



HAL
open science

Intégrer des modèles de viabilité dans les outils d'aide à la gestion participative de territoires protégés

Wei Wei

► To cite this version:

Wei Wei. Intégrer des modèles de viabilité dans les outils d'aide à la gestion participative de territoires protégés. Autre. Université Blaise Pascal - Clermont-Ferrand II, 2012. Français. NNT : 2012CLF22313 . tel-00818900

HAL Id: tel-00818900

<https://theses.hal.science/tel-00818900>

Submitted on 29 Apr 2013

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.

Pour mieux
affirmer
ses missions,
le Cemagref
devient Irstea



N° d'ordre : D.U : 2313

EDSPIC : 596

UNIVERSITE BLAISE PASCAL

U.F.R. Sciences et Technologies

ÉCOLE DOCTORALE DES SCIENCES POUR L'INGENIEUR

THESE

présentée pour obtenir le grade de

DOCTEUR D'UNIVERSITE

Spécialité : Informatique

par **Wei WEI**

Master Professionnel en Informatique décisionnelle et logicielle

**Intégrer des modèles de viabilité dans les
outils d'aide à la gestion participative de
territoires protégés.**

Soutenue publiquement le 11 décembre 2012 devant la commission d'examen.

Rapporteurs : **Patrick SAINT-PIERRE**
Directeur de recherche, Université Paris-Dauphine, Paris
Jean-François PERROT
Professeur émérite, Université Pierre et Marie Curie, Paris

Directeur de thèse : **Guillaume DEFFUANT**
Directeur de Recherche, Irstea, Clermont-Ferrand

Examineurs : **Jean-Pierre BRIOT**
Directeur de Recherche, Université Pierre et Marie Curie/CNRS, Brésil
Raoul MEDINA
Maître de Conférences, Université Blaise Pascal, Clermont-Ferrand

Encadrantes : **Isabelle ALVAREZ**
IPEF, Université Pierre et Marie Curie/Irstea, Paris
Sophie MARTIN
Chargée de Recherche, Irstea, Clermont-Ferrand

Remerciements

Je tiens tout d'abord à remercier le Conseil Régional d'Auvergne pour le co-financement de cette thèse.

Je souhaite aussi adresser mes sincères remerciements à toutes les personnes qui ont participé de près ou de loin à sa réalisation de par leur soutien et leur conseil. En effet, je n'aurais jamais pu réaliser ce travail sans cette aide si précieuse, qu'ils en soient remerciés.

Je tiens à remercier également mon directeur de thèse, M. Guillaume Deffuant, pour la confiance qu'il m'a apporté lors de ce travail doctoral dans son laboratoire. Il s'est montré dès le premier jour, accueillant, chaleureux et disponible. Je le félicite pour ses qualités humaines tout à fait hors du commun qui ont été des ingrédients nécessaires au bon déroulement de ma thèse.

Je remercie très vivement mes deux encadrantes, Mme Isabelle Alvarez et Mme Sophie Martin pour leur attention de tous les instants, pour leurs multiples conseils et pour toutes les heures qu'elles ont consacrées à diriger cette recherche. Je les remercie encore plus particulièrement pour le temps qu'elles ont consacré à la relecture de mon manuscrit.

Je tiens également à remercier M. Jean-François Perrot et M. Patrick Saint-Pierre d'avoir accepté de jouer le rôle de rapporteur de ma thèse. L'attention qu'ils ont portée à ce document et leurs suggestions m'ont permis d'améliorer grandement ce manuscrit.

Je remercie cordialement les membres du jury M. Jean-Pierre Briot et M. Raoul pour l'intérêt qu'ils ont porté à ce travail.

Je voudrais également remercier Benoît Gandar, Claire Bernard, Maxime Lenormand, Nicolas Dumoulin pour m'avoir aidé à corriger ce manuscrit.

Je remercie chaleureusement Thomas Guélon pour avoir joué le rôle de taxi, et pour tous les documents importants qu'il m'a aidé à écrire au cours ces trois années de thèse.

Je remercie tous les membres actuels et anciens du projet SimParc de m'avoir fourni une plate-forme remarquable, et surtout remercier Gustavo Melo, Jean-Pierre Briot et Marta Irving pour leur accueil chaleureux lors de mon voyage au Brésil.

Je n'oublie pas de remercier aussi Xia Peng pour son travail pendant son stage, il m'a fait découvrir plusieurs facettes de l'encadrement et a une part dans ce travail de thèse.

Cette thèse a été réalisée au sein du laboratoire d'ingénierie des systèmes complexes. Je tiens à remercier tous ses membres pour les moments de bonne humeur, qui ont fait que ces trois années ont été si agréables à vivre : Bruno, Clarisse, Charles, Emilie, Franc, Jean-Denis, Sylvie et Thierry.

Ces remerciements ne seraient pas complets sans une pensée pour mes amis de longue date : Jiang, Jing, Song, Thierry et Xiao. Merci de m'avoir aidé et encouragé, et de m'avoir changé les idées quand j'en avais besoin.

Mes dernières pensées iront vers ma famille, et surtout mes parents et ma femme, qui m'ont permis de poursuivre mes études jusqu'à aujourd'hui.

Résumé

Un défi important pour la gestion participative est de gérer des ressources renouvelables et la biodiversité. Dans le cadre de l'aide à la décision pour la gestion participative des territoires protégés, nous proposons dans cette thèse un agent expert qui fournit des informations sur la viabilité des solutions envisagées pour aider les joueurs à prendre une décision. C'est une nouvelle application de la théorie de la viabilité qui propose des concepts et méthodes pour contrôler un système dynamique afin de le maintenir dans un ensemble de contraintes de viabilité.

Tout d'abord, nous avons étudié en quoi la théorie de la viabilité est particulièrement adaptée à la problématique de la gestion durable. L'analyse de viabilité prend en compte à la fois les comportements asymptotiques et transitoires qui doivent tous deux être considérés dans le cadre de la gestion de systèmes environnementaux. C'est l'intérêt de cette étude qui motive l'introduction d'un agent "viabilité" dans le cadre d'aide à la gestion participative.

Nous décrivons ensuite un algorithme d'approximation du noyau de viabilité qui se concentre sur la réduction du temps de calcul. Cet algorithme utilise une procédure de classification du type "plus proche voisin", qui satisfait les conditions nécessaires pour garantir la convergence asymptotique de l'approximation. Nous comparons les résultats obtenus pour des problèmes de viabilité pour lesquels les noyaux théoriques ont été calculés. La fidélité de cet algorithme est toujours supérieure à 90 pourcents. Le temps de calcul, relativement rapide, est très intéressant dans l'objectif d'intégrer ces calculs de noyaux dans des outils d'aide à la décision.

Nous développons également un support logiciel qui est facile à utiliser par les joueurs pour les aider à analyser le problème de viabilité. Les joueurs définissent leur problème de viabilité grâce à une interface graphique utilisateur, puis lancent les simulations. Le logiciel permet aussi de visualiser les noyaux de viabilité et d'observer les trajectoires en 2D et 3D.

Enfin nous étudions un agent expert viabilité dans un jeu de SimParc pour observer ses influences. Nous présentons des résultats fournis par l'agent expert viabilité en utilisant un modèle abstrait qui est axé sur la biodiversité. Puis nous présentons une expérimentation avec et sans agent expert viabilité.

Mots-clés : gestion participative, aide à la décision, agent expert, théorie de la viabilité, système dynamique, gestion de systèmes environnementaux.

Abstract

An important challenge for participatory management is to manage renewable resources and biodiversity. In the framework of the decision support for participatory management of protected areas, we propose in this thesis an expert agent who provides information on the viability of the proposed solutions to help players to make a decision. This is a new application of the viability theory that provides concepts and methods for controlling a dynamic system in order to maintain the system in a set of viability constraints.

Firstly, we study how the viability theory is particularly adapted to the problem of sustainable management. Sustainability analysis takes into account both the asymptotical dynamics and transient behaviors that have both to be considered in the context of environmental management systems. This study motivates the introduction of agent "viability" in the context of support participatory management.

Then, we describe an algorithm of approximation of the viability kernel, which focuses on the reduction of computing time. This algorithm uses the "nearest neighbors" as a classification method, which satisfies the conditions guaranteeing the asymptotic convergence of the approximation. We compare the results obtained in cases where the theoretic viability kernel could be calculated. The accuracy of this algorithm is always higher than 90 percent. The computing time is relatively short, which is very interesting in the aim of integrating these calculations of viability kernel in tools of decision support.

We also develop a software that is easy to use by players to help analyze the problem of sustainability. Players define their sustainability problem through a graphical user interface and launch simulations. The software also allows users to visualize the viability kernels and to observe the trajectories in 2D and 3D.

Finally we study a viability expert agent in a game SimParc to observe these influences. We present the results provided by the viability expert agent using an abstract model which focuses on biodiversity. Then we present an experiment with and without viability expert agent.

Keywords : participatory management, decision support, expert agent, viability theory, dynamical system, environmental management systems.

Table des matières

Introduction	1
I Contexte et Théorie	7
1 Viabilité	9
1.1 La théorie de la viabilité	10
1.1.1 Définitions	10
1.1.2 Noyau de viabilité	10
1.1.3 Formalisme mieux adapté à la gestion environnementale . . .	12
1.2 Le bassin de capture	13
1.3 Contrôleur lourd de viabilité	14
1.4 Applications	15
2 Le Jeu SimParc	17
2.1 Le Projet SimParc	18
2.1.1 Contexte du projet	18
2.1.2 Approche	18
2.2 Le jeu de rôle SimParc	19
2.2.1 L'objectif du jeu	19
2.2.2 Les dispositions générales sur le zonage au Brésil	20
2.2.3 Déroulement du jeu	22
2.3 Le support de l'architecture du Jeu SimParc	24
2.3.1 L'architecture du jeu SimParc	24
2.3.2 Agents artificiels	25
II Viabilité et durabilité : étude théorique	31
3 Analyse de viabilité	33
3.1 Limites de l'analyse asymptotique	34
3.1.1 Modèle de l'impact du tourisme sur la qualité de l'environnement	34
3.1.2 L'information fournie par l'étude asymptotique et le diagramme de bifurcation	36
3.1.3 Les informations manquantes	38
3.2 La théorie de la viabilité pour l'information manquante	42
3.2.1 Le noyau de viabilité pour l'information spatiale	42
3.2.2 Le bassin de capture pour l'information temporelle	44
3.3 Le concept de viabilité et l'analyse de viabilité	46

3.3.1	La dynamique non contrôlée	46
3.3.2	La dynamique contrôlée	46
3.4	Conclusion	52
III	Application	55
4	Algorithme d'approximation du noyau de viabilité	57
4.1	L'approximation du noyau de viabilité discret	58
4.1.1	Contexte et notations	58
4.1.2	Les étapes de l'algorithme	59
4.1.3	Le théorème de convergence de l'algorithme	59
4.2	Choix de la procédure de classification	60
4.2.1	Les fonctions de classification utilisées	60
4.2.2	Démonstration de la convergence	60
4.3	Implémentation et Tests	62
4.3.1	L'utilisation de la méthode de Runge-Kutta	62
4.3.2	Modèle de croissance de population dans un espace limité	62
4.3.3	Modèle de consommation	63
4.4	Conclusion	64
5	Support logiciel	69
5.1	Les motivations	70
5.2	Conception de l'application	70
5.2.1	Utilisation Maven	71
5.2.2	Diagramme des paquetages de l'application	71
5.2.3	Description des paquetages	72
5.3	Interface Graphique Utilisateur	76
5.3.1	Visualisation du noyau de viabilité	77
5.3.2	Observation des trajectoires	79
5.4	Conclusion	82
6	L'expérimentation	85
6.1	Définition du problème de viabilité	86
6.2	Modèle dynamique de fonctionnement du problème	87
6.2.1	Description du modèle	87
6.2.2	Développement du modèle	88
6.3	Les résultats fournis par l'agent expert viabilité	90
6.3.1	Comparaison des noyaux de viabilité	90
6.3.2	Visualisation de l'évolution viable	91
6.4	L'expérimentation de l'agent expert viabilité	93
6.4.1	Conditions d'expérimentation	93
6.4.2	Description de l'expérimentation	95
6.5	Conclusion	98

Table des matières	ix
Conclusion	101
Bibliographie	105
Annexe	113
A Ficher pom.xml du projet	113

Table des figures

1.1	Exemple de noyau de viabilité.	11
1.2	Exemple de bassin de capture.	14
2.1	Rôles dans SimParc.	20
2.2	Carte des différentes zones du parc et leurs ressources.	21
2.3	Les six étapes du jeu SimParc.	23
2.4	L'architecture générale de SimParc.	24
2.5	L'interface de négociation de SimParc.	25
2.6	L'interface de sélection des modes de gestion.	26
2.7	L'interface de la configuration d'une nouvelle session.	27
3.1	Scénarios de développement du tourisme	36
3.2	Scénarios de développement du modèle du tourisme (3.1) avec $a = 6.8$ et $\epsilon = 0.1$	39
3.3	Scénarios de développement du tourisme avec $a = 6$ et $\epsilon = 0.1$	40
3.4	Scénarios de développement du tourisme avec $a = 6$ et $\epsilon = 0.13$	41
3.5	Scénarios de développement du tourisme avec $a = 6$ et $\epsilon = 0.1$ et point initial est ($T = 0.26$, $E = 1$, $C = 0.26$).	41
3.6	Les noyaux de viabilité de la dynamique (3.1) avec $\epsilon = 0.1$	43
3.7	Les noyaux de viabilité de la dynamique (3.1) avec $\epsilon = 0.13$	44
3.8	Les bassins de capture du noyau de la figure 3.6 (c).	45
3.9	Le noyau de la dynamique (3.3).	48
3.10	Les trajectoires des deux évolutions.	49
3.11	Deux comparaisons du noyau de viabilité.	51
4.1	L'exemple de plus proche voisin.	61
4.2	Les approximations du noyau de viabilité du problème de population.	65
4.3	Les approximations du noyau de viabilité du problème de consommation.	67
5.1	Diagramme de paquetage.	71
5.2	Diagramme de classes du paquetage <i>Model</i>	72
5.3	Diagramme de classes du paquetage <i>GeometricBounds</i>	74
5.4	Diagramme de classes du paquetage <i>Algo</i> et <i>Problem</i>	75
5.5	Diagramme de classes du paquetage GUI.	76
5.6	La fenêtre principale du logiciel.	78
5.7	Fenêtre de visualisation de l'approximation du noyau de viabilité en 2 dimensions.	79
5.8	Fenêtre de visualisation de l'approximation du noyau de viabilité en 3 dimensions.	80

5.9	Le panneau de commande pour modifier l'échelle de l'image.	80
5.10	Le panneau des données.	81
5.11	L'exemple de la trajectoire.	82
5.12	L'évolution par dimension.	83
6.1	Deux ensembles de contraintes et leurs noyaux de viabilité.	91
6.2	Les intersections des contraintes.	92
6.3	Visualisation des évolutions viables.	93
6.4	L'évolution par dimension.	94
6.5	L'expérimentation de l'agent expert viabilité à Paris.	95
6.6	La carte de l'unité de paysage 3.	95

Liste des tableaux

4.1	Comparaison avec modèle de population.	64
4.2	Comparaison avec modèle de population.	66
6.1	Ensemble de contraintes des différents modes de gestion recommandés par le comité du Parc.	87
6.2	Les valeurs des paramètres du modèle 6.1.	89

Introduction

Cette thèse consiste à examiner les possibilités de prise en compte des modèles de viabilité, dans le processus de gestion participative (projet SimParc¹ du LIP6² et de l'UFRJ³). Il s'agit de concevoir et d'étudier un agent expert en théorie de la viabilité dans le cadre de la plate-forme SimParc.

Contexte

Un des objectifs du laboratoire dans lequel je travaille, le LISC⁴, est de promouvoir l'utilisation de modèles de l'environnement dans le cadre des problèmes de gestion ou de décision liées à l'environnement. Quand ces modèles sont des équations différentielles, ils peuvent être utilisés par la théorie mathématique de la viabilité. La théorie mathématique de la viabilité initiée dans les années 1990 par [Aubin, 1991], étudie la compatibilité entre des dynamiques et des contraintes. Le concept essentiel est celui de noyau de viabilité qui rassemble l'ensemble des états du système à partir desquels il existe au moins une fonction de contrôle qui assure le respect des contraintes indéfiniment. Les domaines dans lesquels la théorie de viabilité est utilisée sont très variés (économie, sciences cognitives, théorie des jeux, biologie, automatique ..), citons en bio-économie et en écologie [Béné *et al.*, 2001, Bonneuil, 2003, Mullon *et al.*, 2004]. Toutes ces applications encore restent au niveau théorique, les résultats obtenus sur les stratégies viables n'ont pas été testées, exceptées les travaux sur l'affinage du Camembert ([Sicard *et al.*, 2012]). Notre démarche s'inscrit dans cette volonté d'être en mesure de confronter au réel ces scénarios théoriquement viables et de les intégrer aux processus de décisions.

Les jeux sérieux sont une approche nouvelle et efficace pour explorer et tester des possibilités d'évolutions dans un contexte réaliste mais sans coût ni risque [Susi *et al.*, 2007]. Ils nous permettent de quitter le domaine des applications théoriques pour se rapprocher des applications de terrain dans le domaine de l'environnement. En effet, ces jeux sont un substitut intéressant à l'expérience directe dans le monde réel ou sur des infrastructures réelles, qui peuvent permettre des apprentissages relativement rapides et sans danger [Warmerdam *et al.*, 2006].

Dans le domaine de la gestion de la biodiversité [De Lara et Doyen, 2011], un défi important concerne la gestion d'aires protégées comme les parcs nationaux, qui sont soumis à de nombreuses pressions sur l'utilisation et l'accès aux ressources. La

-
1. Simulation participative de la gestion de Parcs
 2. Laboratoire d'informatique de Paris 6
 3. Université fédérale de Rio de Janeiro
 4. Laboratoire d'Ingénierie des Systèmes Complexes

gestion des territoires est perçue comme un compromis entre différents points de vue, correspondant aux différentes fonctions d'usage (habitat, transport, agriculture vivrière ou intensive, loisir, tourisme, conservation de la biodiversité, etc.). Le projet SimParc [Briot *et al.*, 2010] est un projet de recherche interdisciplinaire franco-brésilien. L'inspiration initiale de cette recherche est la modélisation d'accompagnement⁵ [Barreteau, 2003]. SimParc appartient au domaine de l'aide à la gestion participative et a pour but d'aider les différents acteurs à créer une compréhension collective des conflits et à négocier des stratégies pour y faire face en utilisant une plate-forme qui s'appelle aussi SimParc. Cette plate-forme a pour objectif la formation de gestionnaires de parcs nationaux au Brésil, ainsi que l'exploration de scénarios. Les travaux en cours autour de cette plate-forme permettent de l'utiliser en vue d'expérimentations réalistes répétées avec un jeu de rôle. En effet, le jeu est basé sur un processus de négociation qui se déroule au sein du conseil du parc. Les différentes parties prenantes participent au jeu et sont donc des "joueurs". Le gestionnaire du parc agit comme un arbitre dans le jeu, il prend une décision finale sur les types de conservation à attribuer à chaque unité de paysage. Les autres joueurs font des propositions. Il y a également des agents artificiels qui peuvent remplacer certains joueurs [Briot *et al.*, 2011].

Problématique

La gestion des territoires dans lesquels les tensions entre les différentes fonctions d'usage (agriculture vivrière ou intensive, habitat, conservation de la biodiversité, etc.) sont manifestes s'orientent, sous la pression sociétale, vers un processus de décision où l'ensemble des acteurs concernés est impliqué. Dans le passé, ce processus comprenait principalement les représentants de l'état et des experts mandatés pour émettre des recommandations. L'alternative la plus considérée est la gestion participative, processus de décision dans lequel l'ensemble des acteurs élabore la définition du problème et sa solution. Ce mode de décision met l'accent sur la connaissance de chaque acteur, et l'utilisation de modèles formels est plus difficile dans ce cadre. Le problème que nous considérons dans cette thèse est la possibilité d'utiliser, dans le cadre de l'assistance à la gestion participative des territoires, des études de viabilité de modèles de l'environnement soumis à de multiples contraintes. L'objectif est d'amener à une réflexion sur les possibilités d'intégrer des modèles de viabilité dans un processus d'aide à la gestion participative, de manière à apporter aux acteurs des connaissances liées à la viabilité des systèmes.

L'enjeu de la thèse est de faciliter la prise en compte des travaux sur l'analyse de la viabilité des modèles de l'environnement (systèmes écologiques et sociaux). Il s'agit d'examiner la faisabilité de l'utilisation de ces connaissances et leur impact sur la qualité de la solution de gestion, de pouvoir fournir des résultats de viabilité dans un délai acceptable, de permettre aux utilisateurs de formaliser des problèmes

5. ComMod : La modélisation comme outil d'accompagnement

de viabilité et de leur présenter des résultats compréhensibles.

Notre thèse

La thèse est une réflexion sur l'application de la théorie de la viabilité dans le domaine de l'aide à la décision. L'objectif de la thèse est de montrer l'intérêt de construire un agent expert en viabilité, qui permette de fournir des résultats de viabilité pour aider les utilisateurs à prendre en compte ce type de connaissances dans la prise de décision. Idéalement, chaque utilisateur devrait pouvoir faire appel à l'agent expert viabilité pour lui soumettre un problème de viabilité. L'agent expert viabilité calcule des noyaux de viabilité, puis fait l'analyse de viabilité. L'agent expert peut fournir les informations susceptibles d'intéresser les utilisateurs : viabilité actuelle du système (noyau vide ou non), difficulté de contrôle dans l'avenir (étude géométrique), robustesse, etc. Puis l'agent expert restitue ces résultats à l'utilisateur qui a fait la demande. La communication entre l'agent expert et l'utilisateur se fait par une interface, qui permet à l'utilisateur de définir le problème de viabilité qu'il veut poser et fixer un certain nombre de paramètres sur le problème à résoudre.

Plusieurs facettes ont été réalisées pendant la thèse :

- Etude théorique des apports de la viabilité.

Une étude théorique des apports de la viabilité pour la gestion de systèmes environnementaux a été réalisée. Cette étude montre l'intérêt de la prise en compte des comportements transitoires, qui complète l'analyse traditionnelle asymptotique. Elle montre, en s'appuyant sur des modèles connus de la littérature, que la notion de développement durable est mieux prise en compte par la viabilité. Cette étude est la base sur laquelle s'appuient les travaux sur l'agent expert.

- Définition du problème de viabilité.

Une interface graphique utilisateurs a été proposée qui permet aux utilisateurs de construire les éléments nécessaires pour résoudre un problème de viabilité. Les utilisateurs peuvent choisir les modèles prédéfinis, ainsi que modifier les paramètres ou les contraintes de modèle.

- Etude de viabilité.

L'étude de viabilité qui contient le calcul de noyau, le calcul de bassin de capture, l'évolution du système, est réalisée suivant les contraintes d'instanciation qui sont proposées par les utilisateurs et suivant leurs objets d'intérêt. Une interface a été réalisée pour intégrer à terme l'ensemble de ces opérations. Le calcul de noyau et l'évolution du système ont été implémentées et intégrées. (Les autres fonctionnalités sont effectuées manuellement.)

- Présentation des résultats.

La présentation des résultats de l'étude de viabilité est un élément important. Les acteurs ne connaissent pas nécessairement les notions de la viabilité, nous devons donner des résultats compréhensibles essentiellement sous forme graphique. Une version de visualisation graphique a été effectuée.

- Calcul des résultats en temps acceptable.

L'agent expert viabilité doit fournir aux utilisateurs des résultats dans un délai acceptable. Une version simplifiée de l'algorithme de viabilité permettant d'approximer le noyau a été proposée. Cet algorithme satisfait les conditions nécessaires pour garantir la convergence asymptotique de l'approximation.

- Application au cas SimParc.

Pour évaluer l'impact de cet agent expert, nous testons avec le projet SimParc qui a proposé une plate-forme pour créer une compréhension collective des conflits. Nous avons comparé des sessions de jeu avec ou sans agent expert. Une session de test a été effectuée dans ce but. Elle a montré l'importance de la présentation visuelle des résultats.

Organisation du document

La thèse est divisée en trois parties, la partie 1 contient deux chapitres, la partie 2 contient un chapitre et la partie 3 contient 3 chapitres. La plupart des travaux de la thèse ont été publiés ou soumis à des journaux, ou publiés dans des actes de conférence.

Dans la première partie, nous décrivons la théorie de la viabilité et le projet SimParc que nous utilisons pour tester l'agent expert viabilité. Dans le chapitre 1, nous introduisons la théorie de la viabilité et ses principaux outils, nous donnons également quelques applications de la théorie de la viabilité dans plusieurs domaines. Dans le chapitre 2, nous présentons d'abord le contexte et l'approche du projet SimParc. Nous décrivons ensuite le jeu de rôle de SimParc qui est basé sur un processus de négociation. Nous présentons aussi le support de l'architecture du jeu et les différents types d'agents artificiels qui sont en train d'être intégrés dans le jeu. Ce travail a été publié dans les actes de la conférence *Culture and Computing 2011* [Briot *et al.*, 2011].

La seconde partie est une étude théorique qui a permis de montrer que la théorie de la viabilité est adaptée pour répondre aux problèmes de gestion durable de systèmes socio-écologiques pour lesquels les comportements à très longs termes (asymptotiques) ne sont pas assez fiables pour constituer réellement une base de négociation : forêts, espaces protégés, etc. L'étude théorique a montré que les comportements transitoires peuvent être très différents du comportement asymptotique

dans une perspective de gestion durable. Elle a montré, dans le cadre d'un modèle de développement touristique, que la théorie de la viabilité propose des concepts et outils utiles pour caractériser ces comportements transitoires du point de vue de la gestion durable.

Ce sont les résultats présentés dans cette seconde partie qui justifient les travaux de cette thèse en vue d'étudier et de faciliter le développement d'agents experts en viabilité pour l'aide à la gestion durable. Ce travail a été présenté dans *Environment conference 2011* et accepté dans la revue *Ecological modelling*.

Dans la troisième partie, nous décrivons les travaux que nous avons réalisés pour construire l'agent expert viabilité. Dans le chapitre 4, en nous appuyant sur un algorithme d'approximation du noyau de viabilité avec procédure de classification, nous proposons une procédure de classification dont l'objectif est de diminuer le temps d'attente des utilisateurs. Ce travail a été présenté dans *Le dix-huitième congrès francophone sur la Reconnaissance des Formes et l'Intelligence Artificielle (RFIA '12)* et publié dans les actes de cette conférence [Wei *et al.*, 2012b].

Dans le chapitre 5, en collaboration avec un stagiaire [Xia, 2011], nous développons un support logiciel en java qui prévoit les modules permettant d'étudier un problème de viabilité. Actuellement, le logiciel permet de calculer le noyau de viabilité et d'observer l'évolution du système en donnant un point de départ. Le logiciel fournit aussi une interface graphique utilisateur et une visualisation en 2 et 3 dimensions.

Dans le chapitre 6, nous montrons comment un agent expert viabilité pourrait être intégré dans le jeu Simparc. Nous détaillons le modèle simple de l'environnement que nous avons utilisé pour tester l'intérêt de l'agent expert viabilité. Enfin nous présentons les expérimentations réalisées. Ce travail a été présenté dans *IADIS International Conference* et publié dans les actes de la conférence [Wei *et al.*, 2012a].

Première partie

Contexte et Théorie

Viabilité : formalisme adapté à la gestion environnementale

Sommaire

1.1	La théorie de la viabilité	10
1.1.1	Définitions	10
1.1.2	Noyau de viabilité	10
1.1.3	Formalisme mieux adapté à la gestion environnementale	12
1.2	Le bassin de capture	13
1.3	Contrôleur lourd de viabilité	14
1.4	Applications	15

Nous présentons dans ce chapitre la théorie mathématique et numérique de la viabilité concernant l'étude de systèmes dynamiques contrôlés soumis à un ensemble de contraintes. Cette notion fournit un formalisme qui est mieux adapté à la gestion environnementale. Cette démarche est une application de la théorie de la viabilité et a pour but de créer un agent expert en viabilité pour apporter une aide à la décision.

Dans la section 1.1, nous rappelons quelques définitions importantes de la théorie de la viabilité. Elles contiennent le noyau de viabilité qui représente l'ensemble de tous les états initiaux pour lesquels il existe au moins une fonction de contrôle qui permette au système de rester dans l'ensemble des contraintes ; le bassin de capture qui représente l'ensemble de tous les états pour lesquels il existe au moins une fonction de contrôle qui permette d'atteindre la cible donnée ; les contrôleurs de viabilité qui existent et quelques applications dans différents domaines. Dans la section 1.1.3, nous expliquons en quoi le cadre théorique de la viabilité nous paraît particulièrement bien adapté à la problématique de gestion dans le domaine de l'environnement. Premièrement, il permet de se concentrer sur la définition des contraintes, donc sur la définition des états désirables. Deuxièmement, la théorie de la viabilité propose des politiques d'actions possibles, donc fournit des stratégies flexibles pour la gestion environnementale.

1.1 La théorie de la viabilité

1.1.1 Définitions

La théorie de la viabilité [Aubin, 1991] considère un système dynamique défini par son état $\vec{x}(t) \in X \subset \mathbb{R}^N$ et suppose que son évolution peut être influencée par un contrôle $\vec{u}(t)$:

$$\begin{cases} \vec{x}'(t) = f(\vec{x}(t), \vec{u}(t)) \\ \vec{u}(t) \in U(\vec{x}(t)) \end{cases} \quad (1.1)$$

L'ensemble des contrôles admissibles peut dépendre de l'état du système, $\vec{u}(t)$ est choisi dans un sous-ensemble $U(\vec{x}(t)) \subset \mathbb{R}^q$ qui dépend de l'état du système au temps t .

De tout point initial, il peut exister plusieurs évolutions possibles suivant les différentes valeurs des contrôles choisies au cours du temps $t \rightarrow u(t)$. [Aubin, 1991] a défini la notion d'état viable :

Définition 1.1 *Un état initial $\vec{x}(0)$ pour lequel il existe au moins une évolution $\vec{x}(t)$, satisfaisant le système 1.1 avec $\vec{x}(0) = \vec{x}_0$, qui reste dans un ensemble de contraintes $K \subset \mathbb{R}^N$ indéfiniment est un état viable et $\vec{x}(t)$ est appelée évolution viable.*

Les méthodes et outils de la théorie de la viabilité permettent de rechercher les conditions (décisions, états) dans lesquelles les contraintes opérationnelles seront toujours satisfaites et donc dans lesquelles le système pourra fonctionner de manière durable.

1.1.2 Noyau de viabilité

L'ensemble des contraintes de viabilité est un sous-ensemble de X noté K . Le *noyau de viabilité* de K (voir figure 1.1) pour le système (1.1) est défini par :

$$Viab(K) = \{ \vec{x} \in K \mid \exists \vec{u}(\cdot) \mid \vec{x}(t) \in K \forall t \in [0, +\infty[\}. \quad (1.2)$$

$Viab(K)$ rassemble les états dans X à partir desquels on trouve toujours une fonction de contrôles $\vec{u}(t)$ qui permette au système de rester dans l'ensemble des contraintes K . Le noyau de viabilité de K est le plus grand sous-ensemble viable de K pour le système contrôlé (1.1) (la démonstration est dans [Aubin, 1991]).

La méthode principale pour résoudre un problème de viabilité est de déterminer le noyau de viabilité. Les théorèmes de viabilité sont donnés par [Aubin, 1991], ils permettent de déterminer un noyau de viabilité sans considérer l'exploration des combinaisons de contrôles sur l'ensemble du temps. Ces théorèmes sont valides quand les systèmes considérés sont des *systèmes contrôlés Marchaud*.

Définition 1.2 *Soit $U : X \rightarrow Y$ et $f : X \times Y \rightarrow X$. (U, f) est un système contrôlé Marchaud si*

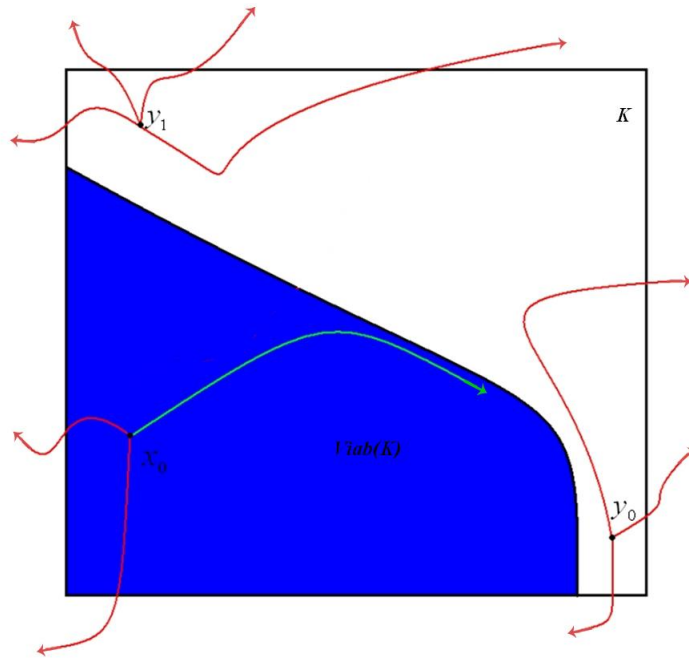


FIGURE 1.1 – Représentation schématique d'un noyau de viabilité. x_0 , y_0 et y_1 sont des états initiaux appartenant à l'ensemble des contraintes K , la trajectoire verte correspond à une évolution viable, et les trajectoires rouges correspondent à des évolutions non viables. Le domaine bleu représente le noyau de viabilité.

- $\text{Graph}(U)$ est fermé ;
- f est continue ;
- f et U sont à croissance linéaire¹
- les ensembles $\{f(x, u) \mid u \in U(x)\}$ sont convexes pour tout $x \in \text{Dom}(U)$ ².

