Estimation des densités d'anguilles jaunes d'Europe (Anguilla anguilla) en France, basée sur leur diffusion dans les bassins versants : Modèle TABASCO (spaTialized Anguilla BASin COLonisation) : rapport intermédiaire année 2

J. Domange, Hilaire Drouineau, Cédric Briand, Laurent Beaulaton, Patrick Lambert

Estimation des densités d'anguilles jaunes d'Europe *(Anguilla anguilla)* en France, basée sur leur diffusion dans les bassins versants : Modèle TABASCO (spaTialized Anguilla BASin COLonisation)

Rapport intermédiaire année 2

Thème 11 - Action 23 : Gestion des poissons migrateurs

Jocelyn Domange[(1)], Hilaire Drouineau[(1)], Cedric Briand[(2)], Laurent Beaulaton[(3)], Patrick Lambert[(1)].

(1) **IRSTEA EABX**
(2) **EPTB-Vilaine**
(3) **ONEMA**

28 janvier 2015

Résumé

Analyse des densités d'anguilles jaunes orientée par le phénomène de dispersion : Bases conceptuelle du modèle TABASCO (spaTialized Anguilla BASin COlonization assessment model).

Les caractéristiques du modèle EDA, et notamment certaines de ses hypothèses, nous ont conduits à proposer une nouvelle modélisation nommée TABASCO pour « spaTialised Anguilla BASin COlonisation assessment model ». TABASCO doit être considéré comme un intermédiaire entre des méthodes strictement statistiques et des modèles mécanistes. Deux approches ont été développées en parallèle. La première repose sur la propagation au travers d'un graphe orienté d'une gaussienne correspondant à la résultante d'une onde de diffusion. La seconde correspond à l'utilisation de matrices de transition pour décrire la dynamique de colonisation des anguilles. La formalisation des calculs a permis d'établir une première version du code informatique. Une phase de correction des erreurs de codage, puis d'amélioration des différentes étapes du calcul a ensuite rendu possible une calibration avec les données de pêche électrique sur un bassin versant de référence (Gironde). Enfin, le code a été adapté pour prendre en compte l'ensemble des bassins versants de France métropolitaine, et tourne actuellement sur 60 bassins versants représentant 70,4 % de la superficie de la France métropolitaine.

Abstract

Analysis of yellow eels density oriented by the dispersal process: conceptual bases of the TABASCO model (spaTialised Anguilla BASin COlonisation assessment model).

Considering the features of the EDA model, in particular certain of its assumptions, we propose a new modelling named TABASCO for « spaTialised Anguilla BASin COlonization assessment model ». TABASCO should be considered as an intermediary between strictly statistical approaches and mechanistic models. We developed two methods in parallel. The first one is based on the spreading of a Gaussian distribution resulting of a diffusive wave through an oriented graph. The second corresponds to the use of transition matrices to describe the colonisation dynamics of eels. We obtained a first version of the computer code thanks to the formalization of the calculations. Then, we performed a calibration on the data of electric fishing from the reference basin (Gironde) following a correction phase to remove the bugs and an improvement of the different calculation steps. Finally, we adapted the code to take into account the whole basins of metropolitan France. The model is currently running on 60 catchments representing 70,4 % of the metropolitan France.

Table des figures

Sommaire

Table des matières

Chapitre 1

Introduction

1.1 Contexte général

Afin de réagir au déclin inquiétant de la population d'anguilles européennes observé depuis les années 1980, la commission européenne a institué en septembre 2007 un règlement qui décrète des mesures de reconstitution du stock d'anguilles et impose à chaque État membre de soumettre un plan de gestion de sauvegarde de l'espèce (règlement européen n° 1100/2007 du 18 septembre 2007).
Conformément à ce règlement communautaire qui astreint chaque état membre concerné à mettre en œuvre pour le 1^{er} juillet 2009 un plan par unité de gestion anguille[1], la France a envoyé son plan national le 3 février 2010, et celui-ci a été approuvé par la Commission européenne le 15 février 2010. Son élaboration a été pilotée par les ministères en charge des pêches maritimes et de l'écologie. Nous rappelons sur la figure 1.1 le découpage retenu pour la définition des unités de gestion anguilles.
Les mesures préconisées portent sur les différents types de pêcheries, les obstacles à la circulation des anguilles, le repeuplement, la restauration des habitats et les contaminations.
Programmées sur le court et le moyen terme (horizon 2012-2015), ces mesures sont porteuses d'objectifs ambitieux en matière de réduction des mortalités par la pêche ou liées aux ouvrages.
Une première évaluation de ces plans a été remise en juin 2012. Le plan français, tant dans sa définition que dans sa post-évaluation, s'est jusqu'à présent principalement appuyé sur la modélisation EDA (Eel Density Analysis) pour calculer l'échappement en anguilles argentées (Jouanin *et al.*, 2012b).
Depuis 2010 en effet, Irstea, l'ONEMA et l'Institut d'aménagement de la Vilaine ont développé une version stable d'EDA qui a été appliquée à plusieurs bassins versants européens et, dans le cadre de la post-évaluation, à l'ensemble des unités de gestion anguille françaises. Un test de fiabilité d'EDA a par ailleurs été réalisé sur une réalité construite à partir du modèle CREPE qui synthétise, sous forme d'une approche individus centrée, la compréhension du fonctionnement d'une fraction de population d'anguilles dans un bassin versant (Lambert, 2012). Au final, il s'avère que dans certains cas, l'extrapolation aux secteurs où la pêche électrique n'est pas possible peut conduire à des estimations d'abondance irréalistes ou des répartitions d'individus dans les réseaux hydrographiques biaisés.
En conséquence, un nouveau type de modélisation est étudiée depuis 2012 en vue d'essayer d'améliorer les estimations du stock d'anguilles jaunes. Il s'agit du modèle TABASCO, qui fait l'objet de ce rapport, et qui a été pensé pour être à mi-chemin entre une approche purement statistique et un modèle mécaniste.

[1] Les « bassins versants anguille » pris en compte comme unités de gestion anguille (UGA) sur les territoires ont été déterminés selon des critères validés par le Comité de Gestion des Poissons Migrateurs compétent sur ces territoires. Les efforts de gestion mis en place sur ces zones doivent contribuer à augmenter la part de population d'anguilles dévalantes.

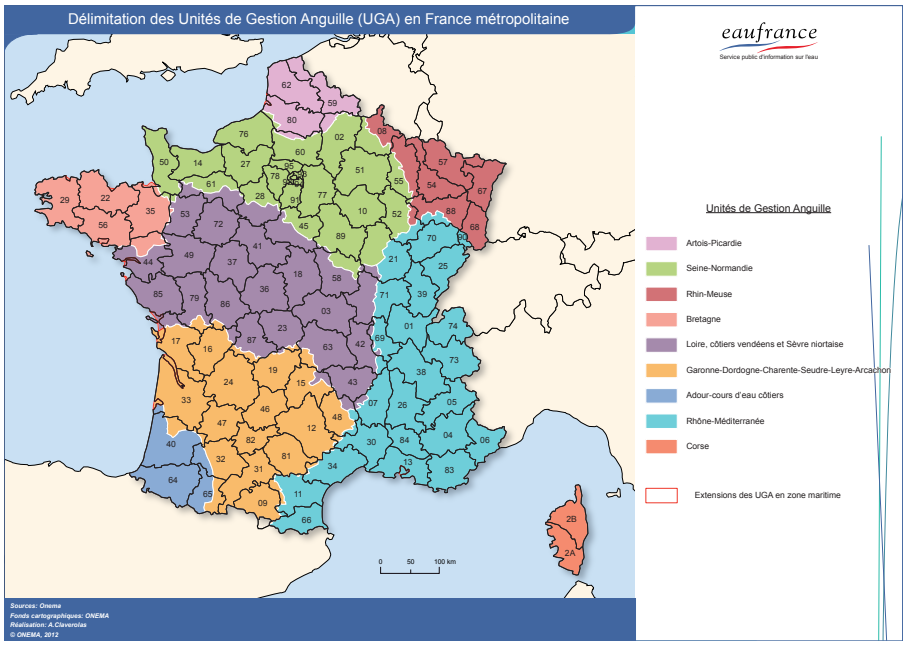

FIGURE 1.1 : Les unités de gestion anguille (UGA) en France.

1.2 Rappel préalable : le modèle EDA

L'objectif du modèle TABASCO est de compléter les résultats de la modélisation du stock d'anguilles en France, principalement obtenus avec le modèle EDA. Ce modèle permettra également d'avoir des données estimées à partir d'approches multiples, ce qui devrait renforcer les prédictions obtenues.

1.2.1 Bases du modèle EDA

Le modèle EDA a été initialement développé pour étudier l'impact des barrages sur la répartition des anguilles dans les cours d'eau bretons (Leprévost, 2007). Il a par la suite été appliqué à l'ensemble du territoire métropolitain dans le cadre de la définition du plan français gestion de l'anguille européenne (2010), puis dans le cadre de l'évaluation de ce même plan (2012). Il a également fait l'objet d'applications dans des unités de gestion en Europe (Walker *et al.*, 2011). Tout au long de ce processus il a fait l'objet d'améliorations successives, en particulier en intégrant des impacts anthropiques au travers de l'occupation du sol (Jouanin *et al.*, 2012b). Son principal atout est la possibilité de son application à large échelle à partir des données de réseau de pêches électriques actuellement disponibles.
Le principe de cette approche (Jouanin *et al.*, 2012b) est :

1. de relier les densités d'anguilles jaunes observées lors de pêches électriques à différents paramètres : méthodes d'échantillonnage, conditions environnementales (distance à la mer, distance relative, température, altitude,...), conditions anthropiques (obstacles, occupation du sol,...) et année d'observation,

2. d'extrapoler les densités d'anguilles jaunes dans chaque tronçon du réseau hydrographique en appliquant un modèle statistique calibré à l'étape 1,

3. de calculer l'abondance totale du stock d'anguilles jaunes en multipliant ces densités par la surface en eau des tronçons et en les additionnant,

4. de calculer un échappement potentiel en convertissant le stock estimé d'anguilles jaunes à l'étape 3 en stock d'anguilles argentées,

5. de calculer un échappement effectif en soustrayant les mortalités d'anguilles argentées (pêcheries, turbines) connues ou estimées,

6. de donner une estimation de l'échappement pristine en considérant les conditions anthropiques mises artificiellement à zéro et un jeu temporel de variables avant 1980.

La prise en compte des mortalités par les turbines a fait l'objet d'un développement particulier en 2012 (Jouanin *et al.*, 2012a).
L'approche retenue dans EDA est basée sur le modèle statistique delta-gamma proposé par Stefánsson (Stefánsson et Pálsson, 1996) qui permet de traiter des données présentant une surreprésentation de valeurs nulles. En effet, les effectifs d'anguilles étant faibles, l'occurrence d'absences d'observations, et donc de densités mesurées nulles, est forte. L'estimation des densités d'anguilles jaunes est ainsi réalisée au travers de la multiplication d'un modèle de présence-absence (modèle Δ) et d'un modèle de densités non nulles (modèle Γ). Les réponses des variables à modéliser n'étant pas linéaires, des modèles additifs généralisés (GAM) (Hastie et Tibshirani, 1990) ont été utilisés, avec une distribution binomiale et un lien Logit[2] pour le modèle Δ et une distribution Gamma[3] et un lien logarithme pour le modèle Γ (Jouanin *et al.*, 2012b).
Parmi les variables explicatives potentielles, n'ont été gardées que celles qui avaient une répartition comparable dans les jeux d'apprentissage (tronçons avec pêches électriques) et d'extrapolation (l'ensemble des tronçons). Ainsi, la distance à la source, la distance relative (ratio de distance à la mer sur la longueur du drain principal), la surface amont du bassin versant et la pente ont été écartées (au risque de fragiliser les prédictions).

1.2.2 Limites du modèle EDA

Dans la version actuelle d'EDA, plusieurs pistes d'améliorations ont été identifiées. Tout d'abord, concernant la distribution naturelle des anguilles dans un réseau hydrographique :

- Le flux de colonisants se répartissant entre tributaires à chaque confluence en plusieurs populations de tailles différentes, les densités de deux tronçons à la même distance de la mer ne devraient pas nécessairement avoir la même densité d'anguilles en fonction du nombre et de la nature des confluences à l'aval (prise en compte de la topologie aval du réseau).

- Deux tronçons à la même distance de la mer ne devraient pas avoir la même densité d'anguilles quelle que soit la surface du bassin versant en amont (non prise en compte de la taille du bassin). On peut supposer que le débit (dont un proxy est la surface de bassin versant amont) influence la colonisation des bassins versants par les anguilles, soit en favorisant cette colonisation avec la notion de débit d'attrait, soit comme facteur bloquant la progression des animaux. Une question encore non tranchée est de savoir si cette taille de bassin versant ne joue qu'au niveau de la surface en eau du tronçon (la largeur évoluant en fonction de la racine carrée de la taille de bassin versant) sans influencer la densité.

- Deux tronçons à la même distance de la mer ne devraient pas avoir la même densité d'anguilles quelle que soit la distance à la source (non prise en compte de la topologie amont). On peut en effet imaginer qu'une station plus proche de la source, donc moins productive, abrite une densité d'anguilles plus faible.

[2]La fonction Logit est une fonction mathématique très utilisée en statistiques et pour la régression logistique. Son expression est :

$$logit(p) = log\left(\frac{p}{1-p}\right)$$

Où p est un nombre compris entre 0 et 1 (typiquement une probabilité). Si le logarithme utilisé est la fonction logarithme népérien, la fonction Logit est la réciproque de la fonction sigmoïde.

[3]La distribution Gamma est une loi de probabilité que l'on peut paramétrer à l'aide d'un paramètre de forme α et d'un paramètre d'intensité β, de telle sorte que sa fonction de densité de probabilité peut se mettre sous la forme :

$$f(x;\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)}\, x^{\alpha-1}\, e^{-\beta\, x}$$

Où α et β sont deux paramètres strictement positifs, et où $\Gamma(x)$ est la fonction Gamma d'Euler.

Concernant maintenant la prise en compte des impacts anthropiques :

- Un barrage ne modifie pas la densité à l'aval. On peut en effet supposer que la densité à l'aval d'un barrage est augmentée (au moins sur une certaine distance) et la densité à l'amont diminuée. Dans une simulation avec la version actuelle d'EDA, la suppression d'un barrage conduit à augmenter l'abondance des anguilles en amont sans modifier les densités à l'aval. Cette apparente création d'anguilles suite à l'effacement de l'obstacle revient implicitement, lors de simulations avec le barrage, à considérer un blocage d'individus à l'aval de l'obstacle équivalent en nombre à cette création et à une mortalité totale des mêmes anguilles.

- Les prélèvements d'anguilles jaunes ne sont pas explicitement traités, ils sont supposés se retrouver dans les densités d'anguilles observées lors des pêches électriques (cependant, ce point ne sera pas non plus pris en compte dans TABASCO).

Il est fort vraisemblable que la méthodologie EDA, moyennant l'ajout de nouvelles variables (surface totale du bassin versant, nombre de confluences à l'aval, ratio entre surface de bassin versant amont et surface du bassin versant total, distance à la source, distance au premier barrage en amont, etc...) et d'un jeu d'apprentissage plus représentatif du réseau total, soit en mesure de lever, au moins partiellement, ces difficultés. Concernant ensuite les techniques statistiques :

- La sensibilité des modèles additifs généralisés à la corrélation entre variables explicatives, qui ne garantit pas une causalité avec la variable à expliquer. C'est en particulier gênant pour les extrapolations, notamment si l'on souhaitait prédire ce que serait la distribution des anguilles sans barrages. L'approche retenue dans TABASCO ne garantit cependant pas nécessairement de meilleurs résultats à ce niveau.

Concernant enfin le jeu de données d'apprentissage :

- La sous-représentation des stations d'échantillonnage dans les milieux profonds (drains principaux à l'aval des bassins versants) rendant les extrapolations hasardeuses. Il est néanmoins certain que l'on sera confronté au même problème dans TABASCO, puisqu'il n'existe de toute façon pas de données dans ces zones.

1.2.3 Proposition d'une nouvelle approche : TABASCO

Il a ainsi été choisi de construire un nouveau modèle, parcimonieux en paramètres, analysant les densités d'anguilles mais décrivant explicitement le processus de colonisation sous-jacent à la distribution dans un réseau hydrographique. Il a été nommé TABASCO pour « spaTialised Anguilla BASin COlonisation assessment model ». TABASCO doit donc être considéré comme un intermédiaire entre des approches strictement statistiques comme EDA et des modèles mécanistes comme GlobAng (Lambert et Rochard, 2007) ou SMEP (Aprahamian *et al.*, 2007). Actuellement, il est admis que ce processus dans un bassin versant est un phénomène diffusif (Ibbotson *et al.*, 2002) avec éventuellement une composante advective[4] vers l'amont sur une courte distance durant les premières années de vie continentale de l'animal (C. Rigaud et al , en préparation). EDA retrouve d'ailleurs une courbe de réponse de l'abondance en fonction de la distance à la mer qui approche une décroissance gaussienne, en accord avec le caractère diffusif de la colonisation[5].
Deux approches ont été poursuivies en parallèle. La première repose sur la propagation

[4] On rappelle que l'advection correspond au transport d'une quantité (scalaire ou vectorielle) par un champ vectoriel. Mathématiquement, l'opérateur advection correspond au produit scalaire du vecteur vitesse par le vecteur gradient (opérateur nabla). En ce qui nous concerne, il s'agit du phénomène de transport orienté des anguilles vers l'amont par leur mouvement propre (*i.e.* par leur nage).

[5] Pour être précis, EDA prédit une décroissance exponentielle de l'abondance. Or, la loi normale est une loi de la famille exponentielle. avec une décroissance dite surexponentielle à droite de la valeur moyenne(Lifschitz, 1995). Il n'est donc pas erroné de comparer les variations de ces deux fonctions sur leur domaine de décroissance.

au travers d'un graphe d'une gaussienne correspondant à la résultante d'une diffusion. La seconde correspond à une vision plus mécaniste de la diffusion, traduite sous forme de matrice de transition élevée à une certaine puissance. Cette seconde approche présente des analogies avec le modèle théorique OMMER (Lambert *et al.*, 2011).
Il nous a semblé intéressant de mener en parallèle le développement de ces deux approches puisque l'on n'a pas d'idée sur leur faisabilité a priori, et qu'elles pourraient être complémentaires en offrant des outils d'interprétation différents (par exemple pour la première approche, une caractérisation de la diffusivité dans un bassin versant, et pour l'approche matricielle, l'identification de tronçons-clefs par une analyse de réseau basée sur ce qui est pratiqué pour les réseaux trophiques (Libralato *et al.*, 2006)).
TABASCO devrait, par sa prise en compte explicite de la structure du réseau hydrographique, résoudre les effets liés à la non prise en compte de la topologie. On peut ainsi espérer que la contrainte liée à l'extrapolation dans EDA (représentativité des variables dans le jeu d'apprentissage) puisse être levée, ce qui offrirait la possibilité de tenir compte de la taille du bassin versant et de la position dans le continuum amont-aval. Par ailleurs, l'impact des barrages doit être mieux traité puisque le processus sous-jacent au niveau du barrage est explicite.
Nous récapitulons dans le tableau 1.1 ci-dessous les points que le modèle TABASCO doit améliorer (notamment par construction), ceux qu'il pourrait améliorer (sous réserve de vérifications), et ceux qui ne seront pas pris en compte ou améliorés.

Améliorations acquises	**Améliorations espérées**	**Problèmes non traités**
Prise en compte de l'effet des obstacles sur l'effectif amont (blocage)	Prise en compte de l'effet des obstacles sur l'effectif aval (rétro-diffusion)	Effets des prélèvements d'anguilles jaunes
Prise en compte de la surface en eau amont (répartition pondérée par l'espace disponible)	Meilleure extrapolation (sensibilité réduite aux variables corrélées)	Manque de données dans les drains principaux
Prise en compte du continuum amont-aval	Détermination des erreurs propagées dans les calculs	
Prise en compte de la topologie du réseau	Cartographie par classe d'âge	

TABLE 1.1 : Tableau récapitulatif des améliorations et des problèmes non traités du modèle TABASCO.

