

REPUBLIQUE DU SENEGAL



Un Peuple – Un But – Une Foi

MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE

Université Gaston Berger de Saint-Louis



U.F.R DE SCIENCES APPLIQUEES ET DE TECHNOLOGIE

SECTION D'INFORMATIQUE

Rapport de Stage

Sujet : conception, implémentation et simulation de la dynamique de l'environnement sahélien lié à l'invasion de la gerbille au Sénégal

de

Moussa SALL

Sous l'encadrement de :

M. Jean Le Fur, Chercheur, Institut de Recherche pour le Développement
M. Jean Marie Dembele, Enseignant-chercheur, Section d'informatique UFR SAT

Remerciements

Alhamdoulilahi Rabil halamine !

J'adresse mes sincères remerciements à:

Mes Parents qui n'ont ménagé aucun effort pour me mettre dans les bonnes conditions de réussite,

mon maître de stage, M. Jean Le Fur pour avoir proposé ce sujet qui m'a permis de me familiariser avec le développement à base d'agents, pour ses longues séances d'explications et de travail à deux pour ma bonne et rapide compréhension, pour son amour pour le travail accompli et surtout pour sa patience. Merci de tout cœur pour m'avoir intégré dans votre équipe,

mon encadreur à l'université Gaston Berger de Saint-Louis, M. Jean Marie Dembele pour sa disponibilité, sa participation incommensurable à la bonne tenue de mon stage, les remarques pertinentes et les passages effectués au labo pour apporter sa participation sur le travail en cours,

M. Pape Adama Mboup, pour son soutien continu et la bonne ambiance durant tout le stage,

M. Laurent Granjon, pour les moments agréables de partage de connaissance et pour m'avoir permis de participer à la mission de caractérisation des communautés de parasites hébergées par les rongeurs présents dans l'écosystème sahélien au Nord du Sénégal,

à tous les doctorants, stagiaires et techniciens du laboratoire Biopass,

à l'ensemble du corps enseignant et administratif,

à tous mes camarades de promotion,

à mes voisins de chambre,

ainsi qu'à toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

Résumé :

Ce rapport présente l'expérience acquise durant les 6 mois de stage que j'ai effectué au Laboratoire de Biologie des Populations Animales Sahélo-Soudaniennes de l'Institut de Recherche pour le Développement. Ce stage portait sur la conception, l'implémentation et la simulation d'un environnement sahélien lié à l'invasion de la gerbille au nord du Sénégal.

Pour atteindre cet objectif il était nécessaire de simuler l'évolution de la pluviométrie en fonction des données mensuelles fournies par les images satellites, d'implémenter une fonctionnalité permettant d'extraire à la demande une zone de questionnement fixée par les biologistes et de mettre en relation l'environnement avec l'évolution de la précipitation. Pour ce faire, j'ai commencé par me familiariser avec la plateforme de développement SimMasto dans la première semaine de mon stage. Ensuite, j'ai pu réaliser les fonctionnalités permettant de convertir indifféremment des coordonnées géographiques en coordonnées discrètes de la grille et vice versa, d'extraire un zoom dans l'emprise de référence, de simuler l'évolution de la pluviométrie et de faire croître ou décroître la végétation suivant la quantité de pluie présente dans la cellule. Au final, nous avons mis en place différentes fonctionnalités permettant de réaliser un environnement propice à l'évolution de la gerbille.

Summary:

This report presents the experience acquired during six months of internship that I did in the Sahelo-Sudanian biological laboratory of animal populations of the Research Institute for Development. The aim of this work was to design, implement and simulate the sahelian environment related to the invasion of gerbil in northern Senegal. To reach this goal I had to simulate the evolution of rainfall based on monthly data from satellite images, implement a feature to extract on demand any area of interest requested by biologists and to link the environment with changes in precipitation. To do so, I started to familiarize myself with the SimMasto development platform at the beginning of my internship. Then, I was able to achieve the functionality to either convert geographic coordinates into discrete grid coordinates and vice versa, extract zoom from a reference grip, simulate the evolution of rainfall and the vegetation growth according to the amount of rain in the cell. Finally, I have implemented various features to achieve a functional environment for gerbil's development.

Table des Matières

Remerciements	1
Résumé :	2
Liste des figures	5
INTRODUCTION.....	6
CHAPITRE 1: PRESENTATION GENERALE	7
I. Contexte	7
II. Présentation du sujet	10
II-1. Problématique.....	10
II-2. Définition du sujet.....	11
II-3. Cahier des charges	11
II-3-1. Formalisation et manipulation de l'espace.....	12
II-3-2. Nature et manipulation des données de l'environnement	12
II-3-3. Mise en relation des objets de l'environnement avec l'évolution des précipitations....	13
CHAPITRE 2: IMPLEMENTATION	14
I. Outils utilisés	14
II. Méthodes	14
II-1. Numérisation des images de la pluviométrie	15
II-1-1. Séquence des opérations.....	16
a. Ré-échantillonnage de l'image	16
b. Reclassification de couleur	17
c. Digitalisation et sauvegarde de l'image.....	17
II-2. Conversion de coordonnées	18
II-2-1. Méthodes de transformations des coordonnées géodésiques de Clarke	18
II-2-2. Méthode de conversion de coordonnées par projection orthographique.....	20
II-3. Extraction de zoom	20
II-3-1. Méthode d'extraction	22
II-4. Mise en relation de la pluviométrie et de la végétation.....	22
II-4-1. Fonction de croissance logistique	22
II-5. Lecture des données	23
III. Mise en œuvre.....	25
III-1. Mise en œuvre de la numérisation des images	26
III-1-1. Traitement de l'image sur Corel Paint	26
III-1-2. Conversion et sauvegarde sur SimMasto.....	26

III-2. Mise en œuvre de la conversion de coordonnées	26
III-2-1. Géographiques en discrètes.....	26
III-2-2. Discrètes en géographiques.....	27
III-3. Mise en œuvre de l'extraction de zoom	27
III-3-1. Extraction de fichier texte.....	27
III-3-2. Extraction de fichier image bitmap.....	28
III-4. Mise en œuvre de la croissance de la végétation	28
III-4-1. Calcul du nombre de jours par pas de temps (<i>oneTick_Uday</i>)	28
III-4-2. Calcul de la capacité de charge en biomasse.....	29
III-4-3. Calcul de la biomasse de la végétation (herbe, arbuste ou culture)	29
III-5. Processus d'exécution du protocole Gerbille.....	29
CHAPITRE 3 : RESULTATS.....	31
I. Simulation des pluies sur le GUI	31
II. Extraction de zooms	32
III. Evolution de la biomasse sur le GUI et sur le graphe de biomasse.....	34
CHAPITRE 4 : DISCUSSION.....	37
CONCLUSION	39
BIBLIOGRAPHIE	40
WEBOGRAPHIE.....	40
ANNEXE : codes sources	41
I. Numérisation images pluviométriques avec main fonctionnel	41
II. Extraction de zoom avec main fonctionnel	44
III. Croissance de la végétation.....	48
IV. Conversion de coordonnées géodésiques en discrètes	48

Liste des figures

Figure 1 : Quelques cas d'utilisation intégrés dans SimMasto.....	8
Figure 2 : Arborecence objet du modèle général de la plate-forme SimMasto telle qu'élaborée par les études de cas représentées.	9
Figure 3 : Organigramme des classes associées à la gestion des simulations formalisées dans le paquetage 'objet concret' de la figure 2 telles que définies par les besoins des études de cas représentées.....	10
Figure 4 : Exemple d'étude réalisée sur la plate-forme SimMasto	11
Figure 5 : Données satellite de la variation de pluies mensuelle de l'année 1999 sur la zone Ouest Africaine, données fournies sous forme d'images Bitmap	15
Figure 6 : Image originale (juillet.1999)	16
Figure 7 : Positionnement du masque de la ME	16
Figure 8 : Image obtenue après extraction et ré-échantillonnage	17
Figure 9 : Image reclassifiée en 9 couleurs	17
Figure 10 : Coordonnées géographiques et cartésiennes d'un point dans un ellipsoïde (Clarke 1880 IGN).....	19
Figure 11 : projection du point sur la largeur et la hauteur du raster	20
Figure 12 : Zones d'intérêt sur la petite emprise (source: Modèle de l'approche petite emprise JLF 12-11-2015)	21
Figure 13 : Zones d'intérêt sur la moyenne emprise (Le Fur, 2015)	21
Figure 14 : Diagramme de la fonction de croissance logistique de Verlhust.....	23
Figure 15 : Principe de simulation de l'environnement sahélien :	24
Figure 16 : Résumé du processus d'exécution du protocole gerbille	30
Figure 17 : Simulation de la pluviométrie du mois d'Août 1999 (a) avec les frontières et (b) sans les frontières.....	31
Figure 18 : Simulation de la pluviométrie de l'année 1999	32
Figure 19 : Simulation de la couverture du sol de la grande emprise	33
Figure 20 : Simulation du zoom petite emprise obtenu après extraction	33
Figure 21 : Zone de questionnement (nord du lac de Guiers) extraite à partir de la petite emprise... ..	34
Figure 22 : zone de questionnement (zone du Ferlo, milieu très herbacé) extraite de la petite emprise	34
Figure 23 : Biomasse initiale de la végétation sur le nord du lac de Guiers	35
Figure 24 : Biomasse de la végétation aperçue à la fin de la saison sèche de l'année 1999 sur le nord du lac de Guiers.....	35
Figure 25 : Biomasse de la végétation aperçue à la fin de la saison des pluies de l'année 1999 sur le nord du lac de Guiers	36
Figure 26 : Graphe d'évolution de la biomasse par pas de temps (a) arbuste, (b) culture, (c) herbe ..	36
Figure 27 : différents aspects du sol et des paysages de la savane sahélienne dans la zone explorée :	38

INTRODUCTION

Dans le cadre du Master 2 d'Ingénierie Informatique et Technologies de l'Information et de la Communication (M2ITIC) de l'Université Gaston Berger (UGB), les étudiants sont amenés à suivre un stage de fin d'étude. L'objectif de ce stage est de donner une idée aux étudiants sur les réalités du monde de professionnel mais aussi de les familiariser avec les outils utilisés en entreprises ou dans les structures professionnelles et les méthodes de réalisation d'un projet complet répondant à une demande bien spécifiée. Mon stage s'est effectué au laboratoire de Biologie des Populations Animales Sahélo-Soudaniennes (BioPass) de l'Institut de Recherche pour le Développement (IRD) de Dakar Bel-Air sur le sujet « *conception, implémentation et simulation de la dynamique de l'environnement sahélien lié à l'invasion de la gerbille au Sénégal* ». Ce sujet s'inscrit dans le cadre du projet CERISE qui est un projet de recherche centré sur les « *scénarios d'invasion de rongeurs au Sahel cas de la Gerbille nigériane (*Gerbillus nigeriae*) et de la souris domestique (*Mus musculus*)* ».

Le travail que présente ce rapport part du besoin de modéliser un environnement sahélien dynamique dans la perspective d'y placer ensuite des gerbilles¹ qui vont se développer en fonction des caractéristiques de cet environnement et de ses changements.

Pour ce faire, nous allons d'abord procéder à la présentation générale dans laquelle nous définissons le contexte et le sujet de ce stage. La partie suivante concernera l'implémentation et sera consacrée aux outils et méthodes utilisés ainsi que la mise en œuvre de ces méthodes. Elle sera suivie par la présentation des résultats obtenus après simulations. Nous discuterons ensuite ces résultats avant de terminer par la conclusion et les perspectives.

¹ Gerbille: rongeur vivant dans des milieux arides semi-arides, grand déprédateur de culture sèche

CHAPITRE 1: PRESENTATION GENERALE

I. Contexte

L'IRD est une institution française qui travaille sur les questions de développement dans les pays à faible taux de croissance. Il regroupe en son sein différents laboratoires travaillant sur des thématiques diverses mais toutes focalisées sur le développement. Parmi ces laboratoires, figure le laboratoire BioPass.

BioPass est un laboratoire de recherche spécialisé dans les populations d'arthropodes, nématodes et rongeurs représentant une participation majeure dans l'agronomie, les forêts, la santé ou la conservation de biodiversité. Les sujets de recherche du laboratoire sont structurés sous forme de projet et parmi ces projets, on peut noter CERISE.

CERISE est un projet sur les scénarios d'invasion de rongeurs dans le sahel. Il est centré autour de deux axes d'études que sont (i) les impacts de la progression récente de la souris domestique au Sénégal d'Ouest en Est suivant le réseau des agglomérations, les infrastructures et les transports en développement et (ii) la progression de la gerbille au Sénégal du Nord vers le Sud avec l'aridification du climat et/ou l'anthropisation du milieu.

Pour apporter un aspect innovant au projet CERISE, des outils de modélisations différents ont été mis en œuvre en vue de produire des scénarios d'expansion d'espèces invasives. C'est ainsi que la plateforme SimMasto a été choisie pour la modélisation de l'invasion de la souris domestique et de la gerbille au Sénégal.

Développé par Jean Le Fur et son équipe, la plateforme de simulation SimMasto est un projet expérimental qui cherche à être un centre de connaissances dynamique sur la coévolution bioécologique des petits rongeurs et de leurs parasites. Elle est conçue avec un souci de robustesse et de flexibilité, ce qui lui procure la possibilité de réunir plusieurs cas de simulation comme (Figure 1): les processus d'hybridation, de capture-marquage-recapture dans l'étude de la dynamique d'une population de rongeurs, de diffusion et de propagation multi-échelles du rat noir, ..., et pour ce sujet, la mise en place d'un environnement sahélien dynamique propice à l'étude de l'invasion de la gerbille.

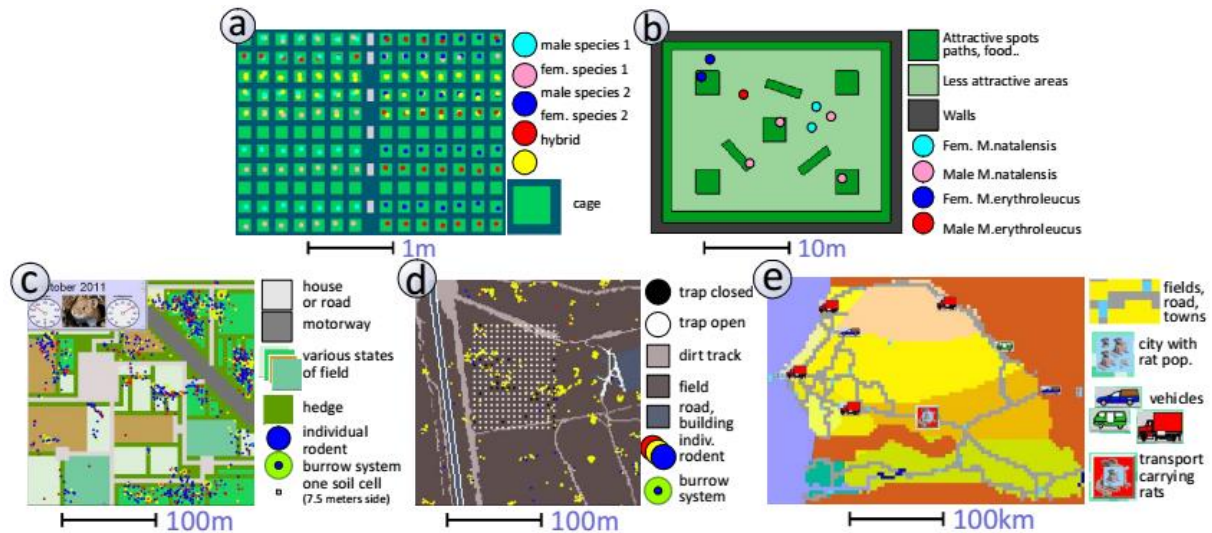


Figure 1 : Quelques cas d'utilisation intégrés dans SimMasto.

(a) expériences d'hybridation en cage et (b) enclos sur des espèces jumelles, (c) population de campagnols évoluant dans un paysage agricole, (d) expérience de capture-marquage-recapture dans une réserve africaine, (e) épidémiologie et transport du rat noir par des véhicules commerciaux au Sénégal (Afrique de l'ouest). (MBOUP, 2015)

La validité de SimMasto réside dans sa capacité à intégrer toute problématique du domaine. En effet, il est nécessaire qu'elle puisse prendre en compte la dimension spatiale, temporelle ou fonctionnelle de la problématique et quelle que soit son actualité, autrement dit qu'elle soit définie il y a dix ans, de nos jours ou dans dix ans. SimMasto est développé sous Repast Symphony² et est construite sur une base définitive de classification proposée par l'approche multi-agent, *i.e.*, « perception, délibération, décision, action/interaction ». La démarche retenue est l'agrégation successive d'étude de cas spécifiques à chaque chercheur et le plus contrastées possibles. Ainsi, à chaque ajout d'étude formalisée, de nouveaux types d'objets sont implémentés tandis que certaines briques déjà existantes peuvent être réutilisées. SimMasto est forgée sur une architecture trois tiers (*Data-Business-Presentation*). Les trois tiers *Data*, *Business* et *Presentation* sont représentés respectivement sur SimMasto par les packages *Données*, *Objet concret* et *Observation*. Le package *Objet concret* représente les divers objets qui peuvent être mis en œuvre dans les différents mondes simulés. Il est scindé en deux branches que sont l'ADN et le 'SQD'. L'ADN représente sur SimMasto les aspects de la génétique du vivant et est composé par des éléments mécaniques (gènes, allèles...) qui permettent les opérations liées aux génomes tels que la méiose, la ségrégation et la fécondation. Inspiré de l'approche système complexe, le 'SQD'³ représente la racine de l'arborescence des agents concrets. Il constitue l'agent proprement dit ou le contenant de l'agent avec ses critères génétiques.