Les noyaux de viabilité peuvent être définis analytiquement dans certains cas extrêmement simples comme dans la section 4.3. Plusieurs algorithmes ont été développés pour approcher le noyau de viabilité dans les cas complexes : l'algorithme de [Saint-Pierre, 1994] utilise une discrétisation de l'espace des états sur une grille régulière. Il permet d'obtenir une approximation numérique de cet ensemble en calculant les noyaux discrets finis exacts pour des systèmes dynamiques discrets en temps et en espace associés au problème initial posé sous forme d'inclusion différentielle. Kviar, l'algorithme de [Deffuant *et al.*, 2007] utilise des Support Vector Machines. Ils ont proposé une approche utilisant des méthodes de discrimination. Leur idée est de ne pas conserver les labels de tous les points de la grille en mémoire, seulement les labels d'une petite partie d'entre eux sélectionnés par la méthode de discrimination pour construire la fonction de discrimination. L'avantage est de gagner de l'espace de mémoire. [Coquelin *et al.*, 2007] proposent

1. F est à croissance linéaire si $\exists c > 0$ tel que $\forall x \in X \quad \sup_{v \in F(x)} \|v\| \leq c(\|x\| + 1)$
 2. $\text{Dom}(U) = \{x \in X \mid U(x) \neq \emptyset\}$

une approche de programmation dynamique qui ne permet pas de gain en espace mémoire mais en temps de calcul. Ils ont défini le noyau comme l'ensemble des états pour lesquels la fonction valeur est en dessous d'un certain seuil. Mais la valeur de ce seuil n'est pas évidente à définir.

La limite des algorithmes d'approximation du noyau de viabilité est : pour approcher les noyaux de viabilité de systèmes continus, ils calculent les noyaux exacts de systèmes discrets définis sur une grille finie. La taille de cette grille croît exponentiellement avec la dimension, ce qui pose des problèmes soit en mémoire, soit en temps de calcul. Et donc limite l'utilisation pratique aux petites dimensions. D'autre part, les algorithmes utilisent souvent des approximations par l'extérieur qui entraînent une surestimation des noyaux réels.

1.1.3 Formalisme mieux adapté à la gestion environnementale

Un problème centré sur les contraintes

Les problèmes de gestion durable doivent concilier les trois piliers économiques, écologiques et sociaux. Dans ces conditions, il est souvent difficile de construire un critère d'optimalité. Il est plus aisé de préciser des seuils souhaités qui constitueront l'ensemble des contraintes à satisfaire. Ainsi, la théorie de la viabilité permet de ne faire qu'un nombre réduit d'hypothèses sur le futur. On fixe uniquement des bornes à l'intérieur desquelles l'état du système doit rester ou des zones à éviter. En pratique, on ne choisit pas de critère d'optimisation ou de scénario d'action prédéfini, mais on choisit les zones de fonctionnement souhaitable et les actions possibles. Cette approche est en particulier appliquée pour les études sur le changement climatique [Petschel-Held *et al.*, 1999].

Dans notre gestion participative du parc, selon la théorie de la viabilité, les acteurs qui veulent faire fonctionner l'agent expert viabilité sont invités à définir ce qu'ils appellent l'ensemble des états désirables pour le parc ou une zone protégée. Le choix des variables d'état du parc et des contrôles admissibles sont également définis par l'acteur, parmi un ensemble de choix possibles. Par exemple, dans un parc, plusieurs actions sont possibles : l'investissement, le nettoyage, la plantation, etc.

Des politiques d'actions possibles

A partir de points appartenant au noyau de viabilité, il existe au moins une politique d'action qui permet au système de rester dans l'ensemble de contraintes. Mais il peut y en avoir plusieurs évolutions viables, toutes satisfieront les contraintes, mais pas de la même façon. Cette caractéristique non seulement permet de diminuer les conflits entre les acteurs, mais aussi de fournir des stratégies flexibles pour la gestion environnementale.

1.2 Le bassin de capture

Un autre concept important de la théorie de la viabilité est *le bassin de capture* [Aubin, 2002]. Nous abordons un problème apparemment différent, celui d’agir sur un système de manière à ce qu’il atteigne une cible tout en respectant des contraintes. La cible correspond alors à un état souhaité, en sachant que le système se détériore dans une certaine région de l’espace.

Dans le calcul du bassin de capture, la dynamique du système est augmentée d’une dimension qui représente le temps. De la même manière que pour le noyau de viabilité, le bassin de capture permet de définir directement des politiques de contrôle, qui fournissent une évolution qui reste toujours dans l’ensemble des contraintes, tout en atteignant la cible en un temps fini ou prescrit. Il donne également les politiques qui permettent d’atteindre la cible en un temps minimal [Chapel, 2007].

Définition 1.3 *Etant donné un ensemble de contraintes K et un objectif fixé C , le bassin de capture $Capt(K, C)$ regroupe tous les états tels qu’il existe une fonction de contrôle qui permet d’atteindre l’objectif fixé tout en restant dans l’ensemble des contraintes :*

$$Capt(K, C) = \{\vec{x} \in K \mid \exists u(\cdot), \exists T \mid \vec{x}(T) \in C \text{ and } \vec{x}(t) \in K \forall t \in [0, T]\}$$

Les évolutions $\vec{x}(t)$ capturant la cible sont les évolutions viables dans K jusqu’à ce qu’elles atteignent la cible C en temps fini. La figure 1.2 présente un exemple de bassin de capture : y_0 appartient au bassin de capture, il existe une évolution qui atteint la cible tout en restant dans K , tandis que y_1 n’appartient pas au bassin de capture, il n’existe aucune fonction de contrôle qui permette au système d’atteindre la cible tout en restant dans K .

Ce problème est étudié notamment dans le domaine de la navigation de robots mobiles [Spiteri *et al.*, 2000], aussi dans la gestion d’actifs en finance. Par exemple, [Pujal et Saint-Pierre, 2004] ont défini les contraintes sur la valeur du portefeuille, et cherché des contrôles de gestion d’actifs qui permettent d’atteindre l’objectif d’obtention de contrats. [Aubin *et al.*, 2005] ont travaillé sur le problème de la gestion d’un portefeuille d’actif, et déterminé des contrôles qui permettent de contrôler l’évolution du portefeuille pour atteindre l’objectif de rentabilité fixé.

Des algorithmes ont été développés pour approcher des bassins de capture. Par exemple, l’algorithme de [Pujal et Saint-Pierre, 2004] ressemble à l’algorithme de [Saint-Pierre, 1994], il permet de construire des stratégies de couverture garantie ; l’algorithme de [Cardaliaguet *et al.*, 2000] permet d’estimer l’erreur de l’approximation. Les algorithmes de [Delanoue *et al.*, 2009] et [Lhommeau *et al.*, 2011] basés sur l’analyse d’intervalle, sont capables de caractériser une approximation intérieure et extérieure.

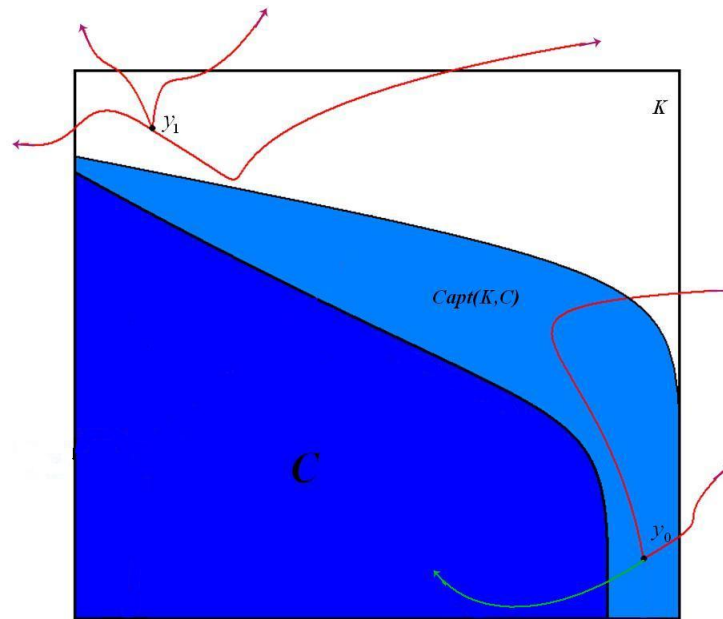


FIGURE 1.2 – Exemple de bassin de capture. La cible C est représentée en bleu foncé. Le bassin de capture $Capt(K, C)$ est représenté en bleu clair. y_0 et y_1 sont des états initiaux de l'ensemble des contraintes K , la trajectoire verte correspond à une évolution viable qui atteint la cible, et les trajectoires rouges correspondent à des évolutions non viables.

1.3 Contrôleur lourd de viabilité

Parmi l'ensemble des solutions viables, l'évolution lourde a été introduite par [Aubin et Frankowska, 1985]. La procédure découle du principe d'inertie, les contrôles gardent toujours les mêmes valeurs si le système ne sort pas des contraintes. Si le système sort des contraintes en conservant les contrôles constants, les valeurs sont modifiées lorsque la frontière du noyau de viabilité est atteinte. Des valeurs de contrôle viables existent nécessairement puisque ce point de la frontière appartient au noyau de viabilité (qui est un fermé lorsque l'ensemble des contraintes l'est aussi (Aubin 1991)). L'évolution viable correspondante sera alors qualifiée d'évolution lourde. L'avantage de cette méthode est très simple à implémenter, [Chapel, 2007] a proposé une procédure dans sa thèse. Sa procédure de contrôle garantit de construire à partir d'un point du noyau de viabilité une trajectoire qui reste indéfiniment dans ce noyau. L'inconvénient de cette méthode est la complexité de calcul. Il oblige à tester les contrôles pour trouver l'action viable, et le temps de recherche croît exponentiellement avec la dimension de l'espace des contrôles.

1.4 Applications

La théorie de la viabilité est appliquée dans de nombreux domaines. Par exemple, en écologie, [De Lara *et al.*, 2007] étudient la stabilité d'un écosystème constitué de merlus et d'anchois dans le golfe de Gascogne, en fonction des niveaux de pêche et du recrutement des espèces. Ils ont retrouvé les configurations qui permettent de maintenir la durabilité du système. En effet, ils ont imposé comme contrainte que la population de chaque espèce reste au-dessus d'un seuil. En génétique, [Bonneuil et Saint-Pierre, 2000] se servent de la notion de noyau de viabilité pour déterminer les fréquences initiales qui conduisent au maintien du polymorphisme. Les facteurs influents comme le facteur de sélection naturelle appelé fitness sont modélisés comme étant des régulateurs car leurs valeurs ne sont pas contrôlées par un agent identifié. En économie, [Aubin *et al.*, 2001] étudient les conditions de pérennité du système de retraites. Le système modélisé est composé des actifs et des inactifs, la dynamique tient compte du fait qu'une partie du revenu des actifs est versée aux retraités. L'objectif est de déterminer des politiques de versement des actifs aux caisses de retraites, qui garantissent un revenu suffisant aux retraités, tout en assurant aux actifs de conserver un pouvoir d'achat garanti. Des questions d'équité et de pouvoir d'achat constituent les contraintes. Le noyau de viabilité de l'ensemble de ces contraintes est composé des états à partir desquels le système des retraites est viable.

Il y a également des applications de la théorie de la viabilité dans la modélisation de problèmes de gestion durable. Par exemple, [Durand *et al.*, 2010] recherchent l'apport des outils de la viabilité dans l'analyse des modèles économiques du développement durable³. [Rapaport *et al.*, 2006] étudient la gestion d'un stock de bois divisé en catégories selon l'âge des arbres. Ils construisent le problème de viabilité avec la variable de contrôle correspondant à l'abattement des arbres, les contraintes sont là aussi d'ordre économique (assurer des revenus grâce à l'abattement des arbres) et écologique (conserver un nombre suffisant d'arbres). Les travaux de [Martinet et Blanchard, 2009] souhaitent satisfaire des contraintes à la fois économique et écologique en utilisant le calcul de noyaux de viabilité. Ces études montrent non seulement que le calcul du noyau de viabilité constitue un indicateur pertinent pour évaluer la durabilité d'une situation, elle montre en plus que l'étude de l'existence d'un noyau non vide en fonction des seuils de contraintes permet d'exhiber des relations entre ces différents seuils. [Bernard, 2011] étudie le cas de la déforestation à Madagascar. Elle a élaboré un modèle mathématique prenant en compte les objectifs environnementaux et économiques de la région de Fianarantsoa, et l'a étudié ensuite dans le formalisme de la théorie de la viabilité.

De nombreux projets sont en cours de développement. Par exemple, le projet de recherche ANR FORGECO⁴ se base sur les principes de gestion intégrée des

3. Cette étude a été réalisée dans le cadre du projet ANR "Déduction".

4. FOret, Gestion et ECOsystème

écosystèmes pour accompagner et organiser l'augmentation des prélèvements de la ressource et une meilleure préservation de la biodiversité et de la qualité des sols. Ce projet traduit une volonté à l'échelle nationale et fait notamment suite à un protocole d'accord signé en 2007 et faisant intervenir les principaux acteurs liés à l'exploitation sylvicole en France : "Produire plus de bois tout en préservant mieux la biodiversité une démarche territoriale concertée dans le respect de la gestion multifonctionnelle des forêts". La théorie de la viabilité y est mobilisée pour étudier la résilience et la viabilité des scénarios de gestions sylvicoles proposés par le projet. Elle est également utilisée pour imaginer de nouveaux scénarios de gestions qui respectent des contraintes d'ordre sociétales (basées par exemple sur le respect du paysage), économiques (basées sur la production de bois), et écologiques (basées sur des indicateurs de biodiversités). Plusieurs modèles de dynamiques forestières sont construits dans le cadre du projet, ces modèles sont simplifiés pour pouvoir être utilisés dans le cadre de la théorie de la viabilité. La théorie de la viabilité consistant à évaluer des "scénarios" sur la base du respect d'un ensemble de plusieurs contraintes se prête particulièrement bien à la gestion multi-fonctionnelle des forêts. Cette approche est facilement appréhendée par les différents membres du projet, écologues, économistes ou responsables de l'animation entre les chercheurs et les acteurs de la filière sylvicole.

Le Jeu SimParc : aide à la gestion participative

Sommaire

2.1 Le Projet SimParc	18
2.1.1 Contexte du projet	18
2.1.2 Approche	18
2.2 Le jeu de rôle SimParc	19
2.2.1 L'objectif du jeu	19
2.2.2 Les dispositions générales sur le zonage au Brésil	20
2.2.3 Déroulement du jeu	22
2.3 Le support de l'architecture du Jeu SimParc	24
2.3.1 L'architecture du jeu SimParc	24
2.3.2 Agents artificiels	25

Dans ce chapitre, nous présentons le projet SimParc, un projet interdisciplinaire visant au développement de méthodologies innovantes, en utilisant l'informatique, pour faciliter la gestion collective et coopérative des parcs nationaux. Ce projet représente un partenariat international et interdisciplinaire vers la construction de la gouvernance démocratique dans les politiques publiques. Ce projet fournit une plate-forme permettant de réaliser un jeu de rôle afin de répéter des expérimentations réalistes et de pouvoir tester facilement l'agent expert viabilité.

La structure de ce chapitre est la suivante. D'abord, nous introduisons le contexte et l'objectif du projet SimParc. Puis, nous décrivons le jeu de rôle de SimParc et les dispositions générales sur le zonage au Brésil. Enfin, nous décrivons le support de l'architecture du jeu SimParc et nous discutons de l'insertion des différents types d'agents artificiels comme des supports supplémentaires.

2.1 Le Projet SimParc

2.1.1 Contexte du projet

Dans le cadre de la gestion des aires protégées, le défi du développement durable est la conciliation d'objectifs concurrents de conservation de l'environnement, de développement économique, et d'inclusion sociale. Le Brésil est confronté à un tel défi, dans les zones rurales mais aussi dans les zones urbaines, comme les restes de la forêt atlantique dans la ville de Rio de Janeiro. Des initiatives visant à régler ces problèmes, notamment par le Ministère brésilien de l'Environnement, sont basées sur les approches participatives, qui permettent de construire des solutions en utilisant les connaissances des acteurs locaux.

Des exemples d'acteurs sociaux impliqués dans ces conflits sont : les gestionnaires du parc, les communautés locales de la zone frontalière, les opérateurs du tourisme, des organismes publics et des organisations non gouvernementales (ONG). Des exemples de conflits liés à la protection de la biodiversité dans la région sont les suivants : occupation irrégulière, l'exploration du tourisme insuffisante, la pollution des eaux, dégradation de l'environnement et l'utilisation illégale des ressources naturelles. Ces conflits se produisent à cause des cultures différentes, des contextes et des pratiques [Irving, 2006].

Le projet de recherche SimParc [Briot *et al.*, 2007] a commencé à la fin 2006. Il se concentre sur l'aide aux différents intervenants pour définir et comprendre collectivement les conflits dans la gestion des parcs et de négocier des stratégies pour les traiter. En particulier, il s'agit d'étudier les utilisations de techniques informatiques pour l'aide à la gestion participative des zones protégées, plus spécifiquement les parcs nationaux. Le prototype actuel SimParc est basé sur un jeu de rôle et des techniques informatiques avancées. L'approche s'inspire de celle du Compagnion Modelling (ComMod) qui a déjà prouvé son efficacité sur un certain nombre de projets de gestion des ressources renouvelables [Barreteau, 2003].

2.1.2 Approche

Depuis la fin des années 1990, un groupe de chercheurs a développé une approche participative pour soutenir les négociations sur les pratiques collectives dans le cadre de la gestion des ressources naturelles. Ils sont réunis au sein de l'association ComMod. Leur méthode qui s'appelle MAS/RPG [Barreteau, 2003], et consiste à coupler des systèmes multi-agents (MAS : Multi-Agent Simulation en anglais) et les jeux de rôle (RPG : Role-Playing Games en anglais) avec les acteurs concernés.

Les jeux de rôles sont particulièrement bien adaptés pour favoriser la communication entre les acteurs. En cas d'utilisation d'un modèle de l'environnement, les joueurs accèdent directement à la simulation, en utilisant une interface utilisateur.

Ils peuvent réviser plusieurs simulations. Les résultats de ces simulations, combinés avec les étapes de discussions entre acteurs, permettent aux joueurs de mieux comprendre les conséquences de leurs décisions et leurs corrélations avec celles des autres joueurs. Cette possibilité de présenter l'impact des comportements individuels sur l'environnement sont exploités par ces chercheurs pour l'aide à la négociation [Bousquet *et al.*, 1999]. Le thème de la négociation peut inclure, par exemple la raréfaction des ressources hydrauliques [Adamatti *et al.*, 2005], le processus de déforestation ou de reboisement [Etienne, 2003]. Cependant, dans les travaux de cette thèse, ce n'est pas un modèle MAS qui est utilisé, un tel modèle n'étant pas disponible actuellement.

SimParc s'inspire de cette approche, en mettant l'accent sur le jeu de rôle, une méthode participative appliquée à la gestion des ressources naturelles renouvelables qui vise à étudier l'impact des comportement individuels sur l'environnement, pour aider les acteurs à comprendre leurs différents points de vue, puis aider à la négociation et pour, enfin, favoriser la décision collective [Briot *et al.*, 2007].

2.2 Le jeu de rôle SimParc

2.2.1 L'objectif du jeu

Actuellement, l'objectif du jeu SimParc est le suivant : aider chaque participant à découvrir et à comprendre les différents facteurs, les conflits et l'importance du dialogue pour une gestion plus efficace des parcs [Briot *et al.*, 2011]. Le jeu n'est pas encore un véritable outil d'aide à la décision. En effet, le parc modélisé est fictif et au cours de la durée de la thèse, il n'est pas prévu de l'appliquer à un parc particulier.

Le jeu est basé sur un processus de négociation qui se déroule dans le conseil du parc. Ce conseil est consultatif. Il comprend des représentants des différents acteurs (par exemple, les communautés, les entreprises touristiques, les associations écologistes non gouvernementales, les agences de l'eau publiques). Le jeu actuel se concentre sur une discussion dans le conseil sur le "zonage" du parc, c'est-à-dire la décision sur un niveau souhaité de conservation pour chaque sous-zone (également appelée "unité de paysage") du parc.

Le jeu définit un certain nombre de rôles des joueurs, chacun représentant un rôle de l'acteur. Chaque joueur, comme tous les jeux de rôle, peut incarner un rôle. Il peut choisir son rôle avec ses postures et objectifs. Pour faciliter l'incorporation du rôle par le joueur, SimParc offre un ensemble de personnages pour représenter les acteurs au cours du jeu (voir figure 2.1). Selon son profil et les éléments concernés dans chacune des unités de paysage (par exemple, lieu de tourisme, les habitants, les espèces menacées ...), chaque joueur va essayer d'influencer la décision sur le type de

conservation pour chaque unité de paysage. Il est évident que les conflits d'intérêts vont apparaître très rapidement, et engendrer souvent différentes stratégies de négociation (par exemple, la formation de coalitions, se soutenir mutuellement, etc).



FIGURE 2.1 – Rôles dans SimParc.

La décision du gestionnaire du parc s'inscrit dans un cadre juridique, elle est également influencée par le processus de négociation entre les acteurs, et par son profil personnel (par exemple, plutôt conservateur ou plutôt ouvert aux préoccupations sociales) [Irving, 2006]. Il a aussi besoin d'expliquer sa décision, si les joueurs en font la demande. Nous prévoyons que les joueurs et le gestionnaire du parc puissent être joués par des humains ou par des agents artificiels.

2.2.2 Les dispositions générales sur le zonage au Brésil

Pour l'instant, nous travaillons sur un parc au Brésil (figure 2.2). Ce parc est divisé en huit unités de paysage. Les joueurs doivent prendre une décision pour chaque unité de paysage. Le zonage est un instrument de planification territoriale, utilisé comme une ressource pour obtenir de meilleurs résultats dans la gestion des aires de conservation. SimParc offre neuf modes de gestion pour les parcs nationaux, qui respectent les lois brésiliennes.

Des zones avec peu ou aucune intervention

- *Intangible* : La nature primitive, rien n'est permis, ce qui représente le plus haut degré de conservation. Cet espace est dédié à la protection totale des écosystèmes, les ressources génétiques et la surveillance environnementale. L'objectif principal de la gestion est de préserver l'environnement.

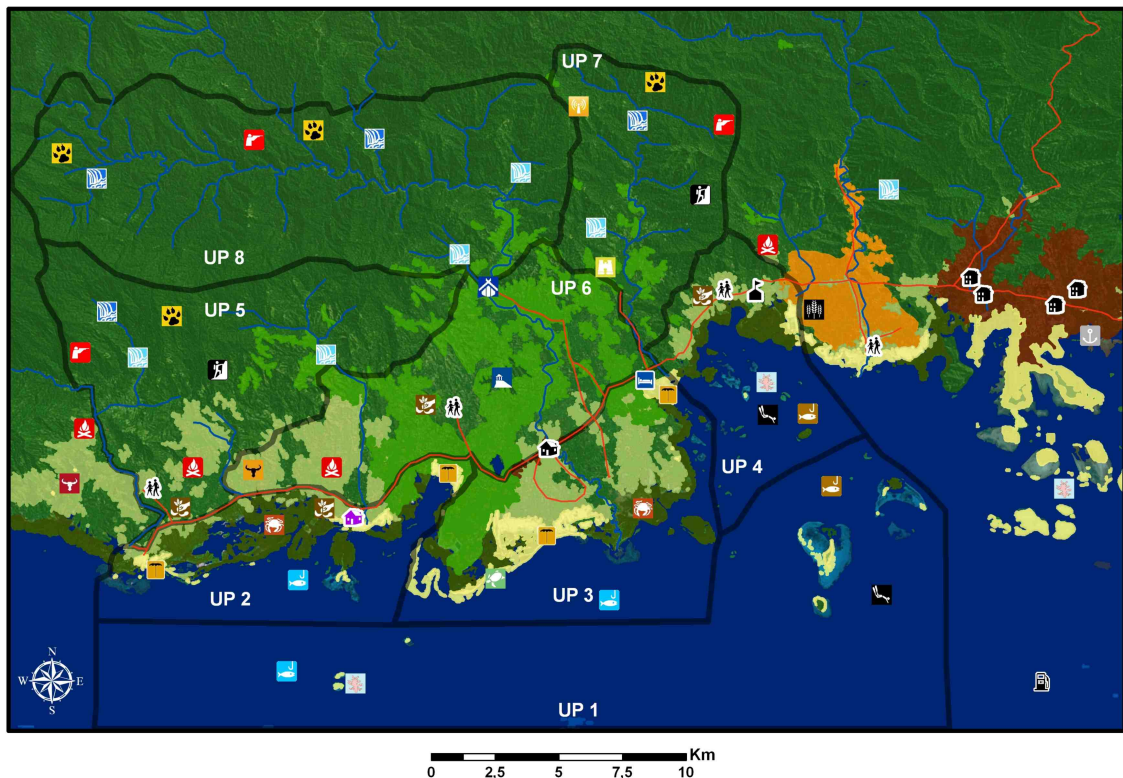


FIGURE 2.2 – Carte des différentes zones du parc et leurs ressources.

- *Primitive* : Il y a très peu d'intervention humaine. Une telle zone contient des espèces de faune et de flore ou des phénomènes naturels de grande valeur scientifique. Elle possède des caractéristiques de transition entre la zone intangible et la zone d'utilisation intensive. L'objectif général de la gestion est de préserver l'environnement naturel tout en facilitant la recherche scientifique et l'éducation.

Des zones avec une intervention d'intensité moyenne

- *Utilisation Extensive* : Elle contient essentiellement des zones naturelles, et peut subir une certaine altération due à l'Homme. Elle se caractérise comme une zone de transition entre la zone primitive et la zone intensive. L'objectif de gestion consiste à maintenir un environnement naturel, avec un impact limité de l'humain, et offre l'accès au public facilement, par exemple pour l'éducation et les loisirs.
- *Historico-culturelle* : Elle contient les zones où ont été découverts des monuments historique ou culturels qui doivent être préservés, étudiés restaurés et exposés au public, au service de la recherche, l'éducation et l'utilisation scientifique. L'objectif principal de la gestion est de protéger les

sites historiques ou archéologiques en harmonie avec l'environnement.

Des zones avec un fort niveau d'intervention

- *Utilisation Intensive* : Elle contient des espaces naturels ou modifiés par l'Homme. Elle doit contenir : le centre d'accueil, des musées, d'autres équipements et services. L'objectif principal de la gestion est de faciliter la récréation intensive et l'éducation environnementale en respectant l'environnement.
- *Utilisation Spéciale* : Elle contient les espaces nécessaires à l'administration, la maintenance et le service des zones de conservation, en particulier les maisons, les ateliers. Ces zones seront sélectionnées et contrôlées pour éviter les conflits. L'objectif global de la gestion est de minimiser l'impact de l'introduction de nouvelles structures ou les effets des travaux dans le milieu naturel.
- *Récupération* : Ce sont des espaces considérés comme anthropisés. Ce sont les zones provisoires, incorporées dans le zonage permanent une fois restaurées. Les espèces exotiques doivent être enlevées et la restauration devrait être naturelle ou induite naturellement. L'objectif global de la gestion est d'arrêter la dégradation des ressources ou de restaurer la zone. Cette zone permet l'usage du public, mais seulement pour l'éducation.
- *Conflits d'utilisation* : Ce sont des espaces situés au sein d'une zone de conservation. Ils sont établis avant la création de l'unité et présentent des conflits avec les objectifs de conservation de la zone protégée. Les zones sont occupées par des entreprises de service public, par exemple pour les gazoducs, les lignes de transmission, les antennes, les prélèvements d'eau, les barrages, les routes, les câbles optiques. L'objectif de gestion est d'améliorer la situation en établissant des procédures pour minimiser les impacts sur les zones protégées.
- *Occupation Temporaire* : Ce sont des espaces situés dans les zones protégées où il y a des concentrations de populations élevées. Ce sont des zonages provisoires, un fois la population dispersée, elles seront incorporées dans les zones permanentes.

2.2.3 Déroulement du jeu

Le jeu est structuré en six étapes représentées dans figure 2.3. Le jeu commence par l'étape 1, chaque participant est associé à un rôle. Puis, un scénario initial est présenté à chaque joueur, y compris les caractéristique des unités de paysage, les types possibles d'utilisation et l'objectif général associé à son rôle. Puis à l'étape 2, chaque joueur émet une première proposition de types d'utilisation de chaque unité de paysage, basée sur sa compréhension de l'objectif de son rôle et sur la configuration initiale. Une fois que tous les joueurs ont fait leur choix, la proposition de chaque joueur est rendue publique.

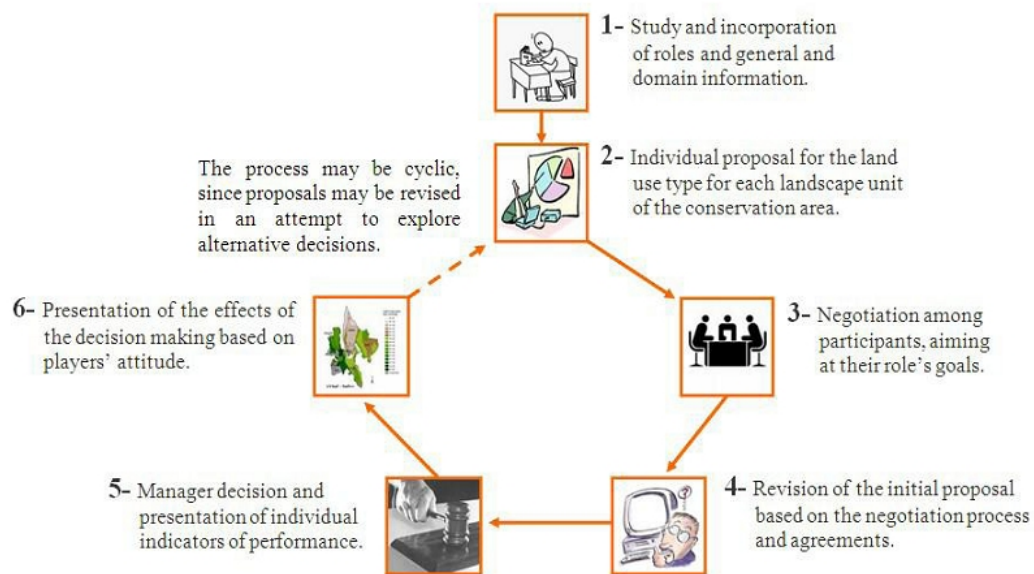


FIGURE 2.3 – Les six étapes du jeu SimParc.

Durant l'étape 3, les joueurs commencent à interagir et à négocier leurs propositions. Cette étape est très importante, car les joueurs construisent leurs connaissances collectivement. Dans l'étape 4, les joueurs révisent leurs propositions et s'engagent sur une proposition finale pour chaque unité de paysage. Dans l'étape 5, le gestionnaire du parc prend la décision finale, en considérant le processus de négociation, mais ses décisions dépendent également de son profil personnel (par exemple, favorise la conservation ou favorise l'exposition au public). Chaque joueur peut alors consulter différents indicateurs de sa performance (par exemple, la proximité de son objectif initial, le degré de consensus, etc.) Il peut également demander une explication des raisons de la décision finale. La dernière étape (étape 6) "ferme" le cycle épistémique en considérant les effets possibles de la décision. Dans le jeu actuel, les joueurs offrent un retour simple sur la décision en indiquant leur degré d'acceptation de la décision.

Un nouveau cycle de négociation peut commencer, ce qui permet de créer une sorte de cycle d'apprentissage. L'objectif principal est pour les participants : de comprendre les différents facteurs, les cultures et les perspectives concernés et comment ils sont liés entre eux ; de négocier, d'essayer de parvenir à un consensus de groupe, et de comprendre les relations de causalité sur lesquelles sont basées les décisions.

2.3 Le support de l'architecture du Jeu SimParc

2.3.1 L'architecture du jeu SimParc

Le prototype du jeu SimParc est basé sur J2E et JSF qui soutiennent le caractère distributif et interactif du jeu ainsi qu'un déploiement facile. La figure 2.4 montre la structure générale de l'architecture et de la communication du prototype actuel SimParc.

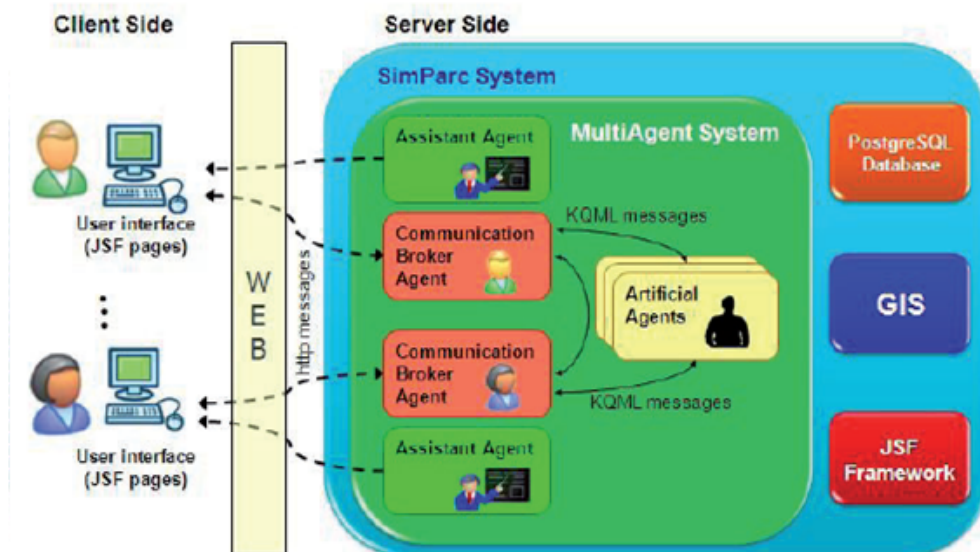


FIGURE 2.4 – L'architecture générale de SimParc.

Les utilisateurs (les joueurs et le gestionnaire du parc) interagissent avec le système de médiation par les Agents Courtiers en Communication (CBA). Un CBA permet que chaque rôle puisse être joué par un humain ou par un agent artificiel. Un CBA traduit également les messages des utilisateurs du format *http* au format KQML¹ multi-agent et vice versa. Pour chaque joueur humain, il y a aussi un agent assistant qui offre de l'aide au cours de la session de jeu (pour plus de détails voir [Vasconcelos *et al.*, 2009]). Au cours de la phase de négociation, les joueurs (humains ou artificiels) négocient entre eux pour essayer de parvenir à un accord sur le type d'utilisation pour chaque unité de paysage (sous-zone) du parc.

Un Système d'Information Géographique (GIS) offre aux utilisateurs différentes couches d'informations (comme la flore, la faune et les caractéristiques des terres) sur la zone du parc géographique. Toutes les informations qui sont échangées au cours de la phase de négociation, constituent les journaux des utilisateurs ; les configurations de jeu, les résultats de jeux et les informations de gestion générales, sont enregistrés et lus à partir d'une base de données PostgreSQL. Il y a une

1. Knowledge Query and Manipulation Language

interface dédiée à la négociation comme dans la figure 2.5. Elle comprend un support avancé pour la négociation, un accès à différents types d'informations sur les autres joueurs, les terres, le droit et l'aide d'un assistant personnel. La figure 2.6 représente l'interface par l'intermédiaire de laquelle les utilisateurs choisissent les types de modes de gestion. Dans cette interface, les joueurs peuvent analyser la zone en fonction de ses différents niveaux de couches (terrain, hydrographique, végétation, etc.). Le prototype informatique de SimParc a été testé lors de deux sessions de jeu par les joueurs experts du domaine en janvier 2009 et en janvier 2010.