Chapitre 2

Principes du modèle TABASCO

2.1 Modélisation du réseau hydrographique français

2.1.1 Objets contenant l'information géographique

Le modèle TABASCO s'appuie principalement sur les sources d'information géographique que sont le réseau hydrographique théorique français et le référentiel des obstacles à l'écoulement, ainsi que sur la base de données sur les milieux aquatiques et les poissons. Le modèle utilise les données géoréférencées dans ces différents objets, et permet un affichage spatialisé des résultats de la simulation par projection sur ces mêmes référentiels.

Le RHT

Le réseau hydrographique théorique (RHT) est un nouveau réseau hydrographique dérivé de la BD Alti® de l'Institut Géographique National (IGN) (Pella *et al.*, 2012). La méthode « Agree » utilisée pour construire ce réseau permet de modifier un modèle numérique de terrain à partir d'un réseau d'écoulement pré-défini. Ainsi, le RHT est développé à partir du modèle numérique de terrain de la BD Alti® re-conditionné par le RHE (Réseau Hydrographique Étendu), qui est lui-même une simplification du réseau hydrographique de référence de l'IGN, la BD Carthage®. La compatibilité entre le RHT et la BD Alti® permet d'identifier les bassins versants et de simuler des écoulements avec une meilleure précision. Cette approche permet d'intégrer un ensemble d'attributs spatialisés et de les cumuler le long du réseau. Ainsi, des attributs topographiques, hydrologiques et climatiques sont calculés et intégrés dans un système d'information géographique.

Le ROE

En France métropolitaine, plusieurs dizaines de milliers d'obstacles à l'écoulement – barrages, écluses, seuils, etc... - ont été recensés sur les cours d'eau. Ils sont à l'origine de profondes transformations de la morphologie et de l'hydrologie des milieux aquatiques, et perturbent fortement le fonctionnement de ces écosystèmes. Dans le cadre de l'évaluation de l'effet de ces obstacles sur les écosystèmes aquatiques, il était nécessaire de les inventorier et de les rassembler dans une base de données nationale. Le Référentiel des Obstacles à l'Écoulement (ROE) recense ainsi l'ensemble des ouvrages inventoriés sur le territoire national en leur associant des informations (code national unique, localisation, typologie) communes à l'ensemble des acteurs de l'eau et de l'aménagement du territoire (Léonard et Zegel, 2010). Il assure aussi la gestion et la traçabilité des informations en provenance des différents partenaires. Le référentiel actuellement mis en ligne est une version 6.0 figée mettant à disposition les données validées et gelées en date du 7 mai 2014. De lourds développements sont actuellement mis en œuvre pour assurer l'interopérabilité et la pérennité du Référentiel des Obstacles à l'Écoulement au travers d'une mise à disposition en temps réel, via le format d'échange défini par le Service d'Administration Nationale des Données et Référentiels sur l'Eau (SANDRE). Le

ROE, dans sa version de septembre 2013, présente 69136 ouvrages référencés en France.

TABASCO repose sur l'adaptation de ces deux derniers objets d'information géographique (RHT et ROE) sous forme d'un réseau formalisé exploitable directement dans le code du modèle.

La BDmap

La Base de Données sur les Milieux Aquatiques et les Poissons (BDmap) fait partie du SIE (Système d'Information sur l'Eau), géré par l'Onema. Une extraction de données donne accès aux informations sur les pêches électriques d'anguilles. La calibration des paramètres du modèle s'appuie sur une optimisation basée sur ces données de terrain (voir plus loin dans ce rapport). L'extraction issue de la BDmap sur laquelle nous nous basons actuellement pour le modèle TABASCO regroupe 19201 opérations de pêches électriques sur 11371 stations.

Formalisation d'un réseau hydrographique

Un réseau hydrographique peut être représenté sous forme d'un **graphe orienté**. En mathématiques, et plus précisément en théorie des graphes, un graphe orienté est un graphe défini par la donnée d'un ensemble de **nœuds** (ou **sommets**, ou **vertex**) V connectés par un ensemble d'**arcs** A, eux-mêmes définis comme étant un couple de sommets (au sens mathématique du terme, c'est-à-dire la donnée des deux sommets dans un ordre déterminé).
Ainsi, tout graphe orienté G s'écrit $G = (V, A)$, avec :
- Les éléments de l'ensemble V qui sont les nœuds, ou sommets, ou encore vertex.
- Les éléments de l'ensemble A qui sont les arcs orientés, représentés par un couple mathématique de sommets.
Un graphe orienté est qualifié de simple s'il ne comprend pas de boucles ni d'arcs multiples (arcs ayant les mêmes nœuds de départ et d'arrivée). Dans le cas contraire on parle de multigraphe.
Le réseau hydrographique que nous modélisons constitue donc un graphe orienté simple.

Adaptation du RHT et du ROE en vue de leur utilisation dans le modèle TABASCO

Le RHT, découpé par le ROE, a été représenté sous forme d'un objet *topology* tel qu'implémenté par le système d'information géographique (SIG) PostGIS (voir la section concernant les outils informatiques du présent rapport pour de plus amples détails sur ce logiciel), en le structurant en tronçons (*reaches* en anglais) et nœuds (*nodes* en anglais).
Un total de 61961 obstacles maîtres[1] du ROE ont pu être projetés sur le RHT. Les tronçons ont été scindés en deux au niveau de chaque obstacle et les attributs des deux sous-tronçons ont été recalculés. Ainsi, les obstacles se situent systématiquement au niveau de nœuds. Cette étape tourne sur la France en environ quatre heures.
10817 stations de pêches électriques ont également pu être projetées sur la topologie créée afin d'identifier les tronçons dans lesquels elles ont été réalisées.
Dans le code, les nœuds et les tronçons du réseau hydrographique ont été récupérés depuis le RHT et le ROE (voir la section outils informatiques pour la méthode utilisée) et un graphe a été créé grâce à l'utilisation d'une bibliothèque logicielle spécifique (BGL, pour Boost Graph Library) en associant les *nodes* et les *reaches* du réseau à des *vertices* et à des *edges* du graphe (Siek *et al.*, 2002).
En outre, le programme a été codé de telle façon qu'il fonctionne bassin versant par bassin versant. Ainsi, tout le réseau hydrographique n'est pas créé d'un seul tenant, mais chaque sous-réseau associé aux différents bassins-versants français est créé successivement. Cela laisse par ailleurs la liberté de ne travailler que sur un bassin versant

[1] Dans le cas d'un ensemble hydraulique comprenant plusieurs obstacles groupés sur un même cours d'eau et pour lesquels l'effet d'un des éléments prévaut sur tous les autres, on ne considère que ce seul obstacle principal, dit obstacle maître.

donné, ou bien sur un ensemble de plusieurs bassins versants, ou encore sur le réseau hydrographique de la France entière. En outre, les calculs peuvent être parallélisés si l'on souhaite optimiser le temps de simulation.
Pour chaque bassin versant, le graphe associé au réseau hydrographique est obtenu par construction à l'aide d'une requête SQL récursive à partir de l'exutoire du bassin, c'est à dire à partir du tronçon situé le plus en aval.

2.1.2 Quels outils d'analyse du réseau ?

Nous disposons, via le RHT et le ROE, d'un certain nombre de caractéristiques du réseau hydrographique. A chaque tronçon est associé un jeu de données comprenant, entre autres, la largeur du tronçon, sa longueur, sa distance à la source, son rang de Strahler, ou encore la surface du bassin amont, ainsi qu'un identifiant unique permettant de l'indexer. Nous recalculons par ailleurs la distance à la mer de chaque tronçon dans le modèle. Ces différents paramètres permettent de visualiser le réseau en fonction de certaines de ses caractéristiques, mais aussi de prévoir quel comportement global aura le modèle dans ce bassin. La figure 2.1) donne par exemple la répartition des tronçons dans notre bassin versant de référence (Gironde), en faisant apparaître les distributions marginales et le barycentre de la distribution globale.
Ce type de représentation graphique nous a notamment permis de constater que la majorité des tronçons d'un bassin versant sont situés à des distances relativement faibles de leurs sources, et ce, quel que soit leur éloignement de la mer.
En outre, on voit facilement apparaître les différents drains du bassin sur ce type de

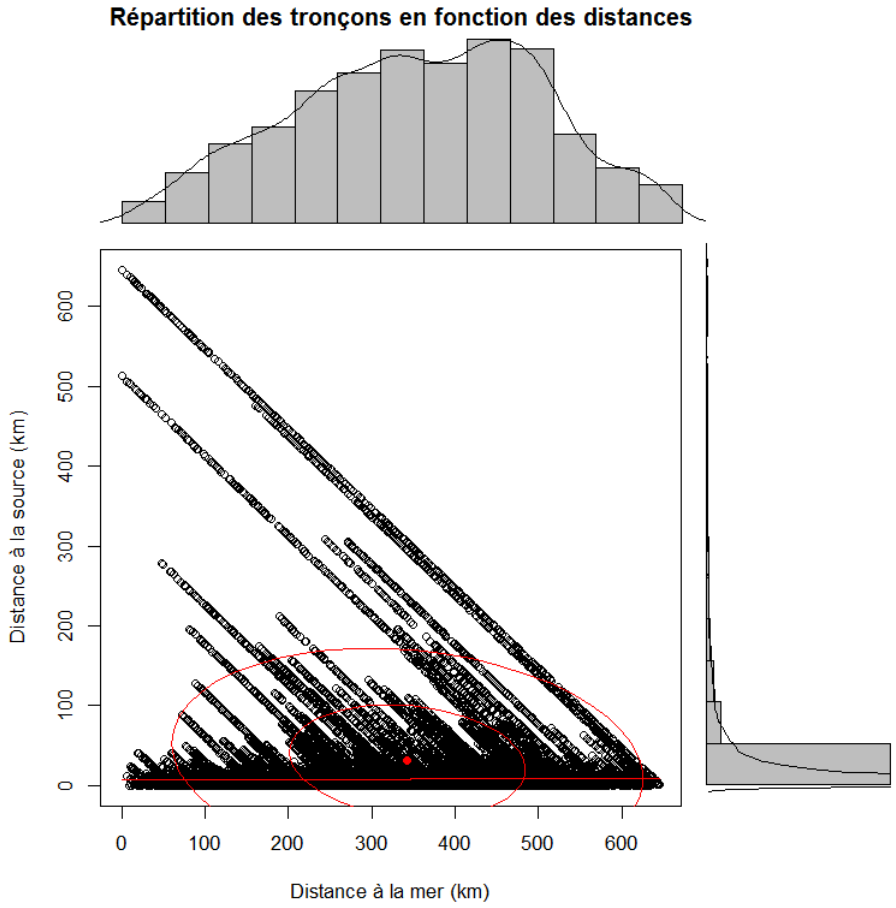

FIGURE 2.1 : Répartition des tronçons du bassin versant de la Gironde en fonction de leur distance à la mer et de leur distance à la source. Les distributions marginales sont également indiquées, ainsi que le barycentre de la distribution globale.

graphe. Tous les tronçons d'un même drain sont sur des droites parallèles d'équation

$D_{source} = L_{tot} - D_{mer}$, où L_{tot} représente la longueur totale du drain dans le réseau (depuis sa source jusqu'à l'exutoire). Tous les points du graphe sont contenus dans le triangle formé par les deux axes et la droite correspondant au drain le plus long du réseau (a priori le drain principal du bassin).

2.2 Deux modèles de diffusion

Le modèle TABASCO, qui s'intéresse au processus de diffusion dans un ou plusieurs bassins versants, se présente sous deux approches développées en parallèle. La première consiste à propager une onde de diffusion gaussienne dans le réseau, et la seconde traite le phénomène de diffusion en utilisant des matrices de transition (ou de transfert) pour modéliser la propagation des effectifs d'anguilles d'un tronçon à l'autre.

2.2.1 Approche gaussienne

Principe général et mise en équations

La première loi que nous utilisons est la loi de conservation locale des anguilles pour un problème unidimensionnel selon l'axe (Ox) :

$$\frac{\partial n}{\partial t} = -\frac{\partial j}{\partial x} \tag{2.1}$$

Où $n(x,t)$ est la concentration d'anguilles, t le temps (qui est lié, mais de façon non triviale, à l'âge des anguilles durant le processus de colonisation du milieu), $\vec{j}(x,t)$ le vecteur densité de courant d'anguilles (c'est-à-dire le produit de la densité d'anguilles par leur vitesse d'ensemble) et x la coordonnée représentative du problème (position dans le bassin versant).
Pour démontrer cette loi de conservation, considérons un tube dans lequel évoluent les anguilles, qui s'appuie sur un cercle d'axe (Ox) et d'aire S. Le tube est un cylindre de génératrices parallèles à (Ox) (figure 2.2).

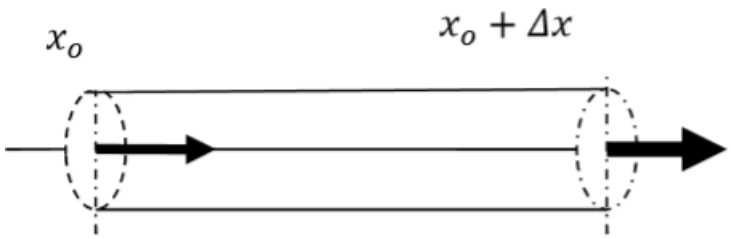

FIGURE 2.2 : Schéma de principe pour la démonstration de la loi de conservation des anguilles selon l'axe (Ox).

Les lignes de courant sont des droites, et le flux d'anguilles au travers d'une section d'abscisse x est :

$$\phi(x,t) \equiv \int\int \vec{j}(x,t).d\vec{S}$$

Soit :

$$\phi(x,t) = S\, j(x,t)$$

Faisons un bilan de matière pour les anguilles qui diffusent entre les instants t_0 et $t_0 + \mathrm{d}t$, pour la tranche comprise entre les abscisses x_0 et $x_0 + \Delta x$:

$$N(t_0 + \mathrm{d}t) - N(t_0) = \phi(x_0, t_0)\,\mathrm{d}t - \phi(x_0 + \Delta x, t_0)\,\mathrm{d}t$$

Où N(t) est le nombre d'anguilles à l'instant t entre les deux sections, soit donc :

$$\frac{N(t_0 + \mathrm{d}t) - N(t_0)}{\mathrm{d}t} = \phi(x_0, t_0) - \phi(x_0 + \Delta x, t_0) = S[j(x_0, t_0) - j(x_0 + \Delta x, t_0)]$$

Soit $n(x,t)$ la densité volumique d'anguilles diffusantes.

$$N(t_0) = \int_{x_0}^{x_0+\Delta x} n(x,t_0) S\, dx$$

$$\frac{N(t_0+\mathrm{d}t)-N(t_0)}{\mathrm{d}t} = S \int_{x_0}^{x_0+\Delta x} \frac{n(x,t_0+\mathrm{d}t)-n(x,t_0)}{\mathrm{d}t} = S \int_{x_0}^{x_0+\Delta x} \frac{\partial n}{\partial t}(x_0,t)\mathrm{d}x$$

$$\frac{S}{\Delta x} \int_{x_0}^{x_0+\Delta x} \frac{\partial n}{\partial t}(x_0,t)\mathrm{d}x = -S\frac{j(x_0+\Delta x)-j(x_0)}{\Delta x}$$

Et en faisant tendre Δx vers 0, il vient :

$$\frac{\partial n}{\partial t}(x_0,t_0) - \frac{\partial j}{\partial x}(x_0,t_0) = 0$$

D'où l'équation de conservation 2.1.
La Loi de Fick à une dimension nous permet ensuite d'exprimer la densité de courant en fonction du gradient de concentration d'anguilles :

$$j = -D\,\frac{\partial n}{\partial x} \tag{2.2}$$

Où D est le coefficient de diffusion, qui s'exprime en $m^2.\,s^{-1}$ dans les unités SI, et habituellement en $km^2.\,années^{-1}$ dans le cas d'une étude sur la colonisation d'un bassin versant par des anguilles. Ce coefficient quantifie le taux de déplacement des anguilles le long d'un gradient de densité, à travers une surface perpendiculaire à la direction du mouvement (Ibbotson *et al.*, 2002).
En combinant les équations 2.1 et 2.2, on obtient l'équation de diffusion unidimensionnelle :

$$\begin{cases} \dfrac{\partial n}{\partial t} = -\dfrac{\partial j}{\partial x} \\ \\ j = -D\,\dfrac{\partial n}{\partial x} \end{cases} \Longrightarrow \frac{\partial n}{\partial t} = D\,\frac{\partial^2 n}{\partial x^2} \tag{2.3}$$

Dans notre problème de diffusion des anguilles dans un bassin versant, nous souhaitons résoudre l'équation 2.3 dans le cas d'un régime non-stationnaire (évolutif) et unidimensionnel (la donnée caractéristique du problème est la distance de pénétration des anguilles dans un bassin versant ramenée sur l'axe (Ox)).
Nous prenons pour l'instant comme hypothèse que le caractère thalassotoque de l'anguille place cette résolution dans le cas simple d'une seule source de diffusion. En effet, les civelles d'anguilles arrivant du milieu océanique, la diffusion dans chaque bassin versant se fait à partir de l'exutoire de ce bassin. Les civelles utilisent les courants de marée montante pour progresser en zone estuarienne. Il serait donc logique de considérer la limite de marée dynamique plutôt que la limite transversale de la mer. Toutefois, la localisation de cette limite de marée dynamique n'est pas une donnée systématiquement renseignée dans les bases de données des réseaux hydrographiques français et européens. Il est donc proposé d'utiliser la distance à la mer comme point de départ du processus de diffusion[2]. On négligera donc, dans la version actuelle du code, les sources secondaires de diffusion des anguilles induites par les obstacles (rétro-diffusion) ainsi que les alevinages. Par ailleurs, la diffusion dans un réseau hydrographique introduit une complexité supplémentaire liée à la topologie du graphe (en fait de l'arbre) que constitue ce réseau hydrographique (Webb et Padgham, 2009, 2013).
Pour résoudre l'équation 2.3, il faut imposer des conditions aux limites. On pose donc

[2]La distance à la mer d'un tronçon est définie dans le modèle comme la distance entre l'exutoire du bassin versant et l'extrémité aval du tronçon.

qu'à $t = 0$ (instant virtuel où les anguilles sont supposées pénétrer dans le bassin versant par l'exutoire) $n(x, 0) = N_0\,\delta(x)$ où δ désigne la distribution de Dirac et $n(x, t)$ la concentration linéique des anguilles. Autrement dit, on suppose que les N_0 anguilles (la totalité) sont présentes à l'exutoire du bassin versant au départ de la simulation. On rappelle que la distribution de Dirac possède la propriété suivante :

$$\int_{-\infty}^{+\infty} \delta(x)\mathrm{d}x = 1$$

On retrouve bien un nombre d'anguilles N_0 à t=0 :

$$\int_{-\infty}^{+\infty} n(x, 0)\mathrm{d}x = \int_{-\infty}^{+\infty} N_0\delta(x)\mathrm{d}x = N_0$$

La solution générale de l'équation 2.3 s'écrit alors :

$$n(x, t) = \frac{N_0}{\sqrt{4\,\pi\,D\,t}}\, e^{-\dfrac{x^2}{4\,D\,t}} \tag{2.4}$$

Avec :

$$\int_{-\infty}^{+\infty} n(x, t)\mathrm{d}x = N_0$$

On peut donc aussi définir une densité de probabilité de présence, qui est alors :

$$p(x, t) = \frac{1}{N_0} n(x, t) = \frac{1}{\sqrt{4\,\pi\,D\,t}}\, e^{-\dfrac{x^2}{4\,D\,t}} = \frac{1}{\sqrt{2\,\pi}\,\sigma}\, e^{-\dfrac{x^2}{2\sigma^2}} \tag{2.5}$$

Il s'agit bien d'une densité de probabilité, car la fonction p vérifie les deux propriétés suivantes :

$$\begin{cases} p(x, t) \geq 0 \quad \forall x \in \mathbb{R} \\ \displaystyle\int_{-\infty}^{+\infty} p(x, t)\mathrm{d}x = 1 \end{cases}$$