² Repast Symphonie : <http://www.repast.sourceforge.net/>

³ SQD : Système quasi décomposable

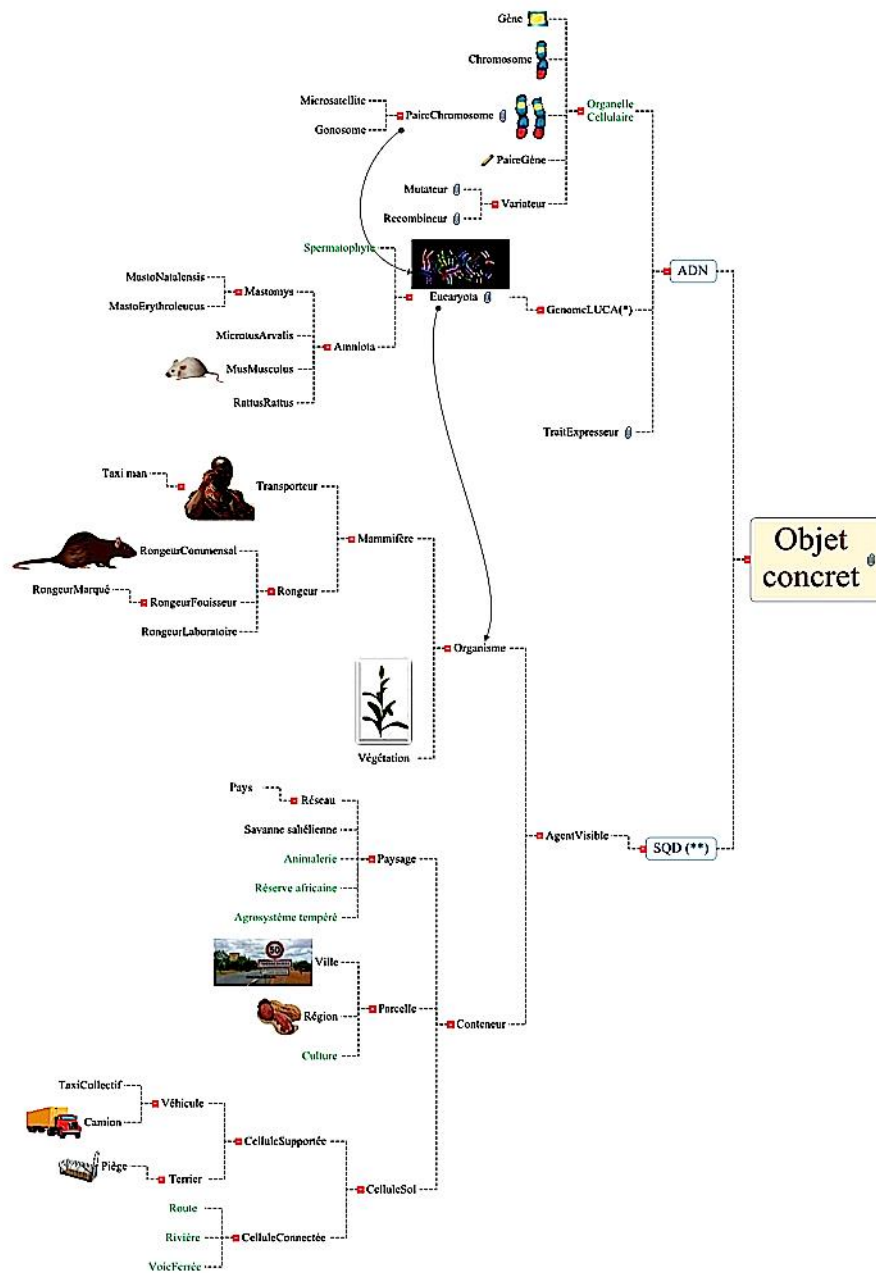


Figure 2 : Arborescence objet du modèle général de la plate-forme SimMasto telle qu'élaborée par les études de cas représentées.

En brun et bleu paquetages, en noir classes, en vert, nœuds non explicitement ou actuellement formalisés. Les relations horizontales sont de nature 'est-un-type-de', verticale en haut : agrégation, en bas : association voir texte). Chaque entité correspond à une classe d'objet au sens informatique du terme. Chacune peut être ou non spécifiée selon l'étude de cas sélectionnée. Seules les classes disposant de fonctionnalités propres à leur rang sont implémentées (fusion de chromosomes dans le cas d'un génome, creusement de terrier pour un rongeur fouisseur...). (*) LUCA: Last Universal Common Ancestor, (**) SQD: Système Quasi Décomposable voir texte (Le Fur, 2016a)

Le package *Objet concret* est rattaché à trois composantes d'importance égale : *donnée*, *observation* et *protocole*, qui font l'objet d'une recherche méthodologique sur la pluridisciplinarité.

Le module *donnée* joue le rôle de gestionnaire des constantes et des données fournies par les thématiciens. Ces dernières sont enregistrées sur un chronogramme qui est un fichier csv jouant le rôle d'intégrateur de données de recherche sur le simulateur suivant une date précise. Le module *Observation* se charge de la restitution du fonctionnement du système sous plusieurs modes d'affichage reliés entre eux. Il reprend le principe du système épiphyte ou conseiller qui modélise les actions de collecte de données (enquêtes, piégeages, ramassages...) effectivement réalisées dans les cas d'études.

Le module *protocole* permet de caractériser et d'automatiser les différentes études de cas agrégées dans le modèle (choix des échelles, entités, indicateurs, données d'entrée ; mise en œuvre...).

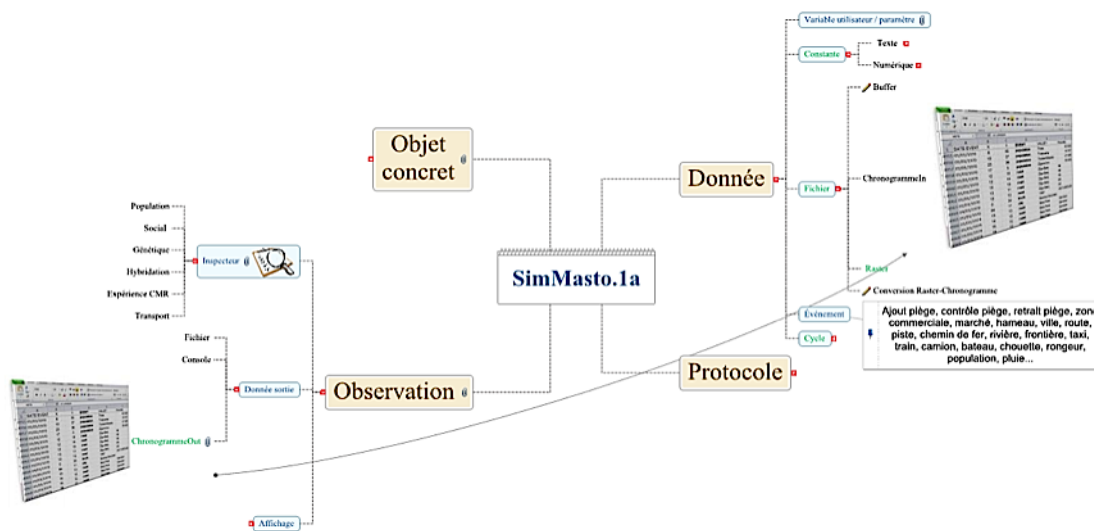


Figure 3 : Organigramme des classes associées à la gestion des simulations formalisées dans le paquetage 'objet concret' de la figure 2 telles que définies par les besoins des études de cas représentées. En brun et bleu paquetages, en noir classes, en vert, nœuds non explicitement ou actuellement formalisés. 'SimMasto.1a' : nom de l'architecture (Le Fur, 2016a)

II. Présentation du sujet

II-1. Problématique

La sécheresse de 1972 au Sénégal a entraîné de grands changements dans le système écologique (Bille, 1974). En effet, l'avancée du désert du nord vers le sud a favorisé l'apparition de nouvelles espèces invasives comme la gerbille nigériane qui est l'un des principaux déprédateurs de cultures sèches (Hima, 2010). La gerbille fait beaucoup de dégât dans les cultures de mil et de Sorgho. Les données recueillies en avril 2016 dans la commune de Tébékrou et de Labgar montrent une avancée importante de l'espèce dans le Ferlo (K. Bâ, C. Diagne, J. Le Fur, A. Sow et moi-même, données non publiées). Ainsi, la compréhension

et le contrôle de la progression de ce rongeur nuisible deviennent une priorité en vue d'atténuer les dégâts causés sur la culture.

II-2. Définition du sujet

Le sujet traité dans ce rapport s'intitule : conception, implémentation et Simulation de l'environnement sahélien lié à l'invasion de la gerbille au Sénégal.

Par conception, on sous-entend l'ensemble des méthodes et procédés mis en place pour une compréhension globale du sujet traité. L'implémentation consisterait à exploiter les connaissances issues de la conception pour les traduire en langage de programmation. Ainsi, la simulation définit l'exécution du programme implémenté en vue de simuler un phénomène complexe par exemple la simulation de l'évolution d'une population de rat noir au Sénégal (Figure 4) sur SimMasto.

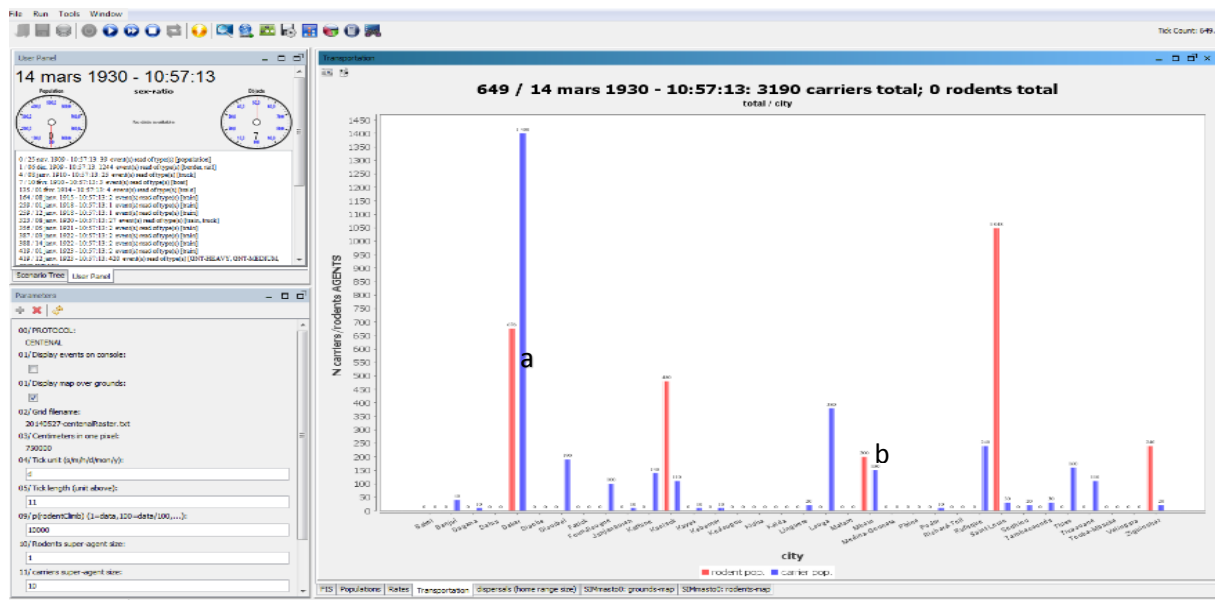


Figure 4 : Exemple d'étude réalisée sur la plate-forme SimMasto

Simulation de l'invasion du rat noir au Sénégal (a) évolution de la distribution des populations de rat dans les villes (b) nombre total de transporteurs présents dans la ville

II-3. Cahier des charges

L'objectif de ce stage est de construire un environnement sahélien dynamique dont la finalité est d'y placer des gerbilles qui vont se développer en fonction des caractéristiques environnementales et des divers changements qui seront notés.

II-3-1. Formalisation et manipulation de l'espace

La construction de l'environnement sahélien dynamique commence par la représentation d'un espace défini par des limites, sud-ouest (13.512°N, 18°W) et nord-est (18.98°N, 9°W) imposées avec une résolution d'un (01) kilomètre carré par cellule. Cependant, le but n'est pas de représenter tout le domaine à couvrir, mais de pouvoir se focaliser sur des zones particulières dont l'intérêt est déterminé par les biologistes et les écologistes qui travaillent sur cette question. Ainsi, trois grands axes de suivi sont requis.

Il est demandé tout d'abord (i) de représenter l'espace sous forme d'une matrice de cellule ainsi que (ii) d'élaborer un système qui permet de passer indifféremment de coordonnées géographiques aux coordonnées discrètes de la grille, et (iii) de pouvoir extraire des zooms, c'est-à-dire extraire la collection de cellules correspondant au zoom et l'ensemble des données qui caractérisent le contenu de la cellule en question.

II-3-2. Nature et manipulation des données de l'environnement

Deux types de données caractérisent l'environnement : les éléments disposés sur le terrain tels que culture, arbre, arbuste, sol nu ou herbe (couverture du sol) et d'autre part, l'intensité des précipitations (pluviométrie) qui conditionne la dynamique de ces divers éléments.

En ce qui concerne la couverture du sol, les données utilisées sont issues de traitement d'images satellite réalisées par ailleurs. Les données sont disponibles sous la forme d'une matrice unique à la résolution retenue pour l'étude (01km). Chaque valeur de la matrice (ou pixel de l'image satellite) code le contenu de la cellule correspondante en fonction des éléments qui s'y trouvent (Ex : culture et arbuste ou sol nu). Le travail consistera à récupérer les valeurs de ces codes à partir du fichier brut et de les transcrire en autant d'éléments distincts qui seront disposés dans chacune des cellules correspondantes en respectant un écartement minimum paramétrable. Chacun de ces éléments sera donc individualisé et devra être visualisé sur une interface graphique et/ou cartographie appropriée.

Les données de la pluviométrie seront mises à disposition elles aussi à partir de données de télédétection. Contrairement aux données de couverture du sol pour lesquelles une seule carte sera disponible, les données de pluviométrie se présenteront sous la forme de cartes distinctes mises à jour mensuellement. Le format disponible ici pour ces données se présente sous la forme d'une image de type raster présentant une continuité de valeur (format jpeg) et couvrant la période de 1999 jusqu'en 2013 donc un nombre de 180 cartes à traiter. Il sera nécessaire de lire ces images et de les convertir en valeur numérique (ASCII), de ré-échantillonner l'image pour la rendre conforme avec la résolution retenue pour le modèle puis de déclassifier les

valeurs en un nombre restreint de classe de valeur de précipitation. Le jeu de cartes mensuelles transformé ainsi obtenu sera lu par le simulateur à la date correspondante à leurs occurrences.

Ces éléments représentent des données pertinentes vis-à-vis des rongeurs.

II-3-3. Mise en relation des objets de l'environnement avec l'évolution des précipitations

Les caractéristiques particulières de différents zooms sélectionnés seront soumises et répondront à l'intensité des précipitations propre à leurs zones (croissance et décroissance de la végétation). Pour se faire, on formalisera une courbe de réponse de type logistique liant la croissance à la pluie selon des taux paramétrables et définis de façon spécifique à chaque type de végétation représentée. Le modèle et ses simulations devront ainsi permettre de restituer l'évolution au cours des années de la nourriture disponible pour les rongeurs. Ces évolutions devront pouvoir être visualisées graphiquement ou sous la forme de série de données standardisées type fichier Excel comprenant les biomasses pour chaque espèce ainsi que la pluviométrie moyenne des zooms sélectionnés. Le développement sera réalisé en utilisant la plateforme SimMasto spécialisée dans la modélisation des rongeurs. Certaines des bibliothèques existantes pourront être utilisées pour le travail (lecture de fichier bitmap, package graphique, etc.).

CHAPITRE 2: IMPLEMENTATION

I. Outils utilisés

Différents outils ont été utilisés dans la réalisation de ce projet. Parmi ces outils, nous pouvons citer : Corel Paint qui est un logiciel de traitement d'image, Repast Symphony qui est un système multi plate-forme interactif de modélisation en Java et qui est incorporé dans l'environnement de développement d'Eclipse. Conçu pour être simple d'utilisation, Repast Symphony fait partie des plateformes qui permettent à des non-informaticiens de pouvoir représenter des cas d'études par programmation de modèle. Elle contient des primitives permettant de créer des systèmes multi-agents (SMA) sans difficultés (Realini, 2011).

Un SMA est un ensemble composé par un environnement délimité par un espace et est utilisé fondamentalement dans l'intelligence artificielle. Cet environnement contient un ensemble d'objets situés et un ensemble d'agents capables de percevoir les objets et les agents présents dans le milieu avec un rayon de perception défini selon l'agent. D'autre part les SMA permettent de lier les agents par un ensemble de relations univoques (relations causales ou à sens unique), rétroactives (cycliques) ou interactives (à double sens) avec leurs voisins. Sur la plateforme SimMasto le concept d'agents est utilisé pour simuler par exemple l'invasion du rat noir au Sénégal (projet CHANCIRA⁴) ou la dynamique de rongeurs dans la réserve de Bandia. Chaque rongeur y est représenté par un agent capable d'acquérir des ressources, de connaître son environnement et possédant une connaissance et des objectifs.

II. Méthodes

La prise en compte des données fournies par les biologistes et les écologistes dans le simulateur SimMasto nécessite un ensemble de traitements. C'est ainsi que suivant le cahier des charges présenté précédemment, il nous a été demandé d'apporter à la plateforme des fonctionnalités permettant de passer indifféremment de coordonnées géodésiques en coordonnées discrètes de la grille, de digitaliser les images satellites de la pluviométrie en matrice numérique, d'extraire à partir des données de référence de la couverture du sol une zone de questionnement fixée par les biologistes (zoom) et de mettre en relation les objets de l'environnement avec l'évolution des précipitations. Ainsi, nous comptons rappeler dans

⁴ **CHANCIRA** : **CH**ANGement environnementaux, **CI**rculation de biens et de personnes : de l'invasion de réservoirs à l'apparition d'anthropozoonoses. Le cas du **Rat** noir dans l'espace Sénégal-malien (projet de recherche ANR-11CEPL-010, janvier 2011, 65p).

chaque point des méthodes, la tâche demandée dans le cahier des charges avant de présenter le travail effectué.