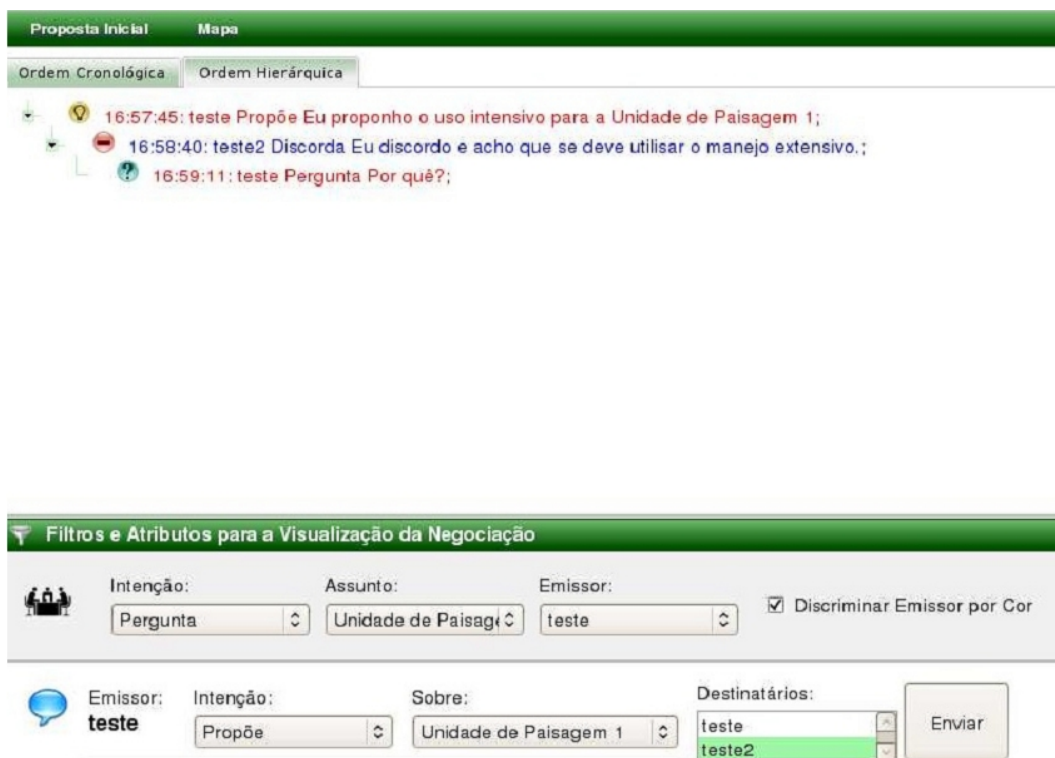


FIGURE 2.5 – L'interface de négociation de SimParc.

2.3.2 Agents artificiels

Le groupe de SimParc est en train d'intégrer différents types d'agents artificiels dans le prototype. L'agent artificiel gestionnaire du parc joue le rôle de gestionnaire du parc. Les agents artificiels joueurs peuvent remplacer des joueurs dans le jeu. L'agent assistant, peut assister les joueurs humains dans le jeu de rôle. L'agent expert peut fournir aux joueurs humains des informations techniques sur la viabilité de leur proposition ou analyser les relations entre les propositions des joueurs.



FIGURE 2.6 – L’interface de sélection des modes de gestion.

L’agent gestionnaire du parc

Comme il est expliqué dans la section 2.2.3, le gestionnaire du parc joue le rôle de l’arbitre dans le jeu, il prend une décision finale pour les types de conservation pour chaque unité de paysage. Il a aussi besoin d’expliquer sa décision à tous les joueurs. Le gestionnaire du parc peut être joué par un humain ou par un agent artificiel. Il sera défini par le gestionnaire du jeu pendant la configuration d’une nouvelle session de jeu (voir figure 2.7).

L’architecture de l’agent gestionnaire du parc est structurée en deux phases. La première étape concerne le processus de la prise de décision individuel. L’agent gestionnaire du parc délibère sur les types de conservation pour chaque unité de paysage. Concrètement, l’agent gestionnaire du parc construit une liste de préférences par rapport aux niveaux autorisés de conservation. Une argumentation [Dung, 1995] a été mise en œuvre pour l’aide à la décision. L’idée principale consiste à utiliser le système d’argumentation pour sélectionner les souhaits. La deuxième étape consiste à prendre en compte les préférences des joueurs, avec la possibilité d’ajuster le profil du gestionnaire du parc, de l’autocratie à la démocratie, et donc l’influence de la voix des joueurs. Ainsi, une méthode originale permettant de combiner voix des joueurs et le vote du gestionnaire du parc, basée sur une fonction d’influence a été conçue.

New Game

Name:

Game Manager:

Phases' Time Definition

Exploration Step: Minutes:

Pre-Negotiation Step: Minutes:

Negotiation Step: Minutes:

Players' Roles Definition

Park Manager:

Artificial Human

Park Manager Profile:

Participatory Attitude:

Selection Type:

Player Selection Manager Selection Random

Select the players for the game!:

Player:

Player:

FIGURE 2.7 – L'interface de la configuration d'une nouvelle session.

L'agent artificiel joueur

Les joueurs artificiels s'inspirent des joueurs virtuels dans les jeux de rôles sur ordinateur. L'idée est de remplacer éventuellement certains des joueurs humains par des agents artificiels. Les deux principales motivations sont les suivantes : (1) nombre insuffisant de joueurs humains pour une session de jeu et (2) le besoin de tester de façon systématique des configurations spécifiques de profils de joueurs. Les joueurs artificiels seront développés à partir de l'agent artificiel gestionnaire du parc avec l'addition de la négociation et de modules d'interaction. Il est prévu d'utiliser les capacités d'argumentation pour générer et contrôler le processus de négociation.

L'agent assistant

Les agents assistants ont été conçus pour assister les joueurs pendant le jeu. L'utilisateur domine complètement son assistant, il peut l'activer ou le désactiver à tout moment. La fonction de base de ces agents est de présenter et d'expliquer chaque étape du jeu. Pendant l'étape de négociation, les agents assistants peuvent également proposer aux joueurs des informations utiles, afin d'améliorer leur analyse concernant la négociation. Par exemple, ils peuvent informer un joueur sur les indicateurs des calculs comme l'affinité (compatibilité) avec des propositions d'autres joueurs, la participation, l'influence, etc.

L'agent expert

C'est dans ce cadre que s'insèrent les travaux de cette thèse. Pour nourrir la réflexion des joueurs, il serait souhaitable de pouvoir solliciter un ou plusieurs agents experts, susceptibles d'apporter aux joueurs un éclairage particulier.

La thèse se concentre sur l'étude d'un agent expert "viabilité". La théorie de la viabilité, [Aubin, 1991] permet d'identifier les politiques qui peuvent maintenir ou restaurer les propriétés souhaitables d'un système dynamique. Les joueurs doivent définir des propriétés désirables comme des contraintes sur les états viables du parc. Les discussions sur les contraintes sont souvent plus faciles à gérer que la discussion sur les objectifs à fixer. L'analyse de viabilité permet d'aider à définir l'ensemble des états dans lesquels le parc doit être conservé afin de garantir ou de rétablir les propriétés désirables, et de déterminer les politiques qui permettent de conserver ces états ou de les atteindre. Notre thèse se concentre sur ce dernier agent artificiel.

Les pré-requis d'un agent viabilité sont donc les suivants :

1) montrer l'intérêt de l'analyse de viabilité pour prendre en compte les dynamiques transitoires dans une optique d'étude de durabilité. C'est un des apports principaux de cette thèse.

2) disposer d'un modèle dynamique du système dont on étudie la viabilité.

Dans la cadre de cette thèse nous avons utilisé un modèle très simple adapté de la littérature pour tester l'agent expert viabilité. L'un des avantages de l'agent expert viabilité est de pouvoir utiliser n'importe quel modèle. La construction du modèle, les choix des variables et les valeurs des paramètres doivent être définis par les acteurs autant que possible, sinon les joueurs n'auront pas confiance en ce modèle. De plus, dans le cadre de la thèse, ce qui a été privilégié c'est la modularité du logiciel et de l'interface, de manière à permettre la modification des modèles.

3) faciliter la définition de la zone d'états souhaitables (choix des contraintes)

et des contrôles.

4) calculer rapidement une approximation du noyau de viabilité.

Les travaux de cette thèse se sont intéressés à ce point crucial pour l'utilisation lors d'un jeu de rôle.

5) présentation des résultats.

Une partie importante de la thèse : les résultats fournis par l'agent expert viabilité doivent être clairs et compréhensibles. Les joueurs peuvent utiliser ses résultats pour convaincre les autres joueurs. Le gestionnaire du parc utilise aussi les résultats pour présenter sa décision finale.

Deuxième partie

Viabilité et durabilité : étude
théorique

Analyse de viabilité : prendre en compte les comportements transitoires et asymptotiques

Sommaire

3.1	Limites de l'analyse asymptotique	34
3.1.1	Modèle de l'impact du tourisme sur la qualité de l'environnement	34
3.1.2	L'information fournie par l'étude asymptotique et le diagramme de bifurcation	36
3.1.3	Les informations manquantes	38
3.2	La théorie de la viabilité pour l'information manquante	42
3.2.1	Le noyau de viabilité pour l'information spatiale	42
3.2.2	Le bassin de capture pour l'information temporelle	44
3.3	Le concept de viabilité et l'analyse de viabilité	46
3.3.1	La dynamique non contrôlée	46
3.3.2	La dynamique contrôlée	46
3.4	Conclusion	52

Nous allons montrer dans cette partie que la théorie mathématique de la viabilité est particulièrement bien adaptée pour étudier le comportement d'un système dynamique sous l'angle de la durabilité (sustainability). La durabilité est un concept essentiel dans le cadre de la gestion de systèmes environnementaux (sans d'ailleurs se limiter à ces systèmes). Par son approche basée sur les contraintes, la théorie de la viabilité propose un éclairage différent sur cette question.

Dans ce chapitre nous étudions liens entre l'intervention de l'analyse de la viabilité et l'analyse asymptotique (qui est une méthode traditionnelle pour l'étude des équations différentielles). Dans ce cadre théorique on considérera que le système environnemental étudié peut être décrit par un système dynamique. Quand cela est possible, un système dynamique est utilisé pour modéliser la dépendance au temps des différentes variables d'état du système. Quand le modèle consiste en un ensemble d'équations différentielles, la durabilité et les concepts associés sont généralement liés à des propriétés asymptotiques : le type et les propriétés des attracteurs (liée par exemple à la valeur des valeurs propres), les propriétés dans l'espace des phases (par exemple, la taille du bassin

d'attraction des différents attracteurs comme dans [Collings et Wollkind, 1990] et [Van Coller, 1997]) et les propriétés du diagramme de bifurcation (voir par exemple [Casagrandi et Rinaldi, 2002] et [Ludwig *et al.*, 1997]).

Cependant, les propriétés asymptotiques ne sont pas suffisantes pour caractériser le problème de durabilité, car ils se concentrent sur le futur. Beaucoup de systèmes naturels n'atteignent jamais un équilibre, car ils sont soumis à des perturbations fréquentes. Souvent, les comportements transitoires peuvent avoir des effets indésirables sur le système. Par exemple, les évolutions peuvent partir très loin avant d'arriver à l'attracteur, ou bien le temps nécessaire pour atteindre un voisinage désirable de l'attracteur peut aussi être très long.

Nous allons décrire ces deux limites de l'analyse de la stabilité asymptotique dans la section 3.1. Puis nous utiliserons la théorie de la viabilité pour contourner ces limites. Dans la section 3.2, nous montrerons d'abord que s'il existe des ensembles de contraintes convenables, les concepts de noyau de viabilité et de bassin de capture permettent d'apporter des informations cruciales sur le comportement spatial ou temporel. Enfin dans la section 3.3, nous discuterons l'approche de la viabilité de la dynamique contrôlée, quand certains paramètres peuvent être interprétés comme des variables de contrôle. En fait, les dynamiques contrôlées sont souvent pour représenter la dynamique de systèmes socio-écologiques confrontés à des problèmes de durabilité influencés par les décisions de gestion ou les comportements humains.

3.1 Limites de l'analyse asymptotique

De part sa proximité thématique avec SimParc, dans cette section nous illustrons notre propos avec un modèle concernant l'influence du tourisme sur la qualité de l'environnement. L'analyse asymptotique de ce modèle a été réalisée dans [Casagrandi et Rinaldi, 2002].

3.1.1 Modèle de l'impact du tourisme sur la qualité de l'environnement

Le modèle décrit dans [Casagrandi et Rinaldi, 2002] traite le problème de l'interaction entre le tourisme et l'environnement, à un niveau très abstrait. Il possède trois dimensions avec douze paramètres. Dix paramètres sont fixes et l'analyse asymptotique est effectuée selon les deux restants, l'attractivité moyenne des sites touristiques et le taux d'investissement. Le modèle (3.1) étudie les interactions entre les touristes $T(t)$ présents dans la zone au temps t , la qualité de l'environnement naturel $E(t)$ et le capital $C(t)$ correspondant aux structures pour les activités touristiques.

$$\begin{aligned}
\dot{T}(t) &= \frac{dT(t)}{dt} = T(t) \left[\mu_E \frac{E(t)}{E(t)+\varphi_E} + \mu_C \frac{C(t)}{C(t)+\varphi_C T(t)+\varphi_C} - \alpha T(t) - a \right] \\
\dot{E}(t) &= \frac{dE(t)}{dt} = E(t) \left[rE(t) \left(1 - \frac{E(t)}{K}\right) - \beta C(t) - \gamma T(t) \right] \\
\dot{C}(t) &= \frac{dC(t)}{dt} = -\delta C(t) + \epsilon T(t)
\end{aligned} \tag{3.1}$$

$\dot{T}(t)$ est notée $f_1(t)$, $\dot{E}(t)$ est notée $f_2(t)$ et $\dot{C}(t)$ est notée $f_3(t)$ dans la suite. En ce qui concerne la première équation, la variation de l'activité touristique est proportionnelle au nombre de touristes actuel avec un facteur correspondant à l'attractivité du site. Cette attractivité est la somme de deux facteurs positifs, l'attractivité de l'environnement et l'attractivité des infrastructures, et de deux facteurs négatifs, la fréquentation et la valeur moyenne de l'attractivité de tous les autres sites touristiques (a peut être considéré comme la compétition exercée par les sites touristiques alternatifs). Les paramètres fixes sont μ_E , l'attractivité associée à haute qualité environnementale, et φ_E , la constante pour 50% de saturation, c'est-à-dire la qualité l'environnement correspondant à une satisfaction des touristes à la moitié du maximum. μ_C et φ_C sont les paramètres correspondants à l'attractivité de l'infrastructure. L'attractivité négative due à la fréquentation est proportionnelle à T avec le facteur $-\alpha$.

La deuxième équation décrit la variation de la qualité de l'environnement. Elle consiste d'abord en une équation logistique qui décrit l'impact sur l'environnement de toutes les activités sauf l'industrie du tourisme. Les paramètres r et K correspondent aux taux de croissance et à la capacité de charge de la fonction logistique. Les deux autres termes représentent le flux de dommages induits par le tourisme. En général, ce flux est positivement corrélé avec le nombre de touristes et le capital avec des facteurs linéaires γ et β .

La dernière équation décrit le taux de variation du capital comme la différence entre l'investissement ϵT et la dépréciation δC (ϵ est le taux d'investissement et δ , le taux d'amortissement).

Le paramètre δ est considéré comme très petit devant r , car la dégradation des structures touristiques est très lente.

Dans [Casagrandi et Rinaldi, 2002], les deux paramètres de l'analyse asymptotique sont a qui mesure la concurrence exercée par les autres sites touristiques et ϵ qui est le taux d'investissement.

Ce modèle simple reproduit fidèlement les scénarios de [Butler, 1980] pour le développement de sites de tourisme. La Figure 3.1 représente trois scénarios d'évolution du tourisme avec le même point de départ, mais avec trois valeurs différentes du taux d'investissement de ϵ au cours de 100 unités de temps. Un scénario conduit à la disparition de l'activité touristique. Les deux autres

conduisent à une activité touristique non nulle mais avec des niveaux différents.

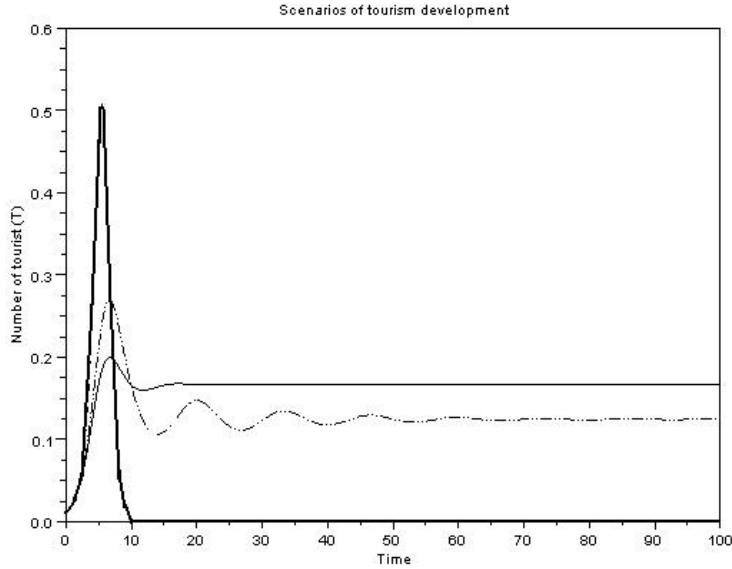


FIGURE 3.1 – Scénarios de développement du tourisme à horizon temporel 100. Tous ces schémas sont obtenus avec le modèle (3.1). Les valeurs des paramètres sont $r = K = \alpha = \beta = \gamma = \varphi_C = 1, \delta = 0.1, \varphi_E = 0.5, \mu_E = \mu_C = 10, a = 6$ et $\epsilon = 0.01$ pour la ligne continue, $\epsilon = 0.1$ pour la ligne en pointillés et $\epsilon = 0.45$ pour la ligne en gras. Le point de départ de toutes les trajectoires est $(T = 0.01, E = 1, C = 0.01)$.

3.1.2 L'information fournie par l'étude asymptotique et le diagramme de bifurcation

L'analyse asymptotique met l'accent sur le comportement du système à horizon infini. Par exemple, pour les trois scénarios décrits dans la figure 3.1, l'analyse asymptotique permet de prévoir les valeurs de convergence du modèle. Ces valeurs peuvent être des équilibres de la dynamique, mais il peut aussi exister des cycles limites comme attracteur. Cette information est évidemment précieuse dans une optique de durabilité, parce qu'elle permet de souligner les dérives futures.

Les équilibres et leur stabilité

Dans le cas général d'un système évolutif décrit par une équation différentielle

$$x'(t) = f(x(t)) \quad (3.2)$$

avec $x \in \mathbb{R}^n$, un équilibre $x_0 \in \mathbb{R}^n$ est un point où la dynamique est nulle ($f(x_0) = 0$). Cet équilibre peut être asymptotiquement stable si la dynamique tend à attirer

les états de son voisinage, ou instable si certains points dans son voisinage ont tendance à être rejetés par elle. Les valeurs d'équilibre sont déterminées par la résolution de l'équation $f(x_0) = 0$ analytiquement si possible, ou avec des méthodes d'approximation (Newton-Raphson, point fixe, l'algorithme de descente de gradient, etc). Ensuite, pour déterminer la stabilité de ces équilibres, nous pouvons utiliser la linéarisation de la dynamique sur ces points d'équilibre :

Theorem 1 Si $A = \left. \frac{\partial f(x)}{\partial x} \right|_{x=x_0}$ est la matrice Jacobienne de $f(x)$ au point d'équilibre x_0 . Le système

$$z' = Az$$

est appelé linéarisé de l'équation (3.2) au point d'équilibre x_0 . Si le linéarisé existe, sa stabilité détermine la stabilité locale de l'équation non linéaire d'origine.

Par conséquent, si la partie réelle de toutes les valeurs propres de A sont strictement négatives, alors l'équilibre est stable.

Nous avons utilisé la méthode de Newton-Raphson pour approximer les valeurs d'équilibre strictement positives du modèle (3.1), ($\bar{T} > 0$, $\bar{E} > 0$, $\bar{C} > 0$), pour les valeurs des paramètres utilisées dans les scénarios présentés dans la figure 3.1. Evidemment, ($\bar{T} = 0$, $\bar{E} = 0$, $\bar{C} = 0$), ($\bar{T} = 0$, $\bar{E} = K$, $\bar{C} = 0$) et aussi ($\bar{T} = \tilde{T} > 0$, $\bar{E} = 0$, $\bar{C} = \tilde{C} > 0$) sont des équilibres de la dynamique. Mais, comme dans [Casagrandi et Rinaldi, 2002], nous considérons les états strictement positifs car ils sont susceptibles de représenter des situations profitables où les activités économiques et la préservation de l'environnement coexistent.

- quand $\epsilon = 0.01$, ($\bar{T} \approx 0.167$, $\bar{E} \approx 0.758$, $\bar{C} \approx 0.017$),
- quand $\epsilon = 0.1$, ($\bar{T} \approx 0.125$, $\bar{E} \approx 0.526$, $\bar{C} \approx 0.125$),
- et quand $\epsilon = 0.45$, ($\bar{T} \approx 0.044$, $\bar{E} \approx 0.402$, $\bar{C} \approx 0.197$).

Pour déterminer la stabilité de ces équilibres strictement positifs, nous avons d'abord besoin de déterminer la matrice Jacobienne $M(\bar{T}, \bar{E}, \bar{C})$ de la dynamique du modèle à ces points positifs d'équilibre, et ensuite de calculer les valeurs propres de ces matrices.

$$M(\bar{T}, \bar{E}, \bar{C}) = \begin{pmatrix} \frac{\partial f_1}{\partial T}(\bar{T}, \bar{E}, \bar{C}) & \frac{\partial f_1}{\partial E}(\bar{T}, \bar{E}, \bar{C}) & \frac{\partial f_1}{\partial C}(\bar{T}, \bar{E}, \bar{C}) \\ \frac{\partial f_2}{\partial T}(\bar{T}, \bar{E}, \bar{C}) & \frac{\partial f_2}{\partial E}(\bar{T}, \bar{E}, \bar{C}) & \frac{\partial f_2}{\partial C}(\bar{T}, \bar{E}, \bar{C}) \\ \frac{\partial f_3}{\partial T}(\bar{T}, \bar{E}, \bar{C}) & \frac{\partial f_3}{\partial E}(\bar{T}, \bar{E}, \bar{C}) & \frac{\partial f_3}{\partial C}(\bar{T}, \bar{E}, \bar{C}) \end{pmatrix} =$$

$$\begin{pmatrix} \bar{T} \left(\frac{-10\bar{C}}{(\bar{C}+\bar{T}+1)^2} - 1 \right) & \bar{T} \left(\frac{5}{(\bar{E}+0.5)^2} \right) & \bar{T} \left(\frac{10\bar{T}+10}{(\bar{C}+\bar{T}+1)^2} \right) \\ -\bar{E} & \bar{E} (1 - 2\bar{E}) & -\bar{E} \\ \epsilon & 0 & -0.1 \end{pmatrix}$$

Nous calculons maintenant les valeurs propres pour les différentes matrices $M(\bar{T}, \bar{E}, \bar{C})$ avec le logiciel Scilab. Nous obtenons alors les valeurs arrondies suivantes :

- $\{-0.289 + 0.612i, -0.289 - 0.612i, -0.100\}$ pour $\epsilon = 0.01$,
- $\{-0.044 + 0.478i, -0.044 - 0.478i, -0.263\}$ pour $\epsilon = 0.1$
- et $\{0.009 + 0.443i, -0.009 - 0.443i, -0.320\}$ pour $\epsilon = 0.45$.

Grâce au théorème 1, nous pouvons déterminer que quand ϵ est égal à 0.01 et 0.1, l'équilibre positif est stable, car les parties réelles de toutes les valeurs propres de la matrice Jacobienne sont négatives. Au contraire pour $\epsilon = 0.45$, le point strictement positif d'équilibre est non stable. Ce résultat est entièrement conforme à la figure 3.1 où les scénarios pour ϵ est égal à 0.01 et 0.1 convergent vers l'équilibre strictement positif, alors que dans le cas ϵ égal à 0.45.

3.1.3 Les informations manquantes

Des études complémentaires dans [Casagrandi et Rinaldi, 2002] se concentrent sur le diagramme de bifurcation en fonction des valeurs des paramètres a et ϵ , pour évaluer si ces attracteurs sont risqués ou non. Mais la détermination des points d'équilibre ou des cycles limites fournit des informations sur le comportement asymptotique du système, et rien sur le comportement transitoire. Le développement durable concerne les générations futures mais aussi les générations contemporaines. Par conséquent, trouver des attracteurs n'est pas suffisant pour étudier le problème du développement durable. Evoluer vers des attracteurs stables, même sans risque du point de vue des bifurcations (c'est à dire en cas d'évolution des paramètres), peut faire passer le système par des états indésirables à cause du comportement transitoire.

3.1.3.1 L'information spatiale : les frontières du bassin d'attraction

Le bassin d'attraction d'un attracteur rassemble les états à partir desquels l'évolution régie par le système (3.2) est telle que la distance à l'attracteur tend vers 0. Quand il y a plusieurs attracteurs, le calcul des bassins d'attraction des différents attracteurs est nécessaire pour déterminer l'attracteur associé à une position initiale donnée.

Par exemple dans le modèle (3.1), quand les valeurs de paramètres sont $\epsilon = 0.1$ et $a = 6.8$, l'équilibre strictement positif $(\bar{T}, \bar{E}, \bar{C})$ n'est pas le seul attracteur. $(T = 0, E = 1, C = 0)$ est aussi un attracteur mais non strictement positif. Si le point de départ est $(0.1, 1, 0.1)$, l'évolution converge vers le point d'équilibre strictement positif, mais si le point de départ est $(0.5, 0.5, 0.5)$, l'évolution converge vers l'attracteur avec activité touristique nulle (voir figure 3.2).

Par conséquent, l'existence d'un équilibre stable strictement positif ne suffit pas pour évaluer le comportement asymptotique de toutes les évolutions. Le calcul du bassin d'attraction est également nécessaire. Il peut par la suite servir de base d'évaluation de la résilience comme dans [Collings et Wollkind, 1990] et

[Van Coller, 1997].

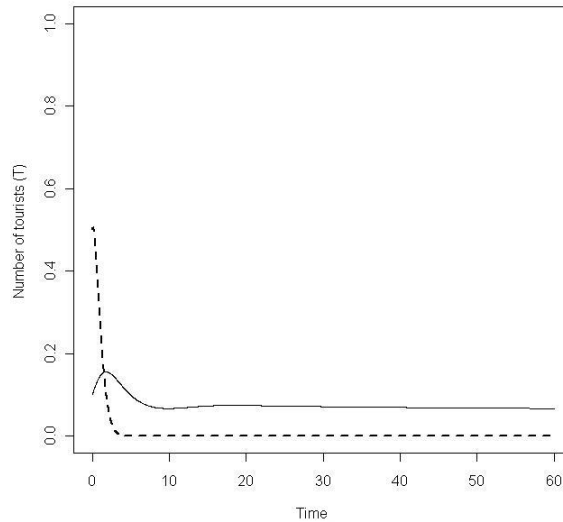


FIGURE 3.2 – Scénarios de développement du tourisme de modèle (3.1). Les valeurs de paramètres sont les même que pour la figure 3.1 avec $a = 6.8$ et $\epsilon = 0.1$. Deux points d'équilibre sont stables : le point strictement positif, notons $A(6.8, 0.1)$ et le point $(0, 1, 0)$. La courbe pleine converge vers $A(6.8, 0.1)$, son point de départ $(0.1, 1, 0.1)$ appartient au bassin d'attraction de $A(6.8, 0.1)$. La courbe en pointillés converge vers 0, son point de départ $(0.5, 0.5, 0.5)$ appartient au bassin d'attraction du point $(0, 1, 0)$.

3.1.3.2 L'information spatiale : la distance maximale à l'attracteur pendant l'évolution

Sachant qu'un point de départ donné appartient au bassin d'attraction d'un équilibre stable peut conduire à croire que la distance entre l'état du système et le point d'équilibre doit diminuer avec le temps. Cette extrapolation n'est pas forcément juste : cette distance peut diminuer puis augmenter et encore diminuer comme dans la figure (3.3.)

Trouver de "bons" attracteurs, qui permettent la coexistence de l'activité touristique et de la préservation de l'environnement, comme dans notre étude de cas, n'implique pas que les évolutions possibles resteront proches de ces attracteurs et présenteront de bons scénarios transitoires. Par conséquent, l'information supplémentaire de la distance maximale à l'attracteur pendant l'évolution (ou une borne supérieure) est nécessaire pour s'assurer que les mauvaises situations sont évitées.

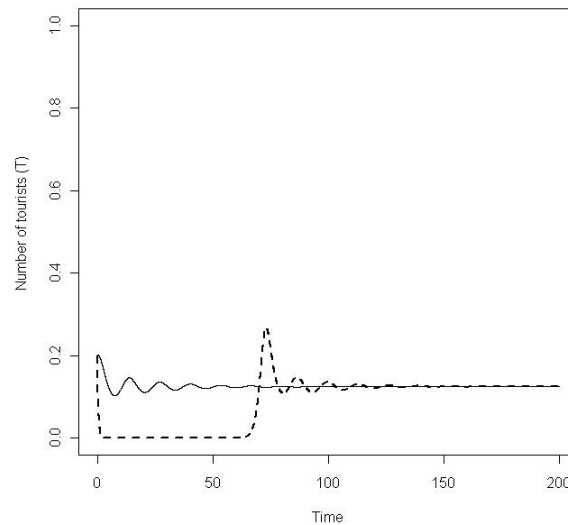


FIGURE 3.3 – Scénarios de développement du modèle de tourisme (3.1). Les valeurs de paramètres sont les mêmes que pour la figure 3.1 avec $a = 6$ et $\epsilon = 0.1$. L'équilibre strictement positif est le seul attracteur. À partir du point initial $(0.2, 0.6, 0.1)$, la distance à la valeur d'équilibre du tourisme (courbe pleine) diminue toujours. À partir du point initial $(0.2, 0.2, 0.2)$, cette distance (courbe pointillée) n'est pas monotone.

En outre, il peut n'y avoir aucun équilibre stable, mais un cycle limite stable, comme c'est le cas lorsque $a = 6$ et $\epsilon = 0, 13$. Il n'y a pas de raison a priori d'écarter les cycles limites. Tout dépend de la définition des états acceptables. Par exemple, c'est le cas de la figure 3.4, si l'activité touristique doit rester comprise entre 0,05 et 0,20. La trajectoire est satisfaisante à partir du temps $t=3$. Au contraire, si les fluctuations acceptables ne doivent pas dépasser une certaine amplitude, l'évolution conduisant au cycle limite ne peut être considérée comme satisfaisante.

3.1.3.3 L'information temporelle : le temps nécessaire pour atteindre l'attracteur

Si la condition initiale d'une trajectoire appartient au bassin d'attraction d'un attracteur A , le temps mis pour atteindre un voisinage de l'attracteur doit être fini. Ce temps fini peut être court ou très long. La courbe en pointillés de la figure 3.1 est une courbe typique qui converge rapidement à l'équilibre stable, mais si nous changeons les conditions initiales au point $(T = 0.26, E = 1, C = 0.26)$, l'évolution représentée dans la figure 3.5, converge aussi vers le même attracteur, mais elle reste encore un temps relativement long ($t > 200$) très proche de 0. Ce scénario n'est donc pas désirable, car la fréquentation touristique reste nulle trop

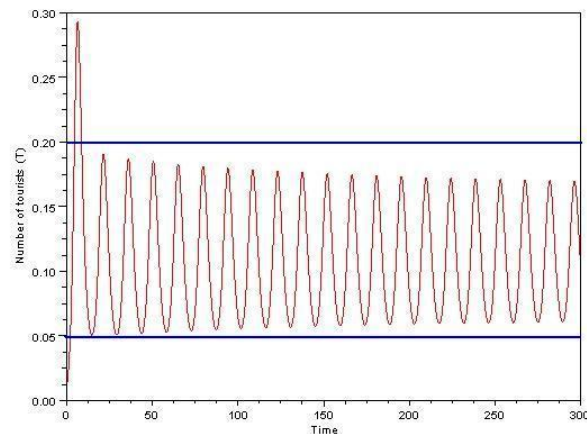


FIGURE 3.4 – Scénarios de développement du tourisme à l’horizon temporel $t=300$. Les valeurs de paramètres sont les mêmes que pour la figure 3.1 avec $a = 6$ et $\epsilon = 0.13$. L’attracteur est un cycle limite. Le point initial est $(T = 0.01, E = 1, C = 0.01)$. Les deux lignes bleues correspondent à $T = 0.05$ et $T = 0.20$.

longtemps.

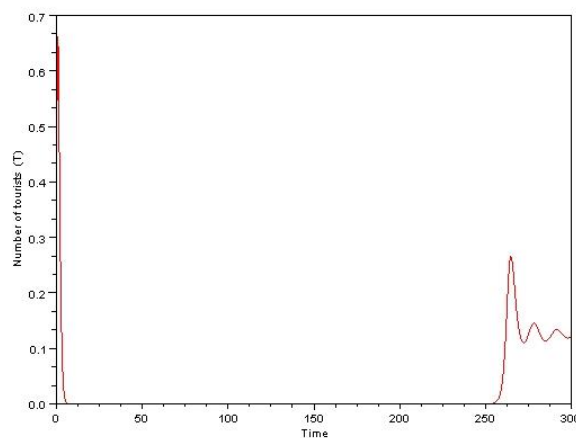


FIGURE 3.5 – Scénarios de développement du tourisme à horizon temporel $t=300$. Les valeurs de paramètres sont les mêmes que dans la figure 3.1 avec $a = 6$ et $\epsilon = 0.1$. Le point initial est $(T = 0.26, E = 1, C = 0.26)$.