On retrouve bien pour $p(x, t)$ l'expression d'une fonction gaussienne de moyenne $\mu = 0$ et de variance $\sigma^2 = 2\,D\,t$. On notera qu'en présence d'advection due à la nage orientée des anguilles (mouvement de masse), l'expression devient simplement :

$$p(x, t) = \frac{1}{\sqrt{4\,\pi\,D\,t}}\, e^{-\dfrac{(x-\lambda)^2}{4\,D\,t}} = \frac{1}{\sqrt{2\,\pi}\,\sigma}\, e^{-\dfrac{(x-\lambda)^2}{2\sigma^2}} \tag{2.6}$$

La moyenne de la gaussienne est alors égale à λ. Dans la suite du rapport, nous traiterons les calculs sans advection par souci de concision, mais ils sont évidemment facilement adaptables à une situation en présence d'advection. Le taux d'advection fait d'ailleurs partie des paramètres optimisés dans le modèle.
Dans TABASCO, nous posons comme hypothèse que le coefficient de diffusion n'est pas constant car il existe un gradient de diffusivité selon l'axe (Ox) (position dans le bassin en fonction de la distance à la mer) qui fait varier ce coefficient (donc selon une seule coordonnée spatiale). Ainsi $D = D(x)$, mais nous le noterons D par souci de clarté dans les équations qui suivent.
Dans le modèle, la dynamique (variation temporelle) et le coefficient de diffusion sont intégrés dans la fonction de calcul de la variance, puisque $\sigma^2 = 2\,D\,t$. A un instant t donné, l'écart-type de la gaussienne σ est constant dans tout le bassin, car dans ce cas on ne regarde que la résultante de la diffusion à cet instant. Par contre, tout au long du

processus de colonisation, l'écart-type évolue, d'abord parce que le temps (ou l'âge des anguilles) augmente, mais également parce que le coefficient D augmente (à mesure que les anguilles progressent dans le bassin, ou de façon équivalente, s'éloignent de la mer). Nous avons vérifié que cela était compatible avec une diminution de la probabilité de sédentarisation avec l'éloignement à la mer, étant donné que les milieux sont moins riches en amont, et qu'en conséquence les anguilles privilégient une sédentarisation dans les tronçons les plus avals. Parallèlement à cela, la largeur et le débit des cours d'eau diminuent lorsque l'on s'éloigne de l'exutoire, ce qui va aussi dans le sens de diminuer la probabilité se sédentarisation. Nous avons donc choisi de traduire cela par une augmentation de la diffusivité lorsque la distance à la mer augmente et lorsque le rang de Strahler d'un cours d'eau diminue. Cette hypothèse sera susceptible d'être modifiée ou abandonnée prochainement s'il s'avérait qu'elle ne rendait pas correctement compte des observations biologiques relatives aux anguilles et de la configuration topologique des bassins. La fonction actuellement choisie dans le modèle pour le calcul de l'écart-type dans chaque tronçon est donc de la forme :

$$\sigma^2 = \alpha \, D_{out}{}^{(\beta - rang/\gamma)} \tag{2.7}$$

Où α, β et γ sont trois paramètres du modèle (dont les domaines de validité seront mentionnés plus loin dans ce rapport), D_{out} est la distance à la mer de l'extrémité aval du tronçon considéré et *rang* est le rang de Strahler du tronçon. Le paramètre α est homogène à une distance, les deux autres paramètres β et γ sont adimensionnels.

Probabilités de sédentarisation

La population d'anguilles $N(i)$ dans le $i^{éme}$ tronçon du réseau est donnée par la proportion d'anguilles qui vont se sédentariser dans ce tronçon compris entre $D_{out}(i)$ et $D_{out}(i) + l(i)$ parmi toutes celles qui sont issues du tronçon précédent. On a ainsi :

$$N(i) = \overline{N(i-1)} \, \frac{\displaystyle\int_{D_{out}(i)}^{D_{out}(i)+l(i)} p(x,i)\,\mathrm{d}x}{\displaystyle\int_{D_{out}(i)}^{+\infty} p(x,i)\,\mathrm{d}x} \tag{2.8}$$

Afin d'implémenter ce calcul du nombre d'anguilles qui se sédentarisent dans le tronçon i, nous allons réécrire l'équation à l'aide de la fonction de distribution cumulative (fonction de répartition) de la distribution normale centrée réduite, définie par :

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-t^2/2}\, dt \tag{2.9}$$

Exprimons l'équation 2.8 en y mettant l'expression explicite de la distribution des anguilles :

$$N(i) = \overline{N(i-1)}\,\frac{\displaystyle\int_{D_{out}(i)}^{D_{out}(i)+l(i)} \frac{1}{\sqrt{2\,\pi}\,\sigma}\,e^{-\frac{x^2}{2\sigma^2}}\,\mathrm{d}x}{\displaystyle\int_{D_{out}(i)}^{+\infty} \frac{1}{\sqrt{2\,\pi}\,\sigma}\,e^{-\frac{x^2}{2\sigma^2}}\,\mathrm{d}x}$$

$$= \overline{N(i-1)}\,\frac{\displaystyle\int_{-\infty}^{D_{out}(i)+l(i)} \frac{1}{\sqrt{2\,\pi}\,\sigma}\,e^{-\frac{x^2}{2\sigma^2}}\,\mathrm{d}x - \int_{-\infty}^{D_{out}(i)} \frac{1}{\sqrt{2\,\pi}\,\sigma}\,e^{-\frac{x^2}{2\sigma^2}}\,\mathrm{d}x}{\displaystyle\int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\,\pi}\,\sigma}\,e^{-\frac{x^2}{2\sigma^2}}\,\mathrm{d}x - \int_{-\infty}^{D_{out}(i)} \frac{1}{\sqrt{2\,\pi}\,\sigma}\,e^{-\frac{x^2}{2\sigma^2}}\,\mathrm{d}x}$$

$$= \overline{N(i-1)}\,\frac{\displaystyle\frac{1}{\sigma}\left(\int_{-\infty}^{D_{out}(i)+l(i)} \frac{1}{\sqrt{2\,\pi}}\,e^{-\frac{x^2}{2\sigma^2}}\,\mathrm{d}x - \int_{-\infty}^{D_{out}(i)} \frac{1}{\sqrt{2\,\pi}}\,e^{-\frac{x^2}{2\sigma^2}}\,\mathrm{d}x\right)}{\displaystyle\frac{1}{\sigma}\left(\int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\,\pi}}\,e^{-\frac{x^2}{2\sigma^2}}\,\mathrm{d}x - \int_{-\infty}^{D_{out}(i)} \frac{1}{\sqrt{2\,\pi}}\,e^{-\frac{x^2}{2\sigma^2}}\,\mathrm{d}x\right)}$$

Et en posant $u = x/\sigma$ ($\mathrm{d}u = \frac{1}{\sigma}\,\mathrm{d}x$) , il vient :

$$N(i) = \overline{N(i-1)}\,\frac{\displaystyle\int_{-\infty}^{(D_{out}(i)+l(i))/\sigma} \frac{1}{\sqrt{2\,\pi}}\,e^{-\frac{u^2}{2}}\,\mathrm{d}u - \int_{-\infty}^{(D_{out}(i))/\sigma} \frac{1}{\sqrt{2\,\pi}}\,e^{-\frac{u^2}{2}}\,\mathrm{d}u}{\displaystyle\int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\,\pi}}\,e^{-\frac{u^2}{2}}\,\mathrm{d}u - \int_{-\infty}^{(D_{out}(i))/\sigma} \frac{1}{\sqrt{2\,\pi}}\,e^{-\frac{u^2}{2}}\,\mathrm{d}u}$$

$$= \overline{N(i-1)}\,\frac{\Phi\big((D_{out}(i)+l(i))/\sigma\big) - \Phi\big((D_{out}(i))/\sigma\big)}{\Phi(+\infty) - \Phi\big((D_{out}(i))/\sigma\big)}$$

Or de plus, $\Phi(+\infty) = 1$ par définition de la fonction de distribution cumulative, d'où finalement :

$$N(i) = \overline{N(i-1)}\,\frac{\Phi\big((D_{out}(i)+l(i))/\sigma\big) - \Phi\big((D_{out}(i))/\sigma\big)}{1 - \Phi\big((D_{out}(i))/\sigma\big)}$$

Cette dernière relation de récurrence peut être utilisée dans le code car la fonction de distribution cumulative de la distribution normale centrée réduite est implémentée dans la bibliothèque de calcul scientifique en C/C++ que nous utilisons (voir dans le chapitre suivant). Une autre possibilité est d'utiliser la fonction d'erreur de Gauss, implémentée dans le fichier d'en-tête <math.h> et définie par :

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp^{-t^2}\,\mathrm{d}t \tag{2.10}$$

En effet, la fonction de distribution cumulative de la loi normale peut s'écrire :

$$\Phi(x) = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)\right] \tag{2.11}$$

On en déduit donc une autre expression de N(i) :

$$N(i) = \overline{N(i-1)}\,\frac{\operatorname{erf}\left(\frac{(D_{out}(i)+l(i))}{\sqrt{2}\,\sigma}\right) - \operatorname{erf}\left(\frac{D_{out}(i)}{\sqrt{2}\,\sigma}\right)}{1 - \operatorname{erf}\left(\frac{D_{out}(i)}{\sqrt{2}\,\sigma}\right)} \tag{2.12}$$

La méthode de calcul suppose la détermination de la proportion d'anguilles qui se retrouvent dans chaque tronçon du réseau hydrographique suite au processus diffusif. Soit donc P_i la proportion d'anguilles et p_i la probabilité conditionnelle de sédentarisation dans le $i^{ème}$ tronçon du réseau (c'est-à-dire la probabilité pour les anguilles de se trouver dans le $i^{ème}$ tronçon sachant qu'elles ne se sont pas réparties dans les tronçons avals). On suppose pour l'instant une branche unique du graphe orienté (on ignore les confluences et les obstacles pour le moment).
Les équations 2.10 et 2.12 s'écrivent aussi :

$$N_i = \overline{N_{i-1}}\,P_i \tag{2.13}$$

Ainsi, la proportion d'anguilles dans le $i^{ème}$ tronçon par rapport à tout le réseau est P_i. Nous donnons pour information la forme de la fonction $P_i(x_{inf})$ pour différentes longueurs de tronçon sur la figure 2.3.

En conséquence, la probabilité conditionnelle que les anguilles se trouvent dans le

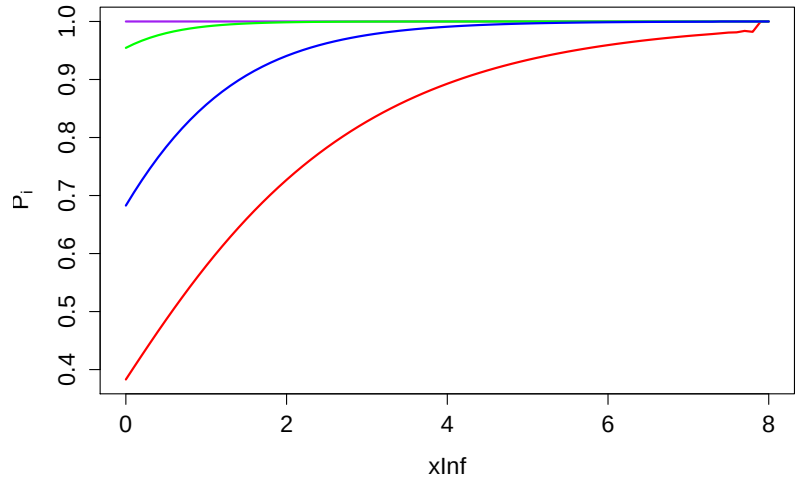

FIGURE 2.3 : Variation de la proportion d'anguilles dans le $i^{ème}$ tronçon en fonction de la valeur de la borne x_{inf} ($x_{inf} = \frac{D_{out}}{\sigma}$, et pour des longueurs de tronçon prises égales à 500 m (rouge), 1 km (bleu), 2 km (vert) et 5 km (violet).

$i^{ème}$ tronçon sachant qu'elles ne se trouvent pas dans les tronçons avals est (et en ne considérant qu'un drain du réseau pour ne pas complexifier les notations) :

$$p_i = P_i(1-P_{i-1})(1-P_{i-2})...(1-P_0) = P_i \prod_{k=0}^{i-1}(1-P_k) \tag{2.14}$$

Connaissant les proportions respectives des (i-1) tronçons avals, on peut donc obtenir la probabilité conditionnelle dans le $i^{ème}$ tronçon. Ce processus itératif nous donne alors les probabilités dans n'importe quel tronçon du réseau. Il reste alors à normaliser

les probabilités. En effet, on a par définition :

$$\sum_{k=0}^{+\infty} p_i = 1 \tag{2.15}$$

Ce qui supposerait un réseau infini. Étant donné que notre réseau est de taille finie, la somme des p_i n'est pas égale à 1 : il faut donc diviser chacune des probabilité p_i par leur somme afin de retrouver une loi de probabilité sur l'ensemble des tronçons. Les nouvelles probabilités $p_i^{'}$ ainsi obtenues vérifient bien :

$$\sum_{k=0}^{M} p_i^{'} = \sum_{k=0}^{M} \frac{p_i}{\sum\limits_{k=0}^{M} p_i} = 1 \tag{2.16}$$

M étant le nombre de tronçons dans le réseau considéré.
En multipliant les probabilités $p_i^{'}$ par le nombre total d'anguilles N_0 qui colonisent le réseau, on obtient le nombre d'anguilles sédentarisées dans chacun des tronçons.

Prise en compte des obstacles

Par construction du modèle, les obstacles sont nécessairement situés à une extrémité de tronçon (par convention au nœud amont), et sont pris en compte de la façon suivante : à chaque obstacle indexé par j est associé un indice de franchissement $\chi(j)$.
Si des obstacle existent entre le tronçon i et le tronçon i+1, alors la probabilité conditionnelle de sédentarisation des anguilles dans le i+1ème tronçon sachant qu'elle ne se sont pas sédentarisées dans les tronçons avals est (et toujours en ne considérant qu'un drain unique, sans embranchements) :

$$\begin{aligned} p_{i+1} &= \chi(j)\,\chi(j-1)\,\ldots\,\chi(1)\,P_{i+1}(1-P_i)(1-P_{i-1})...(1-P_0) \\ &= P_{i+1}\prod_{m=1}^{j}\chi(m)\prod_{k=0}^{i}(1-P_k) \end{aligned} \tag{2.17}$$

Dans la version actuelle du modèle, l'indice de franchissement est considéré comme étant identique pour l'ensemble des obstacles : $\chi(j) = \chi(j-1) = \ldots = \chi(1) = \chi$. Il fait néanmoins partie des paramètres optimisés dans le modèle. L'équation précédente devient donc simplement :

$$p_{i+1} = \chi^N P_{i+1}(1-P_i)(1-P_{i-1})...(1-P_0) = \chi^N P_{i+1}\prod_{k=0}^{i}(1-P_k) \tag{2.18}$$

Où N désigne le nombre d'obstacles présents entre le tronçon 0 et le tronçon i+1.

Prise en compte des confluences

Dans le cas d'une confluence à l'amont (embranchement du réseau), la probabilité de choisir le tronçon amont j est égale à la proportion relative de la surface du bassin versant j par rapport à la somme des surfaces des bassins amonts élevées à une certaine puissance δ :

$$\frac{(S_{BV}(j))^{\delta}}{\sum\limits_{k\in \text{amont}(i)} (S_{BV}(k))^{\delta}}$$

Spécificités de l'ajustement du modèle dans l'approche par propagation d'une gaussienne

Dans le cas du traitement de la diffusion par la propagation d'une onde gaussienne, l'ajustement du modèle se fait grâce à un algorithme de différentiation automatique

(en anglais AD, pour Automatic Differentiation). Un tel algorithme permet d'évaluer les dérivées d'une fonction d'intérêt pour un programme, dite fonction-objectif (par exemple une fonction à minimiser, ou dans notre cas, une vraisemblance à maximiser).Pour un développement détaillé des principes et des outils utilisés dans le cadre de la différentiation automatique, on pourra consulter (Rall, 1981).
Nous avons choisi d'utiliser l'outil ADMB (Automatic Differentiation Model Builder) (Fournier *et al.*, 2012), utilisable en C++ sous licence BSD (la licence BSD (Berkeley Software Distribution License) est une licence libre utilisée pour la distribution de logiciels. Elle permet de réutiliser tout ou une partie du logiciel sans restriction, qu'il soit intégré dans un logiciel libre ou propriétaire.).

2.2.2 Approche par matrices de transition

Cette approche propose de modéliser la probabilité pour une anguille de passer d'un tronçon à un tronçon adjacent au cours d'un pas de temps unitaire. La probabilité est une fonction des caractéristiques du tronçon (surface du tronçon, surface du bassin-versant amont, longueur du tronçon, etc...) et des nœuds entre les tronçons (barrage ou confluence). Les probabilités de mouvement vers un tronçon adjacent pour un pas de temps unitaire sont alors réunies dans une matrice, dite matrice de transition.

Principe général et concept

Il s'agit en fait d'une amélioration du modèle OMMER (Obstacle Mitigation Model for Eel in Rivers) (Lambert *et al.*, 2011).
Initialement, l'idée d'utiliser des matrices de transition dans la dynamique de population revient à P. H. Leslie en 1948 (Leslie, 1948). L'approche retenue était un modèle déterministe basé sur les distributions en âge des populations, qui prédisait la distribution résultante par classe d'âge à intervalles successifs. En 1965, L. P. Lefkovitch proposa une alternative en groupant les populations non plus par classe d'âge mais par stades du développement (Lefkovitch, 1965). Cela évitait notamment d'avoir besoin de connaître l'âge des individus et autorisait des différences dans les taux de développement puisqu'il n'y a pas de relation invariante entre taille et âge. La méthode matricielle a ensuite été appliquée à de nombreux objets d'étude en écologie, jusqu'à la dynamique de population des arbres.
De façon plus formelle, le concept de base est celui de matrice de transition[3]. Il s'agit d'une matrice utilisée pour décrire les transitions d'une chaîne de Markov[4]. Une telle matrice est carrée et chacun de ses coefficients est un nombre réel compris entre 0 et 1, et représentant une probabilité.
Reprenons le cas des anguilles qui colonisent un bassin. Soit $P_{i,j}$ la probabilité pour les anguilles de se déplacer d'un tronçon i vers un tronçon quelconque j du réseau au cours d'un pas de temps. La matrice de transition T est définie par la donnée des $i \times j$ coefficients tels que $T_{i,j}$ est le coefficient de la i^{e} ligne et de la j^{e} colonne de T. Soit :

$$T = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,j} & \cdots \\ p_{2,1} & p_{2,2} & \cdots & p_{2,j} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \ddots \\ p_{i,1} & p_{i,2} & \cdots & p_{i,j} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \ddots \end{pmatrix} \tag{2.19}$$

Étant donné que les anguilles ne peuvent se déplacer que dans un tronçon adjacent à celui dans lequel elles se trouvent à un instant donné, les coefficients $T_{i,j}$ sont nuls pour tous les tronçons j non adjacents au tronçon i. C'est la raison pour laquelle les matrices

[3]On trouve aussi d'autres dénominations : matrice de probabilité, matrice de substitution, matrice stochastique, ou encore matrice de Markov

[4]Une chaîne de Markov désigne de façon générale un processus de Markov à temps discret. En mathématiques, un processus de Markov est un processus stochastique possédant la propriété de Markov, c'est-à-dire que toute l'information utile pour la prédiction du futur est contenue dans l'état présent du processus.

que nous utilisons dans le modèle de diffusion par matrices sont dites "creuses" : elles contiennent en effet beaucoup de zéros.
Puisque la probabilité de déplacement des anguilles d'un tronçon i vers un autre tronçon adjacent (incluant aussi la probabilité qu'elles restent dans le même tronçon au cours du pas de temps) vaut 1, on a alors :

$$\sum_j T_{i,j} = 1 \ \ \forall\ j \text{ adjacent à i} \tag{2.20}$$

Formellement, on dit que la matrice est une matrice stochastique droite.