II-1. Numérisation des images de la pluviométrie

Rappel du cahier des charges : le format disponible ici pour les données de la pluviométrie se présente sous la forme d'une image de type raster présentant une continuité de valeurs (format jpeg) et couvrant la période de 1999 jusqu'en 2013 donc un nombre de 180 cartes à traiter. Il sera nécessaire de lire ces images et de les convertir en valeur numérique (ASCII), de ré-échantillonner l'image pour la rendre conforme avec la résolution retenue pour le modèle puis de déclassifier les valeurs en un nombre restreint de classes de valeur de précipitation.

Travail apporté : la numérisation vise à convertir des fichiers images de la pluviométrie fournis par les géomaticiens du projet en fichiers raster. La pluviométrie constitue l'un des facteurs déterminants de la croissance de la végétation. En effet, la quantité de biomasses de la couche herbacée observée est souvent proportionnelle à la quantité d'eau présente dans le milieu. Ainsi, il devient nécessaire de la prendre en compte dans tout modèle de simulation de la végétation. Dans notre modèle, nous avons utilisé des données satellites du produit TRMM3B42 (Huffman et al., 2007) obtenues sur la presque totalité de l'Afrique de l'Ouest et représentant des cumuls par mois qui sont stockés sous forme de carte (Figure 5). L'exploitation de ces cartes est rendue possible par une suite de traitement que sont :

- ✓ Le ré-échantillonnage de l'image en classe de valeur et en résolution
- ✓ La reclassification de couleur
- ✓ La digitalisation de l'image en matrice numérique et la sauvegarde du résultat

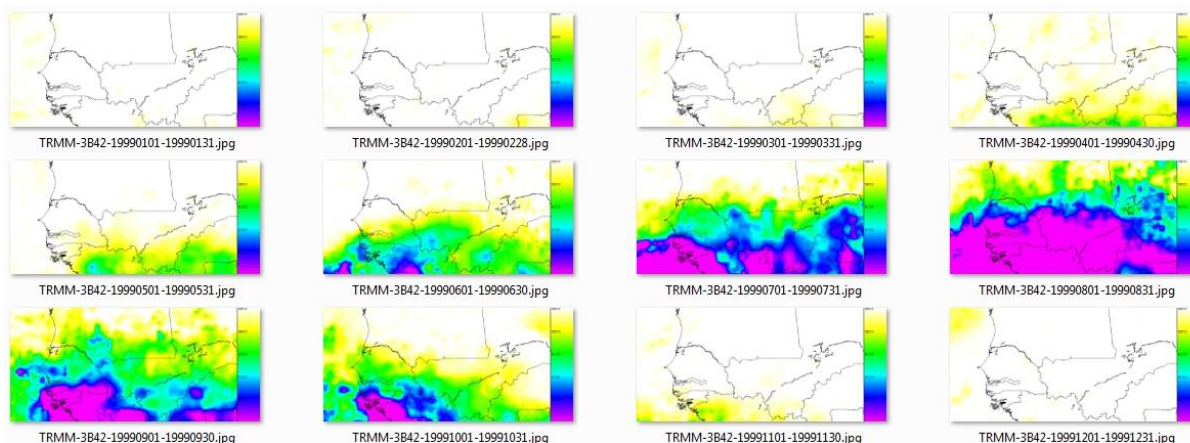


Figure 5 : Données satellite de la variation de pluies mensuelle de l'année 1999 sur la zone Ouest Africaine, données fournies sous forme d'images Bitmap

II-1-1. Séquence des opérations

L'outil Corel Paint est utilisé pour le ré-échantillonnage et la conversion de couleur des données satellite. Nous appliquons ainsi une méthodologie de traitement pour gérer efficacement les 180 cartes de pluviométrie avec les mêmes paramètres.

a. Ré-échantillonnage de l'image

Les données obtenues dépassent la zone de la moyenne emprise⁵ (ME). Ainsi, nous avons défini un masque (Figure 7) pour obtenir le même découpage pour chaque fichier à traiter. Le masque numérique est systématiquement positionné de façon identique au pixel près.

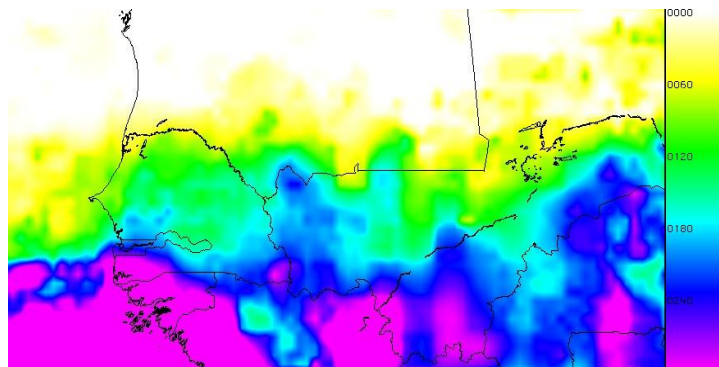


Figure 6 : Image originale (juillet.1999)

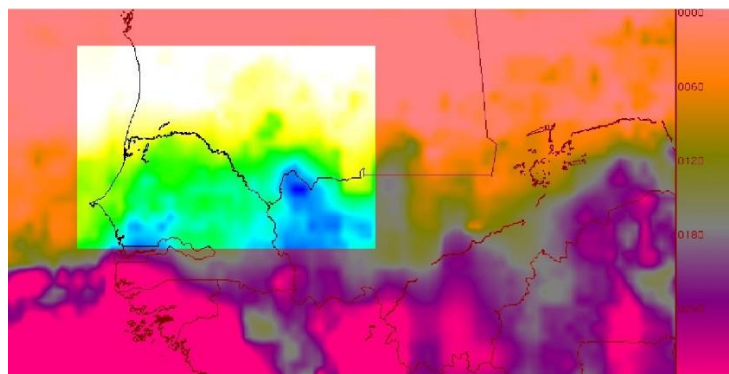


Figure 7 : Positionnement du masque de la ME

Après découpage, nous obtenons la zone de la ME mais l'échantillonnage en pixel ne correspond pas à ce que nous attendons. En effet, le modèle de la ME doit être représenté sur une surface de 976 km de largeur sur 606 km de hauteur pour un pixel par kilomètre carré. D'où l'utilité de ré-échantillonner l'image extraite. D'autre part, l'image jpeg au format 24bits

⁵ Moyenne Emprise (ME) : zone d'étude du modèle d'occupation de site (prédiction, prédiction, probabilité de colonisation-installation)

obtenue après ré-échantillonnage est classifiée en 16,8 millions de couleurs ce qui rendrait leur exploitation sur le simulateur compliquée.

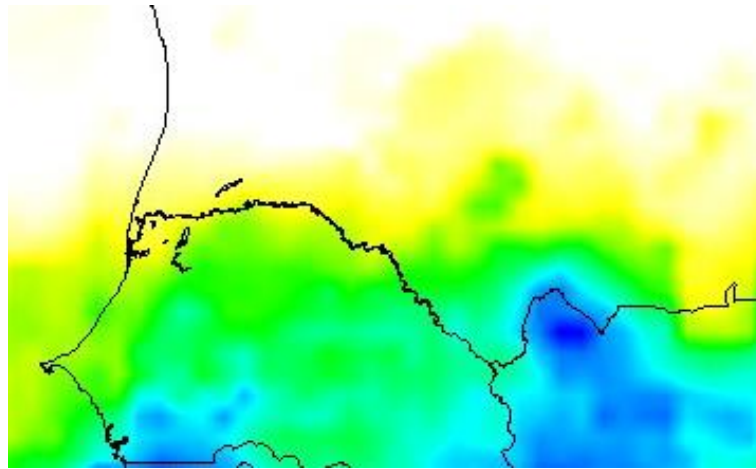


Figure 8 : Image obtenue après extraction et ré-échantillonnage

b. Reclassification de couleur

Nous avons effectué une reclassification en utilisant une palette de 9 couleurs conforme au niveau de résolution effectif du modèle. Après reclassification, l'image est sauvegardée sous un format bitmap avec un nom constitué par le nom du fichier source et la date des deux extrémités du mois concerné. Ainsi, s'achève le traitement de l'image sur Corel Paint.

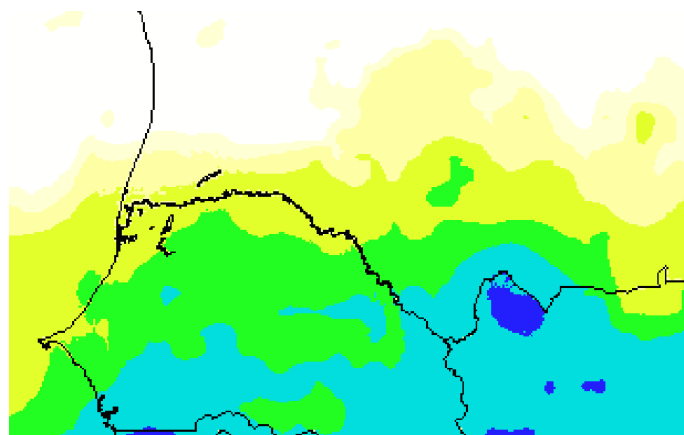


Figure 9 : Image reclassifiée en 9 couleurs

c. Digitalisation et sauvegarde de l'image

La conversion de l'image en code ASCII est réalisée grâce à la fonction *imgRasterloader* de la classe *C_ReadRaster* de SimMasto. Cette fonction reçoit en paramètre le chemin du répertoire de l'image et retourne la matrice numérique correspondante. Ainsi, la matrice obtenue est donnée en paramètre avec le nom du fichier, le nom de l'emprise concernée par le fichier et le chemin du répertoire de sauvegarde sur la fonction *saveBitmapFile*. Cette dernière

calcule le maximum et le minimum de la matrice puis effectue une sauvegarde dans un fichier texte en respectant le format des fichiers raster sur SimMasto.

La digitalisation finalise le traitement de l'image. Ainsi, le simulateur peut l'utiliser pour les simulations (Figure 17).

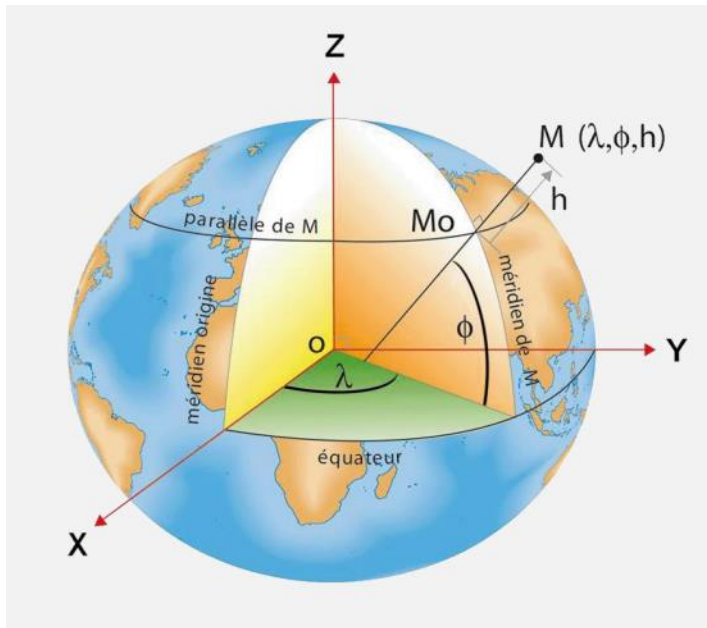
II-2. Conversion de coordonnées

Rappel du cahier des charges : *formalisation et manipulation des données de l'environnement : (ii) élaborer un système qui permet de passer indifféremment de coordonnées géographiques aux coordonnées discrètes de la grille.*

Travail apporté : l'objectif dans cette partie est de convertir indifféremment les coordonnées GPS d'un point du raster en coordonnées cartésiennes d'une part et d'autre part, de calculer sa position par rapport à l'origine du raster. En effet, utilisées couramment dans le monde de la recherche, les coordonnées géographiques permettent de repérer les zones d'études. Elles sont employées sur le projet CERISE pour représenter les points de capture de rongeurs et de ramassage de pelotes de chouette. Ces données de capture et de ramassage sont exploitées par le simulateur SimMasto pour comprendre le scénario d'invasion de la gerbille au Sénégal. Ainsi, ces coordonnées seront prises en compte au fur et à mesure suivant les dates de collecte par la plate-forme pour initialiser la simulation du modèle. Deux méthodes seront utilisées dans la séquence des opérations pour réaliser les conversions.

II-2-1. Méthodes de transformations des coordonnées géodésiques de Clarke

Pour faire la conversion de coordonnées géographiques en coordonnées cartésiennes nous, avons utilisé la méthode de transformation des systèmes géodésiques de Clarke (IGN/SGN, 2013). Cette méthode nous donne les équations permettant de calculer les coordonnées cartésiennes associées au point choisi.



Légende :

M : un point dans l'espace

Mo : Projection de M sur la terre

X, Y, Z : les coordonnées cartésiennes de M

λ, ϕ, h : Coordonnées géodésique de M

O : centre de la terre

Figure 10 : Coordonnées géographiques et cartésiennes d'un point dans un ellipsoïde (Clarke 1880 IGN)

Nous obtenons ainsi, les équations de base pour chacun des axes de coordonnées (X, Y et Z) suivantes :

$$X = (R + h) \cos \phi \cos \lambda \quad (1)$$

$$Y = (R + h) \cos \phi \sin \lambda \quad (2)$$

$$Z = (R(1 - e^2) + h) \sin \phi \quad (3)$$

Avec :

- ✓ h : L'altitude du point qui est assimilée à 0 dans notre cas
- ✓ λ : La longitude du point choisi en radian
- ✓ R : Le rayon de la terre calculé suivant : a le rayon de la terre suivant l'équateur (6 378 249,2 m), b le rayon de la terre suivant les pôles (6 356 515,0 m) et ϕ la latitude en radian d'un point choisi tel que :

$$R = \frac{a}{\sqrt{1 - e^2 (\sin \phi)^2}} \quad (4)$$

- ✓ e : l'excentricité de l'ellipsoïde donnée par l'équation suivante:

$$e = \frac{\sqrt{a^2 - b^2}}{a} \quad (5)$$

On remplace $(\sin \phi)^2$ par $\frac{1 - \cos(2\phi)}{2}$

Le calcul de la distance entre la projection du point choisi suivant la largeur et le point origine du repère nous donne la coordonnée discrète du point suivant la largeur du raster. Celle de la projection du point choisi suivant la hauteur et le point origine nous donne la coordonnée discrète du point suivant la hauteur du raster.

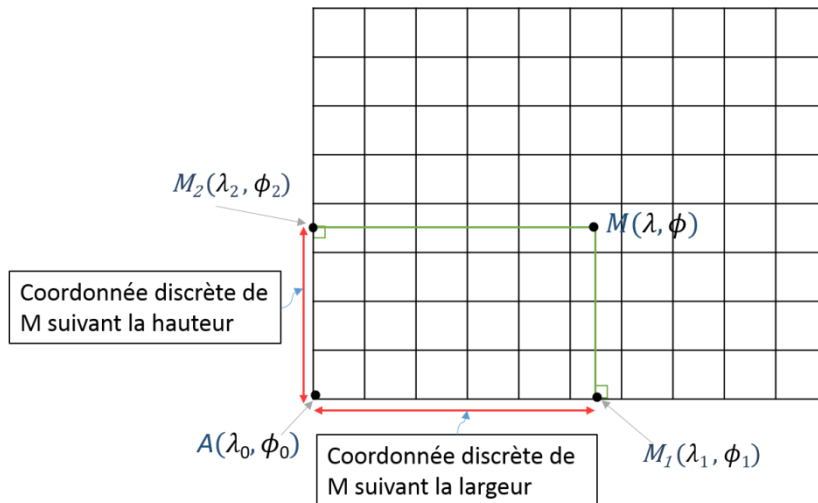


Figure 11 : projection du point sur la largeur et la hauteur du raster

II-2-2. Méthode de conversion de coordonnées par projection orthographique

La conversion de coordonnées d'espace cartésien en coordonnées géodésiques est réalisée à l'aide des formules de la projection orthographique. Une projection orthographique est une projection de perspective par laquelle une sphère est projetée sur un plan tangent.

Les formules de la projection orthographique permettant de calculer la longitude λ et la latitude ϕ du point choisi sur la sphère suivant ses coordonnées cartésiennes (X, Y) dans le plan trigonométrique sont :

$$\begin{cases} \phi = \arcsin\left(\cos(c) \sin(\phi_0) + \frac{Y \sin(c) \cos(\phi_0)}{\rho}\right) \\ \lambda = \lambda_0 + \arctan\left(\frac{X \sin(c)}{\rho \cos(\phi_0) \cos(c) - Y \sin(\phi_0) \sin(c)}\right) \end{cases} \quad (6)$$

Avec :

- ✓ c : l'écart angulaire au centre de la projection
- ✓ λ_0 et ϕ_0 : les coordonnées du centre de la projection (pour notre cas, les coordonnées de l'origine du raster)
- ✓ ρ est une distance calculée tel que :

$$\begin{cases} \rho = \sqrt{X^2 + Y^2} \\ c = \arcsin\left(\frac{\rho}{R}\right) \end{cases} \quad (7)$$

L'écart angulaire c au centre de la projection est calculé pour éliminer les latitudes correspondant à des points en dehors de la carte. Les points sont représentés si et seulement si le cosinus de c est positif.