Au contraire, on peut exhiber des scénarios où le tourisme, l’environnement et le capital présentent des valeurs satisfaisantes au cours d’une longue période dans le cas d’un ”mauvais” équilibre stable (une des variables a une valeur nulle). Selon l’échelle de temps, un tel scénario peut être considéré comme souhaitable. Par exemple, nous pouvons imaginer que la validité du modèle devra être questionnée à nouveau à l’avenir, ou bien d’autres opportunités peuvent se produire avant d’atteindre les

états indésirables.

3.2 La théorie de la viabilité pour l'information manquante

3.2.1 Le noyau de viabilité pour l'information spatiale

Dans le cas de l'existence d'un 'bon' équilibre stable, nous avons montré dans la section précédente que la distance maximale à l'attracteur pendant l'évolution peut être considérée comme une information importante. Le noyau de viabilité permet de déterminer cette distance maximale. En fait, si nous considérons un voisinage donné de l'équilibre stable, le noyau de viabilité de ce voisinage rassemble tous les états à partir desquels démarre une évolution qui reste dans ce voisinage. En revanche, à partir d'un point quelconque en dehors de ce noyau de viabilité, l'évolution va sortir inéluctablement du voisinage et la distance maximale à l'équilibre sera plus grande.

Nous utilisons encore le modèle (3.1) pour illustrer ce point. Nous prenons comme valeurs des paramètres les valeurs utilisées par [Casagrandi et Rinaldi, 2002] : $r = K = \alpha = \beta = \gamma = \varphi_C = 1$, $\delta=0.1$, $\varphi_E=0.5$, $\mu_E=\mu_C=10$, $a=6$ et $\epsilon = 0.1$. Nous avons déjà prouvé dans la section précédente que l'attracteur positif est $\bar{T} \approx 0.125$, $\bar{E} \approx 0.526$, $\bar{C} \approx 0.125$. Définissons une boule (pour la norme sup) autour de cet attracteur avec le paramètre Δ :

$$K = [\bar{T} - \Delta; \bar{T} + \Delta] \times [\bar{E} - \Delta; \bar{E} + \Delta] \times [\bar{C} - \Delta; \bar{C} + \Delta].$$

Cette boule est considérée comme un ensemble de contraintes dans le cadre de la théorie de la viabilité et nous faisons une approximation de son noyau de viabilité selon le modèle (3.1) pour différentes valeurs de Δ . Dans la suite, nous calculons le noyau de viabilité et le bassin de capture en utilisant l'algorithme de [Deffuant *et al.*, 2007].

Si nous prenons $\Delta = 0.075$, le noyau de viabilité n'est pas la boule entière (voir la figure 3.6(a)). Le volume du noyau de viabilité représente environ 25% du volume de la boule. Ce résultat signifie que pour tous les points qui partent de l'extérieur du noyau de viabilité (75% du volume de la boule), l'évolution va sortir de boule et la distance maximale à l'équilibre pendant ces évolutions sera plus grande que Δ le rayon de la boule centrée sur l'attracteur pour la norme sup.

Si nous considérons de plus petites boules, les noyaux de viabilité rassemblent des évolutions qui restent plus près du point d'équilibre. La proportion du volume du noyau de viabilité par rapport au volume de la boule reste environ la même, autour de 24%. (La figure 3.6 (b) et la figure 3.6 (c) représentent les noyaux de viabilité pour des valeurs de Δ égales à 0,05 et 0,025). Nous regardons la figure 3.1

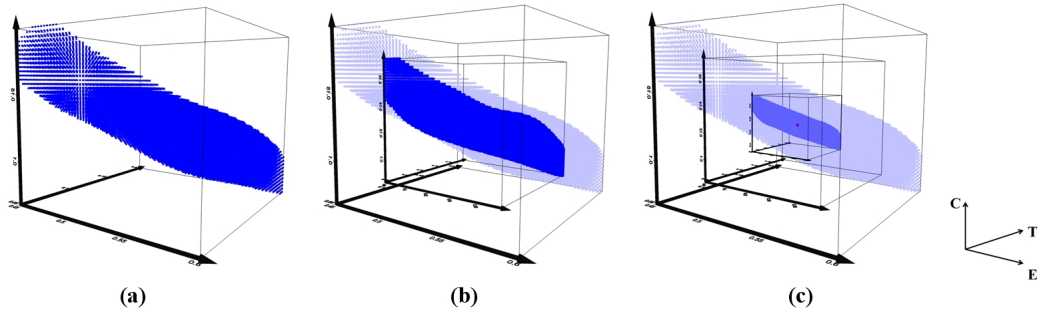


FIGURE 3.6 – Les valeurs de paramètres sont les mêmes que pour la figure 3.1 avec $\epsilon = 0.1$. Les zones bleu foncé représentent le noyau de viabilité de la dynamique (3.1). L'attracteur est $(\bar{T} \approx 0.125, \bar{E} \approx 0.526, \bar{C} \approx 0.125)$ et l'ensemble de contraintes est $K = [\bar{T} - \Delta; \bar{T} + \Delta] \times [\bar{E} - \Delta; \bar{E} + \Delta] \times [\bar{C} - \Delta; \bar{C} + \Delta]$, $\Delta = 0.075$ pour le diagramme (a), $\Delta = 0.05$ pour le diagramme (b) et $\Delta = 0.025$ pour le diagramme (c).

pour expliquer cette observation. La courbe en pointillés fait des oscillations avant de venir près de l'attracteur. L'amplitude de ces oscillations devient de plus en plus petite avec le temps. Mais, lorsque nous réduisons l'ensemble de contraintes, ces oscillations dépassent les contraintes au début de l'évolution. Ce qui montre l'importance de l'information spatiale.

Maintenant, nous choisissons une valeur légèrement plus grande pour ϵ . $\epsilon = 0.13$ correspond à un taux d'investissement légèrement plus grand dans le modèle (3.1). Par l'analyse asymptotique de la section précédente, nous savons qu'il n'y a pas d'équilibre stable, mais plus un cycle limite (figure 3.4). Néanmoins, même si ce n'est plus un équilibre de la dynamique, le point $[\bar{T} \approx 0.125, \bar{E} \approx 0.526, \bar{C} \approx 0.125]$ reste une situation souhaitable avec des valeurs strictement positives pour les trois variables. La question de la détermination de la distance maximale à ce point pendant une évolution reste précieuse dans une perspective de durabilité.

Les figures 3.7 (a), 3.7 (b) et 3.7 (c) représentent les noyaux de viabilité pour les trois mêmes ensembles de contraintes que le paragraphe précédent, mais avec $\epsilon = 0,13$ en paramètre de la dynamique. Les deux premières figures présentent un noyau de viabilité non-vide. Donc, il existe des évolutions qui restent dans l'ensemble des contraintes. Comme toutes les évolutions convergent vers l'attracteur unique qui est le cycle limite, cela implique que le cycle limite est inclus dans la boule qui est centrée en $[\bar{T} \approx 0.125, \bar{E} \approx 0.526, \bar{C} \approx 0.125]$ avec $\Delta = 0.075$ et $\Delta = 0.05$. Au contraire, quand $\Delta = 0.025$, le noyau de viabilité est vide, ce qui signifie que cette petite boule ne contient pas entièrement le cycle limite.

Par conséquent, si les situations souhaitables sont contenues dans la boule centrée sur $[\bar{T} \approx 0.125, \bar{E} \approx 0.526, \bar{C} \approx 0.125]$ avec $\Delta = 0.025$, le modèle (3.1)

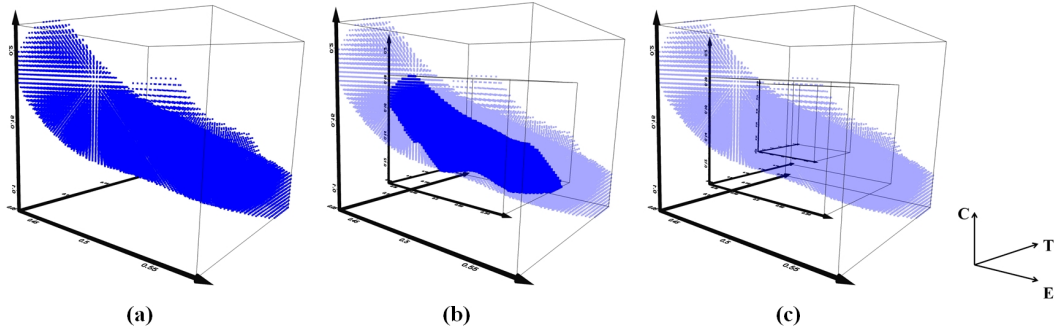


FIGURE 3.7 – Les valeurs de paramètres sont les même que pour la figure 3.1 et $\epsilon = 0.13$. Les zones bleu foncé représentent le noyau de viabilité de la dynamique (3.1), l'ensemble de contraintes est $K = [\bar{T} - \Delta; \bar{T} + \Delta] \times [\bar{E} - \Delta; \bar{E} + \Delta] \times [\bar{C} - \Delta; \bar{C} + \Delta]$, avec ($\bar{T} \approx 0.125$, $\bar{E} \approx 0.526$, $\bar{C} \approx 0.125$) et $\Delta = 0.075$ pour le diagramme (a), $\Delta = 0.05$ pour le diagramme (b) et $\Delta = 0.025$ pour le diagramme (c).

ne fournit aucune trajectoire satisfaisante pour aucune condition initiale. Effectivement, le noyau de viabilité de cette boule est vide, ce qui signifie que pour un point quelconque qui part dans ce domaine, l'évolution sort des contraintes en temps fini. Mais, si des situations souhaitables peuvent être décrites par une plus grande boule $\Delta = 0.05$ par exemple, il existe de nombreux points de départ conduisant à des évolutions souhaitables, même s'il n'y a pas d'équilibre stable, mais un cycle limite.

3.2.2 Le bassin de capture pour l'information temporelle

L'appartenance au bassin d'attraction d'un équilibre stable assure qu'à partir d'un tel point de départ, la trajectoire va converger vers cet équilibre. Mais, nous avons vu dans le paragraphe 3.1.3.3 que le temps nécessaire pour atteindre un voisinage de cet équilibre peut varier énormément selon le point de départ. Cette information peut être fournie par le bassin de capture de la théorie de la viabilité en considérant ce voisinage comme un ensemble cible.

Les situations désirables sont pour cela décrites par une boule autour de l'équilibre stable, ou mieux, par le noyau de viabilité de cette boule : depuis un point quelconque du noyau de viabilité, nous sommes assurés qu'il existe une trajectoire qui restera dans la boule. La figure 3.8 (a) reproduit le noyau de viabilité de la section précédente 3.2.1 qui est représenté dans la figure 3.6 (c). Puis, les huit figures suivantes (Figure 3.8) représentent les bassins de capture du noyau de viabilité pour des temps de capture croissants ($t = 10$, $t = 20$, ..., $t = 80$) à l'intérieur de la zone $K' = [0.4; 0.65] \times [0.05; 0.25] \times [0.05; 0.25] \supset K$. Sur la figure, nous pouvons noter que les points qui appartiennent au bassin de capture sont inégalement répartis autour du noyau de viabilité. Les trajectoires dont le point de départ appartient au bassin de capture représenté dans la fi-

gure 3.8 (i) sont les seules à atteindre le noyau de viabilité en un temps inférieur à 80.

Cette observation nous montre que les points qui sont proches du noyau de viabilité ou même proches de l'attracteur n'atteignent pas nécessairement le noyau de viabilité plus rapidement.

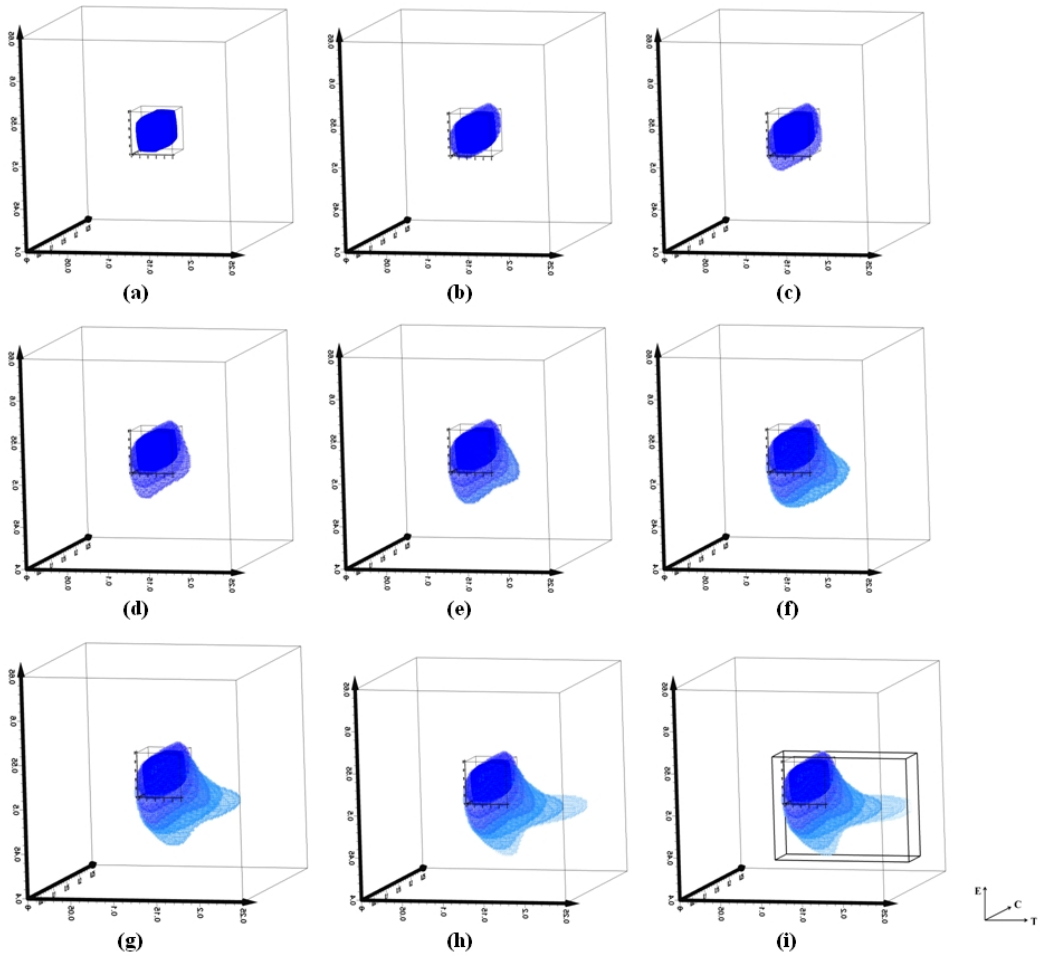


FIGURE 3.8 – (a) est représenté en bleu le noyau de viabilité de la figure 3.6 (c). Les valeurs de paramètres sont les même que la figure 3.6 (c). La grande boule $K' = [0.4; 0.65] \times [0.05; 0.25] \times [0.05; 0.25]$ représente le domaine pour calculer le bassin de capture. (b)... (i) les zones bleu clair sont les bassins de capture de ce noyau de viabilité pour huit différents horizons temporels, $t = 10, \dots, t = 80$.

3.3 Le concept de viabilité et l'analyse de viabilité

3.3.1 La dynamique non contrôlée

Nous avons montré dans la section précédente, la façon dont deux principaux concepts de la théorie de la viabilité, le noyau de viabilité et le bassin de capture, peuvent fournir des informations précieuses qui complètent l'analyse asymptotique dans une perspective d'étude de la durabilité. Après l'étude asymptotique qui fournit les équilibres asymptotiquement stables, il est possible de considérer différents voisinages centrés autour de ces attracteurs (comme les boules de rayons différents) et de calculer le noyau de viabilité de ces voisinages. Les noyaux de viabilité associent des conditions initiales avec la distance maximale à l'attracteur au cours de l'évolution. Le bassin de capture de ces noyaux de viabilité associe des conditions initiales avec le temps nécessaire pour atteindre (et de rester dans) un voisinage donné de l'attracteur. Il est également possible de suivre une approche légèrement différente. Étant donné la dynamique du système, nous pouvons considérer le sous-ensemble de l'espace d'état qui représente l'ensemble des états désirables, qui est un compromis acceptable pour toutes les parties de [Fuentes, 1993] : par exemple, dans le modèle de (3.1), $E \geq \underline{E}$ et $C \geq \underline{C}$ assurent qu'un certain niveau de qualité de l'environnement est préservé ainsi qu'un certain niveau pour les infrastructures. Dans une perspective de durabilité, des informations précieuses sont l'ensemble des points à partir desquels l'évolution reste dans cet ensemble désirable : c'est le noyau de viabilité de cet ensemble désirable qui est considéré comme un ensemble de contraintes. Pour les points à l'extérieur du noyau de viabilité, l'étude de viabilité peut fournir comme information la durée, éventuellement infinie, qui est nécessaire pour rejoindre le noyau de viabilité.

Selon ce point de vue, le concept principal est l'ensemble des états désirables, et non l'attracteur asymptotique : différents comportements asymptotiques peuvent partager le même ensemble d'états désirables, les bifurcations ne sont pas forcément un problème, tant que les attracteurs sont compris dans le même ensemble d'états désirables. L'analyse asymptotique et le point de vue de la viabilité sont cependant liés : si l'ensemble désirable ne contient aucun attracteur (point d'équilibre, cycle limite, etc.), alors le noyau de viabilité sera vide. Mais le paradigme de la viabilité recentre la durabilité sur la définition des états souhaitables, plutôt que sur le choix d'un comportement. C'est pourquoi l'intérêt de l'analyse de la viabilité est encore plus visible quand la dynamique peut être contrôlée, c'est le sujet de la prochaine section.

3.3.2 La dynamique contrôlée

Les problèmes de durabilité impliquent souvent des systèmes socio-écologiques sur lesquelles différentes politiques d'action peuvent être effectuées, l'identification de politiques efficaces constitue donc un enjeu très important pour la gestion. C'est pourquoi il est plus réaliste de considérer des dynamiques contrôlées dans les

questionnements sur la durabilité des systèmes socio-écologiques. Dans ce cadre, il est donc intéressant d'examiner les apports spécifiques de l'analyse de viabilité dans le cas de dynamiques contrôlées.

Dans le modèle de tourisme (3.1), le taux d'investissement ϵ et a en tant que choix du segment touristique (il mesurait la concurrence à l'intérieur de ce segment) peuvent par exemple être considérés comme des variables de contrôle pour le gestionnaire. La dynamique du système est alors décrite par un système dynamique contrôlé :

$$\begin{cases} \dot{T}(t) &= T(t) \left[\mu_E \frac{E(t)}{E(t)+\varphi_E} + \mu_C \frac{C(t)}{C(t)+\varphi_C T(t)+\varphi_C} - \alpha T(t) - a(t) \right] \\ \dot{E}(t) &= E(t) \left[rE(t) \left(1 - \frac{E(t)}{K} \right) - \beta C(t) - \gamma T(t) \right] \\ \dot{C}(t) &= -\delta C(t) + \epsilon(t) T(t) \\ \epsilon(t) &\in [\underline{\epsilon}; \bar{\epsilon}] \\ a(t) &\in [\underline{a}; \bar{a}] \end{cases} \quad (3.3)$$

où $\underline{\epsilon}$ et $\bar{\epsilon}$ (resp. \underline{a} et \bar{a}) sont les limites des taux d'investissement possibles (resp. de la mesure de la compétition exercée par les autres sites touristiques).

Dans ce nouveau modèle de tourisme (3.3), ϵ et a peuvent varier avec le temps donc les évolutions du nouveau système ne sont pas définies au préalable. L'analyse asymptotique, est difficile à utiliser dans ces conditions, la détermination de l'attracteur nécessitant la connaissance des contrôles a priori. L'analyse de viabilité reste pertinente : l'ensemble des états souhaitables peut toujours être défini. L'analyse de viabilité va permettre de calculer l'ensemble qui rassemble des points de départ pour lesquels il existe des fonctions de contrôle $t \rightarrow \epsilon(t)$ et $t \rightarrow a(t)$ qui permettent de maintenir l'état du système dans l'ensemble désirable. C'est le noyau de viabilité. Le second ensemble remarquable rassemble des points de départ pour lesquels il existe des fonctions de contrôle $t \rightarrow \epsilon(t)$ et $t \rightarrow a(t)$, qui permettent d'atteindre l'ensemble K des états souhaitables et de s'y maintenir par la suite. C'est le bassin de capture du noyau de viabilité.

Ainsi, nous considérons l'ensemble défini par :

$$K = [\bar{T} - \Delta; \bar{T} + \Delta] \times [\bar{E} - \Delta; \bar{E} + \Delta] \times [\bar{C} - \Delta; \bar{C} + \Delta]$$

avec $(\bar{T} = 0.15, \bar{E} = 0.55, \bar{C} = 0.15)$ et $\Delta = 0.05$ comme regroupant les états satisfaisants. Nous calculons le noyau de viabilité avec $\underline{\epsilon} = 0.01, \bar{\epsilon} = 0.3, \underline{a} = 6$ et $\bar{a} = 8$ qui permet de déterminer les points initiaux tels qu'il existe des fonctions de contrôle qui maintiennent le système dans des états satisfaisants pendant l'évolution. La figure 3.9 présente ce noyau de viabilité.

Pour tous les points dans le noyau de viabilité, il existe au moins une fonction de contrôle qui permette de maintenir l'évolution dans l'ensemble de contraintes.

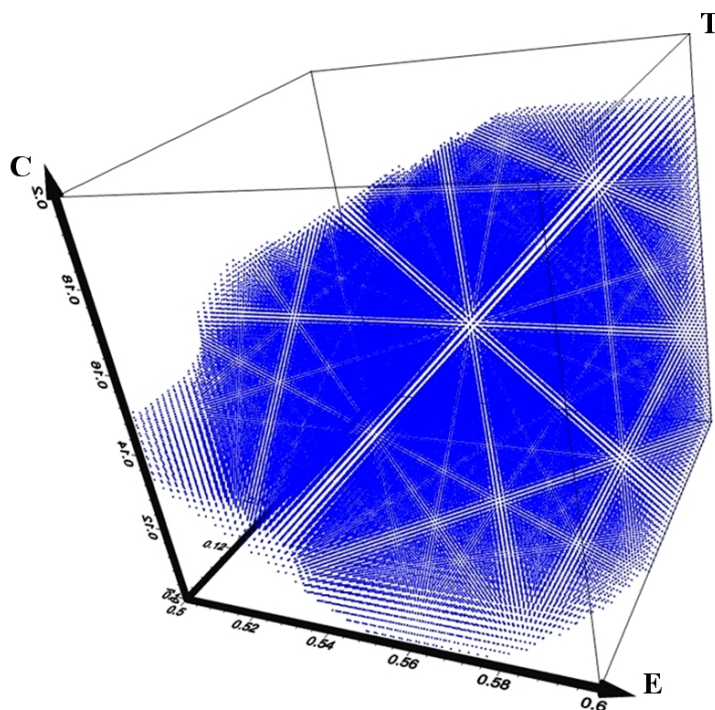


FIGURE 3.9 – La dynamique du modèle (3.3). Les paramètres sont les même que dans la figure 3.1 avec $\underline{\epsilon} = 0.01$, $\bar{\epsilon} = 0.3$, $\underline{a} = 6$ et $\bar{a} = 8$. L'ensemble de contrainte est $K = [\bar{T} - \Delta; \bar{T} + \Delta] \times [\bar{E} - \Delta; \bar{E} + \Delta] \times [\bar{C} - \Delta; \bar{C} + \Delta]$ avec $(\bar{T} = 0.15, \bar{E} = 0.55, \bar{C} = 0.15)$ et $\Delta = 0.05$. Le noyau viabilité est en couleur bleue.

Ces fonctions de contrôle viables sont obtenues comme des conséquences du calcul du noyau de viabilité. La figure 3.10 représente les trajectoires de deux évolutions qui partent du même point : la trajectoire en vert est gouvernée par une fonction de contrôle viable, donc elle reste dans l'ensemble de contraintes. Les points visibles en noir sur cette trajectoire correspondent à des positions où les valeurs du contrôle changent. La trajectoire rouge est obtenue avec le contrôle fixe $\epsilon = 0.01$ et $a = 6$ et sort de l'ensemble de contraintes dans un temps fini.

Évidemment, le noyau de viabilité correspondant à la dynamique contrôlée contient le noyau de viabilité correspondant à la dynamique non contrôlée, car les variations de contrôle offrent des possibilités supplémentaires pour trouver les trajectoires qui restent dans l'ensemble des contraintes.

L'étude de viabilité permet d'obtenir des renseignements précieux sur le comportement du système contrôlé. Dans la suite, on considère le modèle (3.4), pour lequel la valeur de a est fixée. Un seul contrôle est pris en compte, le taux d'investissement ϵ .

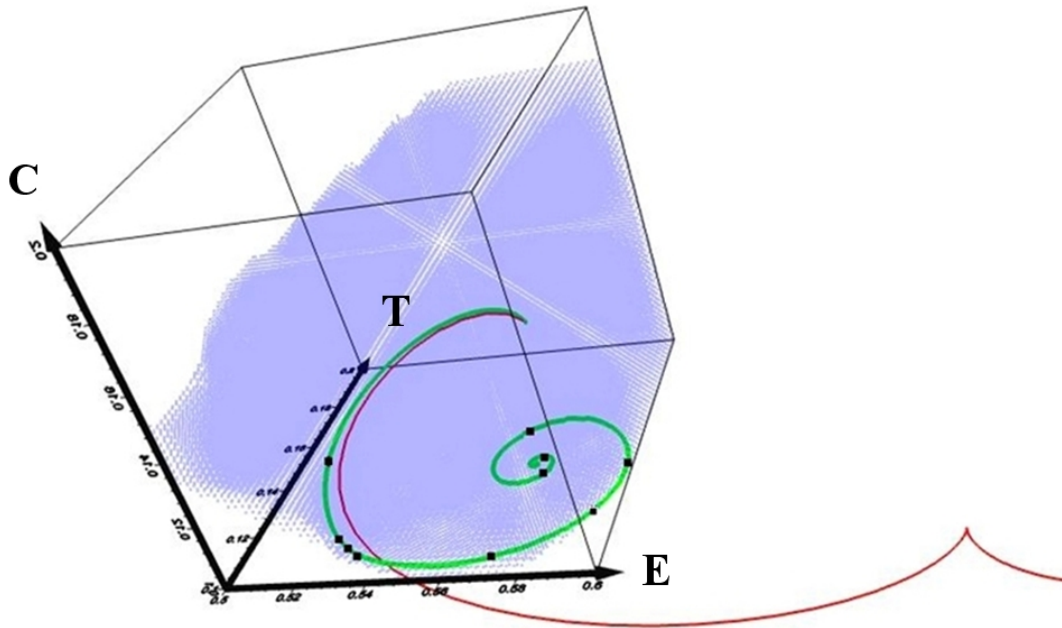


FIGURE 3.10 – Les trajectoires des deux évolutions à partir du point $(T = 0.15, E = 0.58, C = 0.14)$ dans le noyau de viabilité. La trajectoire rouge est obtenue avec les contrôles constants $\epsilon = 0.01$ and $a = 6$. La trajectoire verte est gouvernée par une fonction de contrôle viable. Les points matérialisés sur cette trajectoire correspondent à des positions où les valeurs de contrôle changent.

$$\begin{cases} \dot{T}(t) = T(t) \left[\mu_E \frac{E(t)}{E(t) + \varphi_E} + \mu_C \frac{C(t)}{C(t) + \varphi_C T(t) + \varphi_C} - \alpha T(t) - a \right] \\ \dot{E}(t) = E(t) \left[r E(t) \left(1 - \frac{E(t)}{K} \right) - \beta C(t) - \gamma T(t) \right] \\ \dot{C}(t) = -\delta C(t) + \epsilon(t) T(t) \\ \epsilon(t) \in [\underline{\epsilon}; \bar{\epsilon}] \end{cases} \quad (3.4)$$

On considère des bornes de contrôle de valeur minimale $\underline{\epsilon} = 0.05$ et de valeur maximale $\bar{\epsilon} = 0.3$. Les autres valeurs de paramètres sont les mêmes que pour la figure 3.1.

L'ensemble de contraintes est défini comme :

$$\begin{cases} T \in [0.1; 0.2] \\ E \in [0.4; 0.65] \\ C \in [0.1; 0.2] \end{cases} \quad (3.5)$$

La borne minimale de qualité de l'environnement est désormais égale à 0,4. La valeur de la borne maximale pour E permet d'éviter l'exclusion des situations à haute qualité environnementale. En effet, plus cette borne maximale est élevée,

plus le noyau de viabilité est important. Mais quand cette borne est supérieure à 0.65, le noyau de viabilité n'augmente plus. Le noyau n'atteint jamais la borne supérieure de E : pour des valeurs plus élevées de la borne supérieure de la qualité de l'environnement (au-dessus de 0,65), les évolutions non viables violent d'abord les autres contraintes.

La figure 3.11 présente deux comparaisons entre le noyau de viabilité de la dynamique du modèle contrôlé (3.4) et deux noyaux de viabilité obtenus avec des valeurs fixes de ϵ . Le volume du noyau de viabilité contrôlé (noyau bleu clair) représente environ 19.2% de l'espace de contraintes. Dans le diagramme 3.11 (a), le noyau de viabilité non contrôlé avec $\epsilon = 0.05$ est en bleu foncé. Son volume représente environ 5,6% de l'espace des contraintes. Quand $\epsilon = 0.1$, le volume du noyau de viabilité (couleur bleu foncé dans le diagramme 3.11 (b)) représente environ 6.1% de l'espace des contraintes. Le noyau de viabilité non contrôlé est vide quand $\epsilon = 0.15, 0.2, 0.25$ ou 0.3 .

Calculé pour le système dynamique non contrôlé, le noyau de viabilité permet de distinguer dans l'ensemble des contraintes les points pour lesquels l'évolution restera dans l'ensemble des contraintes, de ceux pour lesquels l'évolution sortira de l'ensemble des contraintes en temps fini. Dans le cas d'un système dynamique contrôlé, le noyau de viabilité sépare les points pour lesquels il existe au moins une fonction de contrôle qui gouverne une évolution qui reste dans l'ensemble de contraintes, et les points pour lesquels le système va quitter l'ensemble de contraintes en temps fini quels que soient les contrôles. Dans une perspective d'analyse de viabilité, les états appartenant au noyau de viabilité peuvent donc être considérés comme durables, à condition qu'une fonction de contrôle viable soit appliquée, puisque l'on sait qu'au moins une telle fonction existe. Les états souhaitables qui ne font pas partie du noyau de viabilité ne sont pas directement intéressants d'un point de vue de durabilité, puisqu'il n'existe aucune fonction de contrôle à partir de ces points, permettant de rester indéfiniment dans l'espace de contraintes.

Dans le cas d'un système dynamique non contrôlé, connaître le temps nécessaire pour atteindre le noyau de viabilité (à partir de conditions initiales qui n'y sont pas) est une information précieuse. Dans le cas d'un système dynamique contrôlé, c'est aussi le bassin de capture du noyau de viabilité qui fournit cette information : plus précisément, pour un temps d'arrivée donnée, le bassin de capture du noyau de viabilité rassemble tous les points à partir desquels il existe une fonction de contrôle qui permet d'atteindre le noyau de viabilité en temps fini inférieur ou égal à ce temps d'arrivée. Le degré de durabilité de ces états peut alors être mesuré par l'inverse du plus petit temps d'arrivée pour lequel l'état appartient au bassin de capture associé : plus le temps est petit, plus le système sera rapidement et définitivement dans un état souhaitable, après une période inéluctable à l'extérieur de l'ensemble de contraintes (à la condition que la bonne fonction de contrôle soit

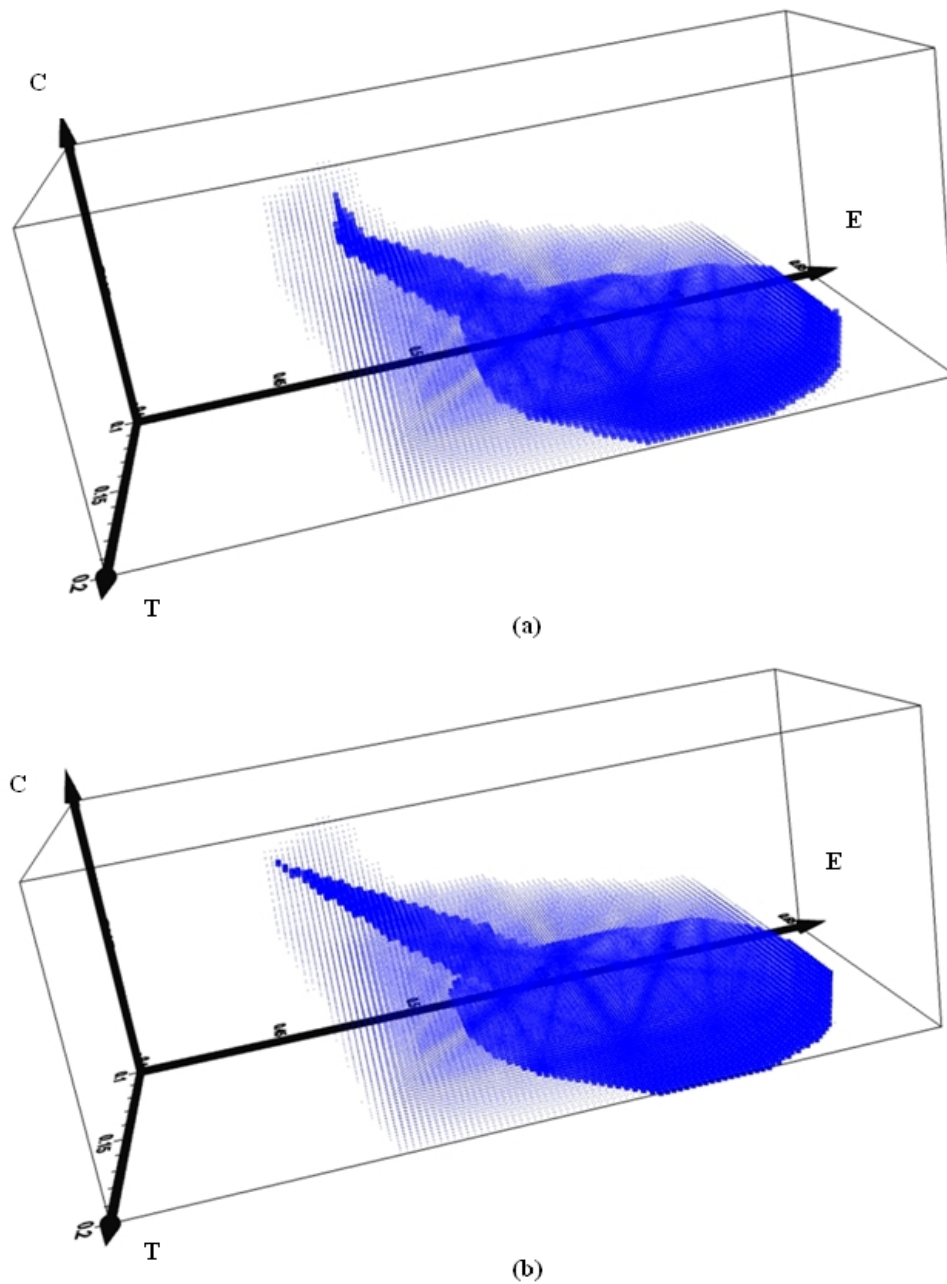


FIGURE 3.11 – En bleu clair, le noyau de viabilité du modèle (3.4). (a) en bleu foncé le noyau de viabilité à contrôle constant $\epsilon = 0.05$, (b) en bleu foncé le noyau de viabilité sans contrôle avec $\epsilon = 0.1$, tous les autres paramètres sont les mêmes que pour la figure 3.1.

appliquée). Dans certains cas, l'étude de viabilité mettra en évidence l'existence d'états à partir desquels il est impossible de rejoindre le noyau de viabilité, quels que soient les contrôles. Ces situations peuvent être considérées comme très dangereuses

du point de vue de la durabilité.