Probabilités de déplacement et sédentarisation

Le bassin versant est représenté sous forme d'un vecteur regroupant les effectifs par tronçon sur les M tronçons :
$N(t) = (N(t,1), N(t,2), \ldots, N(t,M)$
Les individus rentrant initialement à l'aval, on a :
$N(0) = (N_{tot},\ 0\ , \ldots,\ 0)$
On note T la matrice de transition décrivant les probabilités $\mathrm{p}_{i,j}$ de passer du tronçon de départ i à un tronçon d'arrivée j au cours d'un pas de temps unitaire. Cette matrice est creuse (remplie de zéro) puisque les individus ne peuvent se déplacer qu'entre tronçons adjacents.
Le problème revient donc essentiellement à déterminer le modèle d'état. Cela consiste à calculer la matrice T en déterminant les probabilités de passer d'un tronçon à un autre au cours d'un pas de temps unitaire. Les probabilités de passage sont décomposées en étapes successives :

- une probabilité de rester dans le tronçon ou d'essayer de bouger,
- une probabilité de bouger vers l'aval ou vers l'amont sachant que l'anguille change de tronçon,
- une probabilité de franchir le nœud dans la direction choisie,
- une probabilité de choisir un des deux tronçons possibles si j'ai franchi le nœud amont.

Ainsi, la probabilité de ne pas essayer de quitter le tronçon i est :

$$p_i = \frac{1}{1 + \exp^{-(\text{un terme de position relative dans le réseau})}}$$

La probabilité de choisir de partir vers l'amont est P_{AM}, celle de partir vers l'aval $(1 - P_{AM})$.
La probabilité de franchir un nœud vers l'amont est :

$$P_{n,AM}(i) = \begin{cases} 1 \text{ si confluence} \\ \phi_{n,AM}(i) \text{ si barrage} \end{cases}$$

La probabilité de franchir un nœud vers l'aval est :

$$P_{n,AV}(i) = 1$$

Dans le cas d'une confluence à l'amont (embranchement du réseau), la probabilité de choisir le tronçon amont j est égale à la proportion relative des surfaces de bassins versants amonts élevées à la puissance β :

$$\frac{(S_{BV}(j))^{\beta}}{\sum\limits_{k \in \text{amont}(i)} (S_{BV}(k))^{\beta}}$$

En résumé, les probabilités $P_{i\to j}$ sont alors :

$$P_{i\to j} = \begin{cases} 0 \text{ si } (j \notin \text{amont}(i) \mid j \notin \text{aval}(i)) \\ p_i + (1-p_i)[P_{AM}\,(1 - P_{n,AM}(i)) + (1 - P_{AM})] \text{ si } j = i \\ (1-p_i)(1-P_{AM}) \text{ si } j \in \text{aval}(i) \\ (1-p_i)P_{AM}P_{n,AM}(i)\dfrac{(S_{BV}(j))^{\beta}}{\sum\limits_{k\in\text{amont}(i)}(S_{BV}(k))^{\beta}} \text{ si } j \in \text{amont}(i) \end{cases}$$

Spécificités de l'ajustement du modèle pour l'approche par matrices de transition

L'ajustement du modèle pour l'approche par matrices de transition se fait grâce à l'algorithme d'optimisation numérique BOBYQA (Bound Optimization BY Quadratic Approximation) développé par Michael J. D. Powell pour résoudre des problèmes d'optimisation avec contraintes aux limites sans utiliser les dérivées de la fonction-objectif (fonction à minimiser), ce qui le classe dans la catégorie des algorithmes sans dérivées (Powell, 2009). C'est aussi un algorithme itératif à région de confiance, c'est-à-dire qu'il minimise une fonction en procédant par améliorations successives. Au point courant, BOBYQA effectue un déplacement obtenu en minimisant un modèle simple de la fonction (en l'occurrence un modèle quadratique généré par interpolation) sur une région de confiance. Le rayon de confiance (caractérisant la région du même nom) est ajusté de manière à faire décroître suffisamment la fonction à chaque itération.

2.2.3 Ajustement du modèle

Nous décrivons ci-après le principe de l'ajustement du modèle TABASCO, qui est identique dans le cas de l'approche matricielle et dans le cas de l'approche par propagation d'une gaussienne.
Dans un réseau dont la topologie et les caractéristiques des tronçons sont fixées, et connaissant l'effectif total N_{tot}, les composantes diffusives $\sigma(i)$ (qu'elles soient constantes ou qu'elles soient calculées à partir des caractéristiques du tronçon), les probabilités de se sédentariser ou de se déplacer p_i et P_{AM} (calculées au moins à partir de la longueur du tronçon), le coefficient aux confluences β, les probabilités de franchissement vers l'amont $\phi_{AM}(i)$[5] des différents obstacles (constants ou calculés à partir de l'indice de franchissabilité de Steinbach(Steinbach, 2006)), il est possible de calculer de manière récursive tous les effectifs.
Inversement, si l'on dispose d'un jeu d'observation de densités d'anguilles dans une sélection de tronçons, il est possible d'ajuster les paramètres N_{tot}, les $\sigma(i)$ ou p_i et P_{AM}, β, les $\phi_{AM}(i)$ et les $\tau_{AM}(i)$. En particulier, il est intéressant de noter que le paramètre d'intérêt pour la gestion, N_{tot}, est directement exprimé dans les équations, ce qui permet en principe d'en donner un intervalle de confiance. En effet, nous utilisons la méthode du maximum de vraisemblance. Or, l'estimateur du maximum de vraisemblance est asymptotiquement normal[6], on peut donc construire un intervalle de confiance C_n tel qu'il contienne le vrai paramètre avec une probabilité $(1-\alpha)$ (Wasserman, 2004) :

$$C_n = \left(\hat{\theta}_n - \Phi^{-1}(1-\alpha/2)\widehat{\sigma_{\hat{\theta}_n}},\ \hat{\theta}_n + \Phi^{-1}(1-\alpha/2)\widehat{\sigma_{\hat{\theta}_n}}\right)$$

[5]Pour le moment, nous considérons dans le modèle que tous les obstacles ont le même indice de franchissement. A terme, cependant, il est prévu d'améliorer cet aspect en tenant compte de franchissabilités variables en fonction de l'ouvrage.

[6]Un estimateur $\hat{\theta}_n$ d'un paramètre d'intérêt θ avec n observations est dit asymptotiquement normal si et seulement si : $n^{p/2}(\hat{\theta}_n - \theta)\xrightarrow{\mathcal{L}}\mathcal{N}(0,\Gamma)$, où p est la vitesse de convergence (convergence dite en $1/n^p$) et où Γ est la matrice de covariance (asymptotique). La convergence est une convergence en loi, et on rappelle qu'une suite $(X_n)_{n>=1}$ converge en loi vers X si, pour toute fonction φ continue bornée sur un espace métrique E, à valeurs dans $\mathbb{R}$, $\lim\limits_n \mathbb{E}[\varphi(X_n)] = \mathbb{E}[\varphi(X)]$.

Avec $\Phi^{-1}(1-\alpha/2)$ le quantile d'ordre $(1-\alpha/2)$ de la loi normale centrée réduite et $\widehat{\sigma_{\hat{\theta}_n}}$ l'écart-type estimé de $\hat{\theta}_n$. On a alors :

$$\mathbb{P}(\theta \in C_n) \xrightarrow[n\to+\infty]{} 1-\alpha$$

Néanmoins, une difficulté supplémentaire intervient dans l'approche matricielle, car les exposants des matrices de transition sont des entiers. On se retrouve donc avec un paramètre à valeurs discrètes, sans dérivée continue. Or, la matrice hessienne (matrice des dérivées secondes) est nécessaire pour le calcul de l'intervalle de confiance. En conséquence, il ne sera pas possible en pratique de calculer des intervalles de confiance dans l'approche matricielle, sauf si nous parvenons ultérieurement à implémenter l'usage d'exposants non-entiers de matrices (ce qui semble peu probable vu le niveau de complexité de cette tâche).
D'une manière générale, soit $N(i,\theta)$ où θ est le jeu de paramètres qui permet de calculer le nombre d'individus dans le tronçon i. On appellera $\Delta(i,\theta)$ la densité dans le tronçon i, telle que :

$$\Delta(i,\theta) = \frac{N(i,\theta)}{S(i)}$$

Où $S(i)$ est la surface en eau du tronçon i.

2.2.4 Fonction de vraisemblance

Soit q la probabilité de capturer au moins un individu dans un secteur de pêche électrique. Nous postulons pour le modèle que cette loi de probabilité suit une loi Gamma caractérisée par deux paramètres strictement positifs (un paramètre de forme α et un paramètre d'échelle, ou d'intensité, β). La fonction de densité de probabilité de la loi Gamma s'écrit, pour $x > 0$:

$$f(x\,;\alpha,\beta) = x^{\alpha-1}\frac{\beta^{\alpha}\,e^{-\beta x}}{\Gamma(\alpha)}$$

Où $\Gamma(x)$ est la fonction Gamma d'Euler.
La fonction de distribution cumulative associée est :

$$F(x\,;\alpha,\beta) = \frac{1}{\Gamma(\alpha)}\gamma(\alpha,\beta x)$$

Où $\gamma(\alpha,\beta x)$ est la fonction Gamma incomplète inférieure.
La probabilité de présence doit être nulle lorsque la surface prospectée est nulle ou bien lorsque la densité calculée est nulle. Cela est réalisé dans le modèle en prenant comme paramètres :

$$\alpha = \exp(\text{ratePresence}) \times S \times \Delta_{calc}$$
$$\beta = \exp(\text{shapePresence})$$

Où ratePresence et shapePresence sont deux paramètres optimisés, S est la surface prospectée et Δ_{calc} est la densité calculée. D'une manière générale, on écrira $q\left(D\left(i(o),\theta\right),\theta\right)$ cette probabilité d'avoir une observation o non nulle.
L'ajustement se fait par maximum de vraisemblance. La fonction de vraisemblance pour une observation o s'écrit en considérant soit qu'aucune anguille n'est capturée lors de l'opération, soit une densité d'anguilles non nulle. Dans ce deuxième cas, on considère que la densité suit une loi lognormale dont la moyenne correspond à la densité calculée dans le tronçon i(o) où a eu lieu l'observation :

$$L(o,\theta) = \begin{cases} 1 - q\left(D\left(i(o)\right),\theta\right) \text{ si } D_{obs}(o) = 0 \\ q\left(D\left(i(o)\right),\theta\right)\dfrac{1}{D_{obs}(o)\sqrt{2\pi}\sigma}\,\mathrm{e}^{-\dfrac{\left(log\left(D_{obs}(o)\right) - log\left(D\left(i(o)\right)\right)\right)^2}{2\sigma^2}} \\ \qquad\qquad\qquad\qquad\qquad\qquad \text{si } D_{obs}(o) > 0 \end{cases}$$

La fonction de vraisemblance pour l'ensemble des observations s'écrit donc :

$$L(\theta_\Delta, \theta_{LN}, \sigma) = \prod_{o \in (D_{obs}(o)=0)} \left[1 - q\left(D\left(i(o), \theta\right), \theta\right)\right]$$
$$\prod_{o \in (D_{obs}(o)>0)} q\left(D\left(i(o), \theta\right), \theta\right) \left(\frac{1}{D_{obs}(o)\sqrt{2\pi}\sigma} \, \mathrm{e}^{-\frac{\left(log\left(D_{obs}(o)\right) - log\left(D\left(i(o)\right)\right)\right)^2}{2\sigma^2}} \right)$$

Le modèle TABASCO est ensuite ajusté sur les observations en minimisant l'opposé du logarithme de la fonction de vraisemblance.

2.2.5 Paramétrage du modèle

Le tableau 2.1 synthétise tous les paramètres utilisés dans le modèle TABASCO (qu'ils soient optimisés ou non).

2.2.6 Calibration du modèle

Nous disposons des données de pêches électriques réalisées entre les années 1990 et 2009 incluses (soit 20 ans de données).
Dans l'état actuel du code, le modèle basé sur la propagation d'une onde gaussienne est calibré sur l'ensemble des 60 bassins versants dont la liste est fournie en annexe de ce rapport. Ce nombre de bassins correspond à la quasi-totalité des bassins versants dont les exutoires sont situés le long des côtes de la Manche et sur tout l'arc Atlantique. Seuls les plus petits bassins ont été négligés sur ces littoraux. Quelques bassins de Méditerranée ont également été ajoutés.
Nous disposons d'indicateurs numériques pour juger de l'efficacité de l'optimisation et de la rapidité de la convergence : la valeur de la fonction objectif lors de la minimisation (qui est l'opposée de la fonction de vraisemblance), la valeur du gradient de chaque paramètre fournie par l'algorithme d'optimisation, le nombre d'itérations de l'algorithme, et le nombre d'évaluations de la fonction-objectif. Une valeur trop élevée du gradient indique par exemple que l'algorithme ne parvient pas à trouver un minimum local pour le paramètre considéré et ne converge donc pas. La meilleure estimation des paramètres se produit donc lorsque la fonction objectif est minimale et que les coefficients de la matrice des dérivées des paramètres sont minimaux en valeur absolue.
Il est aussi important de rappeler que comme tous les algorithmes qui utilisent la méthode de Quasi-Newton, ADMB ne trouve qu'un minimum local, qui n'est pas nécessairement le minimum absolu de la fonction objectif (mais qui peut l'être). Ainsi, il est nécessaire de vérifier la sensibilité du point de départ de l'optimisation afin de garantir que l'ensemble des paramètres optimisés soit effectivement le meilleur possible.

Définition du paramètre	**Dénomination dans le modèle**	**Symbole usuel**	**Remarques**
Logarithme de la densité surfacique moyenne dans le bassin	logMeanDensity	$\log(D_0)$	optimisé dans le modèle
Taux d'advection	lambdaAdvection	λ	optimisé dans le modèle, en km
Paramètre d'échelle fonction Gamma	shapePresence	$\log(\beta)$	optimisé dans le modèle
Paramètre d'intensité fonction Gamma	ratePresence	$\log\left(\frac{\alpha}{S \times \Delta_{calc}}\right)$	optimisé dans le modèle
Logarithme de l'écart-type de la distribution en densité dans le tronçon	logsigma	$\log(\sigma)$	optimisé dans le modèle
Paramètre 1 pour le calcul de l'écart-type	mused/muDiffusive	-	optimisé dans le modèle, en km
Paramètre 2 pour le calcul de l'écart-type	alpha1sed/ alpha1Diffusive	-	optimisé dans le modèle
Paramètre 3 pour le calcul de l'écart-type	alpha2sed/ alpha2Diffusive	-	optimisé dans le modèle
Pondération des flux aux confluences	d	-	optimisé dans le modèle
Franchissabilité des obstacles	passabilities0	χ	optimisé dans le modèle, facteur entre 0 et 1
Coefficient de diffusion	-	D	en km^2.années^{-1}
Écart-type de la gaussienne	sigma	σ	en km
Distance du tronçon à sa source	Dsource	D_{source}	en km
Distance du tronçon à la mer	Doutlet	D_{out}	en km
Longueur du tronçon	length	l	en km
Longueur totale du drain	-	L_{tot}	en km

TABLE 2.1 : Tableau récapitulatif des paramètres utilisés dans le modèle TABASCO.

Chapitre 3

Outils informatiques pour l'implémentation du modèle

3.1 Langage et normes de programmation

Le modèle TABASCO a été intégralement codé en C++. Le C++ est un langage de programmation compilé, libre d'utilisation, permettant la programmation sous de multiples paradigmes comme la programmation procédurale, la programmation orientée objet et la programmation générique. Ses principaux atouts sont ses fonctionnalités multiples (il dérive d'une amélioration du langage C), et sa portabilité, puisqu'il est compatible avec une grande variété de plateformes matérielles et de systèmes d'exploitation.

3.2 Logiciels et interfaces

3.2.1 Environnement de développement

L'environnement de développement (dont nous utiliserons par la suite l'acronyme en anglais : IDE, pour *Integrated Development Environment*) choisi est le logiciel gratuit, open-source et multi-plateformes Code::Blocks, compatible avec les langages de programmation C, C++, et Fortran.

3.2.2 Base de données

La gestion de la base de données est effectuée grâce à l'outil PostgreSQL. Il s'agit d'un système de gestion de base de données relationnelle et objet (SGBDRO). C'est un outil libre disponible selon les termes d'une licence de type BSD (voir la définition des licences BSD dans la section traitant de l'ajustement).

Interface avec l'utilisateur

Nous utilisons également deux interfaces utilisateur, détaillées ci-dessous.

- psql, qui est une interface en ligne de commande permettant la saisie de requêtes SQL[1], directement ou par l'utilisation de procédures stockées.
- pgAdmin, qui est un outil d'administration graphique pour PostgreSQL, distribué selon les termes de la licence PostgreSQL.

[1] SQL (Structured Query Language, ou langage de requête structurée en français) est un langage informatique normalisé servant à exploiter des bases de données relationnelles. Créé en 1974, et normalisé depuis 1986, ce langage est reconnu par la grande majorité des systèmes de gestion de bases de données relationnelles du marché.

Interface avec le SIG

Le module spatial PostGIS nous permet de travailler en lien avec un système d'information géographique (SIG) sur des données spatialisées ou géoréférencées. C'est ce qui confère à PostgreSQL le statut de SGDBRO spatial.
La version de PostgreSQL que nous utilisons est la version 9.3, celle de PostGIS est la version 2.1.

Système d'Information Géographique

Le SIG que nous utilisons est le logiciel libre et multi-plateformes QGIS. Il gère, via la bibliothèque GDAL3, les formats d'images matricielles (*raster*) et vectorielles, ainsi que les bases de données. QGIS fait partie des projets de la Fondation Open Source Geospatial.
Ses caractéristiques principales sont :

1. La gestion de PostGIS, l'extension spatiale de PostgreSQL.
2. La prise en charge d'un grand nombre de formats de données vectorielles (Shapefile, les couvertures ArcInfo, Mapinfo, GRASS GIS, etc.)
3. La prise en charge d'un nombre important de formats de couches matricielles (GRASS GIS, GeoTIFF, TIFF, JPG, etc.)

3.2.3 Interface du langage avec la base de données

La bibliothèque libpqxx est l'outil nous permettant de faire le lien entre le langage C++ et le logiciel PostgreSQL. Il s'agit de l'interface de programmation (dont nous utiliserons par la suite l'acronyme anglais : API, pour *Application Programming Interface*) standard entre des programmes codés en C++ et des bases de données exploitées avec PostgreSQL. Le code source de libpqxx est lui-aussi disponible sous licence BSD. Un tutoriel détaillant la procédure - assez complexe - d'installation de libpqxx sous Windows est par ailleurs fourni en annexe du présent rapport. L'installation de cette bibliothèque sous Windows nous a en effet posé quelques difficultés lors de la préparation de nos machines en vue de la programmation du modèle TABASCO (notamment car cette bibliothèque est initialement prévue pour fonctionner sous systèmes UNIX).

3.2.4 Bibliothèques pour le calcul et l'algorithmique

Implémentation de graphes

Comme nous l'avons vu précédemment dans ce rapport, la formalisation du réseau hydrographique est réalisé à l'aide de la bibliothèque graphique Boost (en anglais BGL, Boost Graph Library) qui nous permet d'utiliser des outils pour la gestion de graphes. Boost est en fait un ensemble de bibliothèques logicielles libres écrites en C++, qui vise à remplacer la Bibliothèque Standard du C++. L'écriture des modules au sein de cet ensemble est soumise à un comité de lecture. La plupart du code est distribué selon les termes de la licence logicielle Boost, laquelle autorise autant son intégration dans les logiciels libres que propriétaires. La plupart des fondateurs de Boost se trouvent dans le comité du standard C++ et plusieurs de ses bibliothèques ont été acceptées pour faire partie de la base de travail de la norme C++11. Un tutoriel d'installation des bibliothèques non pré-compilées de Boost est fourni en annexe du présent rapport.

