j(x, y, t)}{\sum_{j=1}^p f_j(x, y, t)} - \frac{\sum_{j=1}^p \beta_{ij} f_j(x, y, t)}{\sum_{j=1}^p f_j(x, y, t)} \right) \times f_i(x, y, t) \end{aligned}$$

avec

$$\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad \text{et} \quad \nabla f = \left(\begin{array}{c} \frac{\partial f_i}{\partial x} \\ \frac{\partial f_i}{\partial y} \end{array} \right)$$

avec α_i l'affinité (permettant la migration vers l'habitat préférentiel) de la population de distribution f_i . λ_i et η_i sont des paramètres qui contrôlent l'impact relatif des tropismes antagonistes de l'affinité et de la densité-dépendance et χ_i est le paramètre de croissance de la population. Enfin, $\delta_{i,j}$ et $\beta_{i,j}$ représentent respectivement l'impact de l'alimentation (effet sur la population de prédateurs) et de la prédation (effet sur la population de proies).

L'équation intègre la modélisation du déplacement des poissons basé sur l'affinité au

médium en $\vec{u}$ et compensé par le tropisme à la densité-dépendance. Elle comprend également la croissance naturelle de la population (recrutement et mortalité), qui peut être négative en $\vec{u}$ au temps t . Et finalement, les activités de prédation, si elles existent, sont également prises en compte et externalisées du taux de croissance naturelle. Morcelons l'équation pour faire apparaître les différents termes qui sont alors donnés par :

$$\Theta_i(x, y, t) = \lambda_i \nabla \alpha_i(x, y, t) \cdot \nabla f_i(x, y, t) + \lambda_i \Delta \alpha_i(x, y, t) \times f_i(x, y, t) \quad (\text{C.1})$$

$$\Gamma_i(x, y, t) = -\eta_i \nabla f_i(x, y, t) \cdot \nabla f_i(x, y, t) - \eta_i \Delta f_i(x, y, t) \times f_i(x, y, t) \quad (\text{C.2})$$

$$\Phi_i(x, y, t) = \chi_i \times f_i(x, y, t) \quad (\text{C.3})$$

$$\Omega_i(x, y, t) = \left(\frac{\sum_{j=1}^p \delta_{ij} f_j(x, y, t)}{\sum_{j=1}^p f_j(x, y, t)} - \frac{\sum_{j=1}^p \beta_{ij} f_j(x, y, t)}{\sum_{j=1}^p f_j(x, y, t)} \right) \times f_i(x, y, t) \quad (\text{C.4})$$

Avec Θ_i le terme calculant le tropisme à l'affinité, Γ_i celui calculant le tropisme à la densité-dépendance, Φ_i représente le terme de croissance naturelle et Ω_i celui des activités de prédation (impact de la prédation sur la population i). Il est important de remarquer qu'un poisson à une position donnée au temps $t + \delta t$ peut provenir de différentes positions du temps t et ce à cause des fonctions d'affinité (Θ_i) et de densité-dépendance (Γ_i).

Résumé

La problématique est l'étude de la modélisation pour la simulation de systèmes complexes. Les processus de modélisation et de simulation étant complexes, il faut rationaliser la modélisation et gagner en maîtrise sur elle. Nous proposons de concevoir le modèle complexe comme une société de modèles en interaction. Nous appliquons l'approche *orientée pattern* pour rationaliser la modélisation en la centrant sur les données. Nous utilisons l'approche agent pour distribuer l'observation et le contrôle dans le modèle global. La société de modèles permet alors l'émergence des patterns par l'activité des agents. Les *artefacts*, appliqués à la modélisation, sont le support des données, des patterns et des interactions entre agents. Le couplage des agents se fait ainsi via les *artefacts*, à la différence d'une approche par événements. Nous identifions trois types d'agents : 1° les *agents-modèles*, qui exécutent le modèle d'un phénomène ; 2° les *agents-observateurs*, qui construisent des observables ; 3° les *agents-contrôleurs*, qui contrôlent localement le modèle global. Les agents peuvent également s'adapter à leur environnement avec des opérations pour modifier leurs perceptions et actions. Le modèle complexe peut alors être construit incrémentalement, les comportements des agents peuvent être modifiés facilement, et un agent peut être ajouté ou enlevé du modèle global sans impact sur la structure des autres agents. Enfin, nous identifions quatre types de comportements pour un *agent-modèle* : 1° basé sur des équations différentielles (*intensionnel*) ; 2° centré individu (*émergent*) ; 3° pré-calculé (*extensionnel*) ; 4° joué par un expert (*participatif*). Pour aller plus loin dans la rationalisation de la modélisation, nous exhibons les biais computationnels induits par la traduction de modèles conceptuels en modèles computationnels dans le cas de modèles de peuplements d'agents. Nous montrons que nos modèles sont très sensibles aux différents biais étudiés. Il est nécessaire que les thématiciens connaissent les sources de biais et que l'outil de simulation soit spécialisé afin de permettre : 1° d'explicitier le modèle, 2° de garantir les propriétés identifiées et 3° de permettre des analyses systématiques. C'est ce que nous faisons dans notre simulateur. Enfin, ce travail entre dans le cadre du projet ANR *Chaloupe*. L'enjeu est d'envisager la co-viabilité de peuplements écologiques et de leur exploitation par des groupes sociaux (e.g. pêcheries). Nous présentons les résultats de simulation obtenus validant la société de modèles qui permet une modélisation souple et rigoureuse.

Mots-clés

Système multi-agents ; modélisation, simulation numérique ; informatique

Abstract

We study the modelling process for the simulation of complex systems. The modelling and simulation processes are complex, therefore the modelling has to be rationalized and control gained over it. Our proposition is to conceive the complex model as a society of models in interaction. We apply the *pattern oriented* approach to rationalize the modelling by focusing it on the data. We use the agent approach to distribute the observation and control in the global model. The society of models thus enables the emergence of the patterns through the agents' activities. The *artefacts*, applied to the modelling, support the data, the patterns and the agents' interactions. The coupling of the agents is therefore done through the *artefacts*, which is different from an approach based on events. We identify three types of agents : 1° the *model-agents* that execute the model of a phenomenon ; 2° the observer-agents that build up observables, 3° the controller-agents that locally control the global model. The agents are also able to adapt to their environment using operations to modify their perceptions and actions. The complex model can be therefore incrementally built, the agents' behaviors can easily be modified, and an agent can be added or removed for the global model without impact on the structure of the other agents. At last, we identify four types of behaviors for *model-agents* : 1° based on differential equations (*intensional*) ; 2° individual based (*emergent*) ; 3° based on raw data or pre-calculated (*extensional*) ; 4° played by an expert (*participatory*). To go even further in the rationalization of the modelling, we exhibit computational biases induced by the translation of conceptual models of agent populations into computational models. We show that our models are very sensitive to the studied biases. It is necessary for thematisticians to know the sources of those biases and the simulation tool must be specialized in order to : 1° explicit the model, 2° guarantee the identified properties and 3° make systematic analysis. This is what we have done in our simulator. Finally, this work takes place in the ANR project *Chaloupe*. The challenge here is to envision the co-viability of ecological populations and of their exploitations by social groups (e.g. fisheries). We show our simulation results, confirming by this mean that the society of models enables a flexible and rigorous modelling.

Keywords

Multi-agent systems ; modelling, numerical simulation ; computer science