3.4 Conclusion

Nous avons montré dans ce chapitre que l'analyse asymptotique fournit des informations utiles pour comprendre le comportement asymptotique d'un système dynamique. Elle identifie les attracteurs de la dynamique comme les cycles limites, ou les équilibres, vers lesquels l'évolution du modèle va converger finalement. Le diagramme de bifurcation, qui associe les valeurs des paramètres du modèle avec le nombre et le type des attracteurs, peut également fournir des informations pour évaluer la robustesse des zones dans l'espace d'état ou dans l'espace des paramètres. Mais l'analyse asymptotique ne fournit pas certaines informations essentielles pour établir une analyse pertinente dans un objectif d'analyse de la durabilité. Même si un équilibre est considéré comme satisfaisant, sans l'information sur la frontière du bassin d'attraction, il n'est pas possible de prévoir vers quel attracteur une évolution va converger. De plus le comportement transitoire peut mener une évolution très loin de son attracteur ; une évolution peut atteindre le voisinage de l'attracteur mais dans un temps très long. Ces comportements sont en contradiction avec la durabilité.

Afin de prendre en compte ces facteurs spatiaux et temporels dans la définition de la durabilité, nous avons introduit deux concepts principaux de la théorie de la viabilité : le noyau de viabilité et le bassin de capture, qui permettent de renseigner les informations manquantes. Considérant un ensemble d'états acceptables autour d'un attracteur, le noyau de viabilité rassemble tous les états dans ce voisinage à partir desquels une évolution reste toujours dans le voisinage. Le bassin de capture rassemble tous les états à partir desquels le noyau de viabilité est atteint dans un temps donné. Dans les illustrations, nous avons calculé les noyaux de viabilité et les bassins de capture pour le modèle de développement du tourisme (3.1). Nous avons aussi expliqué comment ces concepts et les informations qu'ils fournissent peuvent être étendus à la dynamique contrôlée, qui est souvent utiles pour représenter des systèmes socio-écologiques. Dans le cas du modèle du tourisme, nous avons utilisé comme contrôles le taux d'investissement et la compétition des autres sites pour montrer comment une certaine évolution peut être conservée dans le noyau de viabilité d'un voisinage donné de l'attracteur.

C'est en s'inspirant de cette étude théorique que nous avons étudié la possibilité d'utiliser un agent expert viabilité. Si un système dynamique peut être utilisé pour modéliser le comportement du système, l'intérêt principal de l'analyse de viabilité est de se concentrer sur la définition d'un ensemble d'états souhaitables et comment rester dans cet ensemble d'états, plutôt que sur l'attracteur asymptotique. En effet, il reste seulement aux joueurs à définir les zones souhaitables. L'agent expert viabilité peut aussi suggérer des possibilités de contrôle selon les variables d'état et les

paramètres, afin de modéliser explicitement ces variables de contrôle dans le modèle dynamique. Les outils de théorie de la viabilité peuvent alors être utilisés pour calculer le noyau de viabilité et son bassin de capture, étant donné un ensemble d'états satisfaisants du système, les contraintes de temps et les contrôles admissibles. Nous allons donc présenter dans la partie suivante, comment nous réalisons l'expert agent viabilité dans SimParc.

Troisième partie

Application

Algorithme d'approximation du noyau de viabilité avec procédure de classification

Sommaire

4.1 L'approximation du noyau de viabilité discret	58
4.1.1 Contexte et notations	58
4.1.2 Les étapes de l'algorithme	59
4.1.3 Le théorème de convergence de l'algorithme	59
4.2 Choix de la procédure de classification	60
4.2.1 Les fonctions de classification utilisées	60
4.2.2 Démonstration de la convergence	60
4.3 Implémentation et Tests	62
4.3.1 L'utilisation de la méthode de Runge-Kutta	62
4.3.2 Modèle de croissance de population dans un espace limité . .	62
4.3.3 Modèle de consommation	63
4.4 Conclusion	64

Dans les chapitres suivants, nous allons montrer des briques qui permettent de réaliser un agent expert viabilité dans le contexte de l'outil d'aide à la gestion participative : d'abord l'agent expert viabilité doit pouvoir fournir rapidement aux joueurs des résultats sur la viabilité des solutions proposées. Nous proposons d'adapter l'algorithme d'approximation du noyau de viabilité en utilisant une procédure de classification conçue pour accélérer le temps de calcul. Ensuite, un support logiciel est proposé pour faciliter la manipulation de l'agent expert viabilité. Le logiciel est aussi développé pour des applications différentes.

Nous allons présenter l'algorithme d'approximation du noyau de viabilité dans ce chapitre qui sera divisé en trois parties. Tout d'abord, nous rappelons l'algorithme de calcul d'une approximation du noyau de viabilité à l'aide d'une procédure de classification et les conditions de sa convergence. Ensuite, nous allons décrire l'ensemble des fonctions de classification que nous proposons d'utiliser et montrer qu'elles respectent bien les conditions de convergence précédentes. Enfin, nous allons comparer les noyaux obtenus pour des modèles dont le noyau théorique peut être calculé.

4.1 L'approximation du noyau de viabilité discret avec une procédure de classification

L'approximation du noyau de viabilité est une étape essentielle pour résoudre le problème de viabilité. En pratique, nous avons souvent discrétisé le système et l'espace de contraintes en utilisant l'approximation par l'extérieur. [Saint-Pierre, 1994] a montré sous de bonnes conditions que le noyau de viabilité discret tend vers le noyau du système initial lorsque le pas de la grille h tend vers 0, si le système dynamique est μ -Lipschitz [Brown *et al.*, 1987]. Dans cette section, nous définissons tout d'abord quelques notations, puis nous présentons les étapes de l'algorithme de classification et le théorème de convergence de cet algorithme.

4.1.1 Contexte et notations

Nous considérons un système dynamique discret en temps défini par la correspondance $G : X \mapsto X$ (qui associe à \vec{x} l'ensemble de ses successeurs) :

$$x^{n+1} \in G(x^n). \quad (4.1)$$

Nous supposons que G est une correspondance μ -Lipschitz à images fermées¹, c'est à dire que les images de deux points quelconques ne peuvent pas être éloignées de manière arbitraire, mais au contraire restent dans un voisinage l'une de l'autre proportionnel à la distance des points de départ.

Notre objectif est d'approcher $Viab_G(K)$ qui est le noyau de viabilité de la dynamique discrète (4.1). D'après les théorèmes de viabilité [Aubin, 1991], $Viab_G(K)$ est le plus grand sous-ensemble E de K tel que :

$$\forall \vec{x} \in E, G(\vec{x}) \cap E \neq \emptyset \quad (4.2)$$

L'algorithme utilisant des fonctions de classification décrit dans [Deffuant *et al.*, 2007] est le suivant.

Soit une grille, K_h , telle que :

$$\forall \vec{x} \in K, \exists \vec{x}_h \in K_h, \text{ tel que } \|\vec{x} - \vec{x}_h\| \leq \beta(h) \quad (4.3)$$

avec $\beta(h) \rightarrow 0$ quand $h \rightarrow 0$. Une telle grille existe puisque K est compact.

De plus,

- l est une procédure d'apprentissage qui associe à un ensemble S de paires $(\vec{x}_i, e_i) \in K \times \{-1, 1\}$, une fonction de classification $l_S(\vec{x}) : K \rightarrow \{-1, 1\}$.
- $d(E, F)$, la distance entre deux sous-ensembles E et F .

1. $\forall \vec{x}, \vec{x}' \in K, G(\vec{x}') \subset G(\vec{x}) + \mu \|\vec{x} - \vec{x}'\| B$ où B est la boule unité et $\forall \vec{x} \in K$, l'ensemble $G(\vec{x})$ est fermé

- $E \setminus F$, l'ensemble complémentaire de F dans E lorsque $F \subset E$.

4.1.2 Les étapes de l'algorithme

A chaque étape, sont définis un ensemble discret $K_h^n \subset K_h^{n-1} \subset K_h$, et un ensemble continu, noté $L(K_h^n)$, qui est une généralisation de cet ensemble discret, et qui constitue l'approximation courante du noyau de viabilité :

1. Initialisation : $K_h^0 := K_h$ et $L(K_h^0) := K$.

2. Récurrence :

- Définition de l'ensemble discret K_h^{n+1} à partir de K_h^n et $L(K_h^n)$:

$$K_h^{n+1} = \{\vec{x}_h \in K_h^n \mid d(G(\vec{x}_h), L(K_h^n)) \leq \mu\beta(h)\}$$

- si $K_h^{n+1} \neq K_h^n$, utiliser la procédure d'apprentissage l avec les points \vec{x}_h de la grille K_h , avec l'étiquette $+1$ si $\vec{x}_h \in K_h^{n+1}$ et l'étiquette -1 sinon. Soit l_h^{n+1} la fonction de classification obtenue de K dans $\{-1, 1\}$, $L(K_h^{n+1})$ est défini ainsi :

$$L(K_h^{n+1}) = \{\vec{x} \in K, \mid l_h^{n+1}(\vec{x}) = +1\} \quad (4.4)$$

- sinon, arrêt et retourner $L(K_h^n)$.

4.1.3 Le théorème de convergence de l'algorithme

La démonstration de ce théorème est donnée dans [Deffuant *et al.*, 2007].

Theorem 2 *S'il existe un réel $\lambda \geq 1$ tel que, pour toutes les itérations n , l'approximation $L(K_h^n)$ satisfait les conditions suivantes :*

$$\forall \vec{x} \in L(K_h^n) \quad d(\vec{x}, K_h^n) \leq \lambda\beta(h) \quad (4.5)$$

$$\forall \vec{x} \in K \setminus L(K_h^n) \quad d(\vec{x}, K_h \setminus K_h^n) \leq \beta(h) \quad (4.6)$$

alors, l'algorithme d'approximation du noyau de viabilité fournit un résultat qui converge vers le noyau de viabilité exact lorsque le pas de la grille h tend vers 0.

Ces conditions signifient que tout point de $L(K_h^n)$ doit être près d'un point de K_h^n et que tout point de $K \setminus L(K_h^n)$ doit être près d'un point de $K_h \setminus K_h^n$.

4.2 Choix de la procédure de classification

Malgré leur utilité pour le calcul de noyaux de viabilité, les SVMs classiques ne vérifient pas toujours les conditions du théorème 2. Nous proposons ici d'utiliser des fonctions de type "plus proche voisin" qui vérifient les conditions du théorème. Le choix de cette procédure de classification est de se concentrer sur la réduction du temps de calcul.

4.2.1 Les fonctions de classification utilisées

Nous nous plaçons dans le cas d'une grille K_h régulière de pas h . Ainsi, $\beta(h) = \sqrt{N}h$, où N est la dimension de l'espace des états.

Nous notons y_0 la fonction "plus proche voisin" :

$$\begin{aligned} y_0 &: K \rightarrow K_h \\ \vec{x} &\rightarrow y_0(\vec{x}) = \operatorname{argmin}(d(\vec{x}, K_h)) \end{aligned} \quad (4.7)$$

C'est le point le plus proche de \vec{x} dans K_h , et nous avons $y_0(\vec{x}) \in K_h$ et $d(\vec{x}, y_0(\vec{x})) \in [0, \sqrt{N}h/2]$.

Nous posons également :

$$\begin{aligned} y_i(\vec{x}) &\subset K_h \\ &:= \{\vec{y} \in K_h, d(y_0(\vec{x}), \vec{y}) = h, d(\vec{x}, \vec{y}) \leq \sqrt{N+3}h/2\} \end{aligned}$$

Soit $S \subset K_h$. Si les points de $S \subset K_h$ sont associés à l'étiquette +1 et ceux de $K_h \setminus S$ à l'étiquette -1, nous proposons d'associer la fonction de classification suivante :

$$\begin{aligned} l_S^*(\vec{x}) &= +1 \\ &\text{si } y_0(\vec{x}) \in S \text{ et } \forall \vec{y} \in y_i(\vec{x}) | \vec{y} \notin S, d(\vec{x}, \vec{y}) > \alpha h \\ l_S^*(\vec{x}) &= -1 \\ &\text{sinon,} \end{aligned} \quad (4.8)$$

avec $\alpha \in [0, 1[$ (voir la figure 4.1).

4.2.2 Démonstration de la convergence

Dans cette section, nous allons vérifier que l^* vérifie bien les conditions du théorème 2 :

Satisfaction de la condition (4.5)

Si $l_S^*(\vec{x}) = 1$ alors $y_0(\vec{x}) \in S$. Or, $d(\vec{x}, y_0(\vec{x})) \leq \beta(h)$ donc $d(\vec{x}, S) \leq \beta(h)$.

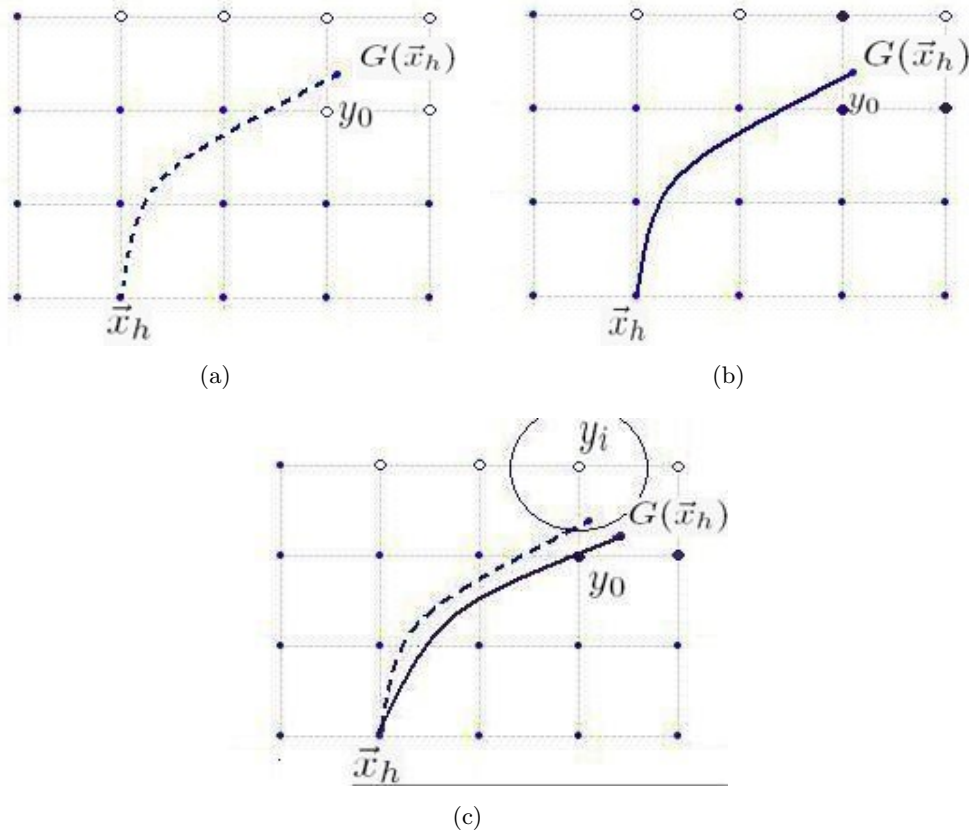


FIGURE 4.1 – Les points noirs appartiennent à K_h^n , les points blancs à son complémentaire dans K_h . Cas 4.1(a), $\vec{x}_h \notin K_h^{n+1}$ car $y_0(G(\vec{x}_h)) \notin K_h^n$. Cas 4.1(b), $\vec{x}_h \in K_h^{n+1}$ car $y_0(G(\vec{x}_h)) \in K_h^n$ et $\forall \vec{y} \in y_i(G(\vec{x}_h)), \vec{y} \in K_h^n$. Cas 4.1(c), $y_0(G(\vec{x}_h)) \in K_h^n$ mais $\exists \vec{y} \in y_i(G(\vec{x}_h)), \vec{y} \notin K_h^n$. Donc pour déterminer si $\vec{x} \in K_h^{n+1}$, il faut vérifier que $d(G(\vec{x}_h), \vec{y}) > \alpha h$.

Satisfaction de la condition (4.6)

Si $l_S^*(\vec{x}) = -1$, nous avons deux possibilités :

- $y_0(\vec{x}) \in S$
- $y_0(\vec{x}) \notin S$

Si $y_0(\vec{x}) \notin S$, $d(\vec{x}, K_h \setminus S) = d(\vec{x}, y_0(\vec{x})) \leq \beta(h)$, et la condition (4.6) est satisfaite.

Si $y_0(\vec{x}) \in S$, $\exists y \in y_i(\vec{x})$ tel que $y \notin S$ et $d(\vec{x}, y) \leq \alpha h$. Ainsi, $d(\vec{x}, K_h \setminus S) \leq \alpha h \leq h \leq \beta(h)$.

4.3 Implémentation et Tests

Dans cette section, nous rappelons d'abord la méthode de Runge-Kutta que nous utilisons pour calculer la solution. Nous comparons ensuite les résultats obtenus pour deux modèles pour lesquels les Noyaux de Viabilité Théoriques (noté NVT) ont pu être calculés.

4.3.1 L'utilisation de la méthode de Runge-Kutta

Le logiciel Kviar utilise la méthode d'Euler pour résoudre les équations différentielles, cette méthode fait des erreurs surtout pour des grandes valeurs de dt . Nous avons donc choisi la méthode Runge-Kutta classique d'ordre quatre pour diminuer les erreurs.

Les méthodes de Runge-Kutta ([Runge, 1895] et [Kutta, 1901]) sont des méthodes d'analyse numérique [Süli et Mayers, 2003] très couramment utilisées pour l'approximation de solutions d'équations différentielles ordinaires [Butcher, 2005]. L'avantage de ces méthodes est d'être faciles à programmer et assez stables pour les fonctions courantes de la physique [Butcher, 1975].

Nous avons choisi la méthode de Runge-Kutta classique d'ordre quatre ($RK4$) qui est un cas particulier d'usage très fréquent. Cette méthode requiert 4 évaluations de $y' = f(x, y)$ par pas d'intégration h : une au point de départ de l'intervalle, 2 aux demi-intervalles, et une à la fin de l'intervalle. Nous considérons $y(t_0) = y_0$, la méthode $RK4$ est donnée par l'équation :

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) + O(h^5)$$

où

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1\right) \\ k_3 &= f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2\right) \\ k_4 &= f(t_n + h, y_n + hk_3) \end{aligned}$$

Le terme d'erreur est alors d'ordre h^5 , l'erreur totale accumulée est de l'ordre h^4 . L'utilisation de méthode de $RK4$ permet d'obtenir une approximation plus fine du noyau réel et les formules sont aussi valables pour des fonctions à valeurs vectorielles, c'est-à-dire en plusieurs dimensions.

4.3.2 Modèle de croissance de population dans un espace limité

Ce modèle a été élaboré par Maltus et Verhulst. L'état du système est décrit par deux variables, $(x(t), y(t))$: $x(t)$ représente la taille de la population et $y(t)$ son

taux d'accroissement. La dynamique à temps discret avec un pas de temps dt est la suivante :

$$\begin{cases} x(t + dt) &= x(t) + x(t)y(t)dt \\ y(t + dt) &= y(t) + u(t)dt \text{ avec } |u(t)| \leq c \end{cases} \quad (4.9)$$

où c est la variation maximale du taux d'accroissement à chaque pas de temps.

[Aubin et Saint-Pierre, 2006] rajoutent des contraintes sur la taille de la population qui doit rester entre deux bornes : l'ensemble des contraintes est ainsi défini par $K = [a, b] \times R$, avec $a > 0$.

L'équivalent continu de ce système a un noyau de viabilité étudié par [Aubin et Saint-Pierre, 2006] :

$$\text{Viab}(K) = \left\{ (x, y) \in R^2 \mid x \in [a; b], y \in [-\sqrt{2.c.\log(\frac{x}{a})}; \sqrt{2.c.\log(\frac{b}{x})}] \right\}$$

Nous prenons comme valeurs des paramètres $a = 0$, $b = 3$, $c = 0.5$. Le tableau 4.1 rassemble les résultats de quatre simulations avec des nombres de points par dimension croissants : tous les noyaux calculés ont un volume qui représente entre 55% et 60% du volume de l'ensemble des contraintes. La fidélité au NVT (mesurée par l'écart en volume entre le NVT et le noyau calculé) est supérieure à 90% et elle augmente avec le nombre des points de la grille. Nous avons également comparé le temps de calcul avec celui du logiciel Kviar [Deffuant et al., 2007], le temps de calcul est toujours plus faible et le gain augmente avec le nombre de points.

La figure 4.2 représente les résultats des simulations du tableau 4.1 avec $dt = 0.1$. Les résultats que nous avons obtenus sont sous forme d'un ensemble de points que nous notons *Noyau.App*. L'approximation est bien sûr meilleure quand le nombre de points augmente (c'est le théorème de convergence).

4.3.3 Modèle de consommation

Ce problème est étudié par [Aubin, 1991], le problème est modélisé en deux dimensions $x(t)$ et $y(t)$. $x(t)$ représente la consommation d'une matière première et $y(t)$ représente son prix. L'ensemble de contraintes est $K = [0; b] \times [0; e]$. La variation des prix entre deux pas de temps est déterminée par un variable de contrôle $u(t)$ et bornée par le paramètre c . La dynamique du système représente la consommation d'une matière première, freinée par les prix :

$$\begin{cases} x(t + dt) &= x(t) + (x(t) - y(t))dt \\ y(t + dt) &= y(t) + u(t)dt \text{ avec } |u(t)| \leq c \end{cases} \quad (4.10)$$

Pour ce système simple, il est possible de déterminer analytiquement le noyau de viabilité [Aubin, 1991] du système continu équivalent :

Simulations	Simu 1	Simu 2	Simu 3	Simu 4
Nb pts dim x	56	112	448	896
Nb pts dim y	80	160	640	1280
Nb pts total	4480	17920	286720	1146880
NoyauApp/K(%)	59.88	57.19	55.34	55.33
Nb Iterations	18	23	34	34
Taux NVT(%)	89.97	93.33	95.16	95.47
Temps	8s	2m09s	22m	1h40m
Temps(Kviar)	19s	2m59s	1h59m	>10h

TABLE 4.1 – Toutes les simulations utilisent les valeurs $dt = 0.1$ et $\alpha = 0.5$. Pour chaque simulation nous avons renseigné le nombre de points selon x , le nombre de points selon y , le nombre de points de la grille ($x*y$), le volume du noyau de viabilité approché en pourcentage du volume de l'ensemble des contraintes (nombre de points de *NoyauApp*/ nombre de points total), le nombre d'itérations avant l'arrêt de l'algorithme, le volume du NVT en pourcentage du volume du noyau de viabilité approché (nombre de points communs au *NoyauApp* et au NVT) / (nombre de points de *NoyauApp*). Les 2 dernières lignes renseignent les temps de calcul respectifs de notre algorithme et de Kviar. Toutes les simulations sont effectuées avec un processeur Core 2,13 GHz d'Intel (TM) 2 CPU, 2 Go RAM.

$$\text{Viab}(K) = \left\{ (x, y) \mid x \in [a; b], y \in [x - c + c.\exp^{-\frac{x}{c}}; x + c - c.\exp^{-\frac{x-b}{c}}] \right\}$$

Nous prenons $b = 2$, $c = 0.5$ et $e = 3$. La figure 4.3 représente deux simulations d'approximation du noyau de viabilité avec $dt = 0.1$.

Dans le tableau 4.2, tous les volumes des noyaux approchés représentent entre 25% et 27% du volume de l'ensemble des contraintes, la fidélité au NVT est supérieure à 97%, et elle augmente avec le nombre des points. Le temps de calcul est beaucoup plus faible que celui de Kviar quand le nombre de points augmente. La figure 4.3 représente les résultats des simulations du tableau 4.2 avec $dt = 0.1$.

4.4 Conclusion

Nous avons présenté un algorithme d'approximation du noyau de viabilité, basé sur une procédure de classification de type "plus proche voisin". Nous avons montré que l'approximation obtenue convergeait vers le noyau de viabilité exact lorsque le pas d'espace tend vers 0. Nous avons comparé les résultats de notre algorithme avec les solutions exactes dans le cas de deux modèles pour lesquelles les noyaux exacts ont pu être calculés. Nous avons montré dans ces deux cas que l'écart au noyau théorique est inférieur à 10 % et décroît avec le nombre de points.

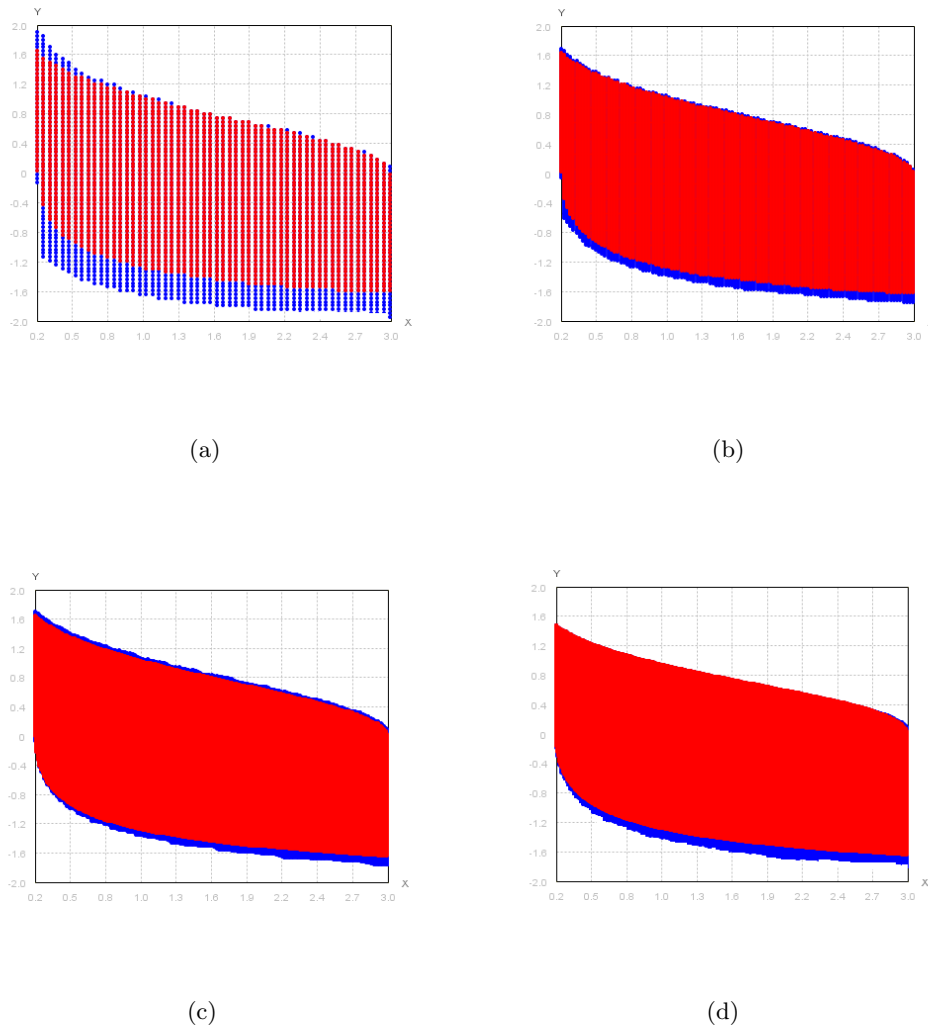


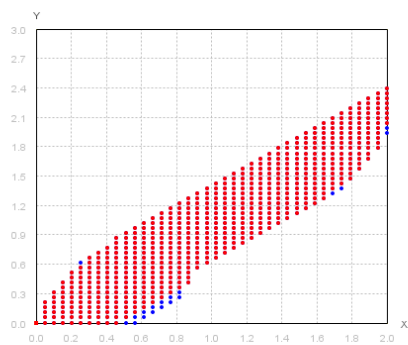
FIGURE 4.2 – Les approximations du noyau de viabilité du problème de population avec $\alpha = 0.5$ et $dt = 0.1$ du tableau 4.1. L'axe horizontal représente la taille de la population (x) et l'axe vertical son taux d'accroissement (y). L'ensemble des contraintes est le rectangle noir. Le noyau de viabilité théorique (NVT) est représenté en rouge. Le noyau de viabilité calculé approche le NVT par l'extérieur, les erreurs sont en bleu. Les graphiques (a) à (d) correspondent aux simulations 1 à 4 respectivement.

En outre, le temps de calcul, pour les simulations effectuées, est divisé par deux et jusqu'à dix suivant les cas, par rapport au logiciel Kviar qui utilise des SVMs comme fonctions de classification. Cela permettra de rendre le jeu sérieux SimParc plus fluide en diminuant le temps d'attente de l'analyse de viabilité. Lors de la séance de test organisée au LIP6, l'utilisation de cet algorithme a permis plusieurs échanges

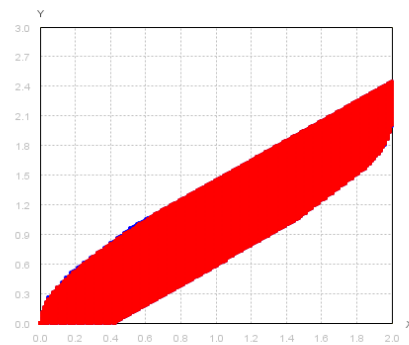
Simulations	Simu 1	Simu 2	Simu 3	Simu 4
Nb pts dim x	40	160	480	960
Nb pts dim y	60	240	720	1440
Nb pts total	2400	38400	345600	1382400
NoyauApp/K(%)	26.29	25.35	25.31	25.28
Nb Iter	13	33	34	36
Taux NVT(%)	97.01	99.5	99.92	99.94
Temps	9s	4m01s	22m	1h13m
Temps(Kviar)	17s	11m54s	2h21m	>10h

TABLE 4.2 – Tous les paramètres du tableau sont les mêmes que pour le tableau 4.1. La méthode de classification proposée ici est plus rapide et ce d'autant plus que le nombre de points augmente.

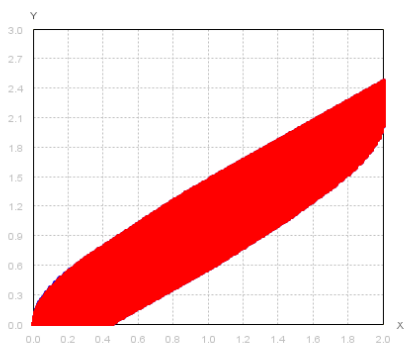
entre les joueurs, le temps d'attente pour obtenir les résultats est acceptable pour ce jeu de rôle.



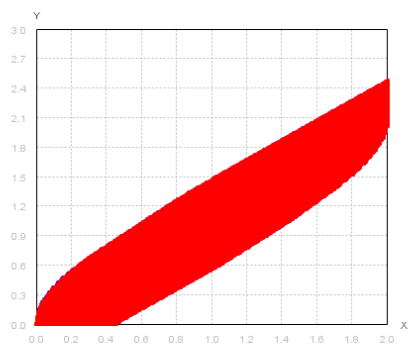
(a)



(b)



(c)



(d)

FIGURE 4.3 – Approximations du noyau de viabilité du problème de consommation avec $\alpha = 0.5$ et $dt = 0.1$ du tableau 4.2. L'axe horizontal représente la consommation (x) et l'axe vertical le prix (y). L'ensemble des contraintes est le rectangle noir. Le noyau de viabilité théorique (NVT) est figuré en rouge. Le noyau de viabilité s'en écarte pour les points bleus. Les graphiques (a) à (d) correspondent aux simulations 1 à 4 respectivement.

Conception d'un support logiciel pour résoudre le problème de viabilité

Sommaire

5.1	Les motivations	70
5.2	Conception de l'application	70
5.2.1	Utilisation Maven	71
5.2.2	Diagramme des paquetages de l'application	71
5.2.3	Description des paquetages	72
5.3	Interface Graphique Utilisateur	76
5.3.1	Visualisation du noyau de viabilité	77
5.3.2	Observation des trajectoires	79
5.4	Conclusion	82

Nous allons présenter dans cette section un nouveau support logiciel pour analyser les différents problèmes de viabilité. Les principaux problèmes de viabilité que nous considérons sont l'approximation du noyau de viabilité, l'approximation du bassin de capture, l'analyse de la résilience [Martin, 2004] et la robustesse [Alvarez *et al.*, 2010]. Nous implémentons et intégrons à l'instant l'approximation du noyau de viabilité l'observation de l'évolution du système. Les autres fonctionnalités est en cours de développer. Ce logiciel est proposé pour mieux conduire des expérimentations numériques afin d'observer les comportements globaux lorsque les paramètres varient. Il comprendra notamment les entrées (définition des systèmes dynamiques et de l'espace de contraintes) et les sorties (visualisation graphique).

Nous allons d'abord parler des motivations dans la section 5.1, à savoir faciliter les utilisations de l'agent expert viabilité, et développer un cadre logiciel pour le développement d'autres algorithmes. Dans la section 5.2, nous allons présenter la conception de l'application en utilisant les diagrammes d'UML¹. Dans la section 5.3, nous allons présenter l'interface graphique utilisateur de ce logiciel et les résultats graphiques fournis par ce logiciel.