Optimisation pour l'approche gaussienne

Dans ce cas, l'optimisation est faite grâce à ADMB (AD Model Builder). C'est un paquetage logiciel, très utilisé notamment pour le développement de modèles statistiques non-linéaires. ADMB est construit autour de la bibliothèque AUTODIF, une extension du langage C++ qui implémente la différentiation automatique en mode inverse[2].

[2]Le mode inverse de différentiation, ou méthode de l'état adjoint, est une approche dans laquelle il n'y a pas besoin de calculer la dérivée de la fonction implicite (qui permet de réécrire le problème initial

Optimisation pour l'approche matricielle

Nous recourons aussi à la bibliothèque gratuite et open-source NLopt pour l'optimisation non linéaire. Contrairement à l'approche par propagation d'une gaussienne, son utilisation a été rendue nécessaire pour l'approche matricielle car dans ce modèle l'un des paramètres à optimiser est à valeurs discrètes (l'exposant de la matrice stochastique), ce qui rend impossible l'utilisation d'ADMB. D'autre part, la capacité de mémoire qui était requise pour les calculs avec ADMB était de toute façon trop importante. NLopt fournit une interface commune pour différentes routines d'optimisation disponibles en ligne, ainsi que des implémentations inédites de plusieurs autres algorithmes. Ses fonctionnalités incluent :

- Une compatibilité avec les langages C, C++, Fortran, Matlab ou GNU Octave, Python, GNU Guile, Julia, GNU R, Lua, et OCaml.
- Une interface commune pour de nombreux algorithmes. Il est possible d'essayer un algorithme différent en changeant simplement un paramètre.
- Un fonctionnement pour l'optimisation large échelle (algorithme avec un très grand nombre de paramètres et de contraintes).
- Une compatibilité avec des algorithmes d'optimisation locale ou globale.
- Une compatibilité avec des algorithmes n'utilisant que les valeurs de la fonction-objectif (sans dérivées) et des algorithmes appliquant des gradients fournis par l'utilisateur.
- Une compatibilité avec des algorithmes destinés à l'optimisation sans contraintes, à l'optimisation avec contraintes aux bornes, et avec contraintes d'égalité/inégalité non linéaires.

L'algorithme BOBYQA, que nous avons déjà mentionné dans ce rapport, et qui assure l'optimisation des paramètres de l'approche matricielle, est implémenté dans NLopt.

Calcul numérique additionnel

Nous nous servons aussi d'Eigen, une bibliothèque C++ de haut niveau contenant des modèles de déclarations (template headers) pour l'algèbre linéaire, le calcul vectoriel et matriciel, ou la résolution numérique de problèmes. C'est notamment cette bibliothèque qui nous permet de gérer efficacement le calcul avec des matrices creuses dans l'approche matricielle.

Parallélisation du code

Par ailleurs, en prévision d'une réduction du temps de calcul si celui-ci s'avérait trop important lors d'une modélisation de la diffusion des anguilles à l'échelle de la France entière, l'interface de programmation OpenMP (Open Multi-Processing) a été intégrée dans le programme pour paralléliser certaines portions du code. OpenMP est une interface de programmation pour le calcul parallèle sur architecture à mémoire partagée. Cette API est supportée sur de nombreuses plateformes, incluant Unix et Windows, pour les langages de programmation C, C++ et Fortran. Il se présente sous la forme d'un ensemble de directives, d'une bibliothèque logicielle et de variables d'environnement. OpenMP est portable et dimensionnable. Il permet de développer rapidement des applications parallèles à petite granularité en restant proche du code séquentiel. Une programmation parallèle hybride peut être réalisée par exemple en utilisant à la fois OpenMP et MPI. OpenMP est une implémentation du multithreading (multi-tâches),

d'optimisation avec contraintes en un problème d'optimisation sans contraintes grâce notamment au théorème des fonctions implicites).

une méthode de parallélisation dans laquelle un thread[3] maître (une série d'instructions exécutées de façon séquentielle) se divise en un nombre donné de threads esclaves avec lesquels le système partage les tâches. Les threads fonctionnent donc en parallèle, et l'environnement d'exécution alloue les threads aux différents processeurs en fonction de leur disponibilité.

[3]Un thread, ou fil (d'exécution), ou tâche (terme et définition normalisés par ISO/CEI 2382-7 :2000, mais d'autres appellations sont connues : processus léger, unité de traitement, unité d'exécution, fil d'instruction, processus allégé, exétron), est similaire à un processus car tous deux représentent l'exécution d'un ensemble d'instructions du langage machine d'un processeur. Du point de vue de l'utilisateur, ces exécutions semblent se dérouler en parallèle. Toutefois, là où chaque processus possède sa propre mémoire virtuelle, les threads d'un même processus se partagent sa mémoire virtuelle. Par contre, tous les threads possèdent leur propre pile d'appel.

Chapitre 4

Exploitation des sorties du modèle

4.1 Présentation des sorties du modèle

4.1.1 Liste et nature des sorties

Nous listons ci-après les variables d'intérêt pour le modèle, en indiquant pour chacune d'elles sa signification et l'unité dans laquelle elle est déterminée.

1. Nombre d'anguilles par tronçon du réseau hydrographique.
2. Nombre d'anguilles, en l'absence d'obstacles, par tronçon du réseau hydrographique.
3. Densité surfacique d'anguilles par tronçon du réseau (nombre d'anguilles par unité de surface, donc individus par km^2).
4. Densité surfacique d'anguilles, en l'absence d'obstacles, par tronçon du réseau (nombre d'anguilles par unité de surface, donc individus par km^2).
5. La matrice de transition (dans le cas de l'approche matricielle seulement). Les coefficients sont des probabilités de transition (nombre compris entre 0 et 1).

D'autres variables ne sont utilisées que pour l'exploration et la calibration du modèle, et ne présentent pas d'intérêt particulier en tant que résultats de la simulation :

1. Probabilité absolue de sédentarisation dans chaque tronçon (il s'agit d'une probabilité, donc un nombre sans unité et compris entre 0 et 1).
2. Bornes d'intégration de la fonction gaussienne dans chaque tronçon (dépendantes de la distance à la mer, de la diffusivité et de la longueur du tronçon ; sans unité).

4.1.2 Sorties graphiques

Comme nous l'avons déjà vu, la totalité des sorties peut être affichée sur le RHT, ce qui permet d'avoir une vision globale (France entière), à l'échelle d'un bassin versant ou même un zoom sur une région ou un ensemble de tronçons particuliers. Un script utilisable par le logiciel R a été mis au point pour récupérer les sorties intéressantes depuis la base de données gérée par PostgreSQL, les afficher graphiquement puis les sauvegarder sous forme de fichier png (ou jpeg, ou pdf, au choix de l'utilisateur). Le script est disponible en annexe à la page 48. A titre d'information, cet affichage est réalisé en moins de 2 minutes sur les 60 bassins versants actuellement dans la simulation.
Une autre possibilité est d'utiliser l'éditeur de cartes de QGIS, qui fournit tous les outils utiles pour l'affichage et la sauvegarde des sorties, si ce n'est que le processus n'est pas automatisé (il faut relancer l'éditeur à chaque fois que l'on souhaite sauvegarder des résultats).

4.1.3 Machine de référence

Dans tout ce rapport, les performances données pour le modèle sont obtenues sur une machine de référence, équipée d'un processeur Intel Core i7-4900MQ cadencé à 2,7 GHz et de 16 Go de RAM.

4.2 Résultats de la calibration année par année

Les résultats présentés ici sont issus de l'approche par propagation d'une gaussienne du modèle TABASCO. Les résultats de l'approche matricielle seront intégrés dans le prochain rapport.
Il est important de noter que ces résultats sont très provisoires, et font partie intégrante du processus de développement du modèle. Néanmoins, il se présagent en rien les résultats définitifs du modèle, qui seront présentés ultérieurement, lorsque toute la phase de calibration sera terminée.

4.2.1 Résultats de l'optimisation des paramètres

Le modèle a tout d'abord été calibré année par année. L'optimisation des paramètres s'effectue donc dans ce cas sur les données de pêches électriques d'une seule année, parmi toutes celles dont nous disposons (1990 à 2009). Nous récapitulons dans le tableau 4.1 les résultats obtenus pour l'optimisation des paramètres du modèle avec une calibration année par année.
Dans le tableau 4.1, le paramètre d'intérêt pour la gestion du stock d'anguilles n'est pas directement exprimé, puisque le paramètre qui est optimisé dans le modèle est le logarithme népérien de la densité moyenne surfacique d'anguilles dans la zone couverte par la simulation ($D0$, qui s'exprime en nombre d'anguilles par km^2). Il est très important de noter que cette densité surfacique est à entendre comme le nombre moyen d'anguilles par unité de surface sur toute la superficie prise en compte dans la simulation (donc dans ce cas sur la superficie totale des 60 bassins versants). De façon différente, les densités surfaciques d'anguilles présentées dans chaque tronçon (notamment sur les cartes) correspondent au nombre d'anguilles par unité de surface sur la seule surface en eau du tronçon.
Afin de rendre le paramètre d'intérêt $N0$ plus explicite, nous avons calculé sa valeur dans le tableau 4.2 pour chaque année entre 1990 et 2008, sur les 60 bassins versants intégrés dans la simulation, et en extrapolant sur l'ensemble de la France métropolitaine à partir des superficies respectives.

Année ↓	**Durée (s)**	**Itér./ Éval.**	**-log(V)**	**log(D0)**	μ	$\alpha 1$	$\alpha 2$	λ	**d**	χ	**rate**	**shape**	**logσ**
Range →	-	-	-	[0 ;15]	[-17 ;-20]	[8 ;20]	[0 ;10]	[0 ;1]	[1 ;5]	[0 ;1]	[-10^3 ;10]	[-10^3 ;10]	[0 ;100]
1990	117.16	58/81	1482.07	8.50539	-19.75	8.72031	4.03131	5×10^{-4}	1	0.1882	-5.07	-1.18	0.7995
1991	86.31	4/13	2065.01	14.138	-18.51	10.5596	1.34008	0.1005	1	0.5077	-12.56	-2.49	2.27885
1992	115.37	44/75	1645.53	8.25776	-18.98	8	5.18711	6×10^{-4}	1.00005	0.1297	-5.46	-1.57	0.8180
1993	129.10	38/103	1329.9	8.07422	-20	8.73691	2.7296	2×10^{-7}	1	0.1416	-5.03	-1.33	0.6659
1994	89.58	3/10	1431.15	12.0607	-18.51	13.516	0.933	0.1003	1	0.5038	-5.80	-1.74	1.78753
1995	137.75	57/107	1731.83	7.77432	-20	8.22731	2.6376	0.00134	1.0011	0.1482	-4.78	-1.25	0.9016
1996	121.08	25/68	1763.11	8.9853	-17	8.92855	1.99008	0.00247	1	0.1368	-5.49	-1.42	0.8649
1997	112.08	22/58	1964.02	9.38987	-17	9.03425	2.1207	0.9999	1	0.1659	-5.63	-1.42	0.9596
1998	123.08	9/91	1934.88	9.27016	-17.38	10.1103	0.92045	0.09211	1	0.2339	-14.73	-2.355	1.4977
1999	87.03	2/4	2109.87	10.9761	-18.5	15.0135	1.01948	0.1	1	0.50001	-14.38	-3.47	1.73381
2000	89.99	7/19	2260.37	12.1644	-17.20	11.2146	1.21807	0.09126	1	0.02246	-5.367	-1.993	1.15573
2001	109.80	30/56	1961.47	8.34582	-17.01	8.32234	2.36263	0.0014	1.00094	0.155	-4.84	-1.26	0.8619
2002	119.08	45/76	1990.46	8.04077	-17.00	8.05816	2.84962	0.99999	1	0.18402	-4.59	-1.25	0.94177
2003	95.55	6/22	1729.16	7.82523	-18.04	8.71077	1.42807	0.12789	1	0.45222	-15.58	-2.27	1.54427
2004	105.16	8/39	1812.08	6.27639	-18.68	8.0508	2.20321	0.0921	1	0.0281	-3.965	-1.242	1.36415
2005	86.14	2/4	2373.25	10.9782	-18.5	15.0077	1.00237	0.1	1	0.49984	-13.56	-3.259	2.53077
2006	95.89	5/15	2146.31	12.2022	-18.52	11.3097	0.92559	0.09954	1	0.50126	-17.58	-2.666	1.65793
2007	100.21	51/86	1385.5	7.9932	-17	8	4.86	0.99999	1	0.14845	-5.545	-1.066	0.6693
2008	119.33	33/69	1208.87	7.12786	-17.00	8.00025	3.85536	2×10^{-7}	1	0.33773	-6.452	-1.196	0.73120

TABLE 4.1 : Tableau récapitulatif des résultats de l'optimisation des paramètres du modèle TABASCO année par année.

Année	log(D0)	N0 (60 BV)	N0 (France)
1990	8.50539	$1{,}92.10^{9}$	$2{,}73.10^{9}$
1991	14.1380	$5{,}37.10^{11}$	$7{,}63.10^{11}$
1992	8.25776	$1{,}50.10^{9}$	$2{,}13.10^{9}$
1993	8.07422	$1{,}25.10^{9}$	$1{,}78.10^{9}$
1994	12.0607	$6{,}72.10^{10}$	$9{,}55.10^{10}$
1995	7.77432	$9{,}24.10^{8}$	$1{,}31.10^{9}$
1996	8.98530	$3{,}10.10^{9}$	$4{,}40.10^{9}$
1997	9.38987	$4{,}65.10^{9}$	$6{,}61.10^{9}$
1998	9.27016	$4{,}13.10^{9}$	$5{,}87.10^{9}$
1999	10.9761	$2{,}27.10^{10}$	$3{,}22.10^{10}$
2000	12.1644	$7{,}46.10^{10}$	$1{,}06.10^{11}$
2001	8.34582	$1{,}64.10^{9}$	$2{,}33.10^{9}$
2002	8.04077	$1{,}21.10^{9}$	$1{,}72.10^{9}$
2003	7.82523	$9{,}73.10^{8}$	$1{,}38.10^{9}$
2004	6.27639	$2{,}07.10^{8}$	$2{,}94.10^{8}$
2005	10.9782	$2{,}28.10^{10}$	$3{,}24.10^{10}$
2006	12.2022	$7{,}74.10^{10}$	$1{,}10.10^{11}$
2007	7.99320	$1{,}15.10^{9}$	$1{,}63.10^{9}$
2008	7.12786	$4{,}84.10^{8}$	$6{,}88.10^{8}$

TABLE 4.2 : Tableau récapitulatif des résultats de l'optimisation du paramètre d'intérêt N0 année par année.

4.2.2 Commentaires sur l'optimisation des paramètres

Il est difficile de tirer des conclusions solides à l'issue de cette étape tant la gamme de variation du paramètre d'intérêt $N0$ est grande (trois ordres de grandeur).
Il est évident, à la lecture du tableau 4.2, que la phase de calibration du modèle demeure pour l'instant trop fragile. En effet, les valeurs de $N0$ varient sur trois ordres de grandeur entre 1990 et 2008, ce qui est énorme et ne semble de plus pas suivre de loi de variation monotone. Cela dénote un mauvais fonctionnement de l'étape d'optimisation, dont les sources peuvent être multiples (plage de variation des paramètres irréalistes au vu des caractéristiques écologiques et/ou topologiques des bassins versants, fonction de calcul de la dispersion des anguilles à revoir, etc:.).
Cependant, on notera que si l'on ne sélectionne que les années pour lesquelles l'algorithme d'optimisation a convergé de façon appropriée (par exemple en imposant une logvraisemblance supérieure à -2000 et un nombre d'évaluations de la fonction-objectif supérieur à 20[1]), on obtient des valeurs de $N0$ plus resserrées variant entre $4{,}84.10^{8}$ et $1{,}92.10^{9}$ pour les 60 bassins.
A titre de comparaison, les données fournies dans le rapport de mise en œuvre du Plan de Gestion Anguille de la France de juin 2012 ((Anonyme, 2012)) indiquent un échappement annuel pour le bassin versant de la Loire de 150000 anguilles en 2008/2009 (suivi du Muséum National d'Histoire Naturelle). En appliquant le taux de conversion

[1] Ces critères sont totalement empiriques.

de 5 % entre stock d'anguilles jaunes et échappement d'anguilles argentées, évoqué dans ce même rapport, on obtient un $N0$ de 3 millions d'anguilles pour la Loire. Soit, en extrapolant à partir de la superficie relative du bassin versant de la Loire (118094 km^2) par rapport à la superficie totale de la France métropolitaine, un $N0$ de 14 millions d'anguilles jaunes pour la France entière. Nous avons donc un désaccord d'un facteur 50 avec l'estimation faite par TABASCO.
Si l'on s'intéresse aux valeurs estimées par le modèle EDA2.1 ((Jouanin *et al.*, 2012b)), on a 2.27 millions d'anguilles argentées qui ont quitté la France en 2009, soit, toujours en appliquant un taux de conversion de 5 %, un stock de 45,4 millions d'anguilles jaunes en 2008. Le désaccord entre les prédictions d'EDA et de TABASCO est donc cette fois-ci dans un rapport de 1 à 15.
Un certain nombre de remarques peuvent également être formulées à partir du tableau récapitulatif 4.1.
Précisons pour commencer que l'année 2009 ne figure pas dans le tableau car un problème dans les données de cette année-là empêche la convergence de l'algorithme d'optimisation. Ce problème sera traité ultérieurement, mais ne remet de toute façon pas en cause le fonctionnement du modèle.
Les deux colonnes "Durée" et "Itérations/Évaluations" nous donnent respectivement des indications sur le temps de simulation (toutes étapes comprises, exprimé en secondes) et sur la façon dont a convergé l'algorithme d'optimisation (en terme de nombre d'itérations et de nombre d'évaluations de la fonction-objectif).
Un critère général de convergence du modèle est très difficile à apprécier. Dans l'approche par propagation d'une gaussienne de TABASCO, nous utilisons deux critères : la valeur absolue maximale des dérivées des paramètres à optimiser (vecteur des gradients) comme critère d'arrêt principal, et le nombre maximal d'évaluations de la fonction objectif lorsque l'algorithme ne progresse plus en tant que critère d'arrêt secondaire. Lorsque l'on descend en-dessous d'une certaine valeur seuil des gradients, l'algorithme s'arrête. Néanmoins, ce seuil d'arrêt est à l'appréciation de l'utilisateur, et il n'existe pas à notre connaissance de valeur adaptée en général : pour un certain jeu de données, soit l'algorithme converge au bout de N itérations et trouve un minimum local avec un vecteur gradient donné, soit il ne converge pas et s'arrête de lui-même au bout d'une ou deux itérations. Ce comportement dépend évidemment très fortement des données de calibration, mais aussi des plages de variation autorisées pour les paramètres, ainsi que de la valeur de départ de chacun des paramètres.
Seule la calibration sur l'année 2007 a permis à l'algorithme de s'arrêter avec le critère d'arrêt principal que nous avions choisi (gradient inférieur à 10^{-2} en valeur absolue). Dans tous les autres cas, l'algorithme s'est arrêté après avoir convergé vers un minimum local puis avoir effectué 50 évaluations de la fonction objectif sans amélioration du résultat.
La meilleure logvraisemblance est obtenue pour l'année 2008, la moins bonne pour l'année 2005, avec un facteur 2 d'écart, ce qui représente une énorme variation de la vraisemblance et qui montre la forte variabilité des résultats en fonction des données annuelles de pêches électriques.
Nous pouvons donner un indicateur de la qualité du modèle (qui pourra ainsi ultérieurement être comparé à d'autres modèles). Nous utiliserons l'AIC (*Akaike Information Criterion*), qui se calcule comme suit :

$$AIC = 2k - 2\ln(L)$$

Où k est le nombre de paramètres à estimer du modèle, et L est le maximum de la fonction de vraisemblance du modèle.
Dans notre cas, nous estimons 10 paramètres, ce qui donne un AIC compris entre 2437,74 et 4766,5.
Les trois paramètres μ, $\alpha 1$ et $\alpha 2$ (sans unité) interviennent pour affiner la fonction de calcul de l'écart-type de la gaussienne, mais ne sont pas des paramètres écologiques au sens strict du terme (en tout cas, ils ne sont pas directement reliés à la quantification d'un paramètre écologique).
Le taux d'advection λ (exprimé en kilomètres dans le modèle) reste très faible, voire nul, quelle que soit l'année considérée. Il oscille entre 0 et 130 mètres.