II-3. Extraction de zoom

Rappel du cahier des charges : formalisation et manipulation des données de l'environnement : (iii) pouvoir extraire des zooms, c'est-à-dire extraire la collection de cellules correspondant au zoom et l'ensemble des données qui caractérisent le contenu de la cellule en question.

Travail apporté : la modélisation de la dynamique de la population de gerbille sur le zoom de la petite emprise (Zoom PE) du projet CERISE vise à lier de façon fonctionnelle et dynamique les populations de rongeurs, la couverture du sol et les données pluviométriques

aux données obtenues *in situ* par piégeage ou analyse de pelotes de chouettes (prédateurs). Ainsi, chacune de ces données doit être prise en compte durant la période de simulation qui s'étend sur 15 années avec un pas d'exécution d'une journée. Par conséquent, le nombre d'objets à représenter par pas de temps et par cellule (23450 cellules au total) se voit multiplié par quatre voire même six. Ce qui diminue de façon très conséquente la vitesse d'exécution du modèle.

Pour contourner cette situation, des zones d'intérêts sont ciblées.

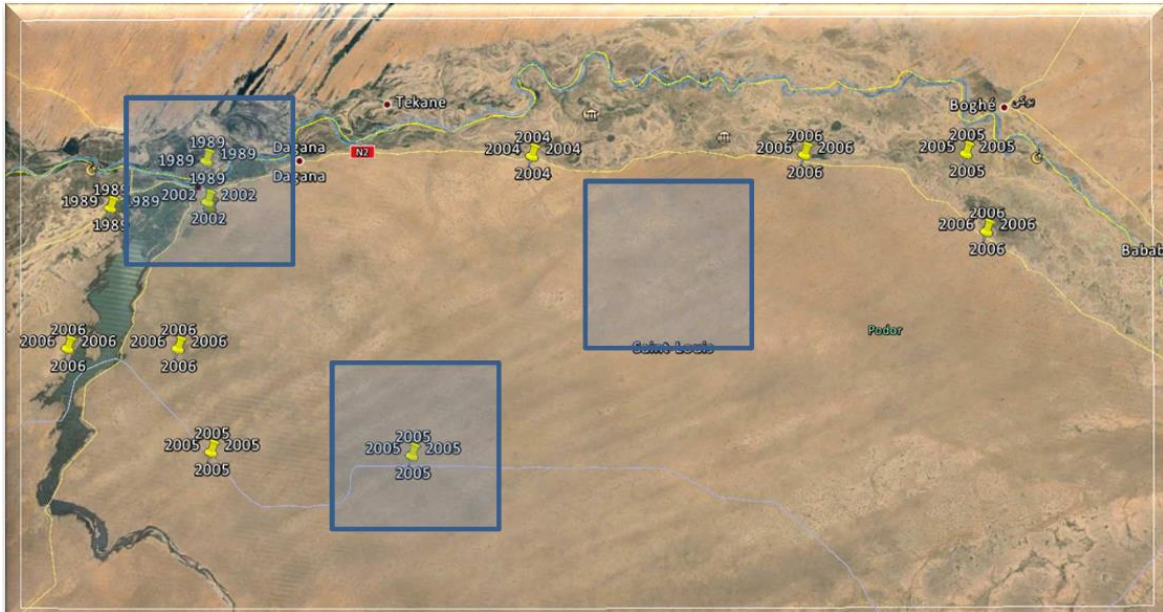


Figure 12 : Zones d'intérêt sur la petite emprise (source: Modèle de l'approche petite emprise JLF 12-11-2015)



Figure 13 : Zones d'intérêt sur la moyenne emprise (Le Fur, 2015)

Ces nouvelles zones d'intérêt représentent des carrés de, par exemple, 30 km² extraits à partir de la petite emprise ou de la moyenne emprise (pluviométrie, couverture du sol). De ce fait, un algorithme mis en place nous permet d'appliquer « à la demande » une extraction d'un zoom selon une origine et un nombre de cellules fixés au préalable. L'algorithme permet d'extraire un zoom à partir de fichiers texte, mais également à partir de fichiers bitmap.

II-3-1. Méthode d'extraction

Pour réaliser l'extraction, nous procédons à la transformation des fichiers source en matrice d'entiers. Ensuite, nous calculons la position de l'origine du raster à extraire (zoom plus petit) en utilisant la méthode de calcul de distance. Pour terminer, nous procédons à l'extraction du zoom tout en vérifiant si les limites sont bien respectées.

II-4. Mise en relation de la pluviométrie et de la végétation

Rappel du cahier de charge : *formalisé une courbe de réponse de type logistique liant la croissance à la pluie selon des taux paramétrables et définis de façon spécifique à chaque type de végétation représentée.*

Travail apporté : la végétation dans les milieux sahélo-soudaniens se présente sous une forme variée avec la présence de certaines espèces d'herbes, d'arbustes, d'arbres et de cultures. Elle constitue une source de nourriture indispensable pour la survie des granivores comme la gerbille. La croissance de cette végétation est conditionnée par plusieurs facteurs écologiques dont la pluviométrie. Cette dernière connaît de fortes fluctuations suivant la saison et constitue ainsi le facteur le plus déterminant de la croissance de la végétation. Des données satellites du produit TRMM3B42 (Huffman et al., 2007) obtenus sur notre zone d'étude (petite emprise) à l'aide d'algorithme de pluie utilisant une combinaison de données de plusieurs capteurs micro-ondes, et des températures de brillance, nous fournit une série de données de pluies mensuelle étalée sur quinze années (1999 à 2013). Le problème posé peut s'exprimer de la façon suivante : comment obtenir la variation de la croissance de la végétation suivant la quantité d'eau journalière présente dans la zone d'enracinement?

II-4-1. Fonction de croissance logistique

Pour répondre à la question posée, nous avons fait appel à un algorithme basé sur la fonction de croissance logistique de Verhulst (Verhulst, 1845). En effet, Verhulst cherche les fonctions f définies et positives sur l'intervalle $[0 ; +\infty$ [vérifiant les deux conditions suivantes :

$$\begin{cases} y(0) = y_0 \\ y' = ry \left(1 - \frac{y}{K}\right), r > 0 \text{ et } K > 0 \end{cases} \quad (8)$$

Avec :

- ✓ y : la croissance obtenue à l'état $t \neq 0$
- ✓ y_0 : la valeur de la croissance initiale ($t = 0$)
- ✓ r : le taux de croissance
- ✓ K : la capacité du milieu, limite les possibilités de la croissance et définit la quantité de biomasse à l'équilibre

La solution analytique de cette équation donne $f(t)$ tel que :

$$f(t) = K \frac{1}{1 + \left(\frac{K}{y_0} - 1\right) e^{-rt}} \quad (9)$$

Ainsi, selon les valeurs de y_0 , la fonction est soit constante (pour $y_0 = K$), soit croissante (pour $y_0 < K$), soit décroissante (pour $y_0 > K$)

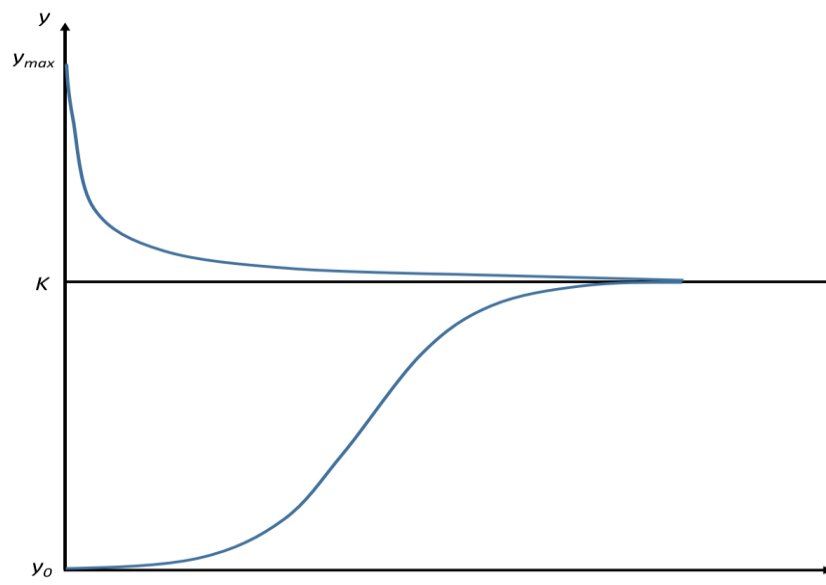


Figure 14 : Diagramme de la fonction de croissance logistique de Verlhust

Si la valeur de Y est petite ou très grande alors respectivement elle croît ou décroît et tend à se rapprocher de la valeur de K .

II-5. Lecture des données

Les biologistes et les écologistes nous fournissent des données (connaissances) variées obtenues après des années de recherche sur le sujet d'étude. Pour intégrer en temps utile ces données dans la simulation, le modèle orienté données est adopté. L'environnement de simulation est initialisé avec les données de *landcover* (couverture du sol) qui découlent de la

numérisation des données satellitaires de couverture du sol (SYNMAP) en code ASCII. L'espace représenté est ainsi discrétisé en cellules enrichies d'objets représentatifs d'un (01) pixel pour 1km^2 . Ces objets de couverture du sol représentent les types de végétation (arbre, arbuste, herbe, culture, sol nu et/ou eau) présents dans la cellule.

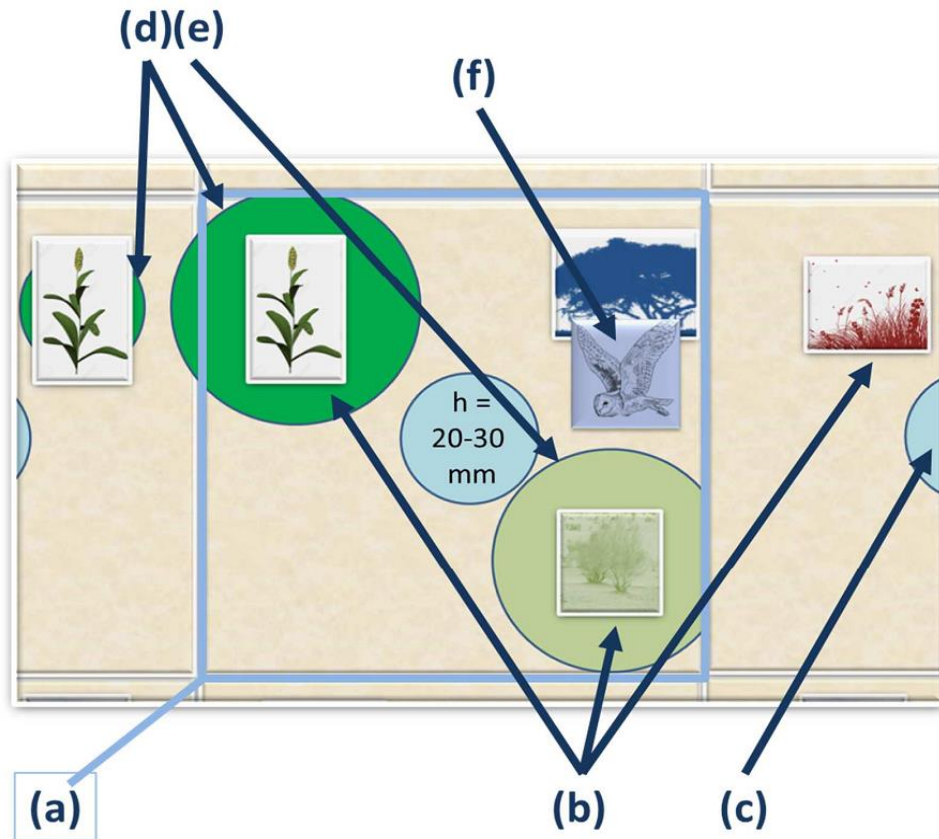


Figure 15: Principe de simulation de l'environnement sahélien :

- (a) le paysage simulé est représenté par un ensemble de cellules jointives de 1km^2 , le carré bleu représente une de ces cellules.
- (b) les données satellitaires de couverture du sol (SYNMAP) caractérisent des zones composites avec arbres, arbustes, herbe, culture, sol nu, eau : pour chaque valeur issue de l'image satellite, le sol est enrichi d'objets représentatifs de la couverture du sol spécifique à cette cellule : culture, herbe, arbuste...
- (c) pour chaque cellule et chaque mois, la hauteur de pluie propre à chaque cellule est lue à partir des fichiers de télédétection,
- (d) et (e) une fonction logistique liée à l'intensité et à l'étendue de la saison des pluies est appliquée (d) aux objets culture (production de biomasse) ainsi que (e) aux objets arbustes (production de graines),
- (f) de même, suivant l'approche orientée données qui a été retenue, les pelotes géo-localisées sont formalisées comme des objets prédateurs (chouette *Tyto alba*). Enfin, les résultats obtenus *in situ* à partir des piégeages et des identifications de pelotes sont formalisés comme autant d'individus ou de groupes d'individus rongeurs géo-localisés (non représenté) susceptibles d'interagir entre eux et avec cet environnement. Cette approche est limitée dans un premier temps à *Gerbillus nigeriae* et *Taterillus sp.* (Le Fur, 2015)

En outre, les données de la pluviométrie, de piégeages de gerbilles et de ramassages de pelotes remplissent le chronogramme. Le chronogramme est un système minimaliste mis au point pour traiter indistinctement des événements de type géographiques, biologiques, écologiques, climatologiques, historiques, etc. Il sert à créer et à mettre à jour l'environnement et est défini *in situ* par un fichier *csv* composé de lignes et de colonnes dans

lequel chaque ligne ou bloc de ligne représente un événement complet. Le chronogramme respecte un format unique et est nécessairement chronologique. Il répond et par ordre aux trois questions suivantes : Quand, Où et Quoi et optionnellement, un complément d'information (comment, combien).

Tableau 1 : Extrait du chronogramme 20151029_GerbilEventsOfficiel.2a.jlf utilisé dans le modèle gerbille

DATE EVENT	X	Y	EVENT	VALUE1	VALUE2	COMMENT
01/01/1999		0	0 rain			2015.10.29 M.Sall & J.Le Fur
17/07/2011	-14.82825	15.81816667	Gerbillus_nij F		489846	20151005-CapturesGnigeriaePE.jlf.ms
17/07/2011	-14.82825	15.81816667	Gerbillus_nij M		489848	20151005-CapturesGnigeriaePE.jlf.ms
01/08/2011		0	0 rain			2015.10.29 M.Sall & J.Le Fur
01/09/2011		0	0 rain			2015.10.29 M.Sall & J.Le Fur
25/09/2011	-14.412	16.495	owl	Gerbillus_nij	2	Dodel/ G. henleyi / nigeriae remplacé par Gerbillus_nigeriae, JLF-MS, 29.10.2015
25/09/2011	-14.412	16.495	owl	Gerbillus_nij	13	Dodel, JLF-MS, 29.10.2015

Ce tableau présente dans ses colonnes les réponses aux différentes questions que doit satisfaire un chronogramme. On note ainsi, la date de l'événement (DATE_EVENT) qui répond à la question du Quand, les coordonnées géographiques (X et Y) de l'événement de piégeage de gerbilles ou de ramassage de pelotes de chouette pour la question du lieu (Où), la colonne EVENT décrivant le type d'événement noté (piégeage, pluies ou ramassage de pelote), la colonne VALUE1 qui renseigne le sexe pour les captures ou le type de rongeurs pour les pelotes, la colonne VALUE2 pour le numéro d'identification des gerbilles capturées ou le nombre de rongeurs (*Gerbillus*, *Taterillus*, etc.) trouvés sur les pelotes ramassées, enfin la colonne COMMENT qui représente les commentaires qui peuvent être notés sur l'évènement. En outre, le 01/01/1999 représente dans notre extrait de chronogramme le début de la simulation mais aussi l'arrivée du premier événement de pluie ; le 17/07/2011, événement pour la création d'un agent gerbille (*Gerbillus_Nigeriae*) femelle aux coordonnées (15,82°N, 14,83°W) ; le 25/09/2011, événement de création de deux agents de gerbilles sur la surface de rayon 2,5 km² et de centre le point de coordonnées (16,495°N, 14.412°W).

III. Mise en œuvre

La mise en œuvre des différents procédés de simulation nécessite une phase d'installation des outils et de familiarisation avec la plateforme SimMasto. C'est ainsi, que la première semaine de mon stage a été consacrée à l'installation des outils et l'initiation dans les règles de codage sur SimMasto. Successivement, nous nous sommes attelés aux tâches qui nous ont été

données (dans le cahier des charges) et parallèlement, nous avons mis en place un nouveau protocole d'étude pour la gerbille.

Le protocole est un objet dérivé de la classe abstraite *A_Protocol* qui coordonne les étapes de la simulation. En effet, il donne la main à la classe *landscape* au début de la simulation pour l'initialisation du raster puis positionne la végétation suivant l'affinité du milieu. À chaque pas de temps, les nouveaux événements qui surviennent sont exploités pour la mise à jour de l'environnement de simulation et chaque agent présent effectue son action du jour.

Le code source est présenté en annexe, version utilisée pour le protocole de simulation de la gerbille (Zoom petite emprise).

III-1. Mise en œuvre de la numérisation des images

Cette partie est effectuée d'abord sur Corel Paint et finalisée sur la plateforme SimMasto.