1. langage de modélisation unifié, en anglais *Unified modeling Language*

5.1 Les motivations

Faciliter l'utilisation

Dans le jeu de rôle d'aide à la gestion participative SimParc [Briot *et al.*, 2011], les joueurs sont amenés à réfléchir sur différents domaines d'une aire protégée, et le jeu comporte deux phases de négociation assez conséquentes. Ceci amène les joueurs à utiliser l'expert viabilité de manière répétée, avec des modifications substantielles des contraintes et éventuellement des paramètres du modèle. Nous avons décidé de développer une interface graphique utilisateur pour jouer le rôle de l'agent expert viabilité qui doit donc être très facile à comprendre et à manipuler.

Un cadre logiciel commun

Les applications de la théorie de la viabilité sont beaucoup étudiées depuis ces dernières années, plusieurs projets se sont développés autour de cette théorie. Par exemple, le projet européen PATRES contient plusieurs cas d'études ([Deffuant et Gilbert, 2011]), l'objectif du projet est de développer des nouvelles méthodes qui définissent les actions favorables pour rétablir le système après des perturbations. L'objectif du projet ANR Dédution est de tester l'hypothèse selon laquelle le formalisme de la théorie de la viabilité associé à l'évaluation de propriétés comme la résilience constitue un outil efficace pour l'ingénierie du développement durable dans le cadre de l'agriculture au sens large [Martin, 2010]. Des nouveaux outils et des algorithmes ont été réalisés pendant ces recherches. Pour éviter de devoir réécrire complètement un logiciel à la conception de chaque nouvel algorithme, il est devenu nécessaire de construire un support logiciel commun, qui puisse être exploité par tout nouvel algorithme. En effet, les utilisateurs peuvent ajouter leur propre composant supplémentaire (les modèles, les contraintes, les algorithmes ou des outils), sans modifier les autres composants.

5.2 Conception de l'application

L'objectif de cet outil logiciel est la réalisation d'une interface graphique qui permet aux utilisateurs de sélectionner le modèle d'un système dynamique et de le paramétrer, puis de sélectionner un type de problème et de paramétrer les contraintes correspondantes. Il permet aussi aux utilisateurs de sélectionner un algorithme et de le paramétrer pour résoudre ce problème et calculer l'approximation du noyau de viabilité ou résilience. Nous avons conçu cette interface en trois composants individuels : le composant du système dynamique, le composant de l'ensemble des contraintes et le composant de l'algorithme. Son avantage est que lorsque nous voulons ajouter ou supprimer un composant du modèle, les autres composants ne seront pas affectés. Si nous voulons construire un nouveau modèle ou mettre en place un nouvel algorithme, nous avons juste besoin de créer le nouvel objet, puis de l'intégrer dans ce logiciel.

5.2.1 Utilisation Maven

L'un des objectifs de ce logiciel est de créer une communauté avec toutes les personnes qui travaillent avec la viabilité. Pour faciliter et d'automatiser certaines tâches de la gestion de notre logiciel, nous utilisons Maven pour développer notre logiciel Java. Apache Maven est un outil logiciel libre pour la gestion et l'automatisation de production des projets logiciels Java en général. Il permet de produire un logiciel à partir de ces sources, en optimisant les tâches réalisées à cette fin et en garantissant le bon ordre de fabrication. L'avantage de cette utilisation pour des agents experts est de pouvoir mettre à jour les différentes versions de modèles ou les valeurs de paramètres dès sont validés.

Maven utilise un paradigme connu sous le nom de *Project Object Model (POM)* afin de décrire un projet logiciel, ses dépendances à des modules externes et l'ordre à suivre pour sa production. Chaque projet ou sous-projet comprend les informations nécessaires à Maven pour traiter le projet (nom du projet, numéro de version, dépendances vers d'autres projets, bibliothèques nécessaires à la compilation, noms des contributeurs, la synchronisation transparente de module, etc.). Ce paradigme est décrit par un fichier *pom.xml* à la racine du projet. L'annexe A décrit le fichier *pom.xml* de ce projet logiciel.

5.2.2 Diagramme des paquetages de l'application

Pour que les codes soient faciles à comprendre et à relire, nous définissons différents paquetages figurant sur la figure 5.1 pour hiérarchiser les classes. Les structures de programmation logiquement liées sont regroupées dans un même paquetage.

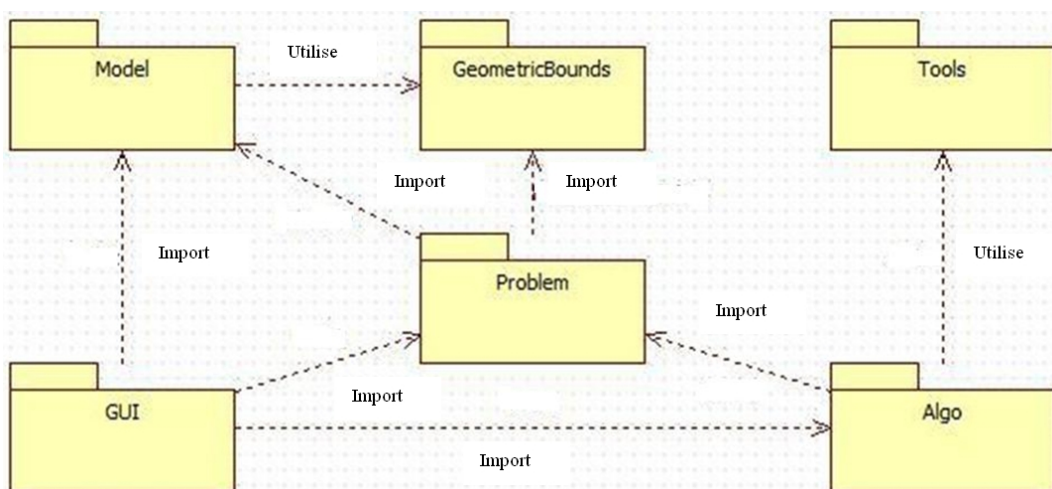


FIGURE 5.1 – Diagramme de paquetage.

Le paquetage *Model* contient tous les modèles de système dynamique. Le paquetage *GeometricBounds* contient les formes géométriques, elles nous servent pour définir les ensembles de contraintes ou les résultats en formes géométriques. Le paquetage *Algo* contient les différents algorithmes pour résoudre le problème de viabilité. Le paquetage *Problem* permet de définir un problème de viabilité et de choisir un algorithme pour le résoudre. Le paquetage *Tools* contient des outils divers. Le paquetage *GUI* contient les composants de l'interface graphique.

5.2.3 Description des paquetages

Model

Un paquetage spécial pour les systèmes dynamiques. Il y a une classe de base abstraite appelé *DynamicSystem* qui déclare les paramètres et les fonctions de l'opération. Chaque classe dérivée hérite de cette classe de base et correspond à un modèle dynamique. Nous implémentons certaines opérations communes dans la classe de base pour éviter les réécritures du code, et nous implémentons les méthodes abstraites dans chaque classe dérivée. La figure 5.2 présente le diagramme de classes du paquetage Model.

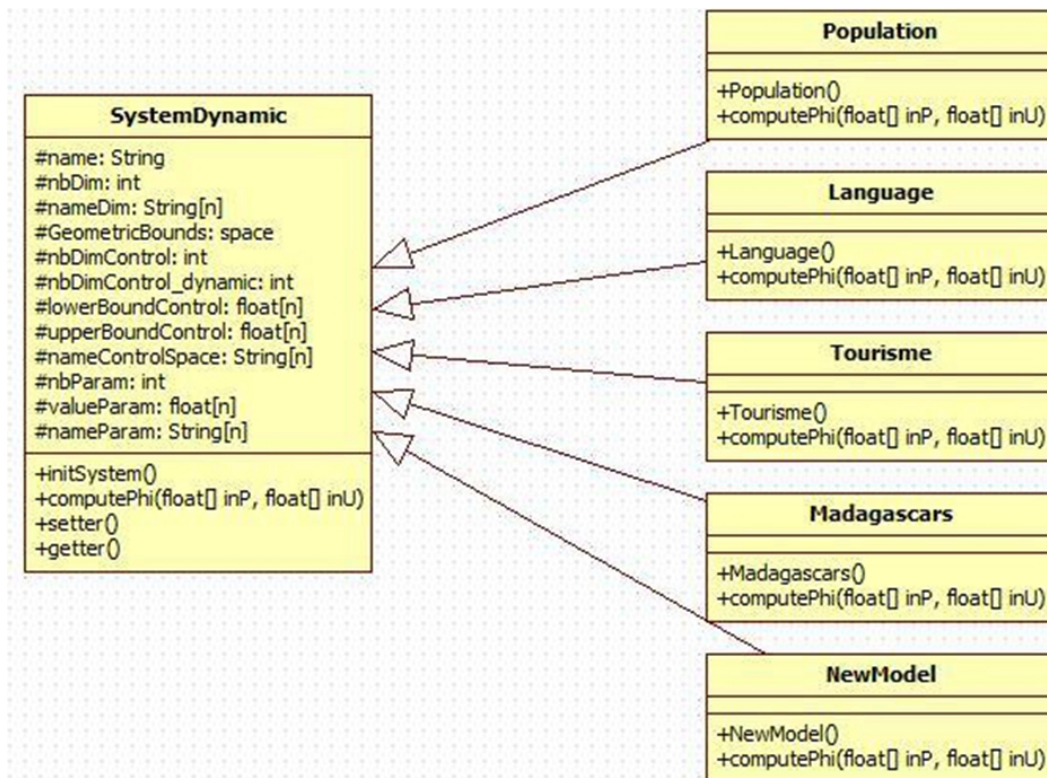


FIGURE 5.2 – Diagramme de classes du paquetage *Model*.

Pour ajouter un nouveau modèle dynamique, nous avons juste besoin de créer une nouvelle classe qui hérite la classe de base *DynamicSystem* comme dans l'exemple suivant, puis de prédéfinir les valeurs pour les paramètres et implémenter les méthodes abstraites.

```

public class NewModel extends DynamicSystem {
    public NewModel(){
        super ();
        this.name = "NewModel"; \\nom du modèle.
        this.nbDim = 2; \\ nombre de dimensions.
        this.nbParam=1; \\ nombre de paramètres.
        this.nbDimControl = 1; \\nombre de contrôles.
        this.initSystem(); \\ initialiser le modèle.
        \\ valeur minimale du contrôle 1.
        this.lowerBoundControl[0] = -0.5f;
        \\ valeur maximale du contrôle 1.
        this.upperBoundControl[0] = 0.5f;
        \\ nom de dimension 1.
        this.nameDim[0]="NameDim1";
        \\ nom de dimension 2.
        this.nameDim[1]="NameDim2";
        \\ nom de paramètre 1.
        this.nameParam[0] = "NameParameter";
        \\ valeur de paramètre 1.
        this.valueParam[0] = value;
        \\ valeur minimale de chaque dimension.
        float [] lower = new float []{value1, value2};
        \\ valeur maximale de chaque dimension.
        float [] upper = new float []{value1, value2};
        \\ nom du contrôle 1.
        this.nameControlSpace[0] = "Name";
    }

    @Override
    public float [] computePhi(float [] inP, float [] inU)
    {
        \\Description du système dynamique.
        ...
    }
    ...
}

```

GeometricBounds

Un paquetage spécial pour les formes géométriques. Il a la même structure que le paquetage Model. La classe de base abstraite est appelé *GeometricBounds*. Chaque classe dérivée correspond à une forme géométrique, car chaque contrainte ou résultat peut être considéré comme un type de forme géométrique. La figure 5.3 présente le diagramme de classes de ce paquetage.

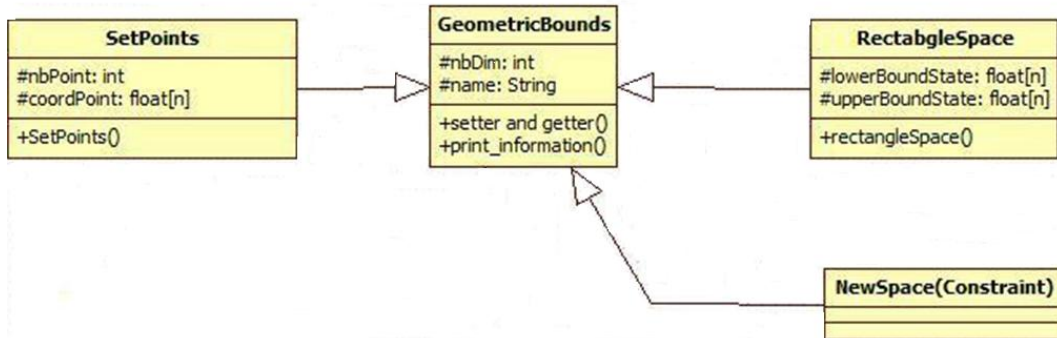


FIGURE 5.3 – Diagramme de classes du paquetage *GeometricBounds*.

Selon les différents besoins de formes géométriques, les utilisateurs peuvent créer des nouvelles classes comme dans l'exemple ci dessous.

```

public class NewGeometric extends GeometricBound {
    public NewGeometric () { ... }
    ...
}
  
```

Algo & Problem

La figure 5.4 présente le diagramme de classes du paquetage *Problem* et *Algo*. Le paquetage *Problem* contient une classe appelée *Problems* qui sert à enregistrer toutes les informations nécessaires du problème de viabilité (le modèle dynamique, les ensembles de contraintes et l'algorithme). Il contient aussi trois classes de base abstraites pour résoudre les problèmes de viabilité (correspondant à l'approximation du noyau de viabilité, l'approximation du bassin de capture et la résilience). Actuellement, nous avons seulement implémenté la classe abstraite *KernelSolve* pour approximer le noyau de viabilité. Nous souhaitons ajouter des nouvelles classes pour résoudre les problèmes du bassin de capture et de la résilience. Les classes dérivées sont implémentées dans le paquetage *Algo*. L'avantage de cette conception est que s'il y a des nouveaux algorithmes qui contiennent de nombreuses classes, nous pouvons créer un nouveau paquetage pour ranger ces codes, et sans modifier l'autre paquetage. Nous avons implémenté pour l'instant l'algorithme qui

a été présenté dans le chapitre 4.

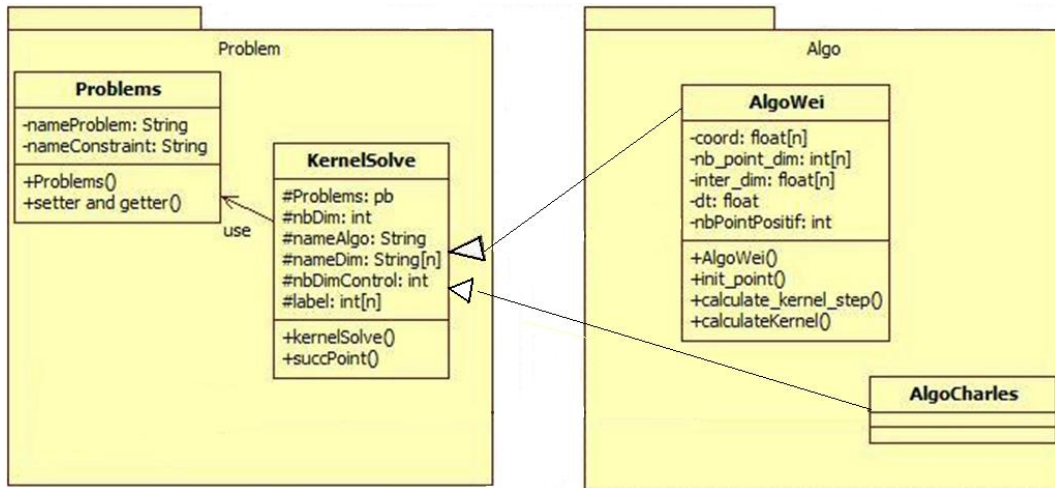


FIGURE 5.4 – Diagramme de classes du paquetage *Algo* et *Problem*.

GUI

Le paquetage *GUI* qui contient tous les éléments pour l'interface graphique utilisateur. La figure 5.5 représente le diagramme de classes du paquetage *GUI*. Il existe une classe *mainFrame* qui décrit le panel principal de l'interface graphique, et quatre sous-panels considérés comme quatre composants de cette interface. Chaque composant contient les classes qui permettent de créer les panels qui affichent les informations. Il contient les composants du panel *Model*, le composant du panel *contraint noyau*, le composant du panel *contraint résilience* et le composant du panel *algorithme*. Le panel *Model* (resp. panel *contraint noyau*, panel *contraint résilience* et panel *algorithme*) est réalisé avec une classe de base abstraite *config-PanelModel* (resp. *configPanelkernelConstraint*, *configPanelResilienceConstraint* et *configPanelAlgo*). Toutes les instances des modèles (resp. panel *contraint noyau*, panel *contraint résilience* et panel *algorithme*) sont réalisées par ses classes dérivées.

Une fois que le nouveau modèle (resp. Géométrique et algorithme) a été créé dans le paquetage *Model* (resp. *Géométrique* et *algorithme*), nous avons besoin de créer des nouveaux panels dans ce paquetage comme dans l'exemple ci-dessous (les formes géométriques peuvent être utilisées pour les contraintes noyau et les contraintes résiliences).

```

public class PanelModelNewModel extends ConfigPanelModel {
    public PanelModelNewModel () { ... }
}
  
```

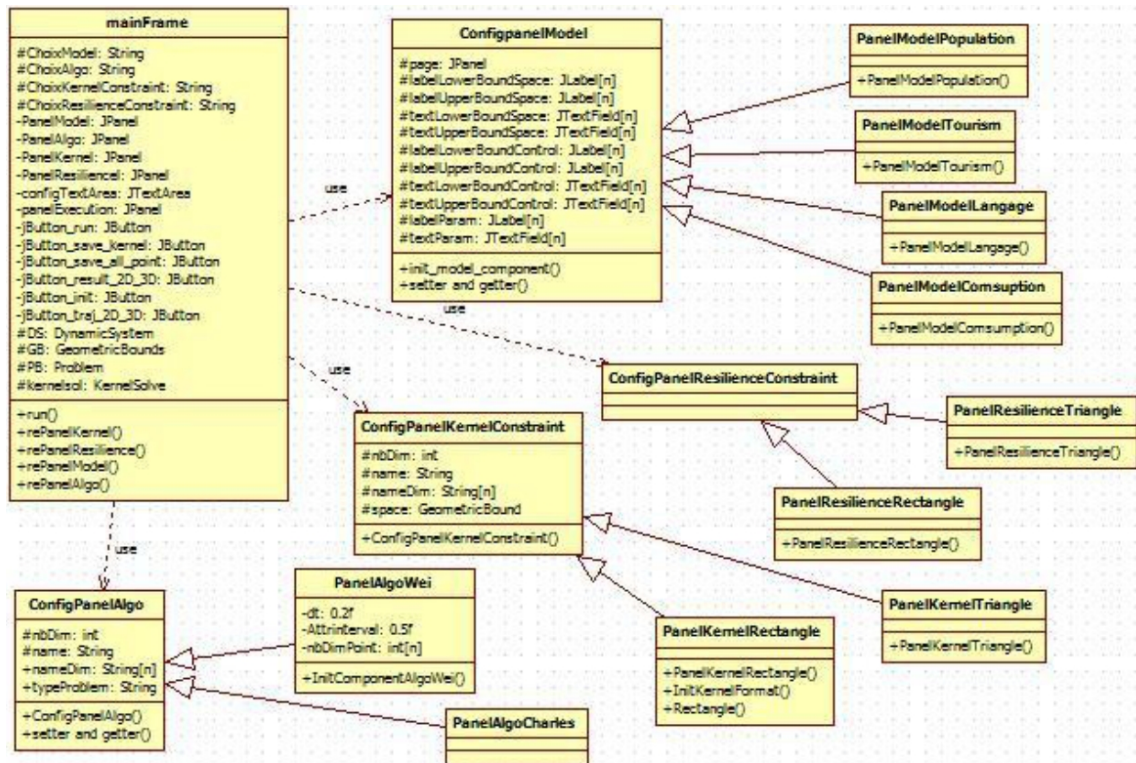


FIGURE 5.5 – Diagramme de classes du paquetage GUI.

```

    ...
}

```

Tools

Nous mettons dans ce paquetage tous les outils ou les méthodes qui aident à la programmation. Par exemple, *Arith.java* sert à faire les calculs entre les tableaux, *Plot.java* dans le paquetage sert à la visualisation, etc.

Comme nous souhaitons partager ces travaux avec d'autres personnes travaillant sur la viabilité, cette manière de ranger les codes devient très importante, il permet aux autres utilisateurs de participer plus facilement au développement.

5.3 Interface Graphique Utilisateur

Dans cette section, nous parlons de l'interface graphique utilisateur. Cette version graphique de l'application permet de visualiser les informations des

problèmes de viabilité, et aussi permet aux utilisateurs de configurer les paramètres du modèle et les contraintes, ainsi que les paramètres pour les algorithmes. Les utilisateurs peuvent aussi exécuter des simulations, puis visualiser et étudier les résultats.

La figure 5.6 présente la fenêtre principale de cette interface graphique utilisateur. La fenêtre est découpée en cinq parties. Dans la partie en haut de l'interface à gauche, les utilisateurs peuvent choisir le modèle dynamique sur lequel ils veulent travailler, et le type de problème qu'ils veulent étudier avec des boîtes déroulantes. Le sous-panel (appelé *Configuration of Tourism* sur la figure 5.6) sous les boîtes déroulantes affiche les variables, les paramètres et les contrôles du modèle dynamique, les utilisateurs peuvent modifier ces valeurs par l'interface sans toucher les codes. Une fois le modèle dynamique choisi et paramétré, les informations du panel vont changer automatiquement selon le modèle. La partie au milieu de l'interface à gauche (appelé *Configuration of Kernel* sur la figure 5.6) permet aux utilisateurs de définir l'ensemble des contraintes. Il contient deux sous panels, les contraintes pour le problème de noyau à gauche, et les contraintes pour le problème de bassin de capture ou de résilience à droite. Les contraintes sont considérées comme des formes géométriques et sont choisies dans les boîtes déroulantes. La partie en bas de l'interface à gauche (appelé *Configuration of AlgoWei* sur la figure 5.6) permet aux utilisateurs de choisir l'algorithme pour résoudre le problème. Chaque algorithme possible possède ses propres paramètres, les utilisateurs peuvent les modifier.

En haut à droite de l'interface, une console résume les différents paramètres choisis, et quelques résultats liés à l'approximation. Une fois le calcul lancé, cette console affiche aussi les informations de chaque itération, comme le nombre d'itérations, le pourcentage du volume du noyau, le temps de calcul, etc. La dernière partie en bas à droite, appelée *Exécution* permet de piloter l'application. Tout d'abord, le bouton *run* lance l'algorithme jusqu'à obtenir l'approximation finale. Le bouton *init* permet d'initialiser les paramètres par défaut. Le bouton *save point* sauvegarde tous les points dans un fichier *'txt'* avec leurs signes. Le nom du fichier est défini par défaut : *nom du modèle_année_mois_jour_heure_minute_seconde.txt*. Le bouton *save kernel* sauvegarde que les points avec un signe positif dans un fichier *'txt'*, c'est-à-dire tous les points du noyau de viabilité. Le nom du fichier est défini par défaut : *nom du modèle_kernel_année_mois_jour_heure_minute_seconde.txt*. Le bouton *vue 2D 3D* permet d'afficher la fenêtre de visualisation en 2D ou en 3D.

5.3.1 Visualisation du noyau de viabilité

Lorsque l'utilisateur appuie sur le bouton *vue 2D 3D*, la fenêtre de visualisation va s'ouvrir automatiquement. Si le modèle dynamique est en 2D, le résultat sera en 2 dimensions. La figure 5.7 donne un exemple (modèle Population, détaillé à la section 4.3.2) de fenêtre de visualisation en 2D.

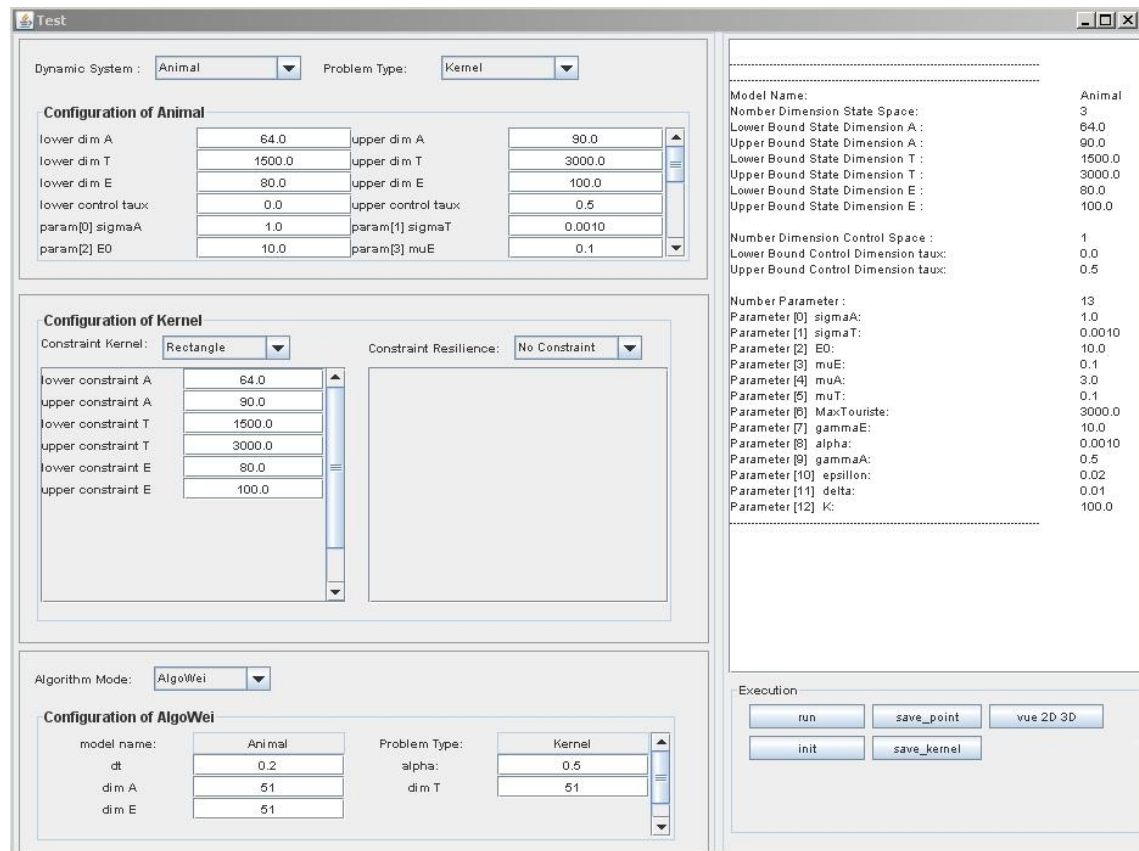


FIGURE 5.6 – La fenêtre principale du logiciel.

Si le modèle dynamique est en 3D, le résultat sera en 3 dimensions. La figure 5.8 donne un exemple de fenêtre de visualisation en 3D.

La fenêtre est découpée en 2 parties, la partie en haut est le panneau de commande, la partie en bas est l'image associée à la simulation. L'image par défaut est le noyau de viabilité. Le noyau n'affiche que les points de sa frontière en couleur bleu clair.

Il y a six options supplémentaires (sept pour un graphique en 3 dimensions) en bas du panneau de commande à gauche qui aident à observer l'image. La première option permet de faire bouger l'image. La deuxième option peut zoomer l'image. La troisième option est pour centrer l'image. La quatrième option permet de modifier les échelles de l'image par le panneau de commande comme dans la figure 5.9. La cinquième option permet de sauvegarder l'image. La sixième option peut observer et enregistrer les données des points par le panneau comme dans la figure 5.10, les données contiennent les bornes des variables, l'ensemble des contraintes, tous les points de la frontière du noyau de viabilité et les coordonnées des trajectoires

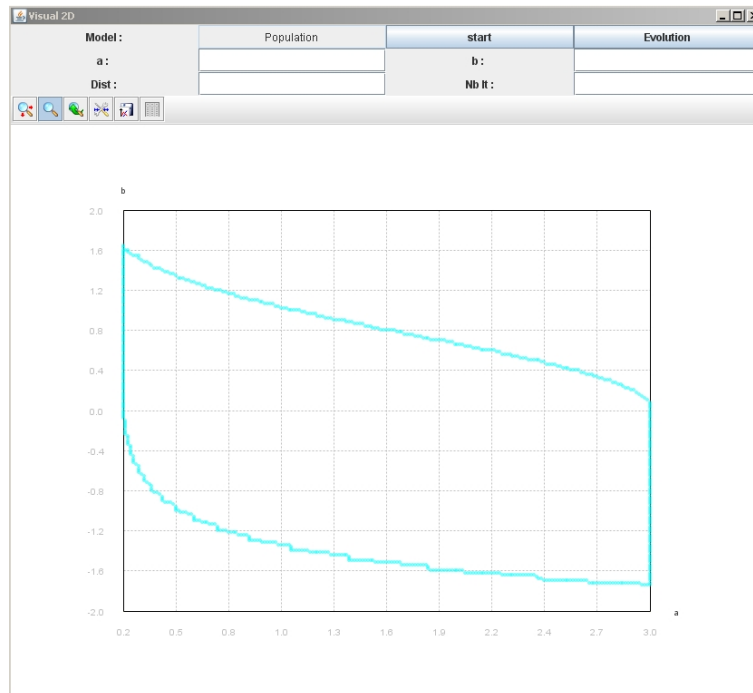


FIGURE 5.7 – Fenêtre de visualisation de l’approximation du noyau de viabilité en 2 dimensions.

à chaque itération. L’option supplémentaire pour 3 dimensions permet de tourner l’image en 3D.

La présentation graphique est plus facile à comprendre que les notions de viabilité. Cette visualisation du noyau de la viabilité permet aux joueurs d’observer la taille et le positionnement des états souhaitables sous les contraintes proposées. Nous avons seulement effectué la visualisation pour 2D et 3D, pour les dimensions supérieures à 3, nous avons prévu de voir des coupes 2D ou 3D en choisissant les variables.

5.3.2 Observation des trajectoires

En utilisant le panneau de commande dans la figure 5.7 ou 5.8, nous pouvons observer une évolution à partir d’un point dans le noyau de viabilité. Les utilisateurs doivent saisir les coordonnées du point de départ (les labels correspondent à chaque nom de dimension), le nombre de pas de la grille² (label *Dist* dans la figure 5.11) et le nombre d’itérations³ (label *Nb it* dans la figure 5.11). Lorsque

2. Le contrôle change la valeur quand la distance de l’évolution à la frontière est inférieure à un seuil. Ce seuil est égal à ce nombre \times pas de la grille. Si les distances de toutes les évolutions sont inférieures à ce seuil, il garde le contrôle qui ramène le plus loin de la frontière.

3. Nous discrétisons le système dynamique en temps dt (voir section 4.1.1). Chaque itération équivalant à un dt .

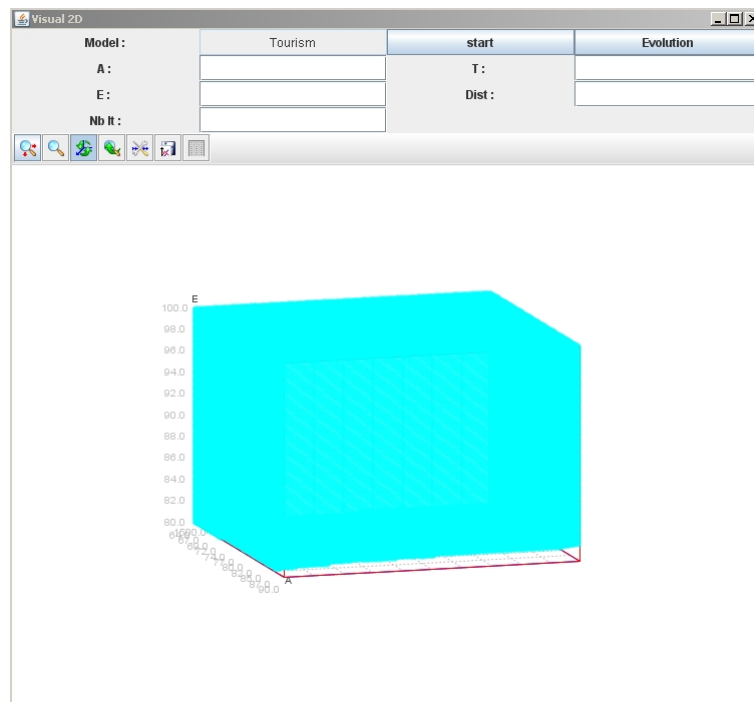


FIGURE 5.8 – Fenêtre de visualisation de l’approximation du noyau de viabilité en 3 dimensions.

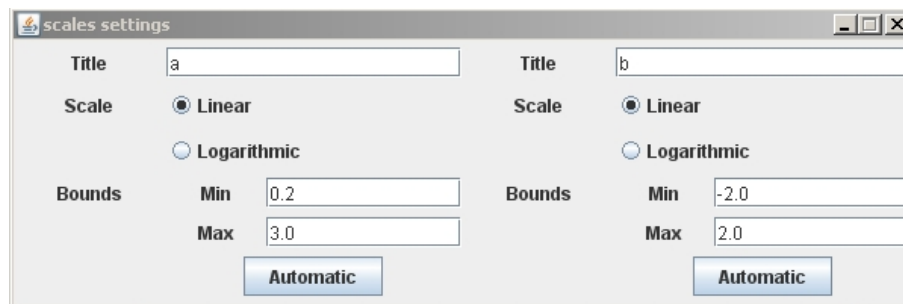


FIGURE 5.9 – Le panneau de commande pour modifier l’échelle de l’image.

ces paramètres sont saisis, le bouton *start* permet de tracer une trajectoire. Si les coordonnées du point de départ saisies appartiennent au noyau de viabilité, il y aura une trajectoire qui s’affiche dans le graphe comme sur la figure 5.11. Sinon, il y aura une fenêtre qui affiche que le point saisi n’est pas dans le noyau. Si aucune valeur n’est saisie, les valeurs sont considérées comme égales à zéro par défaut. Les informations de chaque itération sont affichées dans la console de la fenêtre principale : le numéro de l’itération, à quel moment change le contrôle, quel est le contrôle choisi, etc.