Le paramètre d (sans unité), qui évalue l'importance de la grosseur d'un drain par rapport à un autre lors d'une confluence semblerait indiquer que les anguilles ne privilégient pas le plus important des deux drains (valeur égale à 1), ce qui est très peu probable en réalité. Ce point sera donc à étudier en détail durant les prochains mois.
L'indice de franchissement des obstacles χ (sans unité et compris entre 0 et 1) est lui aussi très variable selon les années considérées, et semble prendre parfois des valeurs irréalistes, même s'il reste souvent évalué entre 10 et 50 %, ce qui n'est pas totalement absurde. Il est ainsi estimée à 2 % seulement en 2000, et atteint presque 51 % en 1991.
Les deux paramètres *rate* et *shape* (sans unité) interviennent respectivement dans le calcul du paramètre d'échelle et du paramètre de forme de la fonction gamma pour la détermination de la probabilité de présence. On peut diviser les résultats de la calibration en deux groupes pour ces deux paramètres : dans un premier cas, on a un paramètre *rate* centré autour de $-5,25$ et un paramètre *shape* centré autour de $-1,38$, et dans le second cas un paramètre *rate* autour de $-14,7$ et un paramètre *shape* autour de $-2,75$. On notera également que ce second groupe semble être la conséquence de mauvaises convergences de l'algorithme et donc d'une mauvaise optimisation (voir les valeurs correspondantes du nombre d'itérations, du nombre d'évaluations de la fonction-objectif, et de la logvraisemblance). Nous considérons donc, à ce stade, que les valeurs moyennes qui ressortent de la calibration année par année pour ces deux paramètres de la fonction gamma sont les premières citées ci-dessus ($-5,25$ et $-1,38$). Ces valeurs sont par ailleurs bien regroupées (écarts-types respectifs de 0.41 et de $0,24$). Nous donnons la forme de la fonction de densité de probabilité de la distribution gamma avec ces deux paramètres sur la figure 4.1, et celle de la fonction cumulative de la distribution gamma sur la figure 4.2.
Enfin, le logarithme de l'écart-type de la distribution en densité dans un tronçon $\log \sigma$

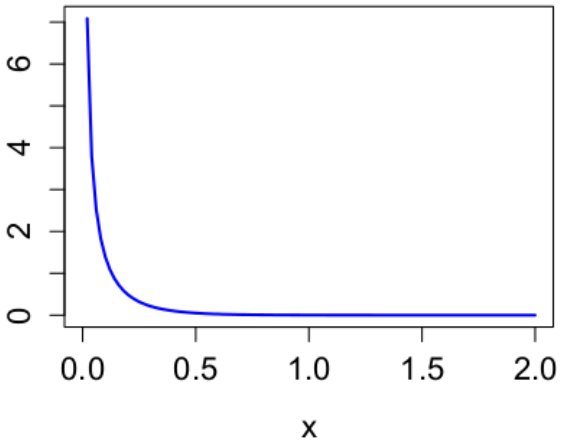

FIGURE 4.1 : Fonction de densité de probabilité de la distribution gamma, pour une surface prospectée de 1 km^2 et une densité calculée de 1000 anguilles par km^2, et avec les paramètres moyens issus de la calibration année par année.

est estimé à une valeur quasi-constante et proche de 1. Ses valeurs estimées restent bien regroupées alors même que sa plage de variation était très grande (de 0 à 100).

4.2.3 Évaluation de la qualité de la simulation

Nous proposons maintenant un ensemble de figures destinées à évaluer la qualité du modèle en fonction de divers paramètres.
Nous nous intéressons tout d'abord aux écarts algébriques obtenus entre les densités observées lors des pêches électriques et les densités prédites par le modèle. L'écart peut être positif (sous-estimation du modèle) ou négatif (sur-estimation du modèle), mais dans tous les cas il doit être centré sur 0 pour que la prédiction du modèle soit acceptable. Nous avons donc tracé plusieurs diagrammes en boîtes de Tukey afin de quantifier cet écart algébrique en fonction de différentes classes de distances à la mer

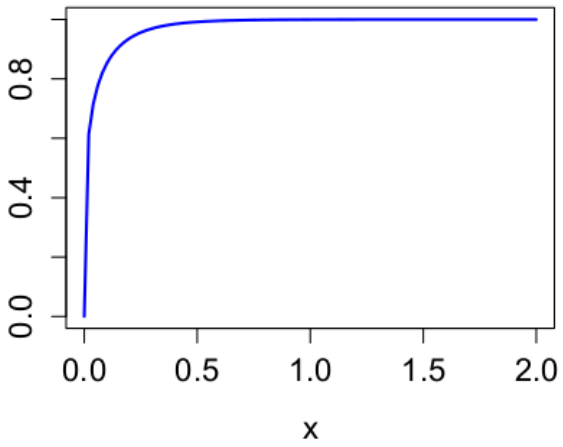

FIGURE 4.2 : Fonction de probabilité cumulative de la distribution gamma, pour une surface prospectée de 1 km^2 et une densité calculée de 1000 anguilles par km^2, et avec les paramètres moyens issus de la calibration année par année.

des tronçons (figure 4.3), en fonction du rang de Strahler des tronçons (figure 4.4) et en fonction des années (figure 4.6).
La figure 4.3 permet de tirer deux conclusions. La première est que le modèle surestime globalement les données (l'écart est globalement négatif), et la seconde est que cette surestimation augmente lorsque l'on se rapproche de la mer. Cette dernière remarque est en fait logique puisque les données de pêches électriques dont on dispose sont essentiellement situées à plus de 250 ou 300 km de la mer. Ainsi la calibration du modèle se fait correctement (par construction) sur toutes les zones pour lesquelles les données sont abondantes, mais le modèle surestime nettement les densités dans les régions proches des exutoires, pour lesquelles les données sont très peu nombreuses voire absentes (en particulier les 200 premiers kilomètres en partant de la mer, qui correspondent à la classe 0 sur la figure).

La figure 4.4 est une autre façon d'évaluer la simulation, mais les observations que l'on peut en tirer sont proches de celles obtenues grâce aux diagrammes représentés en fonction de l'éloignement à la mer. On retrouve une surestimation globale des densités d'anguilles, d'autant plus que le rang de Strahler du tronçon considéré est grand. On aurait tendance, trop rapidement, à dire que les rangs de Strahler les plus élevés correspondent aux tronçons les plus proche de l'exutoire, et que les rangs les moins élevés sont loin de la mer. En réalité, la situation est plus complexe qu'il n'y paraît au premier abord (figure 4.5). Certains petits drains, avec des rangs de Strahler de 1 par exemple, affluent vers les drains principaux même à de faibles distances de l'exutoire. A l'inverse, les drains principaux, qui reçoivent de nombreux affluents depuis leur source, affichent des rangs qui peuvent devenir élevés même à des distances importantes de la mer (par exemple pour le bassin Garonne Dordogne et les données de 2008, on a déjà des tronçons avec des rangs de 6 à plus de 400 km de l'embouchure). Or, il semble, à la lecture de la figure 4.4, que les densités sont d'autant mieux évaluées que les rangs des tronçons sont peu élevés, ce qui conduit à s'interroger sur l'influence des rangs élevés dans les calculs (notamment dans l'expression de l'écart-type de la gaussienne).

La troisième et dernière figure comprenant des diagrammes en boîtes est la figure 4.6. Là encore, on peut vérifier la tendance du modèle à surestimer les densités d'anguilles. On peut par contre directement apprécier la qualité de l'ajustement sur les données année par année, ce qui est très utile pour optimiser la calibration des paramètres. On remarque ainsi que la simulation surestime largement les données des années 1990, 1992 et 1993. Pour toutes les autres années, l'écart algébrique est centré sur 0, ce qui indique un bon ajustement global, avec une qualité qui varie évidemment selon les années.

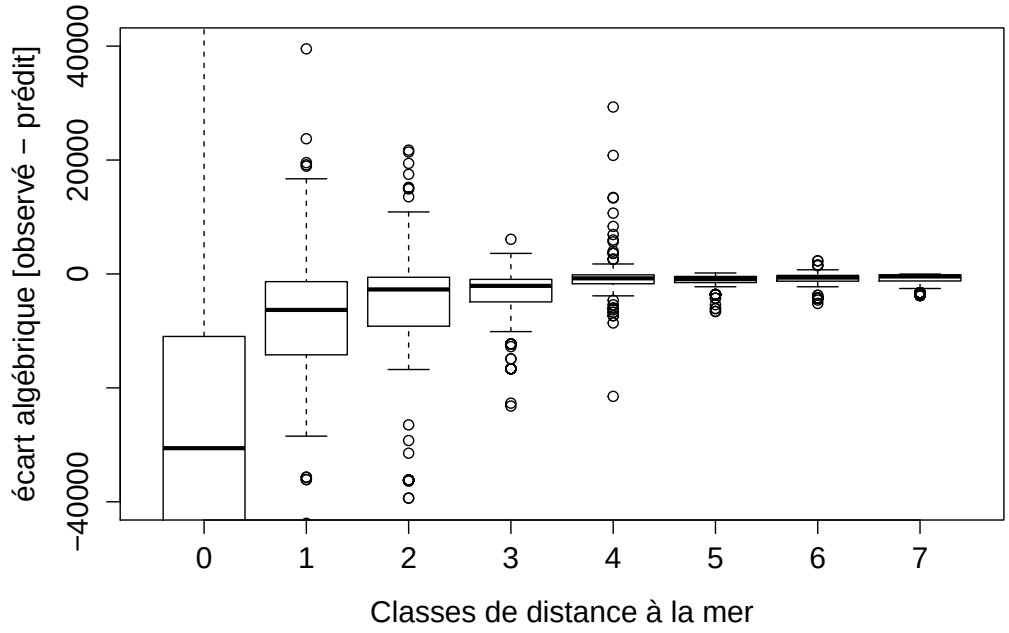

FIGURE 4.3 : Écarts algébriques entre densités d'anguilles observées et densités d'anguilles prédites par tronçon dans le bassin Garonne/Dordogne, en fonction de la distance à la mer. Les points correspondent aux données aberrantes (*outliers*). La ligne épaisse correspond à la médiane, les extrémités de la boîte correspondent au premier et au troisième quartile, et les extrémités des lignes pointillées correspondent respectivement à plus ou moins 1,5 fois l'écart interquartile. Les classes de distance sont définies de la manière suivante : classe 0 si le tronçon est éloigné de la mer d'une distance comprise entre 0 et 200 km, classe 1 pour des distances entre 200 et 250 km, classe 2 pour des distances entre 250 et 300 km, classe 3 pour des distances entre 300 et 350 km, classe 4 pour des distances entre 350 et 400 km, classe 5 pour des distances entre 400 et 450 km, classe 6 pour des distances entre 450 et 500 km, et classe 7 au-delà de 500 km.

Les premiers résultats de TABASCO qui sont ici présentés nous amènent à nous interroger sur la pertinence de la fonction utilisée dans le calcul de l'écart-type de la gaussienne (qui est, on le rappelle, égal à $\sqrt{2\,D\,t}$ où D est le coefficient de diffusion et t le temps). Afin d'apprécier le comportement de cette fonction, nous avons tracé sur la figure 4.7 ce que nous appellerons probabilité surfacique de sédentarisation, c'est-à-dire la probabilité qu'ont les anguilles de se sédentariser dans les tronçons par unité de surface en eau de ces tronçons (c'est une probabilité ramenée à une surface unitaire, qui s'exprime donc en k^{-2}) en fonction de la distance à la mer. Les discontinuités qui apparaissent et qui sont dues à l'utilisation des rangs de Strahler des tronçons dans l'un des calculs (à valeurs discrètes) nous suggèrent de proposer une nouvelle fonction continue (donc sans utiliser le rang de Strahler). de plus, on peut voir que la probabilité "explose" à des distances proches de la mer. Cela nous conduit très probablement à surestimer la population d'anguilles dans ces zones, d'autant plus que nous manquons de données à ces endroits-là, ce qui empêche une bonne estimation des paramètres du modèle. Nous considérons que la surévaluation des densités dans TABASCO provient très certainement d'une mauvaise distribution des anguilles dans ces zones proches de l'exutoire des bassins versants, ce qui doit nous amener à revoir le calcul de la diffusion et de la sédentarisation dans ces secteurs.

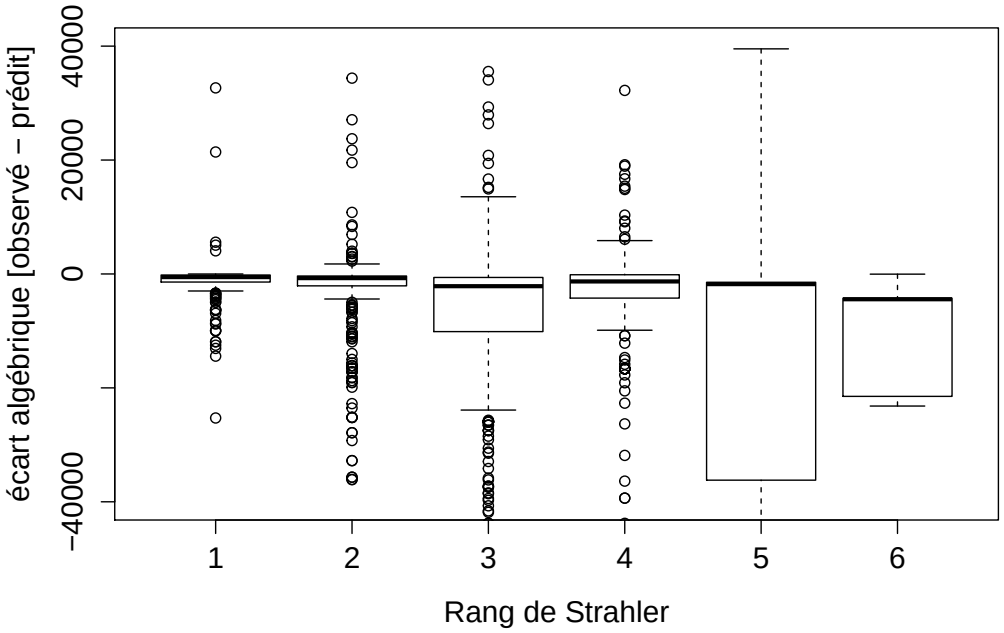

FIGURE 4.4 : Écarts algébriques entre densités d'anguilles observées et densités d'anguilles prédites par tronçon dans le bassin Garonne/Dordogne, en fonction du rang de Strahler. Les points correspondent aux données aberrantes (*outliers*). La ligne épaisse correspond à la médiane, les extrémités de la boîte correspondent au premier et au troisième quartile, et les extrémités des lignes pointillées correspondent respectivement à plus ou moins 1,5 fois l'écart interquartile.

4.3 Nombre et densité d'anguilles à l'échelle d'un bassin versant de référence

Pour l'approche par propagation d'une gaussienne sur le seul bassin versant de la Gironde, avec une calibration sur les données de pêches électriques de 2008, l'algorithme d'optimisation converge en une vingtaine de secondes et la simulation est réalisée en moins de 30 secondes. L'algorithme converge typiquement en une cinquantaine d'itérations et entre 100 et 150 évaluations de la fonction-objectif. Ce temps prend en compte la totalité des étapes du modèle (interaction avec la base de données, calibration et simulation proprement dite).

Nombre d'anguilles en Gironde

Nous présentons sur la figure 4.8 les résultats de la simulation (sous les conditions exposées ci-dessus) pour le nombre d'anguilles par tronçon du réseau dans le bassin Garonne/Dordogne. On vérifie que la colonisation s'effectue préférentiellement dans les drains principaux et à proximité de la mer.

Densité d'anguilles en Gironde

La figure 4.9 est le résultat de la simulation pour l'obtention des densités surfaciques d'anguilles dans le bassin de la Gironde (Garonne/Dordogne). Les conditions sont les mêmes que pour le nombre d'anguilles (données de 2008).

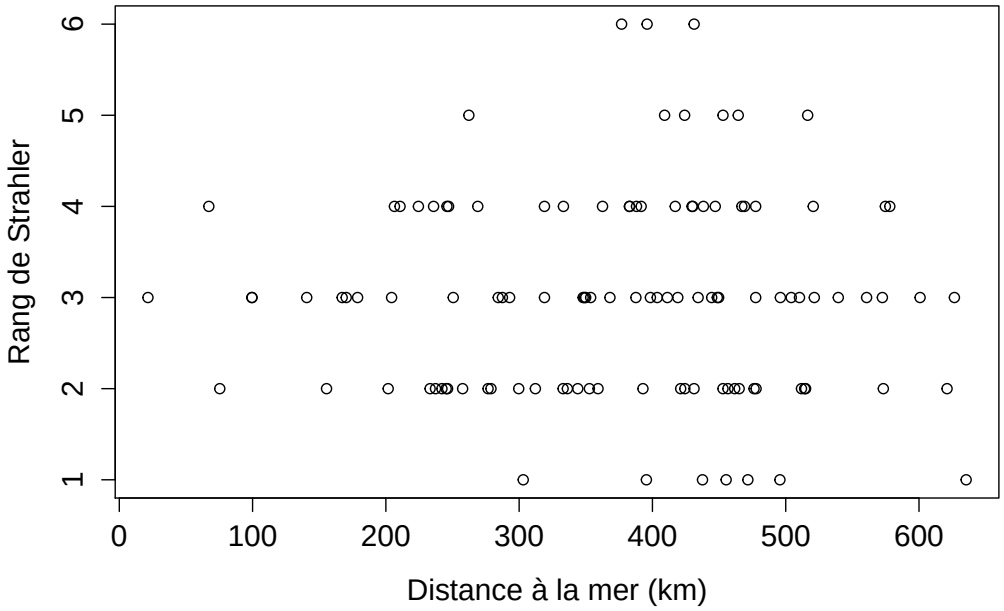

FIGURE 4.5 : Distribution des tronçons du bassin Garonne/Dordogne concernés par les données de pêches électriques de 2008 en fonction de leur rang de Strahler et de leur éloignement à la mer (en km).

4.4 Nombre et densités d'anguilles à l'échelle de la France entière

Pour l'approche par propagation d'une gaussienne, le modèle tourne en moins de 2 minutes 30 secondes sur l'ensemble des bassins versants référencés pour le moment (voir la liste complète en annexe du présent rapport). Il y a actuellement 60 bassins versants pris en compte dans la modélisation, pour une surface totale de 388633 km^2. La France métropolitaine ayant une superficie totale de 551695 km^2[2], le modèle couvre donc $70,4\%$ de la superficie totale de la France métropolitaine.

4.4.1 Nombre d'anguilles en France métropolitaine

Nous présentons les effectifs d'anguilles par tronçon sur les 60 bassins versants sur la figure 4.10, toujours avec les données de pêches électriques de 2008.

4.4.2 Densité d'anguilles en France métropolitaine

Les densités surfaciques d'anguilles par tronçon sur les 60 bassins versants actuellement intégrés dans le modèle sont affichées sur la figure 4.11. Les données utilisées sont toujours celles de 2008.

[2]Surface géodésique de référence selon l'Institut Géographique National (IGN).