III-1-1. Traitement de l'image sur Corel Paint

Le traitement de l'image sur Corel Paint suit les étapes suivantes :

- ✓ Création du masque de la moyenne emprise masqueME.1a.cpt
- ✓ Création de la palette de 9 couleurs Contrainte : deux variations de jaune
 - Ex : TRMM-3B42-19990101-19990131.gif
- ✓ Chargement de la palette de couleur 20150922-palettePluie09Couleurs.xml (8 couleurs + la noire des frontières)
- ✓ Extraction de l'image de la moyenne emprise suivant le masqueME.1a.cpt
- ✓ Récupération du nom initial du fichier (source et date de la carte)
 - Ex : TRMM-3B42-19990101-19990131.gif
- ✓ Ré-échantillonnage de la carte en pixels (976 x 606 en hauteur).
- ✓ Sauvegarde du fichier en format bmp

III-1-2. Conversion et sauvegarde sur SimMasto

Cette partie complète la numérisation des images. L'image traitée sur Corel Paint va être (i) lue et convertie en format ASCII par la fonction *imgRasterLoader* de la classe *C_ReadRaster* puis, (ii) le nom du fichier à sauvegarder est construit avec la fonction *computeFileName* de la classe *C_ConvertRainFile* et enfin, (iii) le raster de la moyenne emprise en format ASCII conçu est sauvegardé avec le nom obtenu dans (ii), par la fonction *saveBitmapFile* de la classe *C_ConvertRainFile*.

III-2. Mise en œuvre de la conversion de coordonnées

III-2-1. Géographiques en discrètes

La conversion de coordonnées géographiques en coordonnées discrètes commence par le calcul des coordonnées cartésiennes en mètres à l'aide de la fonction

convertCoordinate_Umeter de la classe *C_ConvertGeographicCoordinates*. Les coordonnées en mètres obtenues sont utilisées pour déterminer les coordonnées discrètes du point choisi.

i. Calcul de la distance entre deux points

Utilisons le théorème de Thalès pour le calcul de la distance du point choisi à l'origine

$$d = \sqrt{X^2 + Y^2 + Z^2}$$

Ainsi, son implémentation permet à la fonction *distanceToRasterOrigin_Umeter* qui reçoit en paramètre la latitude et la longitude du point choisi, de retourner la distance du point par rapport à l'origine du raster.

ii. Calcul des distances en coordonnées discrètes

Pour la distance AM_1 (Figure 11): nous utilisons la fonction *distanceToRasterOrigin_Umeter* qui nous renvoie la distance du point en mètre. Cette valeur est ensuite divisée par la taille d'une cellule pour obtenir la distance voulue. Le même procédé est appliqué pour le calcul de la distance AM_2 (Figure 10).

iii. Calcul du nombre de cellules en hauteur et en largeur

Le calcul de la distance en espace continu nous renvoie une valeur décimale. Ainsi, pour obtenir le nombre de cellules en hauteur et en largeur, on effectue une conversion en entier de la distance obtenue sur ii.

III-2-2. Discrètes en géographiques

La conversion de coordonnées discrètes en coordonnées géographiques est effectuée sur SimMasto à l'aide la fonction *convertCell_Udegree* suivant les quatre étapes ci-dessous :

- ✓ Calcul du rayon de la terre (*worldRadius*) suivant la latitude du point choisi
- ✓ Calcul du cosinus de l'écart angulaire *c*
- ✓ Calcul des coordonnées géodésiques du point (*coordinateCell_Udegree*) suivant ses coordonnées cartésiennes et les coordonnées géodésiques du point origine du raster.
- ✓ Conversion des valeurs obtenues en radian avec la fonction *convertRadian_Udegree*

III-3. Mise en œuvre de l'extraction de zoom

III-3-1. Extraction de fichier texte

Pour réaliser l'extraction sur les fichiers texte, nous faisons d'abord une extraction du raster (matrice d'entiers) de l'emprise (ME) avec la fonction *txtRasterLoader* de la classe *C_ReadRaster*. Cette fonction charge via le chemin du répertoire du fichier, la grille sous format ASCII en mémoire puis extrait le raster correspondant qu'elle renvoie en sortie.

Le nom du fichier du zoom à créer est construit à partir du nom du fichier en entrée après une vérification du format du fichier par la fonction *getExtension*.

La fonction *extractZoomFromGrid* qui reçoit en paramètre la matrice source, la position de l'origine du zoom en unité cellulaire, la hauteur du zoom, la largeur du zoom et les données issues du fichier source (hauteur, largeur et format) et qui est appelée par la fonction *createZoomFiles*, renvoie la matrice du zoom extraite. La matrice renvoyée est enregistrée par la fonction *saveRainRasterFile* de la classe *C_ConvertRainFiles* dans le répertoire donné en entrée par le paramètre *pathRecording*.

III-3-2. Extraction de fichier image bitmap

Le traitement des fichiers image bitmap présente une légère différence avec celui des fichiers texte sur l'extraction de la matrice source. En effet, l'extraction de la matrice source est réalisée avec la fonction *readRainRaster* de la classe *ConvertRainFiles*. Cette fonction reçoit en entrée le nom complet de l'image puis appelle la fonction *imgRasterLoader* de *C_ReadRaster* qui à son tour va convertir l'image en code ASCII puis retourne la matrice numérique obtenue. Cette matrice numérique renvoyée par *imgRasterLoader* présente des valeurs de couleurs arbitraires, ce qui nous amène à procéder à un redimensionnement de la matrice avec la fonction *rescaleColorValues*. Ainsi, la création et la sauvegarde seront similaires à celles effectuées avec les fichiers texte.

III-4. Mise en œuvre de la croissance de la végétation

L'automatisation du calcul de la biomasse de la végétation sur le simulateur SimMasto est réalisée suivant les trois points présentés ci-dessous.

III-4-1. Calcul du nombre de jours par pas de temps (*oneTick_Uday*)

Dans notre modèle, nous avons des pas de temps paramétrables en secondes, minutes, heures, jours, mois ou années. Ainsi, le calcul du nombre de jours par pas de temps est effectué à l'aide de l'équation suivante :

$$oneDay_Utick = (Tick \times Unit) / mappingDayToUnit$$

Avec : Tick: pas de temps de la simulation

- Unit : Unité de simulation

mappingDayToUnit: correspond à la conversion d'un jour suivant l'unité du pas de temps.

Ex: Unit = 1mn \rightarrow *mappingDayToUnit* = 60 * 24 = 1440

III-4-2. Calcul de la capacité de charge en biomasse

Nous faisons varier la capacité de charge K d'une cellule suivant la quantité de pluie déversée sur la zone d'enracinement de la végétation tel que :

$$K = (K_0 + Rvm \times (Rc - 1)) \quad (10)$$

Avec :

- ✓ K : la capacité de charge de la végétation calculée
- ✓ K_0 : la capacité de charge initiale de la végétation
- ✓ Rvm : le convertisseur des classes de la pluviométrie (1 à 8) à une valeur de capacité de charge par multiplication
- ✓ Rc : classe de la pluviométrie (1 à 8) suivant la quantité d'eau présente dans la cellule

III-4-3. Calcul de la biomasse de la végétation (herbe, arbuste ou culture)

La biomasse à calculer B est donnée par la formule suivante :

$$B = B_0 + \left(\frac{r \times Rc}{oneDay_Utick} \times \left(1 - \frac{B_0}{K} \right) \right) \quad (11)$$

Avec :

- ✓ B_0 : la biomasse initiale de la végétation
- ✓ r : le taux de croissance de végétation
- ✓ Rc : classe de la pluviométrie notée sur la zone d'enracinement

Noter que la capacité de charge et la biomasse sont représentées respectivement dans la fonction de croissance logistique par K et Y .

III-5. Processus d'exécution du protocole Gerbille

L'exécution du protocole gerbille est réalisée suivant différentes phases successives telles que décrites sur la Figure 16.

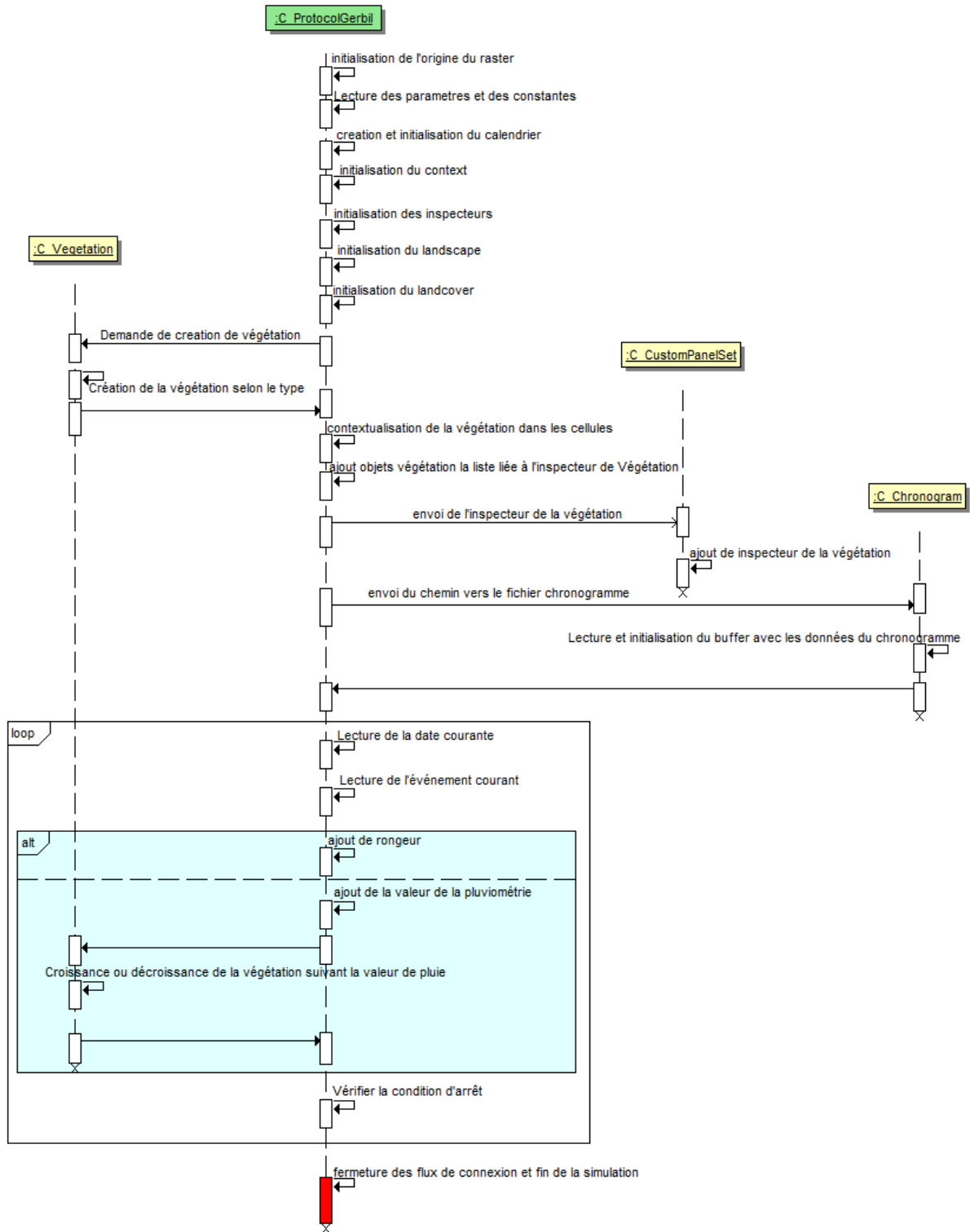


Figure 16 : Résumé du processus d'exécution du protocole gerbille

CHAPITRE 3 : RESULTATS

La visualisation des résultats passe par l'exécution successive de plusieurs fonctionnalités par exemple, la mise en place du raster, l'initialisation de la couverture du sol, etc. La simulation suit une chronologie partant de la lecture des données à la visualisation des résultats en passant par l'exécution des procédés de traitement. Il nous a été demandé sur le cahier des charges (i) de simuler l'évolution de la pluviométrie suivant les données mensuelles fournies par les images satellites, (ii) de mettre en place une fonctionnalité permettant d'extraire à la demande une zone de questionnement (zoom) et (iii) de mettre en relation la pluviométrie et la végétation (croissance et décroissance de la végétation suivant la quantité de pluie présente dans la cellule).

Ainsi, nous comptons présenter les résultats de notre travail sur chacun des trois points du cahier des charges que sont :

- ✓ Le résultat de la simulation des pluies sur le GUI
- ✓ Le résultat de l'extraction de zooms
- ✓ L'évolution de la biomasse sur le GUI et sur le graphe de biomasse

I. Simulation des pluies sur le GUI

La pluviométrie varie de façon mensuelle. A chaque pas de temps, le simulateur vérifie s'il y a un nouveau événement de pluie et met à jour l'environnement. Les images bitmaps originales obtenues après traitement sur Corel Paint contiennent des délimitations de frontière qui sont représentées sur la matrice par la valeur zéro et il est impossible pour le simulateur de déterminer la quantité de pluie exacte dans ces zones. C'est pourquoi nous avons fait appel à une manipulation nous permettant de remplacer les valeurs des cellules touchées par la frontière par celles des autres cellules qui leur sont les plus proches (Figure 17).

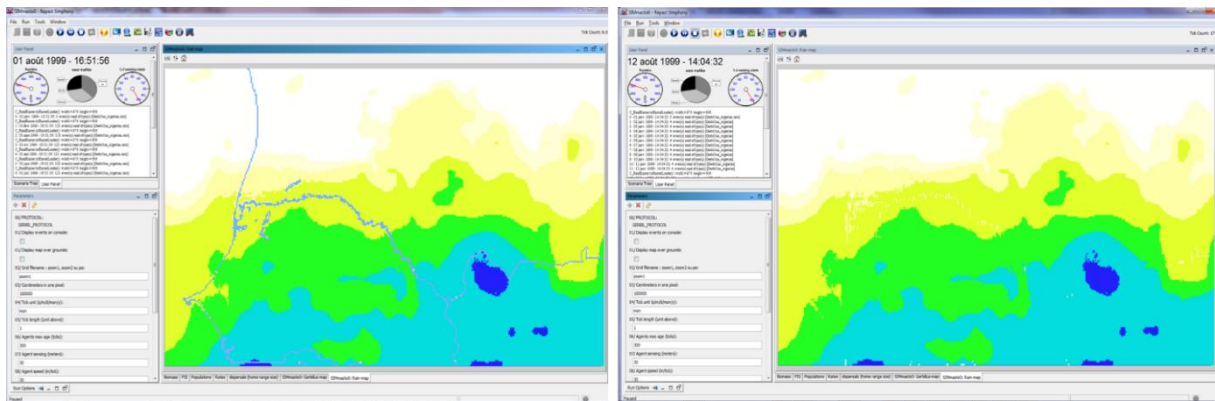


Figure 17 : Simulation de la pluviométrie du mois d'août 1999 (a) avec les frontières et (b) sans les frontières

Les résultats de simulation obtenus après manipulation sur la période de janvier à décembre 1999 sont présentés sur la Figure 18. Ces résultats permettent de mettre à jour la précipitation de la zone d'étude.

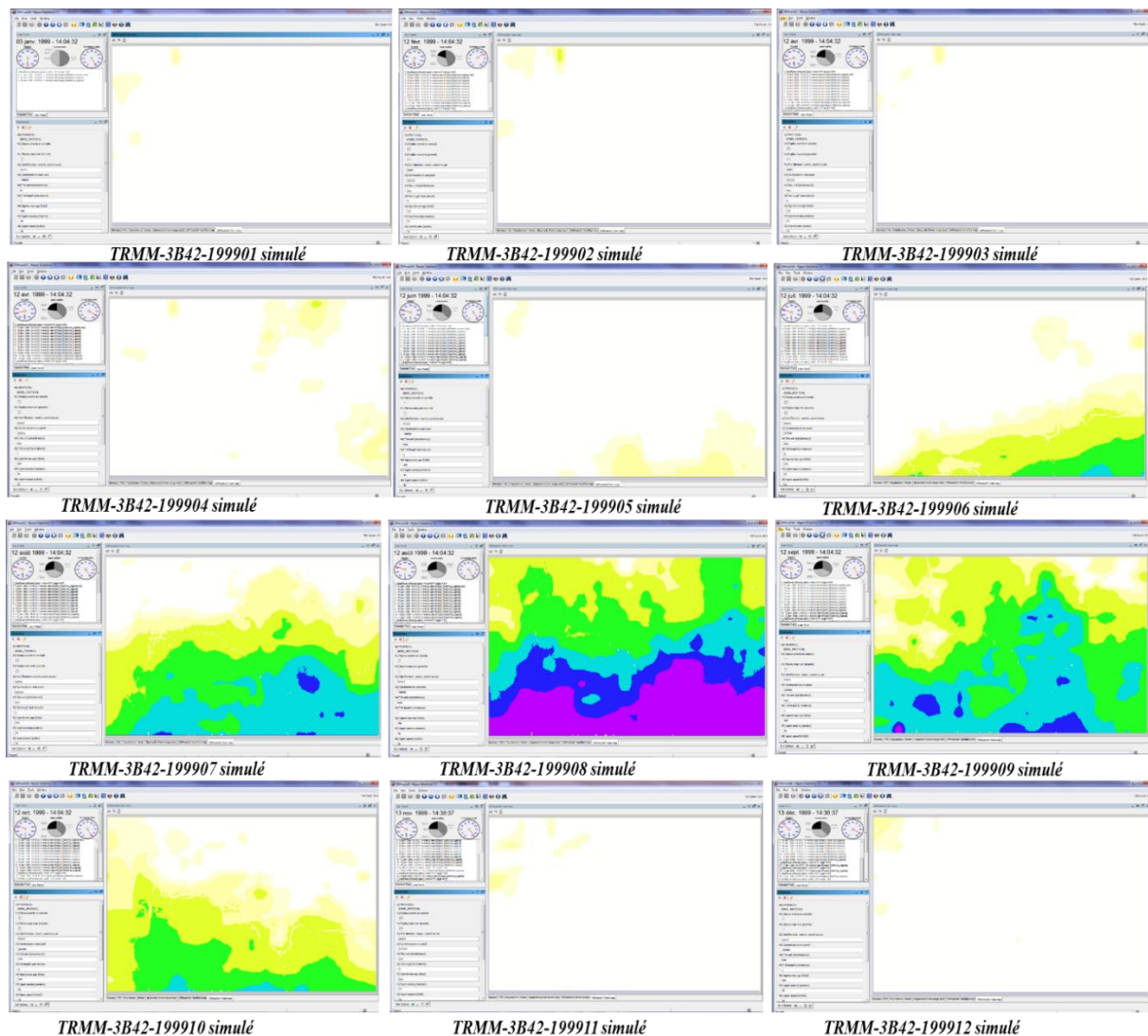


Figure 18 : Simulation de la pluviométrie de l'année 1999

II. Extraction de zooms

L'extraction de zoom est réalisée sur les fichiers de la couverture du sol (*landcover*) et de la pluviométrie. L'initialisation du raster avec les données de la couverture du sol sur la grande emprise donne le résultat de la Figure 19.