Une fois la trajectoire tracée, le bouton *Evolution* permet d’observer l’évolution

0.20000000298023224	-0.07999999821186066
0.20000000298023224	0.0
0.20000000298023224	0.07999999821186066
0.20000000298023224	0.1599999964237213
0.20000000298023224	0.23999999463558197
0.20000000298023224	0.3199999928474426
0.20000000298023224	0.4000000059604645
0.20000000298023224	0.47999998927116394
0.20000000298023224	0.5600000023841858
0.20000000298023224	0.6399999856948853
0.20000000298023224	0.72000000286102295
0.20000000298023224	0.8000000011920929
0.20000000298023224	0.8799999952316284
0.20000000298023224	0.9599999785423279
0.20000000298023224	1.0399999618530273
0.20000000298023224	1.1200000047683716
0.20000000298023224	1.20000000476837158
0.20000000298023224	1.2799999713897705
0.20000000298023224	1.36000000143051147
0.20000000298023224	1.4400000057220459
0.20000000298023224	1.5199999809265137
0.20000000298023224	1.6000000023841858
0.25600001215934753	-0.6399999856948853
0.25600001215934753	-0.5600000023841858
0.25600001215934753	-0.47999998927116394

FIGURE 5.10 – Le panneau des données.

par rapport au pas de temps de chaque dimension, ainsi que les changements du contrôle à chaque pas de temps. Les résultats s'affichent dans une nouvelle fenêtre comme dans la figure 5.12. L'utilisateur peut choisir la dimension dans la boîte déroulante en haut de la fenêtre à droite.

Ce composant permet aux joueurs de tracer une évolution en donnant un point de départ dans le noyau de viabilité. Il permet de prévoir les politiques d'actions, de manière à maintenir l'évolution dans un ensemble de contraintes défini par les joueurs. L'interface contient une fonction importante qui permet de contrôler la distance de l'évolution à la frontière, il est indispensable pour comprendre la forme du noyau de viabilité et choisir des trajectoires intéressantes [Alvarez et Martin, 2011].

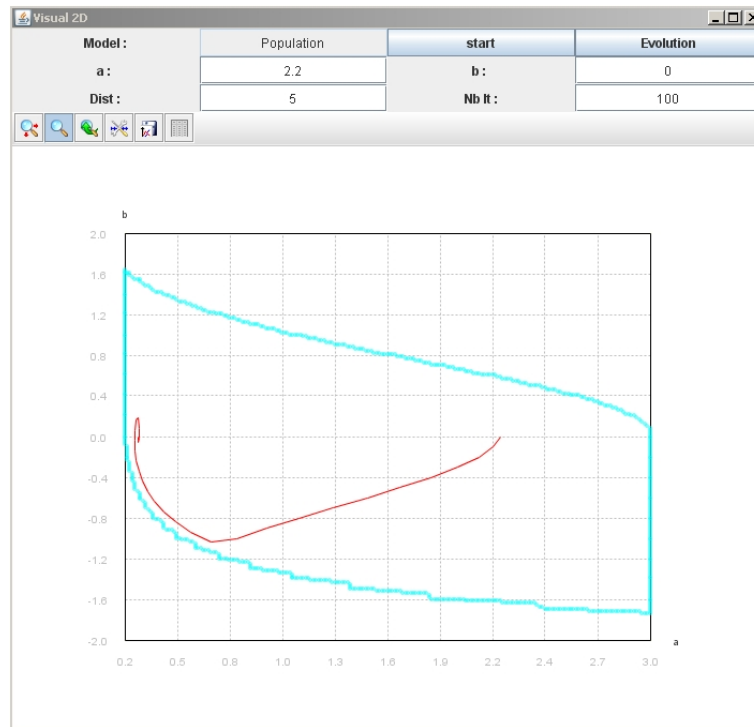
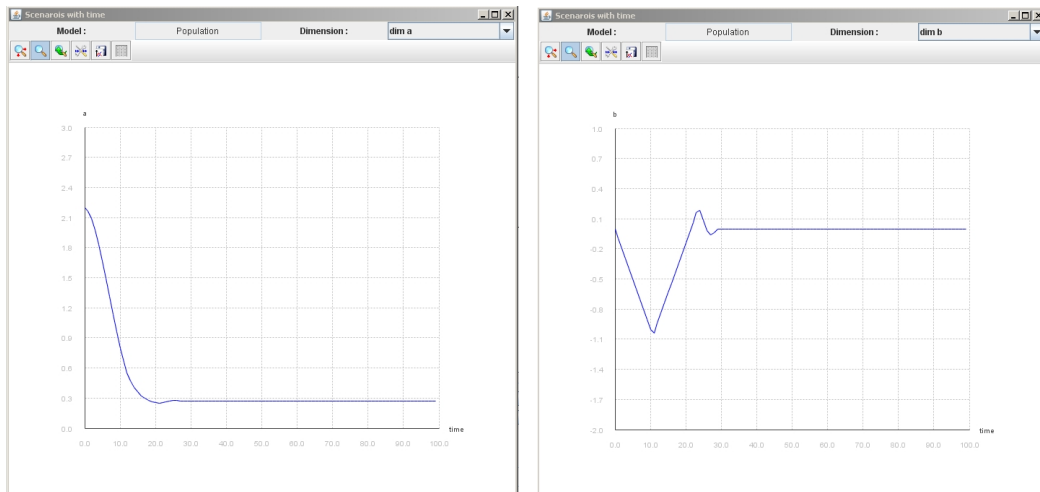


FIGURE 5.11 – L'exemple de la trajectoire du modèle de Population, le point de départ est $a=2.2$ et $b=0$, le nombre de pas de la grille est 5, le nombre d'itération est 100.

5.4 Conclusion

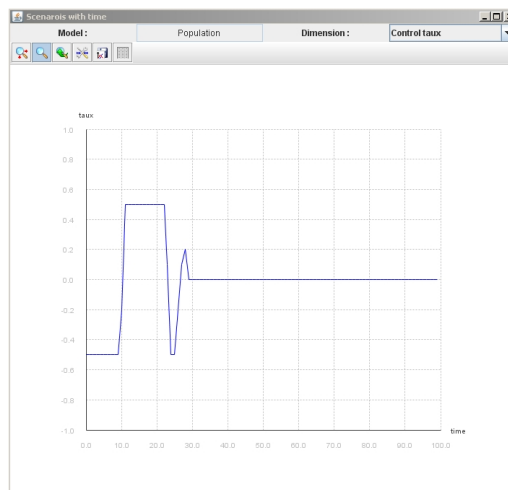
Nous avons réalisé dans cette section une description du logiciel qui permet d'étudier certains problèmes de viabilité. Ce logiciel est développé en Java en tant qu'une interface graphique pour faciliter l'utilisation. Pour utiliser l'agent expert et lancer un calcul de noyau de viabilité, les utilisateurs doivent juste sélectionner un modèle et un type de contraintes (les utilisateurs peuvent modifier les paramètres du modèle ou modifier les bornes des contraintes si besoin), puis définir les paramètres de l'algorithme. Nous avons rangé les classes dans les différents paquetages et nous avons présenté les diagrammes de classes en *UML* pour les expliquer. Nous avons découplé les modules modèles, contraintes et algorithmes pour faciliter la création et l'intégration des nouveaux modèles, contraintes ou algorithmes si nécessaire.

Les utilisateurs peuvent aussi observer les résultats graphiques, car l'observation visuelle est plus facile à comprendre pour les utilisateurs. Les utilisateurs peuvent visualiser le noyau de viabilité, observer une évolution dans le noyau à partir d'un point de départ, étudier les évolutions par rapport au temps de chaque dimension et les contrôles choisis à chaque pas de temps. Les utilisateurs peuvent aussi utiliser les options implémentées pour mieux étudier les résultats. Les utilisateurs peuvent



(a)

(b)



(c)

FIGURE 5.12 – Les évolutions par rapport au temps de chaque dimension, l'évolution est celle de la figure 5.11. Le diagramme (a) présente l'évolution pour la dimension a, le diagramme (b) présente l'évolution pour la dimension b, le diagramme (c) présente l'évolution des contrôles.

observer quelques résultats dans une console, et peuvent sauvegarder les images et les données.

L'expérimentation de l'agent expert viabilité dans le jeu SimParc

Sommaire

6.1	Définition du problème de viabilité	86
6.2	Modèle dynamique de fonctionnement du problème	87
6.2.1	Description du modèle	87
6.2.2	Développement du modèle	88
6.3	Les résultats fournis par l'agent expert viabilité	90
6.3.1	Comparaison des noyaux de viabilité	90
6.3.2	Visualisation de l'évolution viable	91
6.4	L'expérimentation de l'agent expert viabilité	93
6.4.1	Conditions d'expérimentation	93
6.4.2	Description de l'expérimentation	95
6.5	Conclusion	98

Dans les deux chapitres précédents, nous avons effectué des briques qui permettent de construire l'agent expert viabilité. Dans ce chapitre, nous allons étudier la possibilité d'intégrer ces deux briques dans le jeu de SimParc pour tester l'impact de l'agent expert viabilité.

Dans la section 6.1, nous allons montrer comment les joueurs définissent leur problème de viabilité et comment chaque joueur évalue et affine les contraintes, grâce à l'agent expert viabilité. Dans la section 6.2, nous décrivons un modèle simple, adapté à la problématique d'une zone du parc support de SimParc. Ce modèle nous a permis d'effectuer les premiers tests de l'agent expert viabilité. Dans la section 6.3, nous présentons les premiers résultats fournis par l'agent expert viabilité qui permettent d'aider les joueurs à interpréter leurs simulations. Nous présentons ensuite deux sessions de test avec comparaison entre le jeu sans l'agent expert viabilité et le jeu avec l'agent expert viabilité pour observer l'influence de l'agent expert viabilité.

6.1 Définition du problème de viabilité

Dans le chapitre 5, nous avons présenté une interface graphique permettant aux utilisateurs d'utiliser l'agent expert viabilité. Un problème reste à résoudre : comment les joueurs définissent leur problème de viabilité ?

Selon la théorie de la viabilité, les joueurs qui veulent faire fonctionner l'agent expert viabilité sont invités à définir ce qu'on appelle l'ensemble des états souhaitables pour le parc ou la zone protégée. Les choix des variables d'états et de l'évolution possible du parc peuvent également être définis par le joueur parmi un ensemble de choix possibles. Le joueur doit aussi décider de ce qu'il ou elle considère comme les contrôles possibles pouvant être utilisés pour modifier l'état du parc (plutôt que de laisser le parc évoluer par lui-même). Par exemple, différents types d'actions sont possibles : l'investissement, le nettoyage, la plantation, etc.

Dans la pratique, nous construisons à l'avance une famille de systèmes dynamiques. Les joueurs peuvent ainsi choisir entre ces modèles, ils peuvent tester ces modèles et définir la valeur de leurs paramètres. De plus, les joueurs fournissent leurs contraintes sur les variables. L'expérimentation a été faite avec une version adaptée du modèle du chapitre 3, mais dans ce cadre, et compte tenu de la réglementation brésilienne, ce modèle était trop axé développement, et donc nous avons développé un nouveau modèle. Nous avons proposé un modèle adapté à une zone particulière du parc (unité de paysage 3) fictif servant de base au jeu SimParc actuel. Cette unité de paysage est choisie car elle est relativement simple par rapport aux autres. Le modèle décrit les interactions à un niveau abstrait entre l'activité touristique, la qualité de l'environnement et le développement d'espèces animales.

Dans le tableau 6.1 nous présentons un ensemble de contraintes correspondant à différents modes de gestion recommandés par le comité du Parc. Elles peuvent être considérées comme une référence pour les joueurs. Cependant, les joueurs peuvent modifier les contraintes s'ils ne sont pas d'accord (Il faut l'autorisation du gestionnaire). Au cours du jeu (voir section 2.2.3), chaque joueur propose ses propres contraintes pour chaque unité de paysage suivant son intérêt et ses propres motivations. Les joueurs peuvent également demander à l'agent expert de faire une analyse de viabilité. Si les joueurs obtiennent des résultats intéressants, ils peuvent utiliser ces résultats pour discuter avec d'autres joueurs à la troisième étape. Il est possible que les joueurs n'obtiennent pas les résultats souhaités. En effet, certains joueurs peuvent obtenir un noyau vide, ce qui signifie que le parc ne peut pas rester dans un état qui respecte l'ensemble des contraintes de leur choix. Au cours de la quatrième étape, le gestionnaire du parc examine les propositions et engage une proposition finale pour chaque unité de paysage. L'utilisation de l'expert viabilité permet au joueur de discuter directement des contraintes. La négociation entre les acteurs concerne la définition de la zone souhaitable. Il est plus facile de discu-

ter de la position de la frontière plutôt que du fait de restreindre le mode de gestion.

Modes	Intangible	Primitive	Extensive	Historico-culturelle	Intensive	Conflit d'utilisation
T	0	<200	200↔700	400↔1500	1500↔3000	0↔1000
E	100%	>90%	>80%	>70%	>60%	>70%
A	100%	>95%	>90%	>85%	>80%	>85%

TABLE 6.1 – Ensemble de contraintes des différents modes de gestion recommandés par le comité du Parc. Les autres modes sont temporaires.

Nous pouvons ajouter un rôle : *assistance agent expert viabilité* pendant le jeu de SimParc. Les utilisateurs peuvent faire appel à l'agent expert viabilité eux-mêmes, ou demander une assistance agent expert viabilité pour les aider.

6.2 Modèle dynamique de fonctionnement du problème

6.2.1 Description du modèle

L'objectif principal du jeu SimParc est de concilier la préservation de la biodiversité et l'inclusion sociale. Le modèle étudié au chapitre 3 de la thèse ne convenait pas à la réglementation très restrictive des parcs brésiliens. C'est pourquoi, dans la zone test de l'agent expert viabilité, nous avons proposé de travailler dans le cadre d'une problématique simplifiée. Le système considéré prend en compte l'activité touristique (inclusion sociale), la qualité de l'environnement et la préservation d'une espèce animale (biodiversité). Le modèle comporte trois variables : la qualité de l'environnement E , les animaux à protéger A et l'activité touristique T . Nous nous intéressons à la qualité de l'environnement et à la présence d'animaux. En effet, le système modélisé concerne le cas où les touristes viennent dans le parc principalement pour observer les animaux dans leur habitat naturel.

Les animaux

Les animaux A considérés sont des superprédateurs (au sommet de la chaîne alimentaire, ils n'ont pas de prédateurs. Ce sont par exemple des oiseaux, des tortues, des mammifères supérieurs). Ils se multiplient librement suivant une loi de Verhulst [Verhulst, 1847] avec un taux d'accroissement maximal des animaux γ_A . La capacité d'accueil des animaux $\sigma_A(E - E_0)$ dépend de la qualité de l'environnement (plus l'environnement est bon, plus la capacité d'accueil est élevée) où σ_A est la valeur de référence. En revanche, en dessous d'un seuil propre à l'espèce (E_0), les animaux cessent de se reproduire quand l'environnement est mauvais.

$$A' = A * (\gamma_A(\sigma_A(E - E_0) - A))$$

Comme dans un modèle proie prédateur, la mortalité des animaux croit avec la fréquentation touristique avec un facteur σ_T ($\sigma_T \ll 1$). L'équation est la suivante :

$$A' = A * (\gamma_A(\sigma_A(E - E_0) - A)) - \sigma_T T A$$

Les touristes

Les touristes T viennent pour voir les animaux. En revanche, les touristes n'aiment pas la surfréquentation touristique et le départ des touristes est d'autant plus important que le nombre de touristes est élevé. $\varpi_E, \varpi_A, \varpi_T$ sont les poids relatifs aux contributions de E, A, T sur la variation du tourisme. Nous avons donc :

$$T' = T * (\varpi_E E + \varpi_A A - \varpi_T T)$$

L'environnement

L'environnement E se régénère tout seul avec un ralentissement lorsqu'il se rapproche de sa valeur maximum K (nous utilisons encore une équation de Verhulst). Le taux d'accroissement maximal est γ_E .

$$E' = E * \gamma_E \left(1 - \frac{E}{K}\right)$$

Les touristes abîment l'environnement (avec leurs déchets par exemple) avec un coefficient α .

$$E' = E * \gamma_E \left(1 - \frac{E}{K}\right) - \alpha T E$$

Dans une première version du modèle, on définit le contrôle comme une restauration de l'environnement $\mu(K - E)$. Cette restauration est moins efficace quand E s'approche du maximum.

$$E' = E * \gamma_E \left(1 - \frac{E}{K}\right) - \alpha T E + \mu(K - E)$$

où μ , le taux d'investissement de l'état.

Le modèle obtenu est le suivant :

$$\begin{aligned} A' &= A * (\gamma_A(\sigma_A(E - E_0) - A) - \sigma_T T) \\ E' &= E * (\gamma_E \left(1 - \frac{E}{K}\right) - \alpha T) + \mu(K - E) \\ T' &= T * (\varpi_E E + \varpi_A A - \varpi_T T) \end{aligned} \tag{6.1}$$

6.2.2 Développement du modèle

L'étude de stabilité

Le modèle 6.1 reste à niveau abstrait, il contient dix paramètres et un contrôle. Nous avons étudié les points d'équilibres de ce modèle et leur stabilité. Nous

proposons comme capacité de l'environnement $K = 100$ et nous supposons que les animaux ne peuvent pas se reproduire quand la qualité de l'environnement descend en dessous d'un certain seuil, fixé à $E_0 = 10$. La capacité de A est $\sigma_A(E - E_0) = 90$ si nous prenons comme référence $\sigma_A = 1$. Les autres paramètres sont définis dans le tableau 6.2.

Paramètres	Valeur	Description
γ_A	0.5	le taux d'accroissement maximal est de 50%.
σ_T	0.001	la probabilité de décès de l'animal qui rencontre un touriste est faible.
ϖ_E	0.1	le poids relatif de E .
ϖ_A	3	le poids relatif de A .
ϖ_T	0.1	le poids relatif de T .
γ_E	10	l'environnement se régénère de 1/10 au maximum.
α	0.001	les dommages causés par les touristes sont faibles.

TABLE 6.2 – Les valeurs des paramètres du modèle 6.1.

Pour calculer le point d'équilibre, nous supposons que $\mu = 0$. Nous trouvons que le point d'équilibre est $\bar{P} = [A=67.015, T=2089.55, E=79.10]$ en utilisant la méthode de *Newton-Raphson*. Avec la théorème 1 du chapitre 3, nous avons calculé dans un premier temps la matrice Jacobienne $M(\bar{A}, \bar{E}, \bar{T})$ de la dynamique du modèle 6.1 à ce point d'équilibre strictement positif.

$$M(\bar{A}, \bar{E}, \bar{T}) = \begin{pmatrix} -\gamma_A & \gamma_A \sigma_A & -\sigma_A \\ 0 & -\gamma_E/K & -\alpha \\ \varpi_A & \varpi_E & -\varpi_T \end{pmatrix}$$

Nous avons calculé les valeurs propres de la matrice $M(\bar{A}, \bar{E}, \bar{T})$ avec le logiciel Scilab. Nous obtenons les valeurs arrondies suivantes :

$$\begin{aligned} & - 40 \\ & - 0.0999999 + 0.0458804i \\ & - 0.0999999 - 0.0458804i \end{aligned}$$

A l'aide du théorème 1, nous savons que ce point d'équilibre \bar{P} est stable car les parties réelles de toutes les valeurs propres de la matrice Jacobienne sont négatives. Le modèle a un seul point d'équilibre strictement positif et stable le point $\bar{P} = [A=67.015, T=2089.55, E=79.10]$.

Amélioration du modèle

Cette première version du modèle n'est pas facilement adaptable car l'activité touristique tend vers $T=2089$. En effet, si nous comparons avec le tableau 6.1, il n'y a que le mode intensif qui permet d'obtenir un noyau non vide. Nous proposons de contrôler les entrées du parc pour limiter le nombre de touristes.

Pour nous rapprocher de la situation réelle, nous ajoutons un paramètre T_{max} permettant de limiter le nombre de touristes. C'est-à-dire dans le système dynamique, nous prenons la valeur maximale entre T et T_{max} . Nous obtenons le nouveau modèle :

$$\begin{aligned}
 A' &= A * (\gamma_A(\sigma_A(E - E_0) - A) - \sigma_T \text{Min}(T, T_{max})) \\
 E' &= E * (\gamma_E(1 - \frac{E}{K}) - \alpha \text{Min}(T, T_{max})) + \mu(K - E) \\
 T' &= 0 \quad \text{si } (T \geq T_{max}) \\
 T' &= \text{Min}(T, T_{max}) * (\varpi_E E + \varpi_A A - \varpi_T \text{Min}(T, T_{max})) \quad \text{sinon}
 \end{aligned} \tag{6.2}$$

6.3 Les résultats fournis par l'agent expert viabilité

Dans cette section nous présentons les résultats fournis par l'agent expert viabilité qui permettent d'aider les joueurs à prendre des décisions. Tous les résultats sont obtenus avec le modèle 6.2.

6.3.1 Comparaison des noyaux de viabilité

Le plus important pour les joueurs, est de définir ce qu'ils considèrent être la zone d'évolution souhaitable pour le parc. Les contraintes ainsi définies peuvent être utilisées pour calculer le noyau de viabilité. Si le noyau de viabilité est vide, c'est-à-dire que le système est non durable avec les contraintes choisies, les joueurs sont amenés à changer l'ensemble de contraintes ou le mode de gestion. S'il existe un état souhaitable qui respecte les contraintes, les joueurs vont proposer des stratégies viables aux autres joueurs.

Quand les joueurs proposent des noyaux non vides, l'agent expert peut fournir le noyau de viabilité correspondant à l'intersection des états souhaités par chacun des joueurs. La figure 6.1 représente deux ensembles de contraintes et les noyaux de viabilité associés proposés par deux joueurs différents. Le noyau de viabilité de la figure 6.1(a) est contenu dans le noyau de viabilité de la figure 6.1(b). Le fait que l'intersection soit non vide ne signifie pas que le noyau de viabilité correspondant à cet ensemble est l'intersection, car certaines contraintes peuvent être inopérantes (par exemple comme dans la figure 3.11 du chapitre 3, le noyau ne grandit pas quand on déplace le seuil de qualité de l'environnement au-delà de 0.65).

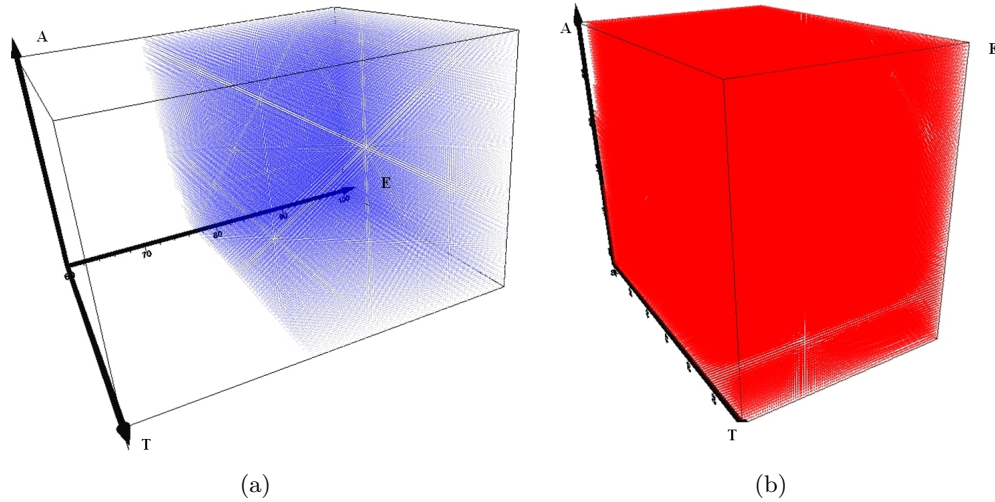


FIGURE 6.1 – Deux ensembles de contraintes et leur noyau de viabilité. (a) Les contraintes sont $A \in [64, 90]$, $T \in [1400, 3000]$, $E \in [60, 100]$. Le noyau de viabilité est représenté en bleu et son volume correspond à environ 64% du volume du grand cube. (b) Les contraintes sont $A \in [64, 90]$, $T \in [1500, 3000]$, $E \in [80, 100]$. Le noyau de viabilité est représenté en rouge et son volume correspond à environ 97% du volume du petit cube.

Dans la figure 6.2, le noyau du diagramme (a) n'est contenu ni dans le noyau de la figure 6.1(a) ni dans le noyau de la figure 6.1(b). Mais l'intersection est non vide. Les joueurs peuvent choisir parmi les stratégies viables au cours d'un cycle de négociations afin de réduire l'ensemble des conflits grâce à ces informations. Par exemple, les joueurs peuvent recalculer le noyau en utilisant l'intersection des contraintes, ou des unions des contraintes.

6.3.2 Visualisation de l'évolution viable

Une fois que les joueurs ont obtenu leur noyau de viabilité, l'agent expert viabilité peut leur fournir de véritables évolutions viables et les fonctions de contrôle qui les régissent. Pour observer une évolution viable, les utilisateurs peuvent utiliser la fenêtre comme dans les figures 5.7 en 2D et 5.8 en 3D. Les joueurs saisissent les coordonnées du point de départ et le nombre d'itérations comme dans la figure 5.11. Une fois la trajectoire tracée, les informations qui correspondent à cette évolution sont affichées en haut à droite de l'interface de la figure 5.6. Les informations contiennent les coordonnées des points de chaque itération, ainsi que tous les contrôles testés jusqu'au contrôle adéquate (c'est-à-dire celui qui permet à la trajectoire de rester dans le noyau).

Dans la figure 6.3, le noyau de viabilité est représenté en rouge. Partant d'un point appartenant à ce noyau de viabilité, la trajectoire verte représente

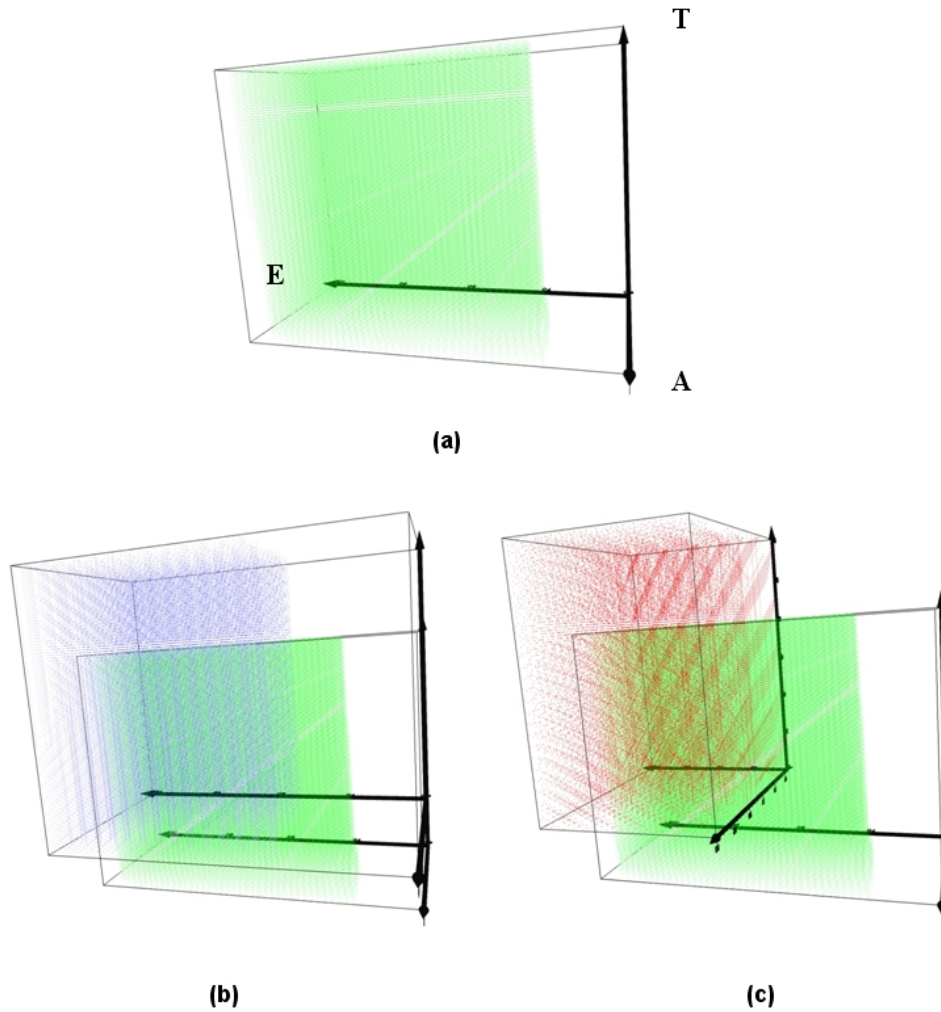


FIGURE 6.2 – (a) Les contraintes sont $A \in [60, 80]$, $T \in [1000, 2500]$, $E \in [60, 100]$. Le noyau de viabilité est représenté en vert et son volume correspond à environ 70% de ses contraintes. (b) représente l'intersection du noyau de (a) et du noyau de la figure 6.1(a). (c) représente l'intersection du noyau de (a) et du noyau de la figure 6.1(b).

une évolution viable. Partant du même point, la trajectoire noire suit la même évolution que la trajectoire verte dans le noyau puis quitte l'état souhaitable car aucun contrôle viable n'a été appliqué.

Nous observons sur la figure 6.4 que la trajectoire verte tend vers un voisinage du point $D = [A \approx 65.89, T \approx 2056.68, E \approx 80]$, car le contrôle change pour que la trajectoire ne sorte pas de l'ensemble de contraintes. Cette information a deux intérêts : d'une part, elle donne des exemples de fonctions de contrôle viables pour

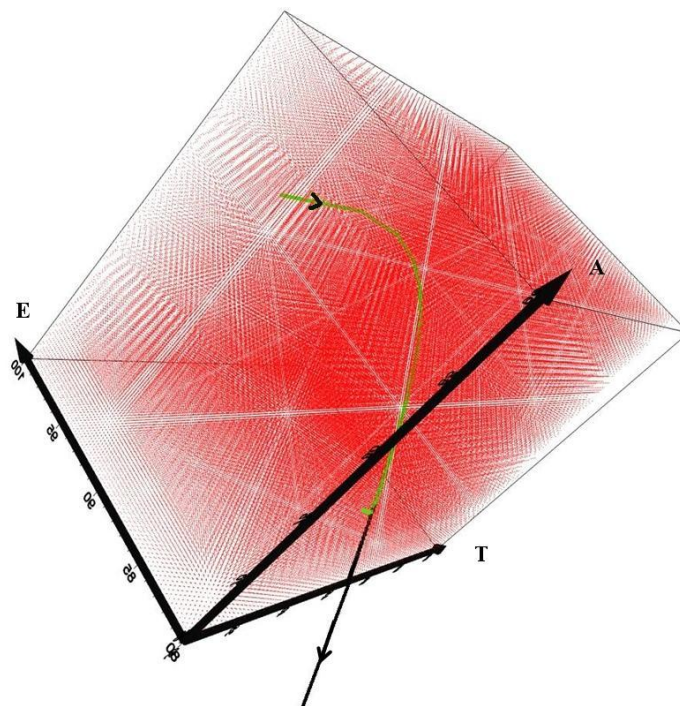


FIGURE 6.3 – Le noyau de viabilité est le même que celui représenté figure 6.1(b) et il est représenté en rouge. Les deux trajectoires partent du même point ($A=80$, $T=2000$, $E=100$) qui appartient au noyau de viabilité. La trajectoire verte reste dans le noyau de viabilité et la trajectoire noire sort de l'espace de contraintes.

les joueurs; d'autre part, elle leur permet de tester la viabilité des fonctions de contrôle qu'ils proposent.

6.4 L'expérimentation de l'agent expert viabilité

Une expérimentation de l'agent expert viabilité a été effectuée le 30 juin 2011 au LIP6, expérimentation visant à l'analyse de l'intégration de l'agent expert viabilité dans le jeu SimParc. Le test est composé de deux sessions. Dans la première, nous avons joué sans l'agent expert viabilité et dans la seconde session nous avons joué avec l'agent expert viabilité.

6.4.1 Conditions d'expérimentation

La figure 6.6 représente la carte de l'unité de paysage 3, cette unité de paysage contient des portions côtières, marines et terrestres. Sur ses côtes se trouvent deux plages ouvertes aux touristes. La plage de la partie ouest de l'unité de paysage accueille un grand nombre de touristes, parfois trop pendant les weekends. Il y a un hôtel de 15 ans qui est situé en face de la plage de sud.

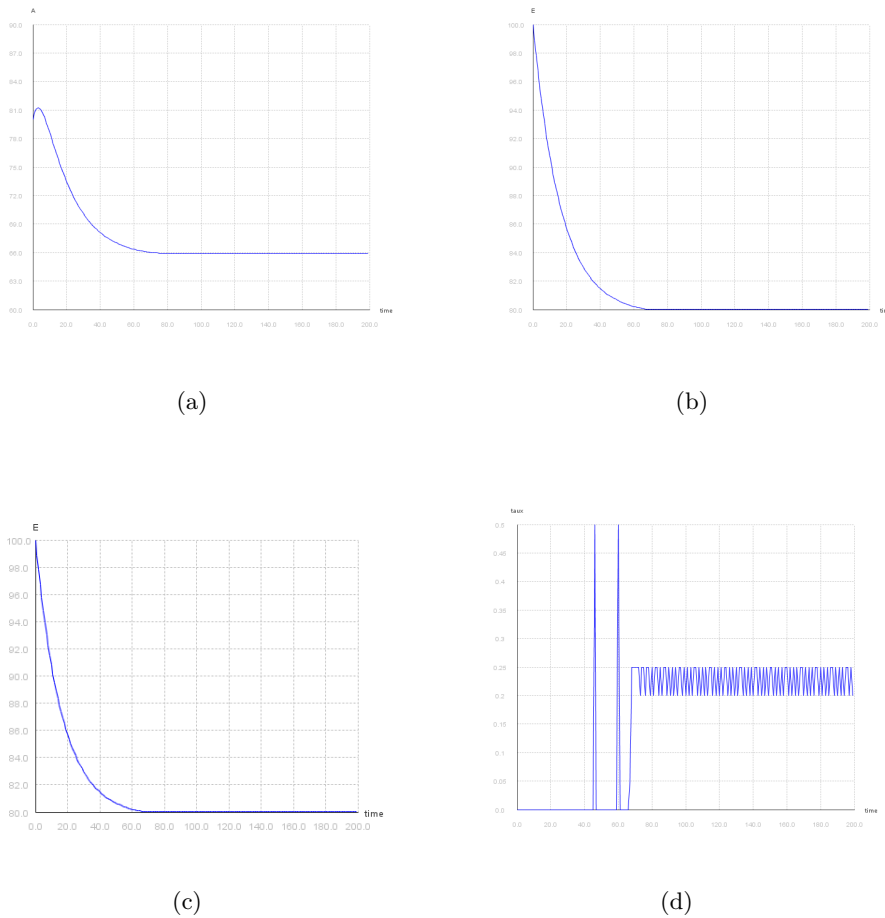


FIGURE 6.4 – Les diagrammes (a), (b) et (c) représentent l'évolution par rapport au temps des dimension A , T et E de la trajectoire verte. Le diagramme (d) représente les changements de la valeur de contrôle à chaque itération.