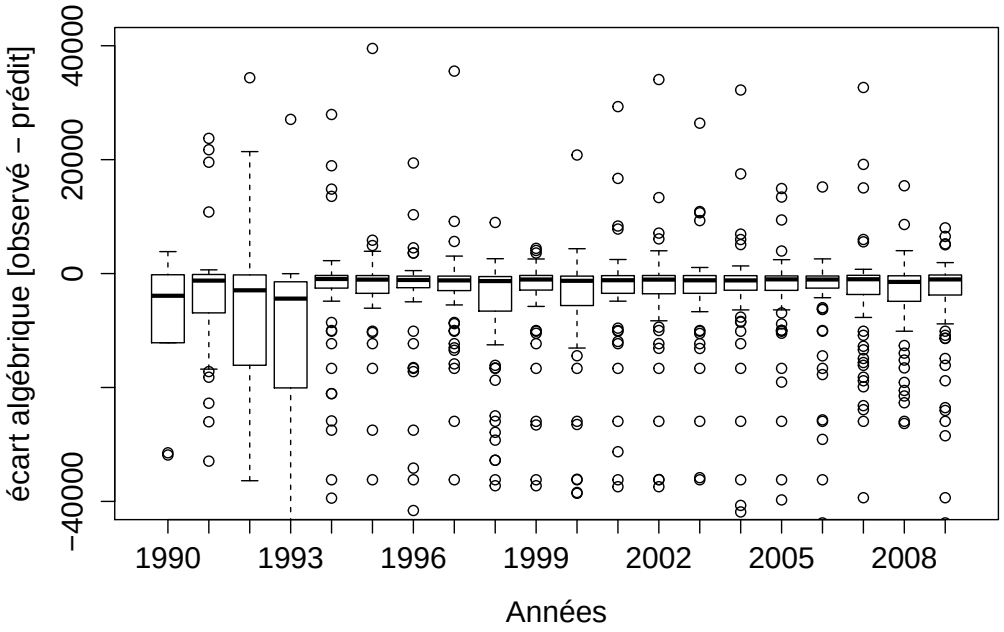

FIGURE 4.6 : Écarts algébriques entre densités d'anguilles observées et densités d'anguilles prédites par tronçon dans le bassin Garonne/Dordogne, en fonction des années, entre 1990 et 2009. Les points correspondent aux données aberrantes (*outliers*). La ligne épaisse correspond à la médiane, les extrémités de la boîte correspondent au premier et au troisième quartile, et les extrémités des lignes pointillées correspondent respectivement à plus ou moins 1,5 fois l'écart interquartile.

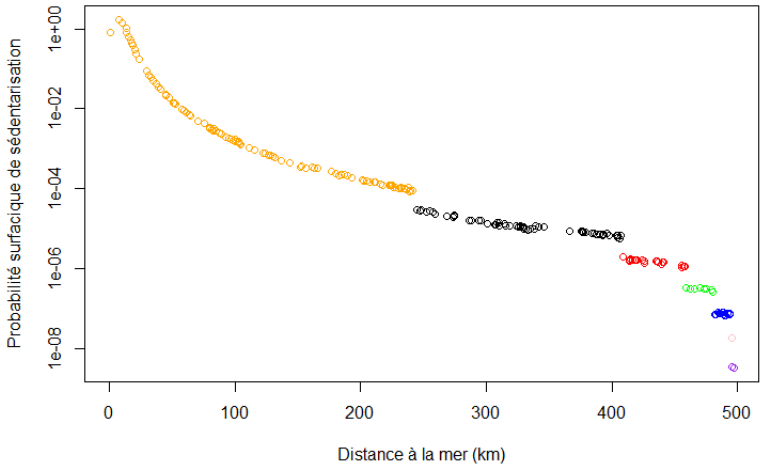

FIGURE 4.7 : Probabilité de sédentarisation par tronçon, ramenée à une surface unitaire (km^{-2}) en fonction de la distance à la mer, pour la Garonne. Les discontinuités sont dues à l'expression de l'écart-type de la gaussienne, où figure le rang de Strahler des tronçons (à valeurs discrètes). Les couleurs correspondent respectivement à un rang de 7 pour l'orange, un rang de 6 pour le noir, un rang de 5 pour le rouge, un rang de 4 pour le vert, un rang de 3 pour le bleu, un rang de 2 pour le rose et un rang de 1 pour le violet. L'échelle des ordonnées est logarithmique.

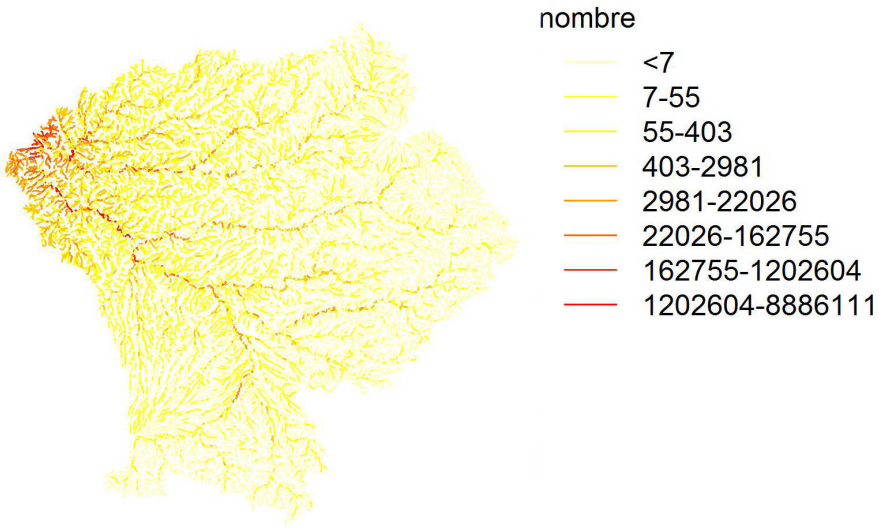

FIGURE 4.8 : Nombre d'anguilles par tronçon dans le bassin de la Gironde (Garonne/-Dordogne), à partir d'une calibration sur les données de pêches électriques de l'année 2008.

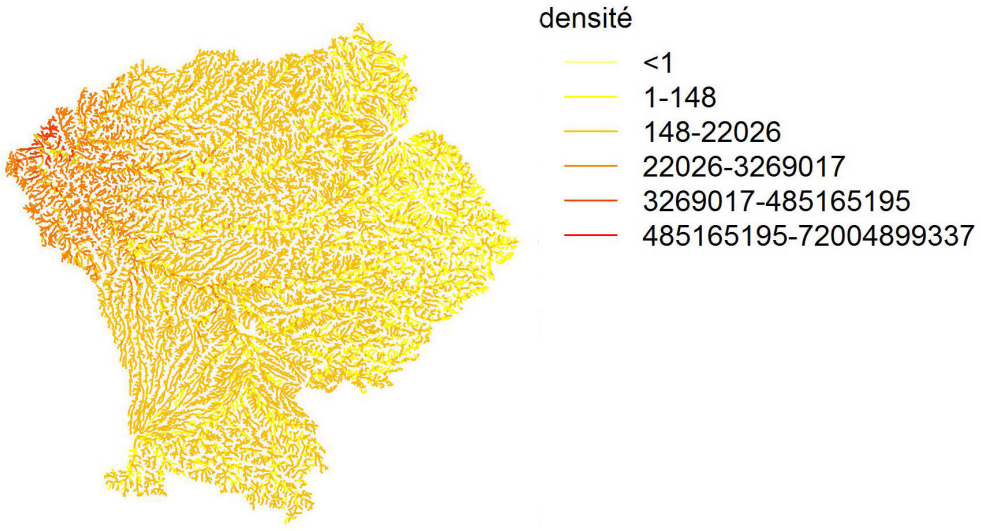

FIGURE 4.9 : Densité surfacique d'anguilles par tronçon dans le bassin de la Gironde (Garonne/Dordogne), à partir d'une calibration sur les données de pêches électriques de l'année 2008 (en individus/km^2).

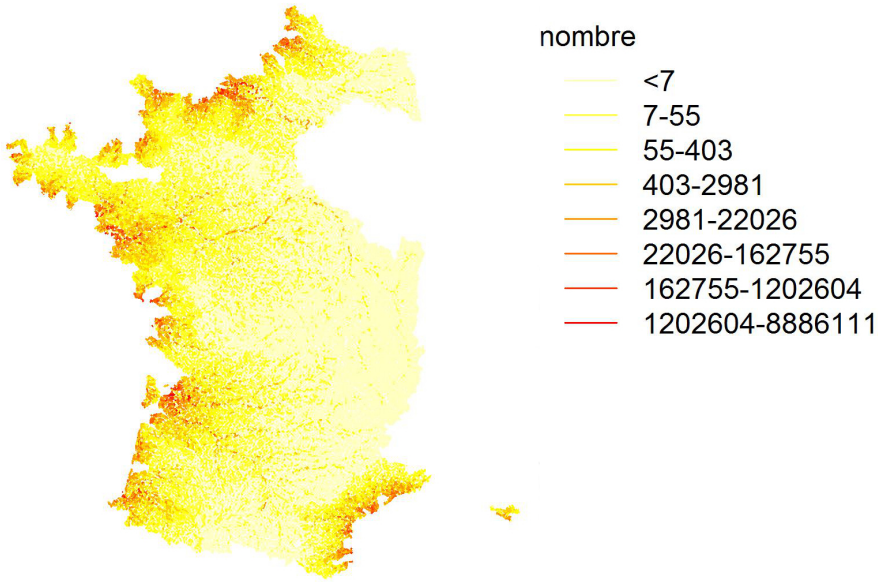

FIGURE 4.10 : Nombre d'anguilles par tronçon sur les 60 bassins versants de référence actuellement considérés dans la simulation.

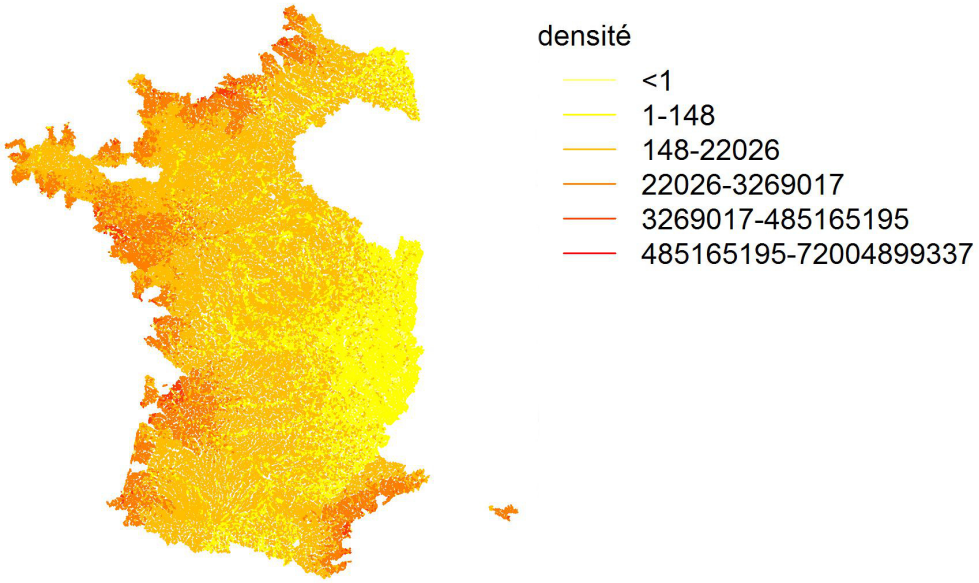

FIGURE 4.11 : Densité surfacique d'anguilles par tronçon sur les 60 bassins versants de référence actuellement considérés dans la simulation (en individus/km^2).

Chapitre 5

Bilan et perspectives

5.1 Bilan du développement de TABASCO

Durant la première phase du projet (2013), un important travail d'intégration des données dans les outils successifs a été conduit. Une première version du code pour deux hypothèses de calcul (approche par propagation d'une gaussienne et approche par matrice de transition) a été effectuée. Il a été vérifié que l'approche par propagation d'une gaussienne sans diffusion à l'aval des obstacles tournait rapidement, ce qui a pu initier la phase de calibration. L'approche matricielle a dû résoudre un problème de gestion de la mémoire lié à la taille de la matrice de transition (nombre de tronçons au carré). L'utilisation d'outils adaptés aux matrices creuses a permis de résoudre le problème. La présence d'un exposant entier a par ailleurs entraîné l'abandon de la différentiation automatique pour le modèle matriciel. La solution finalement retenue a été d'utiliser un algorithme dit d'optimisation sans dérivées sous contraintes (BOBYQA). Comme il n'a pas besoin de calculer les dérivées de la fonction objectif, on peut travailler avec des fonctions à valeurs discrètes (non continues ou non dérivables).
La seconde phase de développement du projet (2014) a tout d'abord permis la calibration du modèle sur le bassin versant de référence que nous avions choisi, à savoir le bassin versant de la Gironde (Garonne/Dordogne). Nous avons ensuite intégré une soixantaine de bassins versants dans la modélisation, représentant un peu plus de 70 % de la superficie de la France Métropolitaine. Cette modification du nombre de bassins a nécessité certaines adaptations dans le code.
Concernant l'approche par propagation d'une gaussienne, une longue phase de test a aussi été menée afin d'améliorer la prise en compte des différentes variables topologiques dans le calcul de la diffusivité. Un travail de débogage a également été effectué pour améliorer le code et corriger certaines erreurs.
Enfin, un travail de préparation et de normalisation des sorties graphiques a été réalisé pour la présentation des résultats actuels et futurs.

5.2 Perspectives

5.2.1 A court terme

Nous avons programmé pour le début de l'année 2015 un certain nombre d'améliorations et d'ajouts au modèle, dont la plupart ont été abordés et validés lors d'un comité de pilotage le 1er octobre 2014 :

- Programmer le calcul de l'intervalle de confiance des paramètres en récupérant la matrice hessienne, au moins pour l'approche par propagation d'une gaussienne. Cela doit être fait en particulier pour l'abondance totale d'anguilles, qui est non seulement un paramètre du modèle mais également une variable d'intérêt pour la gestion.

- Corriger si nécessaire la fonction de calcul de la diffusivité (approche gaussienne) pour limiter l'apparition de certains artefacts de calcul, comme par exemple des effets de bord au niveau des extrémités des bassins (estuaire ou sources des drains), des discontinuités ou encore une surestimation des abondances dans certaines zones.
- Vérifier que l'on tient bien compte de la surface en eau de chaque tronçon pour représenter sa capacité d'accueil.
- Tester un algorithme qui stoppe la diffusion des anguilles lorsque l'effectif à sédentariser passe en-dessous d'un certain seuil.
- S'assurer que l'on utilise le même jeu de données dans TABASCO et dans EDA. Vérifier la version utilisée du RHT. Vérifier la possibilité d'exprimer les résultats du modèle sur un tronçon RHT de base.
- Faire apparaître dans les sorties graphiques les données de pêches électriques, le paramétrage utilisé pour le calcul, un critère de qualité (AIC) et une analyse des résidus.
- Évaluer la fiabilité et la robustesse de TABASCO de plusieurs manières : en utilisant des données issues du modèle opératoire CREPE (Lambert, 2012), en calculant un critère de qualité, et en définissant une procédure systématique de comparaison des modèles. En particulier, la réponse de TABASCO à différents niveaux d'abondance totale dans CREPE sera testée pour mesurer la capacité du modèle à détecter une amélioration ou une dégradation du stock d'anguilles dans un hydrosystème.
- S'assurer, dans le modèle, du critère de détermination de la proportion d'anguilles entrant dans un bassin versant donné.

5.2.2 A long terme

Dans la dernière phase du projet (deuxième moitié de 2015), il est prévu de :

- Implémenter, si cela s'avérait possible (complexité non négligeable), l'utilisation d'exposants réels pour la matrice stochastique dans l'approche matricielle (actuellement, on rappelle que les exposants sont des entiers).
- Appliquer le modèle sur les abondances d'anguilles par classes de taille (par exemple en cm :]15,30],]30,45] et]45,+∞[).
- Soumettre un article scientifique dans une revue à comité de lecture.
- Présenter le modèle dans une conférence internationale.

Chapitre 6

Annexes

6.1 Liste par ordre alphabétique des bassins versants intégrés dans la modélisation

Chaque nom de bassin versant est suivi de la mer ou de l'océan dans lequel se jette le fleuve qui lui est associé, ou dans lequel il se vide au niveau de son exutoire (dans le cas d'étangs par exemple).

1. Adour (Océan Atlantique)
2. Agly (Mer Méditerranée)
3. Arguenon (Manche)
4. Arques (Manche)
5. Aude (Mer Méditerranée)
6. Aulne (Océan Atlantique)
7. Authie (Manche)
8. Bidassoa (Océan Atlantique)
9. Blavet (Océan Atlantique)
10. Bourdigoul (Mer Méditerranée)
11. Bourret/Boudigau (Océan Atlantique)
12. Bresle (Manche)
13. Brivet (Océan Atlantique)
14. Canche (Manche)
15. Charente (Océan Atlantique)
16. Couesnon (Manche)
17. Courant de Contis (Océan Atlantique)
18. Courant de Lège (Océan Atlantique)
19. Courant de Mimizan (Océan Atlantique)
20. Courant de Soustons (Océan Atlantique)
21. Dives (Manche)

22. Douve (Manche)
23. Elorn (Océan Atlantique)
24. Falleron (Océan Atlantique)
25. Fremur (Manche)
26. Gapeau (Mer Méditerranée)
27. Gironde (Océan Atlantique)
28. Grand Isaka (Océan Atlantique)
29. Hérault (Mer Méditerranée)
30. Jalle du Cartillon (Océan Atlantique)
31. Laïta (Océan Atlantique)
32. La Palme (Mer Méditerranée)
33. Lay (Océan Atlantique)
34. Léguer (Manche)
35. Leyre (Océan Atlantique)
36. Libron (Mer Méditerranée)
37. Loire (Océan Atlantique)
38. Nivelle (Océan Atlantique)
39. Odet (Océan Atlantique)
40. Orb (Mer Méditerranée)
41. Orne (Manche)
42. Rance (Manche)
43. Riviere d'Auray (Océan Atlantique)
44. Sée (Manche)
45. Seine (Manche)
46. Sélune (Manche)
47. Seudre (Océan Atlantique)
48. Seulles (Manche)
49. Sèvre niortaise (Océan Atlantique)
50. Sienne (Manche)
51. Somme (Manche)
52. Tech (Mer Méditerranée)
53. Têt (Mer Méditerranée)
54. Thau/Mosson/Lez (Mer Méditerranée)
55. Touques (Manche)
56. Trieux (Manche)
57. Uhabia (Océan Atlantique)
58. Untxin (Océan Atlantique)
59. Vilaine (Océan Atlantique)
60. Vire (Manche)

6.2 Tutoriel d'installation des fichiers non pré-compilés de la bibliothèque graphique Boost

Prérequis :

1. Télécharger (par exemple en allant sur http://sourceforge.net/projects/mingw/files/) et installer MinGW, de préférence à la racine d'un disque local (ce qui donne par exemple le chemin d'accès "C:\MinGW" si on choisit de l'installer dans le répertoire "MinGW" à la racine du disque local C:).
2. Télécharger la dernière version stable de Boost sur http://www.boost.org/users/download/.
3. Télécharger la dernière version de PostgreSQL sur http://www.postgresql.org/download/.

Installation des bibliothèques non pré-compilées de Boost :

Ouvrir l'invite de commande de Windows (Sous Windows 7, aller dans Tous les programmes→Accessoires→Invite de Commande). Si vous n'êtes pas administrateur de votre machine, ouvrez l'invite en mode administrateur (clic droit, et "Exécuter en tant qu'administrateur"). L'utilisateur doit maintenant choisir parmi les méthodes décrites ci-après celle qui fonctionnera sur sa machine et avec sa configuration logicielle.