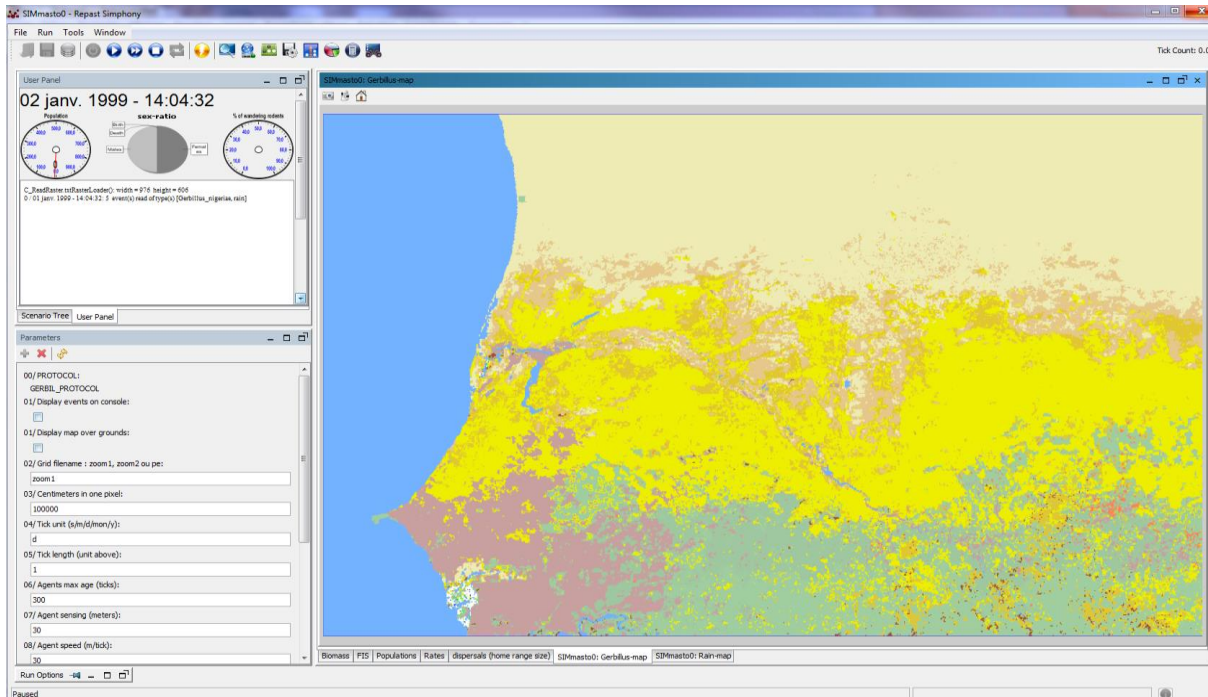


Figure 19 : Simulation de la couverture du sol de la grande emprise

L'utilisation de notre algorithme d'extraction sur la grande emprise nous permet alors d'obtenir le raster de notre zone d'étude (zoom petite emprise) présenté sur la Figure 20.

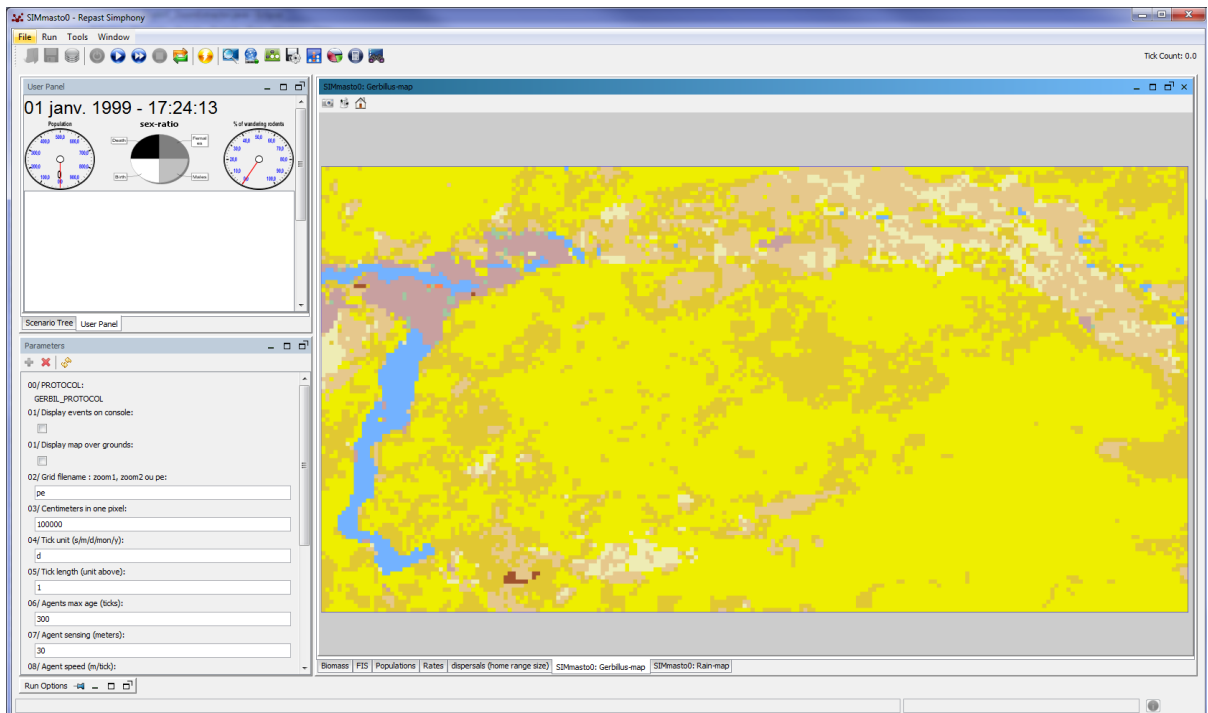


Figure 20 : Simulation du zoom petite emprise obtenu après extraction

Ce même algorithme, appliqué sur la petite emprise permet d'extraire des petites zones de questionnement (voir Figure 21 et Figure 22).

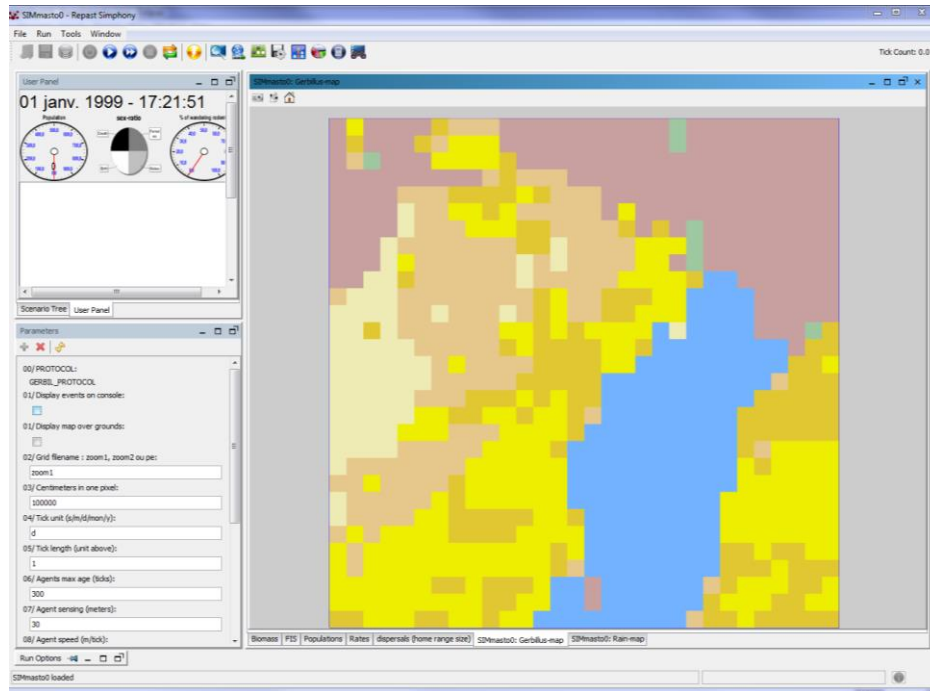


Figure 21 : Zone de questionnement (nord du lac de Guiers) extraite à partir de la petite emprise

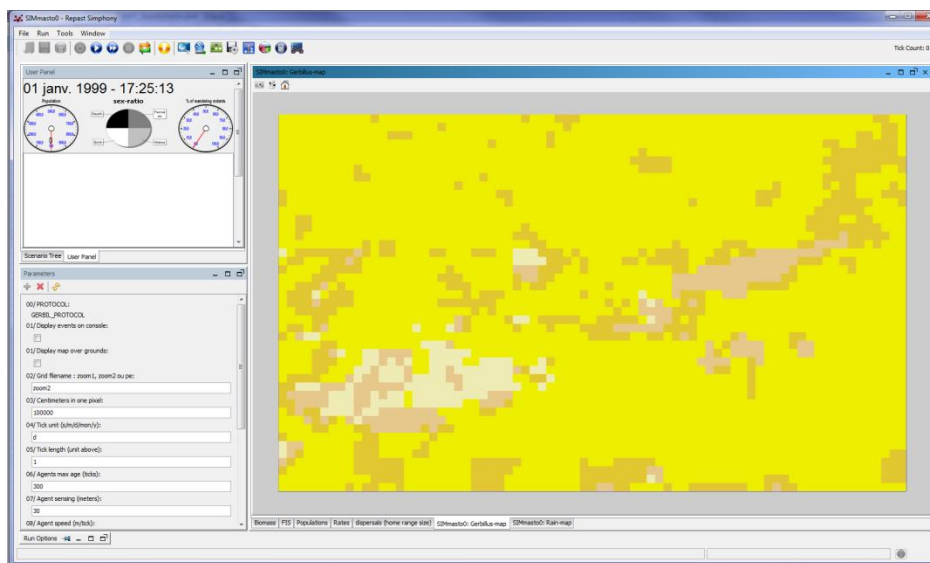


Figure 22 : zone de questionnement (zone du Ferlo, milieu très herbacé) extraite de la petite emprise

III. Evolution de la biomasse sur le GUI et sur le graphe de biomasse

L'initialisation de la couverture du sol s'accompagne de celle de la végétation avec une biomasse initiale attribuée à chaque objet de végétation suivant son type. Ainsi, la visualisation initiale de la végétation nous donne la Figure 23.

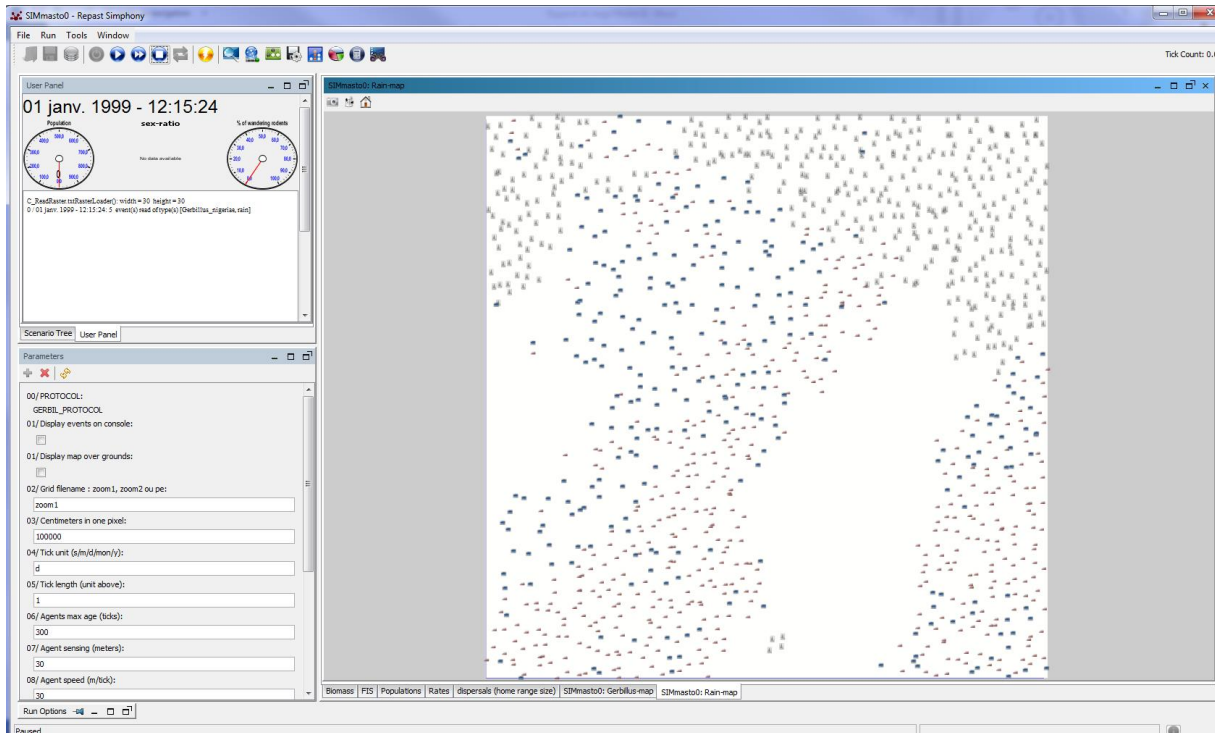


Figure 23 : Biomasse initiale de la végétation sur le nord du lac de Guiers

Pour chaque pas de temps, la quantité d'eau présente dans la cellule est mise à jour, entraînant ainsi la variation de la biomasse. Plus la quantité de pluie sur la cellule est importante, plus la biomasse croît et inversement. La simulation durant la saison sèche et la saison des pluies nous donnent les résultats suivants :

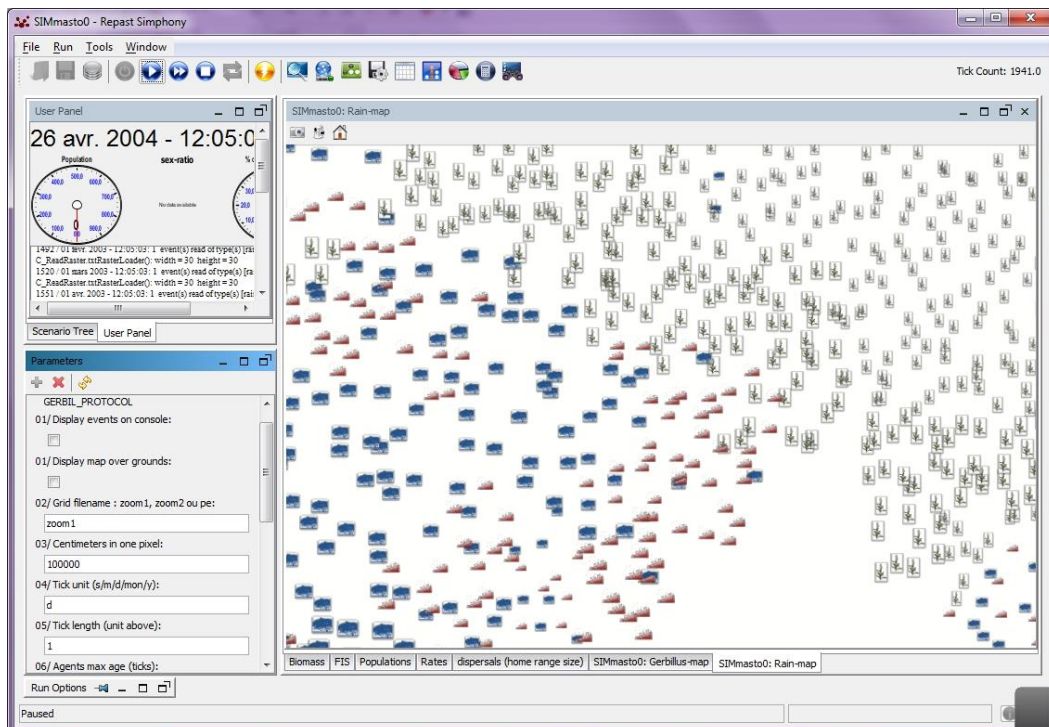


Figure 24 : Biomasse de la végétation aperçue à la fin de la saison sèche de l'année 1999 sur le nord du lac de Guiers

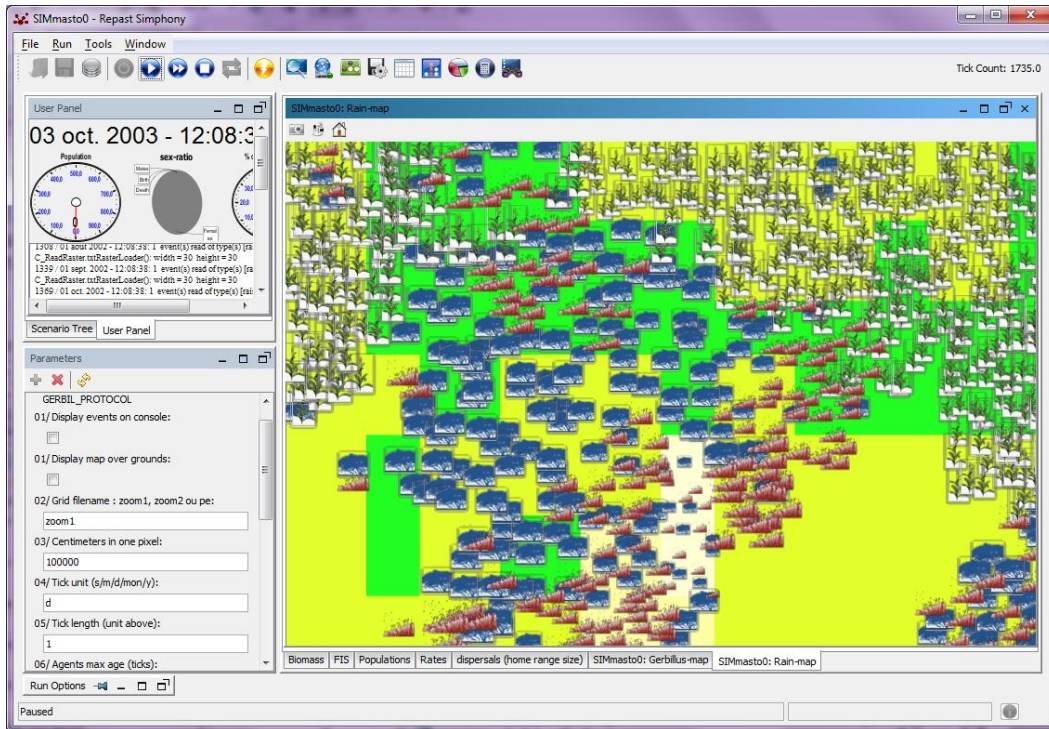


Figure 25 : Biomasse de la végétation aperçue à la fin de la saison des pluies de l'année 1999 sur le nord du lac de Guiers

La variation de cette quantité de biomasse est observée sur le graphe d'évolution de la biomasse présenté sur la Figure 26.