Cet hôtel accueille des touristes venant de différentes régions du Brésil et du monde.

Il y a aussi des activités maritimes dont la pêche artisanale pratiquée par des pêcheurs vivant dans la zone du parc. En raison de l'existence de zones de mangrove, les pêcheurs ramassent aussi des crabes, vendus aux touristes qui fréquentent les plages près de la mangrove. Une activité importante développée dans cette unité de paysage est la recherche scientifique et la gestion des tortues. La protection et la préservation des tortues se situent principalement dans la partie gauche de la plage où il y a du sable blanc. Pour simplifier l'expérimentation, nous ne prenons pas en compte les pêcheurs artisanaux.



FIGURE 6.5 – L'expérimentation de l'agent expert viabilité à Paris.

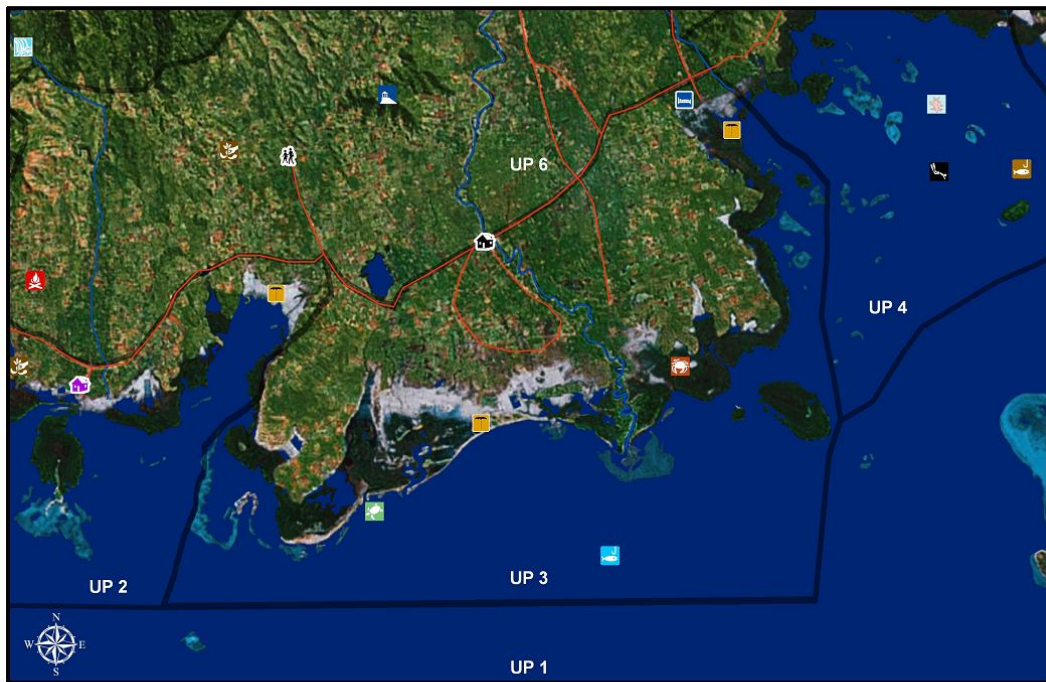


FIGURE 6.6 – La carte de l'unité de paysage 3.

6.4.2 Description de l'expérimentation

L'objectif de ce test est de comparer le jeu sans l'agent expert viabilité et le jeu avec l'agent expert viabilité, de savoir est-ce que l'agent expert viabilité peut aider les joueurs à prendre une décision, et d'observer la satisfaction des joueurs concernant l'agent expert viabilité. Le test commence avec une introduction du

contexte global de la gestion des zones protégées et du projet de recherche SimParc. L'expérimentation était joué manuellement (c'était une simulation de SimParc, et non une utilisation de simParc directement).

Introduction du contexte

1. Présentation des fonctionnalités du jeu. (2 minutes)
2. Présentation de la carte du parc. (3 minutes)
3. Affectation de chaque profil des joueurs. (2 minutes)
4. Présentation des caractéristiques spécifiques de l'unité de paysage. (3 minutes)

La session sans l'agent expert viabilité

Nous commençons la première session qui n'utilise pas l'agent expert viabilité. A part le gestionnaire du parc, le jeu contient trois autres joueurs : un représentant de l'association du service hôtelier qui veut développer l'activité touristique ; un écologiste membre d'une ONG¹ qui veut protéger l'environnement ; un représentant de l'association des opérateurs du tourisme, son objectif est aussi développer l'activité touristique. Les processus détaillés et le temps passé est décrit ci-dessous.

1. Analyse des matériaux (la carte et les profils) fournis à chaque joueur. (8 minutes)
2. Soumission des propositions pour chaque joueur. (2 minutes)
3. Les échanges entre les acteurs sur les propositions présentées. (15 minutes)
4. Retraitement des propositions finales de chaque joueur. (10 minutes)
5. Présentation de la décision du gestionnaire sur la base des propositions et des discussions. (10 minutes)
6. Débat final entre les participants et le gestionnaire du parc sur les résultats. (5 minutes)

Pendant cette session de test, chaque joueur a choisi son mode de gestion (l'association du service hôtelier et l'association des opérateurs du tourisme ont choisi le mode d'*utilisation intensive* et l'écologiste de l'ONG a choisi le mode *Primitive*). Ils sont ensuite entrés dans le processus de négociation. Pendant 15 minutes de négociation, aucun représentant n'a changé de décision. Enfin, le gestionnaire du parc a choisi le mode *Primitive* mais les joueurs ayant une opinion différente n'étaient pas d'accord avec cette décision. Les joueurs ont négocié pendant un certain temps mais aucun différend n'a pu être réglé. Le jeu ne nous a pas permis d'atteindre notre objectif de comprendre les différents facteurs et

1. Organisation non gouvernementale

les conflits permettant une gestion plus efficace. Les joueurs maintiennent leurs opinions car en absence d'arguments suffisants ils n'ont pas convaincu les autres.

La session avec l'agent expert viabilité

Après une présentation de théorie de la viabilité, nous commençons la deuxième session de test, nous avons joué le jeu avec l'agent expert viabilité. Un rôle qui s'appelle assistance viabilité a été ajouté pour aider les joueurs à manipuler l'agent expert et à répondre aux questions techniques, le jeu toujours contient trois joueurs : l'analyste de l'environnement qui veut protéger l'environnement ; une représentant de l'association des résidents qui veut développer le zone, mais ne veut pas trop endommager le terrain, un représentant de l'association des opérateurs du tourisme qui veut développer l'activité touristique. Les processus détaillés et les temps sont présentés ci-dessous.

1. Analyse des matériaux (la carte et les profils) fournis à chaque joueur. **Les joueurs font l'appel à l'agent expert viabilité.** (13 minutes)
2. Soumission des propositions pour chaque joueur. (2 minutes)
3. Les échanges entre les acteurs sur les propositions présentées. (10 minutes)
4. Retraitement des propositions finales de chaque joueur. **Les joueurs font l'appel à l'agent expert viabilité.** (10 minutes)
5. Présentation de la décision du gestionnaire sur la base des propositions et des discussions. **Présentation des résultats avec les propositions de l'agent expert viabilité.** (10 minutes)
6. Débat final entre les participants et le gestionnaire du parc sur les résultats. (5 minutes)

Dans cette session de test, les trois joueurs ont demandé à l'agent expert viabilité de faire les simulations. L'assistance de l'agent expert aide les joueurs à résoudre les problèmes techniques. Les joueurs choisissent un mode de gestion et proposent leurs contraintes. Ils étudient ensuite les résultats. Certains joueurs ont obtenu des noyaux vides, soit ils changent le mode de gestion, soit ils relaxent des contraintes (les utilisateurs utilisent l'interface de la figure 5.6 pour toutes ses opérations). Pendant le processus de négociation, les joueurs font des propositions avec les résultats pour convaincre les autres joueurs (l'analyste de l'environnement a proposé mode *Primitive*, l'association des résidents a proposé mode *Historico-culturelle* et l'association des opérateurs du tourisme a proposé mode *intensive*). Après la discussion, l'analyste de l'environnement et l'association des résidents se sont mis d'accord sur la même mode (*Historico-culturelle*), car ils ont réussi à trouver des contraintes communes. Enfin, pendant le retraitement des propositions finales de chaque joueur, il reste seulement deux choix controversés

comme modes de gestions (*Historico-culturelle* et *intensive*). Le gestionnaire du parc a décidé le mode *Historico-culturelle*. Il a ensuite expliqué son choix avec les propositions des joueurs et les résultats fournis par l'agent expert viabilité. L'association des opérateurs du tourisme n'a pas contesté cette décision.

Pendant cette session de test, les joueurs ont bien compris les fonctionnements de l'agent expert viabilité, et ils n'ont pas eu trop de difficultés pour l'utiliser. Les joueurs sont satisfaits des résultats fournis par l'agent expert viabilité. Si nous comparons ces deux sessions de test, nous n'avons pas pris plus de temps avec l'agent expert viabilité, et les conflits ont été moindres. Cependant, la visualisation graphique a été implémentée après l'expérimentation. Les joueurs n'ont donc malheureusement pas vu les résultats graphiques. Pendant l'expérimentation, les joueurs ont seulement eu le pourcentage de volume du noyau ainsi que les coordonnées des trajectoires. Nous avons, par la suite, continué de développer le logiciel. Grâce à cette expérimentation, nous avons décidé de fournir des résultats graphiques aux joueurs (ceux-ci ont été fortement demandés par les joueurs).

6.5 Conclusion

Nous avons présenté dans ce chapitre les fonctionnements de l'agent expert viabilité dans le jeu de SimParc, et les moments du jeu pendant lesquels les joueurs utilisent l'agent expert viabilité. Nous avons utilisé un modèle simple et abstrait qui décrit l'interaction entre la qualité de l'environnement, les activités touristiques et les animaux à protégés. Cette version de modèle n'est pas parfaite, elle est en cours de développement mais le modèle a été adapté à l'objectif de la gestion du parc.

Nous avons ensuite illustré l'information donnée par le calcul du noyau de viabilité qui permet facilement de distinguer les situations où l'ensemble désirable peut être préservé. De plus, nous avons montré la possibilité de fournir aux joueurs des exemples d'évolutions viables et les fonctions de contrôle ou encore la possibilité de vérifier la viabilité des fonctions de contrôle qu'ils pourront proposer. Enfin, nous avons testé l'agent expert viabilité dans une session étendue du jeu SimParc. Nous rappelons que l'agent expert viabilité ne prend pas la décision pendant le jeu, son rôle est seulement de proposer des résultats de viabilité pour aider les joueurs à évaluer la qualité de la décision.

Conclusion

Bilan

Dans cette thèse, nous avons étudié l'intérêt d'un agent expert viabilité dans un jeu sérieux d'aide à la gestion participative. Cet agent artificiel fournit aux joueurs des informations sur la viabilité d'un système. Grâce à la théorie de la viabilité, les joueurs étudient la compatibilité entre la dynamique du système et des contraintes ce qui leur permet de se concentrer sur les ensembles d'états du système qu'ils considèrent comme souhaitables. Ces discussions ont été fructueuses dans la cadre du projet SimParc.

Nous avons tout d'abord étudié la possibilité d'utiliser l'agent expert viabilité dans le cadre de la gestion de systèmes environnementaux. Cette étude théorique a montré que l'analyse asymptotique permet d'identifier les attracteurs de la dynamique, mais qu'elle manque d'informations spatiales et temporelles pour établir une analyse dans un objectif d'analyse de la durabilité. Nous avons proposé l'approximation des noyaux de viabilité et des bassins de capture qui permettent de renseigner les informations manquantes. L'agent expert viabilité fournit des analyses de viabilité mieux adaptées à la gestion environnementale. Il reste pour les joueurs à définir les ensembles d'états souhaitables. L'agent expert viabilité permet aussi d'identifier les possibilités de contrôle selon les variables d'état et les paramètres du modèle dynamique.

Nous avons ensuite présenté un algorithme d'approximation du noyau de viabilité avec une procédure de classification de type "plus proche voisin". L'algorithme satisfait les conditions de convergence lorsque la résolution de la grille tend vers 0. Cet algorithme nous a permis de gagner du temps de calcul. Cela nous a permis de diminuer le temps d'attente de l'analyse de viabilité.

Nous avons également développé un support logiciel qui permet d'approximer des noyaux de viabilité et d'observer des évolutions des systèmes. Ce logiciel possède une interface graphique utilisateur très facile à utiliser. Il est aussi facile de créer des nouveaux modèles ou des nouvelles contraintes sans modifier les autres classes. Le logiciel permet de visualiser les noyaux de viabilité en 2D ou 3D, de tracer les trajectoires d'évolution, d'observer les résultats et d'enregistrer les données et les images. De plus, c'est un support logiciel commun qui est facile à exploiter avec tout nouvel algorithme. Le logiciel est disponible sur le site : <http://motive.cemagref.fr/lisc/software>.

Nous avons ensuite expliqué le fonctionnement de l'agent expert viabilité en utilisant un modèle abstrait. Ce type de résultats est susceptible d'intéresser les

acteurs. Cependant, un des avantages majeurs de l'agent expert viabilité est qu'il peut s'utiliser sur d'autres modèles dynamiques. Nous avons aussi présenté les résultats fournis par l'agent expert viabilité. Ces résultats permettent aux joueurs de facilement distinguer les situations désirables, et ainsi proposer des fonctions de contrôle pour la gestion du parc par anticipation. Enfin nous avons comparé deux sessions de test, une sans l'agent expert viabilité et une autre avec l'agent expert viabilité, nous avons trouvé que les résultats fournis par l'agent expert viabilité peuvent aider les joueurs à évaluer la qualité de leur choix et à proposer des arguments assez forts pour la négociation. Les conflits sont plus faciles à gérer grâce à l'agent expert viabilités et la décision finale est plus facile à accepter pour les joueurs.

Perspectives

Notre travail garde un caractère exploratoire, et les perspectives qu'il ouvre nous paraissent aussi importantes que ses réalisations bien établies.

Comparaison avec les autres algorithmes

Dans le chapitre 4, nous avons comparé le temps de calcul entre notre algorithme et celui de [Deffuant *et al.*, 2007] qui utilise les SVMs, la comparaison avec les autres algorithmes existants devra être effectuée; par exemple, le modèle malgache de [Bernard, 2011] en 3 dimensions, les études de base sur l'algorithme de [Saint-Pierre, 1994] qui est très connu dans ce domaine.

Développement du logiciel

Le logiciel que nous avons présenté dans le chapitre 5 reste à compléter. Actuellement, le logiciel n'approche que les noyaux de viabilité, les algorithmes permettant d'approcher les bassins de capture et calculer le coût de la résilience devront être implémentés. Nous souhaitons aussi implémenter les algorithmes existants, pour que les utilisateurs puissent choisir différents algorithmes selon leurs besoins (plus rapide ou utilisant moins de mémoire, etc). De plus, la visualisation en 3D n'est pas encore parfaite, la présentation des résultats reste à améliorer.

Un test avec les vrais joueurs

Nous attendons de tester l'agent expert viabilité avec le vrai jeu de SimParc pour mieux comprendre son influence. Cela nous permettrait d'améliorer l'agent expert viabilité. Nous pourrions aussi ajuster les résultats fournis par l'agent expert viabilité grâce à ce test. Par exemple, s'il y a des informations supplémentaires qui peuvent encore intéresser les joueurs. De plus, comme la théorie de la viabilité est une théorie mathématique peu connue, ce test nous permettrait d'évaluer la compréhension de cette théorie pour les joueurs qui viennent de différents domaines.

Il manque aussi un modèle dynamique lié au parc, car le modèle animal qui a été présenté dans la section 6.2 est un modèle abstrait. Avec ce modèle, il est difficile de comprendre les paramètres et surtout difficile de définir les paramètres selon les informations et les ressources du parc. Une assistance peut être créée pour aider les joueurs à faire appel à l'agent expert.

Relaxation des contraintes

Le jeu SimParc est destiné à être joué avec de nombreux joueurs, il est possible de trouver des noyaux vides même en faisant plusieurs simulations. Pour éviter un trop grand nombre de manipulations, nous souhaitons que l'agent expert viabilité puisse fournir des propositions de relaxation des contraintes. [Jussien, 1997] a proposé une méthode utilisant un arbre binaire de recherche, ce serait est une piste à travailler.

Bibliographie

- [Adamatti *et al.*, 2005] ADAMATTI, D. F., SICHMAN, J. S., BOMMEL, P., DUCROT, R., RABAK, C. et CAMARGO, M. E. (2005). Jogoman : A prototype using multi-agent-based simulation and role-playing games in water management. *Simulation*. 19
- [Alvarez et Martin, 2011] ALVAREZ, I. et MARTIN, S. (2011). Geometric robustness of viability kernels and resilience basins. In DEFFUANT, G. et GILBERT, N., éditeurs : *Viability and Resilience of Complex Systems*, volume 46 de *Understanding Complex Systems*, pages 193–218. Springer Berlin / Heidelberg. 81
- [Alvarez *et al.*, 2010] ALVAREZ, I., MARTIN, S. et MESMOUDI, S. (2010). Describing the result of a classifier to the end-user : Geometric-based sensitivity. In *European Conference on Artificial Intelligence, Frontiers in Artificial Intelligence and Applications.*, volume 215, pages 835–840. 69
- [Aubin et Saint-Pierre, 2006] AUBIN, J. et SAINT-PIERRE, P. (2006). An introduction to viability theory and management of renewable resources. *Advanced Methods for Decision Making and Risk Management.*, pages 52–96. 63
- [Aubin, 1991] AUBIN, J.-P. (1991). *Viability Theory*. Birkhäuser. 1, 10, 28, 58, 63
- [Aubin, 2002] AUBIN, J.-P. (2002). Viability kernels and capture basins of sets under differential inclusions. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 4, pages 4605 – 4610. 13
- [Aubin *et al.*, 2001] AUBIN, J.-P., BONNEUIL, N., MAURIN, F. et SAINT-PIERRE, P. (2001). Viability of pay-as-you-go systems. *Journal of Evolutionary Economics*, 11:555–571. 15
- [Aubin et Frankowska, 1985] AUBIN, J.-P. et FRANKOWSKA, H. (1985). Heavy viable trajectories of controlled systems. *Annales de l'institut Henri Poincaré (C) Analyse non linéaire*, pages 371–395. 14
- [Aubin *et al.*, 2005] AUBIN, J.-P., PUJAL, D. et SAINT-PIERRE, P. (2005). Dynamic management of portfolios with transaction costs under tyochastic uncertainty. In BRETON, M. et BEN-AMEUR, H., éditeurs : *Numerical Methods in Finance*, pages 59–89. Springer US. 13
- [Barreteau, 2003] BARRETEAU, O. (2003). The joint use of role-playing games and models regarding negotiation processes : Characterization of associations. *Artificial Societies and Social Simulation.*, 6. 2, 18
- [Bellam, 1957] BELLAM, R. (1957). *Dynamic Programming*. Princeton University Press.
- [Bernard, 2011] BERNARD, C. (2011). *La théorie de la viabilité au service de la modélisation mathématique du développement durable Application au cas de la forêt humide de Madagascar*. Thèse de doctorat, Université Blaise Pascal, U.F.R. Sciences et Technologies, école doctorale des sciences fondamentales. 15, 102

- [Bertsekas, 2005] BERTSEKAS, D. (2005). Dynamic programming and optimal control. *Athena Scientific*.
- [Béné *et al.*, 2001] BÉNÉ, C., DOYEN, L. et GABAY, D. (2001). A viability analysis for a bio-economic model. *Ecological Economics*, 36(3):385–396. 1
- [Bonneuil, 2003] BONNEUIL, N. (2003). Making ecosystem models viable. *Bulletin of Mathematical Biology*, 65:1081–1094. 1
- [Bonneuil et Saint-Pierre, 2000] BONNEUIL, N. et SAINT-PIERRE, P. (2000). Protected polymorphism in the two-locus haploid model with unpredictable fitnesses. *Journal of Mathematical Biology*, 40:251–277. 15
- [Bousquet *et al.*, 1999] BOUSQUET, F., BARRETEAU, O., LE PAGE, C., MULLON, C. et WEBER, J. (1999). An environmental modelling approach, the use of multi-agent simulations. In *Advances in Environmental and Ecological Modelling*, pages 133–122. 19
- [Briot *et al.*, 2010] BRIOT, J., SORDONI, A., VASCONCELOS, J., SEBBA PATTO, V., ADAMATTI, D., IRVING, M. et MELO, G. (2010). A design of an artificial decision maker for a human-based social simulation - experience of the simparc project. In *Activity-Based Modeling and Simulation*, pages 17–35, Presses Universitaires Blaise-Pascal,. 2
- [Briot *et al.*, 2007] BRIOT, J.-P., GUYOT, P. et IRVING, M. (2007). Participatory simulation for collective management of protected areas for biodiversity conservation and social inclusion. In *AIS-CMS07 International Modeling and Simulation Multiconference (IMSM07)*, pages 183–188. 18, 19
- [Briot *et al.*, 2011] BRIOT, J.-P., IRVING, M., MELO, G., VASCONCELOS, E., ALVAREZ, I., MARTIN, S. et WEI, W. (2011). A serious game and artificial agents to support intercultural participatory management of protected areas for biodiversity conservation and social inclusion. In *The Second International Conference on Culture and Computing (Culture and Computing 2011)*. Kyoto University, IEEE. 2, 4, 19, 70
- [Brown *et al.*, 1987] BROWN, B., ELLIOTT, D. et PAGET, D. (1987). Lipschitz constants for the bernstein polynomials of a lipschitz continuous function. *Journal of approximation theory*, 49(2):196–199. 58
- [Butcher, 2005] BUTCHER, J. (2005). *Numerical Methods for Ordinary Differential Equations*. John Wiley & Sons. 62
- [Butcher, 1975] BUTCHER, J. C. (1975). A stability property of implicit runge-kutta methods. *BIT Numerical Mathematics*, 15:358–361. 62
- [Butler, 1980] BUTLER, R. W. (1980). The concept of a tourist area cycle of evolution implication for management of resources. *Canadian Geographer*, 8:5–12. 35
- [Cardaliaguet *et al.*, 2000] CARDALIAGUET, P., QUINCAMPOIX, M. et SAINT-PIERRE, P. (2000). Numerical schemes for discontinuous value functions of optimal control. *Set-Valued Analysis*, 8:111–126. 13

- [Casagrandi et Rinaldi, 2002] CASAGRANDE, R. et RINALDI, S. (2002). A theoretical approach to tourism sustainability. *Conservation Ecology*, 6(1):13. 34, 35, 37, 38, 42
- [Chapel, 2007] CHAPEL, L. (2007). *Maintenir la viabilité ou la résilience d'un système : les machines à vecteurs de support pour rompre la malédiction de la dimensionnalité?* Thèse de doctorat, Université Blaise Pascal, U.F.R. Sciences Pour l'Ingénieur. 13, 14
- [Collings et Wollkind, 1990] COLLINGS, J. et WOLLKIND, D. (1990). A global analysis of a temperature-dependent model system for a mite predator-prey interaction. *SIAM Journal on Applied Mathematics*, 50(5):1348–1372. 34, 38
- [Coquelin et al., 2007] COQUELIN, P.-A., MARTIN, S. et MUNOS, R. (2007). A dynamic programming approach to viability problems. In *Approximate Dynamic Programming and Reinforcement Learning, 2007. ADPRL 2007. IEEE International Symposium on*, pages 178–184. 11
- [De Lara et Doyen, 2011] DE LARA, M. et DOYEN, L. (2011). *Sustainable Management of Natural Resources*. Springer. 1
- [De Lara et al., 2007] DE LARA, M., DOYEN, L., GUILBAUD, T. et ROCHET, M.-J. (2007). Is a management framework based on spawning-stock biomass indicators sustainable? a viability approach. *ICES Journal of Marine Science*, 64(4):761–767. 15
- [Deffuant et al., 2007] DEFFUANT, G., CHAPEL, L. et MARTIN, S. (2007). Approximating viability kernels with support vector machines. *Automatic Control, IEEE Transactions on*, 52(5):933–937. 11, 42, 58, 59, 63, 102
- [Deffuant et Gilbert, 2011] DEFFUANT, G. et GILBERT, N. (2011). *Viability and Resilience of Complex Systems : concepts, methods and case studies from ecology and society*. Springer. 70
- [Delanoue et al., 2009] DELANOUE, N., JAULIN, L., HARDOUIN, L. et LHOMMEAU, M. (2009). Guaranteed characterization of capture basins of nonlinear state-space systems. In FILIPE, J., CETTO, J. A. et FERRIER, J.-L., éditeurs : *Informatics in Control, Automation and Robotics*, volume 24 de *Lecture Notes in Electrical Engineering*, pages 265–272. Springer Berlin Heidelberg. 13
- [Dung, 1995] DUNG, P. M. (1995). On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357. 26
- [Durand et al., 2010] DURAND, M.-H., MARTIN, S. et SAINT-PIERRE, P. (2010). Viabilité et développement durable. *Les cahiers du GERAD*. 15
- [Etienne, 2003] ETIENNE, M. (2003). Sylvopast : a multiple target role-playing game to assess negotiation processes in sylvopastoral management planning. *Journal of Artificial Societies and Social Simulation*, 6. 19
- [Fuentes, 1993] FUENTES, R. E. (1993). Scientific research and sustainable development. *Ecological Applications*, 3:576–577. 46

- [Irving, 2006] IRVING, M. (2006). Areas protegidas e inclusao social : Construindo novos significados. 18, 20
- [Jussien, 1997] JUSSIEN, N. (1997). *Relaxation de contraintes pour les problèmes dynamiques*. Thèse de doctorat, Université de Rennes 1, école doctorale des sciences pour l'ingénieur. 103
- [Kutta, 1901] KUTTA, M. (1901). Beitrag zur näherungsweise intergration oder differentialgleichungen. *Zeit. Math. u. Phys.*, 46:435–453. 62
- [Lhommeau et al., 2011] LHOMMEAU, M., JAULIN, L. et HARDOUIN, L. (2011). Capture basin approximation using interval analysis. *Control Signal Process.*, 25:264–272. 13
- [Ludwig et al., 1997] LUDWIG, D., WALKER, B. et HOLLING, C. (1997). Sustainability, stability, and resilience. *Conservation Ecology*, 1(1):7. 34
- [Martin, 2004] MARTIN, S. (2004). The cost of restoration as a way of defining resilience : a viability approach applied to a model of lake eutrophication. *Conservation Ecology*, 9(2). 69
- [Martin, 2005] MARTIN, S. (2005). *La résilience dans les modèles de systèmes écologiques et sociaux*. Thèse de doctorat, L'école normale supérieure de Caen.
- [Martin, 2010] MARTIN, S. (2010). Compte-rendu final du projet anr déduction. *Report.*, 34. 70
- [Martinet et Blanchard, 2009] MARTINET, V. et BLANCHARD, F. (2009). Fishery externalities and biodiversity : Trade-offs between the viability of shrimp trawling and the conservation of frigatebirds in french guiana. *Ecological Economics*, 68(12):2960 – 2968. 15
- [Mullon et al., 2004] MULLON, C., CURRY, P. et SHANNON, L. (2004). Viability model of trophic interaction in marine ecosystems. *Natural Resource Model*, 17. 1
- [Petschel-Held et al., 1999] PETSCHEL-HELD, G., SCHELLNHUBER, H.-J., BRUCKNER, T., TOTH, F. et HASSELMANN, K. (1999). The tolerable windows approach : Theoretical and methodological foundations. *Climatic Change*, 41:303–331. 12
- [Pujal et Saint-Pierre, 2004] PUJAL, D. et SAINT-PIERRE, P. (2004). L'algorithme du bassin de capture appliqué à l'évaluation d'options. *Finance*, 25(1):75–106. 13
- [Rapaport et al., 2006] RAPAPORT, A., TERREAUX, J. et DOYEN, L. (2006). Viability analysis for the sustainable management of renewable resources. *Mathematical and Computer Modelling*, 43(5–6):466 – 484. 15
- [Runge, 1895] RUNGE, C. (1895). Über die numerische auflösung von differentialgleichungen. *Math. Annalen*, 46:167–178. 62
- [Saad, 2003] SAAD, Y. (2003). *Iterative methodes for Sparse Linear Systems*. Industrial and Applied Mathematics.

- [Saint-Pierre, 1994] SAINT-PIERRE, P. (1994). Approximation of the viability kernel. *Applied Mathematics and Optimization*, 29:187–209. 11, 13, 58, 102
- [Sicard *et al.*, 2012] SICARD, M., REULLON, R., MESMOUDI, S., PERROT, N., ALVAREZ, I. et MARTIN, S. (2012). Viability theory to explore complex food process : Application to cheese ripening optimization. *Food Control*, 23(2):312–319. 1
- [Süli et Mayers, 2003] SÜLI, E. et MAYERS, D. (2003). *An Introduction to Numerical Analysis*. The university of Cambridge. 62
- [Spiteri *et al.*, 2000] SPITERI, R., PAI, D. et ASCHER, U. (2000). Programming and control of robots by means of differential algebraic inequalities. *Robotics and Automation, IEEE Transactions on*, 16(2):135–145. 13
- [Susi *et al.*, 2007] SUSI, T., JOHANNESSON, M. et BACKLUND, P. (2007). Serious games – an overview. *University of Skövde*. 1
- [Van Coller, 1997] VAN COLLER, L. (1997). Automated techniques for the qualitative analysis of ecological models : continuous models. *Conservation Ecology*, 1(1):5. 34, 39
- [Vasconcelos *et al.*, 2009] VASCONCELOS, E., MELO, G., BRIOT, J.-P., SEBBA PATTO, V., SORDONI, A., IRVING, M., ALVAREZ, I. et LUCENA, C. (2009). A serious game for exploring and training in participatory management of national parks for biodiversity conservation : Design and experience. In *VIII Brazilian Symposium on Games and Digital Entertainment (SBGAMES'09)*, pages 93–100, PUC-Rio, Rio de Janeiro, Brazil. Sociedade Brasileira de Computação (SBC), Brazil. 24
- [Verhulst, 1847] VERHULST, P. (1847). Notice sur la loi que la population poursuit dans son accroissement. *Mémoires de l'Académie Royale des Sciences, des Lettres et des Beaux-Arts de Belgique*, pages 1–32. 87
- [Warmerdam *et al.*, 2006] WARMERDAM, J., KNEPFLTÉ, M., BIDARRA, R., BEKEBREDE et MAYER, I. (2006). Simport : a multiplayer management game framework. In *9th International Conference on Computer Games (CGAMES06)*, Dublin, Ireland. 1
- [Wei *et al.*, 2012a] WEI, W., ALVAREZ, I., MARTIN, S., BRIOT, J.-P., IRVING, M. et MELO, G. (2012a). Integration of viability models in a serious game for the management of protected areas. In Antonio Palma dos REIS, Patrick S.P. Wang, A. P. A., éditeur : *IADIS Intelligent Systems and Agents Conference 2012*, pages 55–62, Lisbonne, Portugal. IADIS (International Association for Development of the Information Society). isbn : 978-972-8939-69-4. 5
- [Wei *et al.*, 2012b] WEI, W., ISABELLE, A. et SOPHIE, M. (2012b). Algorithme d'approximation du noyau de viabilité avec procédure de classification. In *Actes de la conférence RFIA 2012*, pages 978–2–9539515–2–3, Lyon, France. Session "Posters". 5
- [Xia, 2011] XIA, P. (2011). Conception et test d'un support logiciel au développement d'algorithmes de viabilité. *Mémoire de Master II Informatique, Université Blaise Pascal*. 5

Annexe

ANNEXE A

Ficher pom.xml du projet

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.mycompany</groupId>
  <artifactId>DemoGUIMaven</artifactId>
  <packaging>jar</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>DemoGUIMaven_WEI</name>
  <url>http://maven.apache.org</url>
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>2.3.1</version>
        <configuration>
          <source>1.6</source>
          <target>1.6</target>
        </configuration>
      </plugin>
      <plugin>
        <artifactId>maven-assembly-plugin</artifactId>
        <executions>
          <execution>
            <phase>package</phase>
            <goals>
              <goal>single</goal>
            </goals>
            <configuration>
              <descriptorRefs>
                <descriptorRef>jar-with-dependencies</descriptorRef>
              </descriptorRefs>
              <archive>
                <manifest>
                  <mainClass>com.mycompany.demoguimaven.Main</mainClass>
                </manifest>
              </archive>
            </configuration>
          </execution>
        </executions>
      </plugin>
    </plugins>
  </build>
  <repositories>
    <repository>
      <url>http://download.java.net/maven/2/</url>
      <id>beans-binding</id>
    </repository>
  </repositories>
</project>
```

```
<layout>default</layout>
  <name>Repository for library Library[beans-binding]</name>
</repository>
<repository>
  <id>maven2-repository.netbeans.maven2</id>
  <name>Netbeans Maven Repository</name>
  <url>http://bits.netbeans.org/maven2/</url>
  <layout>default</layout>
</repository>
<repository>
  <id>fr.cemagref.lisc.maven2</id>
  <name>Lisc repository</name>
  <url>http://trac.clermont.cemagref.fr/nexus/content/groups/public/</url>
</repository>
</repositories>
<dependencies>
  <dependency>
    <groupId>com.google.code.jmatharray</groupId>
    <artifactId>jmatharray</artifactId>
    <version>2009.07.25</version>
  </dependency>
  <dependency>
    <groupId>com.google.code.jmathplot</groupId>
    <artifactId>jmathplot</artifactId>
    <version>2011.09.1</version>
  </dependency>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>3.8.1</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>org.netbeans.api</groupId>
    <artifactId>org-openide-util-lookup</artifactId>
    <version>RELEASE691</version>
  </dependency>
  <dependency>
    <groupId>org.netbeans.external</groupId>
    <artifactId>AbsoluteLayout</artifactId>
    <version>RELEASE701</version>
  </dependency>
</dependencies>
</project>
```