Méthode 1

1. Dans l'invite de commande, déplacez-vous dans le répertoire de téléchargement de Boost, puis dans "tools\build\v2\engine". Exemple de commande à taper :
   cd boost_1_55_0\tools\build\v2\engine
2. Garder l'invite de commande ouverte. A l'aide d'un éditeur de texte (idéalement Notepad++), ouvrir le fichier "build.bat". Dans la routine intitulée " :Guess_Toolset", trouver la structure contenant le mot-clef "mingw". Éditer le fichier en conséquence (avec le bon chemin d'accès vers votre répertoire d'installation de MinGW, et effacer toutes les autres structures similaires. Au final, votre routine " :Guess_Toolset" doit contenir quelque-chose comme :
   :Guess_Toolset REM Try and guess the toolset to bootstrap the build with...
   REM Sets BOOST_JAM_TOOLSET to the first found toolset.
   REM May also set BOOST_JAM_TOOLSET_ROOT to the REM location of the found toolset.

   call :Clear_Error
   call :Test_Empty %ProgramFiles%
   if not errorlevel 1 set ProgramFiles=C :\Program Files

   if EXIST "C :\MinGW\bin\gcc.exe" (
   set "BOOST_JAM_TOOLSET=mingw"
   set "BOOST_JAM_TOOLSET_ROOT=C :\MinGW\"
   goto :eof)
   call :Clear_Error
   if NOT "_%CWFolder%_" == "__" (
   set "BOOST_JAM_TOOLSET=metrowerks"
   set "BOOST_JAM_TOOLSET_ROOT=%CWFolder%¨
   goto :eof)
   call :Clear_Error
   call :Test_Path mwcc.exe
   if not errorlevel 1 (

```
set "BOOST_JAM_TOOLSET=metrowerks"
set "BOOST_JAM_TOOLSET_ROOT=%FOUND_PATH%..\..\"
goto :eof)
call :Clear_Error
call :Error_Print "Could not find a suitable toolset."
goto :eof
```

3. Dans l'invite de commande, vérifiez que vous êtes toujours dans le sous-répertoire "\tools\build\v2\engine", puis tapez la commande "build.bat".Cela va normalement créer le fichier "bjam.exe" pour MinGW/gcc.

4. Ajouter le chemin d'accès vers ce fichier dans la variable d'environnement PATH de Windows (au début de la variable, et séparé des autres chemins par un point-virgule). En principe, cela doit ressembler à :
"C :\boost_1_55_0\tools\build\v2\engine\bin.ntx86 ;"

5. A la racine de votre répertoire d'installation de Boost, taper alors dans l'invite de commande :
bjam –toolset=gcc –layout=system –with-thread install

En cas de difficultés, quelques explications additionnelles sont disponibles sur ce forum : http://www.developpez.net/forums/d1096176/c-cpp/cpp/bibliotheques/boost/aide-installation-boost-version-1-46-1-a/.

Méthode 2

1. Allez dans le répertoire "tools\build\v2\"

2. Tapez la commande : bootstrap.bat gcc

3. Tapez la commande : b2 install –prefix=PREFIX (sans espace entre les deux signes moins, et où PREFIX est le répertoire dans lequel vous souhaitez que Boost.Build soit installé).

4. Ajoutez "PREFIX\bin" dans votre variable d'environnement PATH.

Méthode 2 bis

1. Choisir un répertoire d'installation. Boost.Build placera tous les fichiers intermédiaires qu'il génère lors de l'installation dans le répertoire d'installation. Si votre répertoire racine de Boost est accessible en écriture, cette étape n'est pas strictement nécessaire : par défaut, Boost.Build créera pour cela un sous-répertoire "bin.v2\" au sein de votre répertoire de travail courant.

2. Invoquer b2. Changer votre répertoire courant en vous plaçant dans le répertoire racine de Boost, et invoquer b2 en tapant dans votre invite de commandes :
b2 –build-dir=build-directory toolset=toolset-name –build-type=complete stage
Pour une description complète de cette utilisation et des différentes options, consultez la documentation Boost.Build.
Votre session devrait ressembler à cela :
%C :\WINDOWS> cd C :\Program Files\boost\boost_1_55_0
%C :\Program Files\boost\boost_1_55_0> b2 ^
%More ? –build-dir="C :\Documents and Settings\dave\build-boost" ^
%More ? –build-type=complete msvc stage
L'option "–build-type=complete" fera que Boost.Build installera toutes les versions supportées des différentes bibliothèques. Pour savoir comment n'installer que certaines versions spécifiques, référez-vous à l'aide en ligne de Boost.Build.

Infos supplémentaires (en anglais pour le moment) :
Building the special stage target places Boost library binaries in the stage\lib\subdirectory of the Boost tree. To use a different directory pass the –stagedir=directory option to b2.
Note : b2 is case-sensitive ; it is important that all the parts shown in bold type above be entirely lower-case. For a description of other options you can pass when invoking b2, type : b2 –help
In particular, to limit the amount of time spent building, you may be interested in : • reviewing the list of library names with –show-libraries • limiting which libraries get built with the –with-library-name or –without-library-name options • choosing a specific build variant by adding release or debug to the command line.
Note : Boost.Build can produce a great deal of output, which can make it easy to miss problems. If you want to make sure everything is went well, you might redirect the output into a file by appending ">build.log 2>&1" to your command line. 5.3 Expected Build Output During the process of building Boost libraries, you can expect to see some messages printed on the console. These may include • Notices about Boost library configuration—for example, the Regex library outputs a mes-sage about ICU when built without Unicode support, and the Python library may be skipped without error (but with a notice) if you don't have Python installed. • Messages from the build tool that report the number of targets that were built or skipped. Don't be surprised if those numbers don't make any sense to you ; there are many targets per library. • Build action messages describing what the tool is doing, which look something like : • toolset-name.c++ long/path/to/file/being/built • Compiler warnings.

6.3 Tutoriel d'installation de la bibliothèque libpqxx

Le présent tutoriel indique comment installer et compiler la bibliothèque libpqxx sur une machine Windows.

1. Installer la dernière version de PostgreSQL sur votre ordinateur. Cela ne pose aucun problème particulier en utilisant l'exécutable pgInstaller.

2. Après l'installation, vous devez pouvoir trouver dans votre répertoire d'installation le sous-répertoire *lib* qui contient la bibliothèque libpq. Exemple de chemin d'accès : "C :\Program Files\PostgreSQL\Version\lib".

3. Télécharger la dernière version de lipqxx, et décompresser les fichiers dans le répertoire de votre choix. Ensuite, il faut préparer la compilation de libpqxx (ce qui suppose d'avoir déjà sur votre machine la bibliothèque libpq sous la forme de ses fichiers binaires et de ses fichiers d'en-tête). Pour cela, copiez le fichier "win32\common-sample", renommez-le en " win32\common", et éditez-le pour qu'il tienne compte des chemins d'accès à vos fichiers « include » et « lib » de PostgreSQL. Vérifiez pour cela que la variable PGSQLSRC pointe bien vers votre installation de PostgreSQL, puis suivez les instructions du fichier "win32\common" pour commenter ou décommenter certaines lignes. Ensuite, vous devez créer les fichiers d'en-tête de configuration, de la forme "include\pqxx\config-*-*.h". Le plus simple pour cela est de copier les fichiers-types dans "config\
sample-headers". Dans ce dernier répertoire, trouvez les sous-répertoires correspondant le mieux à votre compilateur et à votre version de la bibliothèque libpq respectivement. Prenez les fichiers config-*-*.h dans ces répertoires, et copiez-les dans le répertoire "include\pqxx\".

4. Il reste à compiler et installer la bibliothèque lipqxx. Dans cette optique, installer (si ce n'est pas déjà fait) MinGW sur votre machine (directement à la racine du disque, par exemple dans "C :\MinGW"). Ensuite, installer MSYS (mais attention, pas dans l'arborescence de MinGW, par exemple dans "C :\MSYS"). Puis, lancer MSYS (aller dans Démarrer → Tous les Programmes → MinGW → MSYS → msys). Dans l'invite de commande de MSYS, aller dans le répertoire principal d'installation de lipqxx (celui dans lequel vous avez décompressé les

fichiers), puis taper :
export LDFLAGS=-lws2_32.

5. Vérifiez que le chemin vers le répertoire "\bin" de MinGW est bien dans votre variable d'environnement PATH (le rajouter au besoin).

6. Tapez enfin :
./configure –prefix="C :/MinGW/local" –enable-static && make &&
make install

Il y a alors de grandes chances pour que la compilation ne fonctionne pas. Pas d'inquiétude, cela est normal sous Windows, et vient du fait qu'il faut éditer manuellement les fichiers d'en-tête, ce qui conduit inexorablement à des erreurs lors de la compilation de la bibliothèque (en général, il s'agit d'erreurs spécifiant des fichiers d'en-tête introuvables ou manquants). Une erreur typique indique ainsi que le fichier d'en-tête <sys/select.h> est manquant. Vous trouverez alors une variable de configuration nommée PQXX_HAVE_SYS_SELECT_H, qui ne doit pas être définie si votre système d'exploitation ne possède pas de fichier sys/select.h (ce qui est le cas sous Windows). Dans ce cas, supprimez-la ou supprimez les lignes définissant cette variable dans le fichier d'en-tête de configuration (a priori le fichier config-internal-compiler.h), et retentez une compilation. L'erreur aura normalement disparu, et procédez de la même façon pour d'autres nouvelles erreurs qui apparaîtraient. Il n'est pas anormal de supprimer plusieurs lignes dans le fichier de configuration pour que tout fonctionne au final !

6.4 Script R pour l'affichage graphique des sorties

SCRIPT DE GESTION DES SORTIES GRAPHIQUES DU MODELE TABASCO
(© Hilaire Drouineau, Jocelyn Domange and Patrick Lambert / IRSTEA Bordeaux / Janvier 2015)

```
    # Chargement des bibliothèques
library(RPostgreSQL)
library(rgeos)
library(sp)
library(squash)
library(RColorBrewer)
library(classInt)
library(maptools)
library(graphics)

    # Fonction qui prend le mot de passe de connexion à la base de données
getPass=function(){
print("password : ")
pass=scan(n=1,what=character(),quiet=TRUE)
cat(" 014 ")
return(pass)
}

    # Connexion à la base de données
m <- dbDriver("PostgreSQL")
con <-dbConnect(m,host="xx.xx.fr",port=XXXX,dbname="xxxx",
user="xxxx.xxxx",password=getPass())

    # Récupération des résultats de la modélisation
print(Sys.time())
res=dbGetQuery(con,"select *, st_ astext(geom) wkt from xxxx.xxx")
print(Sys.time())
```

```
    # Création des lignes géoréférencées avec attributs
row.names(res)=res$edge_ id
tot=do.call("rbind",mapply(readWKT,res$wkt,res$edge_ id))
res_ final=SpatialLinesDataFrame(tot, res[-length(row.names)])
print(Sys.time())

    # Tracé et sauvegarde de la carte
#x11(height=16/2.54,width=16/2.54)
graphics.off()
jpeg("C :/Mes_ programmes/carte_ densite_ 60BV.jpeg",height=16/2.54,
width=16/2.54,units="in",res=300)
par(mar=c(2.5,2.5,.5,13),mgp=c(1.5,.5,0))

    nombre_ classe_ couleur=5
bornes=pretty(c(0,max(log(res$density))),n=5, min.n=5)
couleurs=rev(heat.colors(length(bornes)-1))

    plot(res_ final,col=(sapply(res_ final$density,getcolor)))

    getcolor=function(x) {
if (log(x)<=bornes[1]) {
return(couleurs[1])
} else{
return(couleurs[max(which(log(x)>bornes))])
}
}

    box()
par(xpd=NA)

    text(grconvertX(1,"npc"),grconvertY(.75,"npc"),"densité",adj=c(0,0))
legend(grconvertX(1,"npc"),grconvertY(.75,"npc"), c(paste("<",
round(exp(bornes[2])),sep=""),paste(round(exp(bornes[2 :(length(bornes)-1)])),
round(exp(bornes[3 :(length(bornes))])),sep="-")), lty=rep(1,length(couleurs)),
col=couleurs,bty="n",xjust=0,yjust=1)
dev.off()
print(Sys.time())
```

AUTRES OPTIONS D'AFFICHAGE

```
    # color = function(x)rev(heat.colors(x))
# color = function(x)heat.colors(x)
mycol <- colorRampPalette(c("lightcyan", "blue4"))
plot(res_ final)
plot(res_ final, col=res_ final$number)
plot(res_ final, col=res_ final$number, lwd=res_ final$strahler)
map<-makecmap(res_ final$number, breaks = c(0,100,10000,1000000,
100000000,10000000000), colFn = mycol)
pl.color<-cmap(res_ final$number, map = map)
plot(res_ final, col=pl.color, lwd=res_ final$strahler, main="Nombre d'anguilles par
tronçon")
mycol <- colorRampPalette(c("yellow", "red4"))
map<-makecmap(res_ final$concentration, breaks = c(0,100,10000,1000000,
100000000,1000000000000000), colFn = mycol)
pl.color<-cmap(res_ final$concentration, map = map)
plot(res_ final, col=pl.color, lwd=res_ final$strahler/2)
print(Sys.time())
```

```
    # discrétisation en 7 classes (quantiles)
distr <- classIntervals(res$concentration,7,style="quantile",dataPrecision=5)$brks

    # choix d'une gamme de couleurs
# pour voir les palettes disponibles : display.brewer.all()
colours <- brewer.pal(7,"YlOrRd")

    # optionnel - codes des couleurs utilisées
colours

    # attribution des couleurs aux régions
colMap <- colours[(findInterval(res$concentration,distr,all.inside=TRUE))]

    # Affichage de la carte
par(xpd=T, mar=par()$mar+c(6,0,0,0))
plot(res_ final, col=colMap, lwd=res_ final$strahler/2)

    # affichage de la légende
# legend("bottom", inset=c(-0.05,0),legend=leglabs(distr,under="Inférieur à",
over="Supérieur à", between="-"), fill=colours, bty="n",
title="Densité surfacique d'anguilles par tronçon", cex=0.5,
pt.cex=0.5)
legend(x=c(2.0, 2.0), y=c(2.0, 2.0), legend=leglabs(distr),
fill=colours, bty="n", title="Densité surfacique d'anguilles par tronçon")
legend(locator(1), legend=leglabs(distr), fill=colours, bty="n",
title="Densité surfacique d'anguilles par tronçon", cex=0.7,
pt.cex=0.7)
# l'introduction de la chaine de caractère «  n » entraine un saut de ligne dans
# le texte à afficher

    # titre et sous titres
title(main="Prédiction par le modèle TABASCO de la densité surfacique d'anguilles
par tronçon ",
sub="auteurs : Jocelyn Domange, Hilaire Drouineau, Patrick Lambert, IRSTEA Bor-
deaux (équipe EABX/PMA)"
```

Bibliographie

Anonyme : Plan de gestion anguille de la france, rapport de mise en oeuvre - juin 2012. Rapport technique, Ministère de l'écologie, de l'énergie, du développement durable et de la mer / Ministère de l'alimentation, de l'agriculture et de la pêche / Onema, 2012.

M. W. Aprahamian, A. M. Walker, B. Williams, A. Bark et B. Knights : On the application of models of European eel (*Anguilla anguilla*) production and escapement to the development of Eel Management Plans : the River Severn. *ICES Journal of Marine Science*, 64:1472–1482, 2007.

David A. Fournier, Hans J. Skaug, Johnoel Ancheta, James Ianelli, Arni Magnusson, Mark N. Maunder, Anders Nielsen et John Sibert : AD Model Builder : using automatic differentiation for statistical inference of highly parameterized complex nonlinear models. *Optimization Methods and Software*, 2012.

T. J. Hastie et R. J. Tibshirani : *Generalized Additive Models*, volume 43 de *Monogrpahs on Statistics and applied Probability*. CRC Press, 1990.

Anton Ibbotson, Jim Smith, Peter Scarlett et Miran Aprahamian : Colonisation of freshwater habitats by the European eel *Anguilla anguilla*. *Freshwater Biology*, 47(9):1696–1706, September 2002.

C. Jouanin, P. Gomes, C. Briand, V. Berger, F. Bau, H. Drouineau, P. Baran, P. Lambert et L. Beaulaton : Évaluation des mortalités d'anguilles induites par les ouvrages hydroélectriques en France. Projet SEA-HOPE (Silver Eels escApment From HydrOPowEr). Convention ONEMA-Irstea. Rapport final., 2012a.

Céline Jouanin, Cédric Briand, Laurent Beaulaton et Patrick Lambert : Eel Density Analysis (EDA 2.x), Un modèle statistique pour estimer l'échappement des anguilles argentées (Anguilla anguilla) dans un réseau hydrographique, Rapport Final. Rapport technique, Irstea Bordeaux, Institut d'Aménagement de la Vilaine, ONEMA, 2012b.

Patrick Lambert : Développement d'outils de modélisation de la population d'anguille européenne prenant en compte la diversité des paramètres de dynamique par grande fractiond'aire de répartition continentale de l'espèce. Rapport technique, Irstea Bordeaux, ONEMA, 2012.

Patrick Lambert et Eric Rochard : Identification of the inland population dynamics of the european eel using pattern-oriented modelling. *Ecological Modelling*, 206:166–178, 2007.

Patrick Lambert, Guy Verreault, Brigitte Lévesque, Valérie Tremblay, Jean-Denis Dutil et Pierre Dumont : Détermination de l'impact des barrages sur l'accès de l'anguille d'Amérique (Anguilla rostrata) aux habitats d'eau douce et établissement de priorités pour des gains en habitat. Rapport technique, Institut Maurice-Lamontagne, 2011.

L. P. Lefkovitch : The study of population growth in organisms grouped by stages. *Biometrics*, 21(1):1–18, March 1965.

Aurélie Léonard et Pascale Zegel : Référentiel des obstacles à l'écoulement, Version 1, Descriptif de contenu. Rapport technique, ONEMA, 2010.

Gaëlle Leprévost : Développement d'un indicateur pour caractériser l'impact migratoire sur le stock d'anguille européenne à l'échelle des bassins. Rapport technique, ONEMA / IAV, 2007.

P. H. Leslie : Some further notes on the use of matrices in population mathematics. *Biometrika*, 35(3 and 4):213–245, December 1948.

Simone Libralato, Villy Christensen et Daniel Pauly : A method for identifying keystone species in food web models. *Ecological Modelling*, 195:153–171, 2006.

M. Lifschitz : *Gaussian Random Functions*. Kluver Academic Publishers, 1995.

P. T. Manders : A transition matrix model of the population dynamics of the clan-william cedar (widdringtonia cedarbergensis) in natural stands subject to fire. *Forest Ecology and Management*, 20:171–186, 1987.

Hervé Pella, Jérôme Lejot, Nicolas Lamouroux et Ton Snelder : Le réseau hydrographique théorique (RHT) français et ses attributs environnementaux. *Géomorphologie : relief, processus, environnement*, 2012.

Michael J. D. Powell : The bobyqa algorithm for bound constrained optimization without derivatives. Rapport technique, Department of Applied Mathematics and Theoretical Physics, Cambridge University, 2009.

Louis B. Rall : *Automatic Differentiation : Techniques and Applications*, volume 120 de *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1981.

Jeremy G. Siek, Lie-Quan Lee et Andrew Lumsdaine : *The Boost Graph Library, User Guide and Reference Manual*. C++ In-Depth. Addison-Wesley / Pearson Education, 2002.

Gunnar Stefánsson et Ólafur K. Pálsson : Statistical evaluation and modelling of the stomach contents of icelandic cod (gadus morhua). *Can. J. Fish. Aquat. Sci.*, 54 (1):169–181, 1996.

P. Steinbach : Expertise de la franchissabilité des ouvrages hydrauliques transversaux par l'anguille dans le sens de la montaison. Rapport technique, Conseil Supérieur de la Pêche, 2006.

A.M. Walker, E. Andonegi, P. Apostolaki, M. Aprahamian, L. Beaulaton, P. Bevacqua, C. Briand, A. Cannas, E. De Eyto, W. Dekker, G. De Leo, E. Diaz, P. Doering-Arjes, E. Fladung, C. Jouanin, P. Lambert, R. Poole, R Oeberst et M Schiavina : Pilot projects to estimate potential and actual escapement of silver eel. Rapport technique, The European Commission Directorate-General for Maritime Affairs and Fisheries, 2011.

Larry Wasserman : *All of Statistics : A Concise Course in Statistical Inference*. Springer-Verlag, september 2004.

J. Angus Webb et Mark Padgham : How does network structure and complexity in river systems affect population abundance and persistence ? *Limnologica*, 43:399–403, 2013.

J.A. Webb et M. Padgham : Patterns of dispersal through stream networks respond simply to multiple structural modifications. *In 18th World IMACS / MODSIM Congress, Cairns, Australia*, pages 1809–1815, 2009.