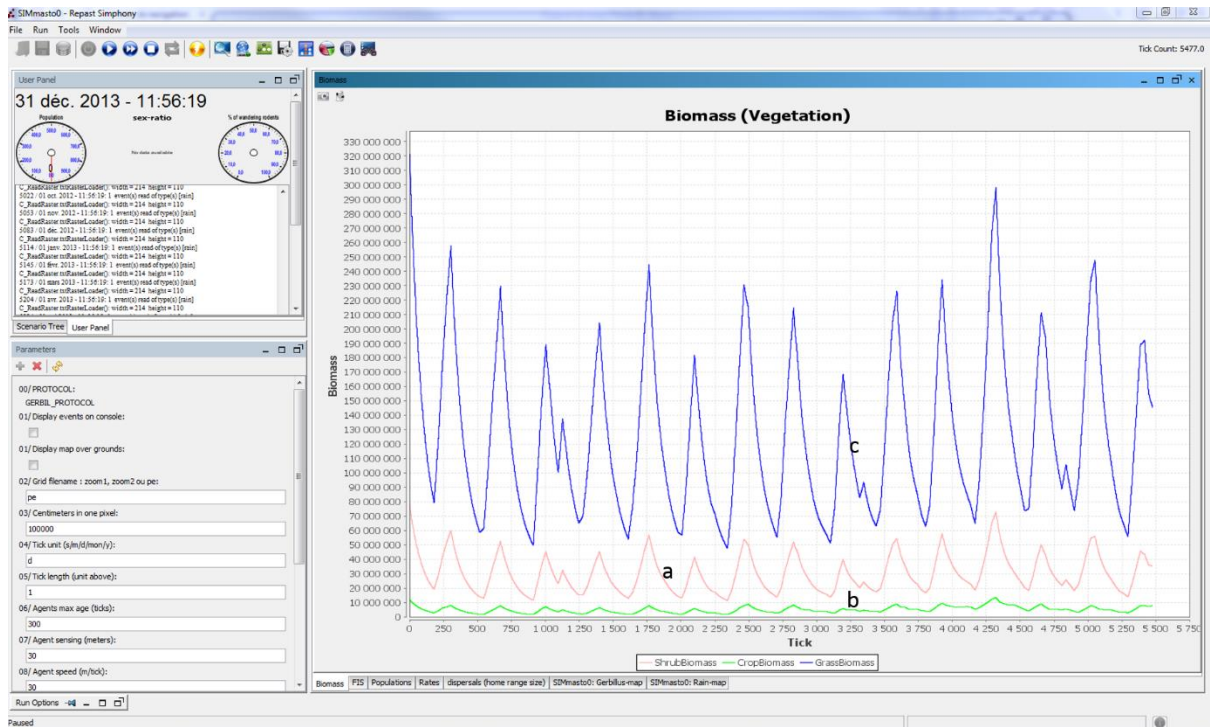


Figure 26 : Graphe d'évolution de la biomasse par pas de temps (a) arbuste, (b) culture, (c) herbe

CHAPITRE 4 : DISCUSSION

L'objectif de ce rapport est de présenter le travail réalisé sur la mise en place d'un environnement de simulation propice à l'évolution de la gerbille en se basant sur des données fournies par les thématiciens ayant travaillé sur le domaine. L'approche adoptée et les algorithmes réalisés nous ont permis d'atteindre les objectifs spécifiés sur le cahier des charges, c'est-à-dire (i) de simuler l'évolution de la pluviométrie en fonction des données mensuelles fournies par les images satellites, (ii) d'implémenter une fonctionnalité permettant d'extraire à la demande une zone de questionnement et (iii) de mettre en relation la pluviométrie et la végétation.

Plusieurs traitements alternatifs auraient pu être appliqués pour atteindre cet objectif. En effet, l'utilisation de l'équation du modèle de Clarke ou de la projection orthographique permettrait de convertir dans les deux sens les coordonnées géodésiques en coordonnées discrètes, mais pour vérifier la validité de la conversion, nous avons décidé d'utiliser le modèle de Clarke pour la conversion de coordonnées géodésiques en discrètes et la projection orthographique pour faire la conversion inverse. Cette approche utilisant deux algorithmes distincts nous garantit ainsi la validité des conversions.

Les objets de végétation sont placés dans les cellules suivant les données de couverture du sol issues des images satellites retenues pour le projet. Ainsi, deux types de végétation sont positionnés au maximum sur chaque cellule couvrant 1km^2 sur la zone d'étude. Ce pattern issu de l'extraction de ces images nous paraît un peu incorrect vis-à-vis de ce qui a été observé durant la mission du 8 au 16 avril 2016 sur le Ferlo. En effet, bien que les zones visitées soient effectivement herbacées, elles sont toutes composées d'un biotope diversifié avec la présence d'arbustes, d'arbres, d'herbes, de cultures pluviales et comportent une part importante de sol nu et ceci nous est apparu très homogène sur l'ensemble de la zone parcourue (Figure 27). Il semble ainsi que la réduction de catégories effectuée par Synmap soit trop importante par rapport à la composition du milieu.



Figure 27 : différents aspects du sol et des paysages de la savane sahélienne dans la zone explorée :

(a) nature du sol dans un champ de culture pluviale. (b) Le champ est clôturé par une haie d'épineux dont l'intérieur constitue un habitat pour *Arvicanthis niloticus*, le reste pouvant être colonisé par *Gerbillus nigeriae* ou *Taterillus* sp. Le champ peut être labouré, en jachère... En avril, on y trouve des traces de mil ou de culture d'arachide (K. Ba, obs.,pers.) (c) pose d'une ligne de pièges à l'intérieur de la grande muraille verte ; les points noirs sont des bouses sèches. (d) savane typique sur la route entre Lagbar et Richard-Toll. Ces zones identifiées dans les données de couverture du sol comme « herbacées » (Figure 2) rentreraient plutôt dans la catégorie « arbustes et sol nu » (c et d) ou culture et arbustes (a et b). (Le Fur, 2016b)

CONCLUSION

La tâche principale à accomplir dans ce stage était de mettre en place un environnement de simulation propice au développement de la gerbille. Sur la base d'un cahier des charges, nous avons pu réaliser des fonctionnalités permettant d'extraire à la demande une zone de questionnement sur le raster de la grande emprise, de convertir dans les deux sens des coordonnées géodésiques en coordonnées discrètes et de numériser des images de capture satellitaire, mais aussi de mettre en relation la pluviométrie et la végétation qui est une source de nourriture importante pour la gerbille.

Ce stage a été très bénéfique pour nous. Il nous a permis de mettre en pratique nos connaissances théoriques et en même temps de les approfondir qualitativement en découvrant par nos recherches et l'encadrement un nouveau domaine du système complexe qu'est la modélisation à base d'agents. On a pu participer aussi à une mission dont l'objectif pour nous était de nous familiariser avec l'espèce étudiée (la gerbille), d'être en contact avec l'environnement dans lequel elle vit et de constater effectivement l'étendue de sa présence dans le milieu 17 ans après sa découverte au Sénégal. Cette mission nous a permis de voir qu'il est nécessaire d'apporter une amélioration sur la méthode de répartition de la végétation actuellement utilisée sur SimMasto en utilisant peut-être une autre source de données de la couverture végétale comme celle de FAO-2005 ou GLC-20096 (Le Fur, 2016b).

Le taux de croissance paramétrable spécifique à chaque objet de végétation dépend de l'espèce et de l'environnement. Comme le temps ne nous a pas permis d'approfondir nos recherches, il sera nécessaire dans la suite de consulter les chercheurs dans ce domaine afin d'avoir à notre disposition des valeurs plus spécifiques pour chaque type de végétation.

La mise en place de l'environnement n'est utile que si elle est suivie par la mise en œuvre de l'évolution des populations de gerbilles dans le milieu. Ainsi, il paraît nécessaire voire même crucial de poursuivre le développement du modèle (i) en positionnant les gerbilles capturées par piégeages ou trouvées sur les pelotes de chouettes et (ii) en mettant en place les aspects liés à la survie de l'espèce dans le milieu (reproduction, alimentation, exploration de l'espace non occupé, etc.).

⁶ 20141217-LeFur_BilanDonnéesLandCoverGerbillus.jlf.pdf

BIBLIOGRAPHIE

- Bille, J.. (1974). Recherches écologiques sur une savane sahélienne du Ferlo septentrional, Sénégal : 1972, année sèche au sahel (*Station d'Ecologie ORSTOM, B.F. '20 Richard Toll, Sénégal.*). 16, 5–20
- Hima, K. (2010). Biologie évolutive de *Gerbillus nigeriae* (Rodentia, Gerbillinae), principal ravageur des cultures céréalières au Niger : aspects chromosomique, morphologique et populationnel. Thèse Univ.A.M. Niamey, 208p.
- Huffman, G.J., Bolvin, D.T., Nelkin, E.J., Wolff, D.B., Adler, R.F., Gu, G., Hong, Y., Bowman, K.P., and Stocker, E.F. (2007). *The TRMM multisatellite precipitation analysis (TMPA): Quasi-global, multiyear, combined-sensor precipitation estimates at fine scales. J. Hydrometeorol. 8, 38–55.*
- IGN/SGN (2013). Transformation entre systèmes géodésiques.
- Le Fur, J. (2015). Modification de l'approche Petite Emprise (PE). rapp.interne projet FRB Cerise, nov. 2015, 3p.
- Le Fur, J. (2016a). Une architecture de simulation pour l'articulation incrémentale de savoirs pluridisciplinaires. Application à la connaissance des rongeurs et de leurs populations. coll. JFSMA 2016, 10p. soumis.
- Le Fur, J. (2016b). Modèle Gerbille – Compte rendu d'une mission dans le Ferlo – Sénégal en avril 2016. rapp.interne projet FRB Cerise, avr. 2016, 8p.
- Mboup, P.A. (2015). Des Connaissances à la Simulation Multi-agents: Modélisation orientée événements de la Colonisation du Rat noir au Sénégal par les Transports Humains sur un Siècle. p.
- Realini, A. (2011). Conception et mise en œuvre d'une interface de visualisation des résultats de simulation d'une plate-forme de simulation individus-centrée. Rapport DUT, IUT informatique Montpellier, 49p.
- Verhulst, P.F. (1845). Recherches mathématiques sur la loi d'accroissement de la population. *Nouv. Mém. Académie R. Sci. B.-Lett. Brux. 18, 14–54.*

WEBGRAPHIE

- <http://repast.sourceforge.net/>.
- https://fr.wikipedia.org/wiki/Mod%C3%A8le_de_Verhulst.
- https://fr.wikipedia.org/wiki/Projection_orthographique.
- http://vminfotron-dev.mpl.ird.fr:8080/masto2_2/.

ANNEXE : codes sources

I. Numérisation images pluviométriques avec main fonctionnel

```
/* This source code is licensed under a BSD licence as detailed in file SIMmasto_0.license.txt */
package simmasto0.util;

import java.io.File;
import java.util.ArrayList;
import presentation.dataOutput.C_FileWriter;
import data.I_ConstantGerbil;
import data.I_ConstantString;

/** Read the rain bitmaps, rescale colors and save the grid in ASCII
 * @author M.Sall 10.2015, rev. MS&JLF 04.2016 */
public class C_ConvertRainFiles implements I_ConstantString, I_ConstantGerbil {
    //
    // METHODS
    //
    /** Read the rain raster bitmap file and build the corresponding matrix */
    public int[][] readRainRaster(String fileNameInRain) {
        int[][] rainMatrix = C_ReadRaster.imgRasterLoader(RASTER_PATH + fileNameInRain);
        rainMatrix = rescaleColorValues(rainMatrix);
        System.out.println(fileNameInRain + " bitmap read ");
        return rainMatrix;
    }
    /** Save the grid in the file following the name passed in args */
    public void saveRainRasterFile(int[][] matriceLue, String fileNameInRain, String path) {
        int nbColumns;
        int nbLines;
        C_FileWriter rasterFile_Utxt;
        // Create or recreate empty gerbilSavannah file
        nbColumns = matriceLue.length;
        nbLines = matriceLue[0].length;
        rasterFile_Utxt = new C_FileWriter(RASTER_PATH + path + "/" + fileNameInRain + ".txt", false);
        // Compute max and min values of the matrix
        int maxValue = 0;
        int minValue = Integer.MAX_VALUE;
        for (int i = 0; i < nbColumns; i++) {
            for (int j = 0; j < nbLines; j++) {
                if (maxValue < matriceLue[i][j]) maxValue = matriceLue[i][j];
            }
        }
    }
}
```

```

        if (minValue > matriceLue[i][j]) minValue = matriceLue[i][j];
    }
}
// Raster file add values
System.err.println("C_ConvertRainFiles.saveRainRasterFile(): Writing file");
rasterFile_Utxt.writeln("DSAA");
rasterFile_Utxt.writeln(nbColumns + "");
rasterFile_Utxt.writeln(nbLines + "");
rasterFile_Utxt.write(minValue + " " + maxValue);
for (int i = nbLines - 1; i >= 0; i--) {
    rasterFile_Utxt.writeln("");
    rasterFile_Utxt.write(matriceLue[0][i] + "");
    for (int j = 1; j < nbColumns; j++)
        rasterFile_Utxt.write(" " + matriceLue[j][i]);
}
rasterFile_Utxt.closeFile();
System.err.println("C_ConvertRainFiles.saveRainRasterFile(): " + fileNameInRain + ".txt conversion complete");
}
/** Compute the name of the resulting file from the original name with date and suffix<br>
 * run this one time for all */
public String computeFileName(String fileNameInRain) {
    // ex: fileNameInRain = TRMM-3B42-19990101-19990131 to 19990101
    return fileNameInRain.substring(10, 16) + "-ME-Rain";
}
/** Elaborate a list with the whole series of file names (one for each month) contained in the rain folder */
public ArrayList<String> buildFileNameList(String directoryPath) {
    ArrayList<String> fileNameList = new ArrayList<String>();
    // Find all of file name in the directory
    File directory = new File(directoryPath);
    if (!directory.exists()) System.err.println("C_ConvertRainFiles.buildFileNameList());
    Folder " + directoryPath + " does not exist");
    else {
        File[] subfiles = directory.listFiles();
        int length = subfiles.length;
        for (int i = 0; i < length; i++)
            fileNameList.add(subfiles[i].getName());
    }
    return fileNameList;
}
/** Replace all occurrences of old value with new value in the grid */
public int[][] replaceValueInMatrix(int oldValue, int newValue, int[][] grid) {
    if (checkOccurrenceInMatrix(newValue, grid)) {

```

```

        System.err.println(newValue + " already exist, impossible to change with " + oldValue);
        return null;
    }
    int oldValueSaved = 0, frontierValue = 0;
    for (int i = 0; i < grid.length; i++) {
        for (int j = 0; j < grid[0].length; j++) {
            if (grid[i][j] == oldValue) {
                grid[i][j] = newValue;
                oldValueSaved = newValue;
            }
            // When frontierValue is encountered, replace with preceding cell value. Used only once
            else {
                if (grid[i][j] == frontierValue) grid[i][j] = oldValueSaved;
                else oldValueSaved = grid[i][j];
            }
        }
    }
    return grid;
}
/** @return true if the requested value exist in matrix */
public boolean checkOccurrenceInMatrix(int value, int[][] matrix) {
    for (int i = 0; i < matrix.length; i++)
        for (int j = 0; j < matrix[0].length; j++)
            if (matrix[i][j] == value) return true;
    return false;
}
/** Rain bitmap values are arbitrary color values, order them to the corresponding levels of rain intensity */
public int[][] rescaleColorValues(int[][] matriceLue) {
    int[][] newMatrix = matriceLue;
    // Reorder bitmap values to conform to rain intensity scale
    newMatrix = replaceValueInMatrix(8, 17, newMatrix);
    newMatrix = replaceValueInMatrix(3, 16, newMatrix);
    newMatrix = replaceValueInMatrix(4, 15, newMatrix);
    newMatrix = replaceValueInMatrix(1, 14, newMatrix);
    newMatrix = replaceValueInMatrix(2, 13, newMatrix);
    newMatrix = replaceValueInMatrix(5, 12, newMatrix);
    newMatrix = replaceValueInMatrix(6, 11, newMatrix);
    newMatrix = replaceValueInMatrix(7, 10, newMatrix);
    // Rescale values from 1 to 8
    newMatrix = replaceValueInMatrix(10, 1, newMatrix);
    newMatrix = replaceValueInMatrix(11, 2, newMatrix);
    newMatrix = replaceValueInMatrix(12, 3, newMatrix);
}

```

```

        newMatrix = replaceValueInMatrix(13, 4, newMatrix);
        newMatrix = replaceValueInMatrix(14, 5, newMatrix);
        newMatrix = replaceValueInMatrix(15, 6, newMatrix);
        newMatrix = replaceValueInMatrix(16, 7, newMatrix);
        newMatrix = replaceValueInMatrix(17, 8, newMatrix);
        return newMatrix;
    }
    //
    // MAIN
    //
    public static void main(String[] args) {
        int[][] matriceMELue;
        String rainBitmapFolder = "rain-bitmaps-ME/";
        C_ConvertRainFiles converter = new C_ConvertRainFiles();
        ArrayList<String> fileNameList = converter.buildFileNameList(RASTER_PATH + rainBitmapFolder);
        // build all ME matrix
        int k = 0;
        for (String fileNameInRain : fileNameList) {
            fileNameInRain = fileNameList.get(k);
            matriceMELue = converter.readRainRaster(rainBitmapFolder + fileNameInRain);
            converter.saveRainRasterFile(matriceMELue, converter.computeFileName(fileNameInRain), "rain-ASCII-ME");
            k++;
        }
        System.out.println("conversion complete!!!");
    }
}

```

II. Extraction de zoom avec main fonctionnel

```

package simmasto0.util;

import java.awt.Point;
import java.util.ArrayList;

import com.vividsolutions.jts.geom.Coordinate;

import data.I_ConstantGerbil;
import data.I_ConstantString;
/** Extract the input raster files (rain series or landcover) of a zoom or TPE ('très petite emprise') landscape at the desired size from the
reference
* grid of ME ('moyenne emprise') and save the resulting files / Used in Gerbil protocol only

```

```

* @author M.Sall & J.Le Fur 12.2015, rev. JLF&MS 03.2016 */
public class C_ZoomExtractor implements I_ConstantString, I_ConstantGerbil {
//
// METHODS
//
/** Extract the zoom matrix from the reference grid (ME) */
public void createZoomFiles(Coordinate zoomOrigin_Udegree, int zoomWidth_Ucell, int
zoomHeight_Ucell, String pathRecording, String pathFolder) {
    C_ConvertRainFiles rainFilesConverter = new C_ConvertRainFiles();
    int[][] zoomMatrix = new int[zoomWidth_Ucell][zoomHeight_Ucell];
    int[][] referenceMatrix = null;
    String zoomFileName = "";
    int fileIterator = 0;
    Point zoomOrigin_Ucell = computeZoomOrigin_Ucell(zoomOrigin_Udegree);
    ArrayList<String> fileNameList = rainFilesConverter.buildFileNameList(RASTER_PATH + pathFolder);
    if (fileNameList.isEmpty()) {
        System.err.println("C_ZoomExtractor.extractZoom(): folder " + pathFolder + " is empty.");
        return;
    }
    String format;
    // Extract the zoom matrix from the reference matrix
    for (String fileName : fileNameList) {
        fileName = fileNameList.get(fileIterator);
        format = getExtension(fileName);
        switch (format) {
            case "bmp" : {
                referenceMatrix = rainFilesConverter.readRainRaster(RASTER_PATH + pathFolder + "/" + fileName);
                zoomMatrix = extractZoomFromGrid(referenceMatrix, zoomOrigin_Ucell, zoomWidth_Ucell, zoomHeight_Ucell, "bmp");
                if(fileName.contains("landcover")) zoomFileName = "landcover";
                else zoomFileName = (fileName.split("-")[0]).substring(0, 6) + "-Zoom-Rain";
                rainFilesConverter.saveRainRasterFile(zoomMatrix, zoomFileName, pathRecording);
            }
            break;
            case "txt" : {
                System.out.println(RASTER_PATH + pathFolder + "/" + fileName);
                referenceMatrix = C_ReadRaster.txtRasterLoader(RASTER_PATH + pathFolder + "/" + fileName);
                zoomMatrix = extractZoomFromGrid(referenceMatrix, zoomOrigin_Ucell, zoomWidth_Ucell, zoomHeight_Ucell, "txt");
                if (fileName.contains("landcover")) zoomFileName = "landcover";
                else zoomFileName = (fileName.split("-")[0]).substring(0, 6) + "-Zoom-Rain";
                rainFilesConverter.saveRainRasterFile(zoomMatrix, zoomFileName, pathRecording);
            }
            break;
        }
    }
}

```

```

    }
    fileIterator++;
}
}
/** Extract zoom matrix from ME reference matrix; all units are cell units<br>
 * @version MS&JLF 03.2016 */
public int[][] extractZoomFromGrid(int[][] referenceMatrix, Point zoomOrigin_Ucell, int
zoomWidth_Ucell, int zoomHeight_Ucell, String format) {
    int[][] zoomMatrix = new int[zoomWidth_Ucell][zoomHeight_Ucell];
    try {
        for (int i = zoomOrigin_Ucell.x; i < zoomOrigin_Ucell.x + zoomWidth_Ucell; i++) {
            for (int j = zoomOrigin_Ucell.y; j < zoomOrigin_Ucell.y + zoomHeight_Ucell; j++) {
                switch (format) {
                    case "bmp" :
                        zoomMatrix[i - zoomOrigin_Ucell.x][j - zoomOrigin_Ucell.y] = referenceMatrix[i][j];
                        break;
                    case "txt" :
                        zoomMatrix[i - zoomOrigin_Ucell.x][j - zoomOrigin_Ucell.y] = referenceMatrix[i][j];
                        break;
                }
            }
        }
    } catch (Exception e) {
        System.err.println("C_ZoomExtractor.extractZoomFromGrid() : zoom extraction failed.");
    }
    return zoomMatrix;
}
/** Convert zoom origin in degrees into continuous space then into cell coordinates of the ME
reference grid */
public Point computeZoomOrigin_Ucell(Coordinate zoomOrigin_Udegree) {
    Point zoomOrigin_Ucell;
    C_ConvertGeographicCoordinates converter = new C_ConvertGeographicCoordinates(new
Coordinate(I_ConstantGerbil.gerbilMELongitudeWest_Udegree, I_ConstantGerbil.gerbilMELatitudeSouth_Udegree));
    Coordinate zoomOrigin_Ucs = converter.convertCoordinate_Ucs(zoomOrigin_Udegree.x, zoomOrigin_Udegree.y);
    zoomOrigin_Ucell = new Point((int) zoomOrigin_Ucs.x, (int) zoomOrigin_Ucs.y);
    return zoomOrigin_Ucell;
}
//
// GETTER
//
/** Retrieve the extension of the input file */
public String getExtension(String fileName) {

```

```
        return fileName.substring(fileName.lastIndexOf(".") + 1);
    }
    //
    // MAIN
    //
    public static void main(String[] args) {
        C_ZoomExtractor zoomExtractorTest = new C_ZoomExtractor();
        // Folder to save extracted files
        String pathRecording;
        String referencePathFolder = "rain-ASCII-ME"; // Average grip

        // CASE Zoom 001 08.12.2015
        pathRecording = "Zoom_001_12.2015";
        Coordinate zoom1Origin_Udegree = new Coordinate(-16, 16.2); // here Longitude_Udegree, Latitude_Udegree
        width_heightRaster_Ukilometer.set(0, 30);
        width_heightRaster_Ukilometer.set(1, 30);
        zoomExtractorTest.createZoomFiles(zoom1Origin_Udegree, width_heightRaster_Ukilometer.get(0), width_heightRaster_Ukilometer.get(1),
        pathRecording, referencePathFolder);

        //CASE Zoom 002 08.12.2015
        pathRecording = "Zoom_002_12.2015";
        Coordinate zoom2Origin_Udegree = new Coordinate(-15.45, 15.76); // here Longitude_Udegree, Latitude_Udegree
        width_heightRaster_Ukilometer.set(0, 75);
        width_heightRaster_Ukilometer.set(1, 45);
        zoomExtractorTest.createZoomFiles(zoom2Origin_Udegree, width_heightRaster_Ukilometer.get(0), width_heightRaster_Ukilometer.get(1),
        pathRecording, referencePathFolder);

        //CASE PE
        pathRecording = "Zoom_PE_12.2015";
        zoom2Origin_Udegree = new Coordinate(-16., 15.758572); // here Longitude_Udegree, Latitude_Udegree
        width_heightRaster_Ukilometer.set(0, 214);
        width_heightRaster_Ukilometer.set(1, 110);
        zoomExtractorTest.createZoomFiles(zoom2Origin_Udegree, width_heightRaster_Ukilometer.get(0), width_heightRaster_Ukilometer.get(1),
        pathRecording, referencePathFolder);
    }
}
```


III. Croissance de la végétation

```

/** Compute the change of biomass given the level of rain in the cell.<br>
 * The model follow a logistic law: biomass <- biomass * (1 + growth_rate * (1 - biomass / carrying_capacity)) * @version J.Le Fur & M.Sall,
 02.2016*/
@Override
public void growOlder_Utick() {
    double oneDay_Utick = C_ConvertTimeAndSpace.convertTimeDurationToTick(1, "d");
    double rainValueMultiplier = 1.E8;// TODO number in source: Convert rain class value (1 to 8) to carrying capacity
    double sensitivity = 1.;// TODO number in source sensitivity to rain
    // Compute carrying capacity given the class of rain intensity
    double carryingCapacityWithRain_Ugram = VegetationCarryingCapacity_Ugram +
    (rainValueMultiplier * (((C_SoilCellSavanna) this.currentSoilCell).getRainLevel() - 1));
    // Compute biomass change using the logistic law
    if (this.vegetationType != TREES && this.vegetationType != BARREN) {// No growth for these vegetation types
        this.biomass_Ugram += (this.growthRate_UgramPerDay * ((C_SoilCellSavanna) this.currentSoilCell).getRainLevel() * sensitivity) /
        oneDay_Utick * (1 - this.biomass_Ugram / carryingCapacityWithRain_Ugram);
    }
}

```

IV. Conversion de coordonnées géodésiques en discrètes

```

package simmasto0.util;
import java.util.HashMap;
import java.util.Map;
import com.vividsolutions.jts.geom.Coordinate;

import data.C_Parameters;
import data.I_ConstantGerbil;
import data.I_ConstantMusTransport;

/** Utility to convert from geographic coordinates to meters and compute distances in meter from the raster origin? <br>
 * Calculation depends on the raster used "author Moussa Sall, oct.2015 <br>
 * source: http://geodesie.ign.fr/contenu/fichiers/documentation/pedagogiques/TransformationsCoordonneesGeodesiques.pdf */
public class C_ConvertGeographicCoordinates {
    //
    // CONSTANTS
    //
    private static final double earthRadiusAtEquator_Umeter = 6378249.2;// half of great diameter

```

```

private static final double earthRadiusAtGreenwich_Umeter = 6356515.; // half of little diameter
//
// FIELDS
//
public Coordinate rasterOrigin_Udegree;
public Coordinate rasterOrigin_Uradian;
public Coordinate rasterEnd_Udegree;
private double ellipsoidEccentricitySquared_Umeter2;
//
// CONSTRUCTOR
//
public C_ConvertGeographicCoordinates(Coordinate rasterOrigin_Udegree) {
    this.rasterOrigin_Udegree = rasterOrigin_Udegree;
    this.rasterOrigin_Uradian = new Coordinate(convertDegree_Uradian(rasterOrigin_Udegree.x),
        convertDegree_Uradian(rasterOrigin_Udegree.y));
    this.ellipsoidEccentricitySquared_Umeter2 = (Math.pow(earthRadiusAtEquator_Umeter, 2) - Math.pow(earthRadiusAtGreenwich_Umeter, 2)) /
        Math.pow(earthRadiusAtEquator_Umeter, 2);
    // Compute radius of the normal curvature
    this.rasterOrigin_Umeter = convertCoordinate_Umeter(this.rasterOrigin_Uradian);
}
// TODO MS initialize the constructor with origin Coordinate and end Coordinate in degree
public C_ConvertGeographicCoordinates(Coordinate rasterOrigin_Udegree, Coordinate endRaster_Udegree)
{
    this(rasterOrigin_Udegree);
    rasterEnd_Udegree = endRaster_Udegree;
}
//
// METHODS
//
/** computeRadiusOfNormalCurvature */
public Coordinate convertCoordinate_Umeter(Coordinate coordinate_Uradian) {
    double tmp1 = Math.sqrt(1 - this.ellipsoidEccentricitySquared_Umeter2 * ((1 - Math.cos(2 * coordinate_Uradian.y)) / 2));
    double radius_Umeter = earthRadiusAtEquator_Umeter / tmp1; // raster origin radius
    return new Coordinate(radius_Umeter * Math.cos(coordinate_Uradian.y) * Math.cos(coordinate_Uradian.x), radius_Umeter *
        Math.cos(coordinate_Uradian.y) * Math.sin(coordinate_Uradian.x), radius_Umeter * (1 - ellipsoidEccentricitySquared_Umeter2) *
        Math.sin(coordinate_Uradian.y));
}
/** Compute distance to origin of a geographic position in meters */
public double distanceToRasterOrigin_Umeter(double longitude_Udegree, double latitude_Udegree) {
    Coordinate coordinate_Uradian = new Coordinate(convertDegree_Uradian(longitude_Udegree), convertDegree_Uradian(latitude_Udegree));
    Coordinate coordinate_Umeter = convertCoordinate_Umeter(coordinate_Uradian);
    // compute distance using Pythagorean theorem

```

```

        return Math.sqrt(Math.pow((rasterOrigin_Umeter.x - coordinate_Umeter.x), 2) + Math.pow((rasterOrigin_Umeter.y - coordinate_Umeter.y),
2)
+ Math.pow((rasterOrigin_Umeter.z - coordinate_Umeter.z), 2));
    }
    public double convertDegree_Uradian(double value_Udegree) {
        return value_Udegree * Math.PI / I_ConstantGerbil.straightAngleValue;
    }
    public double convertRadian_Udegree(double value_Uradian) {
        return value_Uradian * I_ConstantGerbil.straightAngleValue / Math.PI;
    }
    /** "return latitude and longitude geographic coordinates had with the cell conversion by the rule
of three */
    public Coordinate convertCell_Udegree(Coordinate oneCell) {
        Coordinate coordinateCell_Udegree = new Coordinate();
        double worldRadius = earthRadiusAtEquator_Umeter
/ Math.sqrt(1 - this.ellipsoidEccentricitySquared_Umeter2 * ((1 - Math.cos(2 * rasterOrigin_Uradian.y)) / 2));
        // C is an intermediate variable used to simplify the expression
        double cosinus_C = Math.sqrt(1 - ((Math.sqrt(oneCell.x * oneCell.x + oneCell.y * oneCell.y)) / worldRadius) * (Math.sqrt(oneCell.x *
oneCell.x + oneCell.y * oneCell.y)) / worldRadius);
        double sinus_C = (Math.sqrt(oneCell.x * oneCell.x + oneCell.y * oneCell.y)) / worldRadius;
        coordinateCell_Udegree.x = this.rasterOrigin_Uradian.x + Math.atan((oneCell.x * sinus_C)
/ ((Math.sqrt(oneCell.x * oneCell.x + oneCell.y * oneCell.y)) * Math.cos(this.rasterOrigin_Uradian.y) * cosinus_C) - ((oneCell.y *
Math.sin(this.rasterOrigin_Uradian.y))) * sinus_C));
        coordinateCell_Udegree.y = Math.asin((cosinus_C * Math.sin(this.rasterOrigin_Uradian.y))
+ (oneCell.y * Math.cos(this.rasterOrigin_Uradian.y) * sinus_C) / (Math.sqrt(oneCell.x * oneCell.x + oneCell.y * oneCell.y)));
        coordinateCell_Udegree.x = this.convertRadian_Udegree(coordinateCell_Udegree.x);
        coordinateCell_Udegree.y = this.convertRadian_Udegree(coordinateCell_Udegree.y);
        return coordinateCell_Udegree;
    }
    /** "Return the line and column of the corresponding cell */
    public Coordinate convertCoordinate_Ucs(double longitude_Udegree, double latitude_Udegree) {
        if (C_Parameters.UCS_WIDTH_Umeter == 0.0) C_Parameters.UCS_WIDTH_Umeter = I_ConstantGerbil.UCS_WIDTH_Umeter;
        double coordinate_Umeter_y = this.distanceToRasterOrigin_Umeter(this.rasterOrigin_Udegree.x, latitude_Udegree);
        double y = coordinate_Umeter_y / C_Parameters.UCS_WIDTH_Umeter;
        double coordinate_Umeter_x = this.distanceToRasterOrigin_Umeter(longitude_Udegree, this.rasterOrigin_Udegree.y);
        double x = coordinate_Umeter_x / C_Parameters.UCS_WIDTH_Umeter;
        return new Coordinate(x, y);
    }
    public Coordinate convertValueCoordinate_Umeter(Coordinate coordinate, double UcsWidth) {
        return new Coordinate(coordinate.x * UcsWidth, coordinate.y * UcsWidth);
    }
}

```