

VALÉRIE CARETTE

**AMÉLIORATION DE LA REPRÉSENTATION
GÉOMÉTRIQUE 2D ET 3D DES AGRÉGATIONS DE
POISSONS EN SUPPORT À L'ÉTUDE DE LEUR
ÉVOLUTION SPATIO-TEMPORELLE**

Mémoire présenté
à la Faculté des études supérieures de l'Université Laval
dans le cadre du programme de maîtrise en sciences géomatiques
pour l'obtention du grade de maître ès sciences (M.Sc.)

DÉPARTEMENT DES SCIENCES GÉOMATIQUES
FACULTÉ DE FORESTERIE ET DE GÉOMATIQUE
UNIVERSITÉ LAVAL
QUÉBEC

2008

Résumé

Les systèmes d'information géographique (SIG) constituent des outils performants pour la gestion, l'analyse et la représentation de données spatiales. Ils sont utilisés dans de nombreux domaines dont en biologie marine. Parmi les phénomènes spatiaux étudiés en milieu marin, la dynamique des poissons fait l'objet de nombreuses recherches, étant donné l'importance des pêches dans l'économie des régions côtières et la nécessité de gérer ces ressources de façon durable. Un comportement clé des poissons est leur capacité à se regrouper pour former des agrégations. Une compréhension de ces agrégations est nécessaire afin d'établir des stratégies de reconstruction efficaces des ressources halieutiques en déclin. Cependant, les méthodes existantes utilisées pour représenter les agrégations de poissons ne modélisent pas ces agrégations explicitement en tant qu'objets spatiaux. De plus, malgré les capacités intéressantes offertes par les SIG actuels, ces outils sont limités en raison de la nature tridimensionnelle, dynamique et floue des agrégations de poissons dans le milieu marin.

L'objectif principal de ce mémoire est de proposer de nouvelles approches pouvant permettre d'améliorer l'identification et la représentation spatiale des agrégations de poissons à des échelles régionale et locale. En 2D, l'approche proposée repose sur les modèles d'objets spatiaux flous découlant de la théorie des ensembles flous. Elle utilise une structure de données vectorielle pour représenter les limites des étendues minimale et maximale des agrégations de poissons et une structure de données matricielle pour modéliser la transition graduelle existant entre ces limites. Par ailleurs, en 3D, l'approche développée se base sur la triangulation Delaunay 3D et dynamique ainsi que sur l'algorithme de *clustering* 3D des *α -shapes*. Elle permet de détecter les agrégations contenues dans un jeu de données et de les reconstruire sous la forme d'objets 3D afin de pouvoir en étudier, par exemple, les propriétés morphologiques. L'application des approches proposées à des données halieutiques révèle plusieurs avantages et limitations qui sont discutés tout au long du mémoire.

Abstract

Geographic information systems (GIS) are powerful tools to manage, analyse and represent spatial data. They are used in various disciplines, including marine biology. One of the most important phenomena intensively studied by marine biologists is the dynamics of fish. This is partly because there is an increasing need for sustainable management of fisheries which are very important in the economy of coastal zones. Fish aggregations are a fundamental component of these dynamics and should be better understood to establish efficient recovery strategies in the context of declining aquatic resources. However, the traditional representations of fish aggregations do not model those aggregations explicitly as spatial objects. Moreover, despite many interesting capabilities of current GIS, these tools are unable to handle the tridimensional, dynamic and fuzzy nature of fish aggregations.

The main objective of this thesis is to propose new approaches to improve the representation of fish aggregations at the regional and local scales. In 2D, the proposed approach is based on the fuzzy spatial objects models, which are based on the fuzzy sets theory. It uses a vector data structure to delineate the maximal and minimal extents of fish aggregations and a raster data structure to model the gradual transition which exists between these boundaries. In 3D, the proposed approach for the representation of fish aggregations is based on the dynamic Delaunay tetrahedralisation and the 3D α -shapes clustering algorithm. The integrated algorithm allows automatic detection of the fish aggregations contained in a dataset. 3D models also allow amongst other things the study of the morphological properties of the different aggregations. Testing these approaches with fisheries data (e.g. datasets from scientific surveys, acoustic datasets) revealed several benefits and limitations which are discussed throughout this thesis.

Avant-Propos

Ce mémoire est le résultat de plusieurs mois de travail et d'efforts. Je tiens à saluer toutes les personnes qui y ont contribué de près ou de loin.

Je remercie sincèrement mon directeur de recherche, Mir Abolfazl Mostafavi, sans qui je n'aurais probablement jamais entrepris des études graduées. Merci de m'avoir proposé ce projet de recherche alors que je terminais mon baccalauréat, de m'avoir encouragée à poursuivre mes études et de m'avoir accordé ta confiance. Je désire également remercier mon co-directeur de recherche, Rodolphe Devillers, pour ses conseils et commentaires qui m'ont souvent éclairée, particulièrement lors de la rédaction de ce mémoire.

Je remercie aussi George Rose pour avoir pris le temps de répondre à mes nombreuses questions concernant la biologie marine et Hugo Ledoux avec qui j'ai pu échanger régulièrement.

Lors de ces deux années de recherche, j'ai eu la chance de croiser le parcours de nombreux étudiants lors de cours, de conférences ou de réunions. Je remercie en particulier les étudiants du CRG et du projet GeoCod pour leur bonne humeur et leur amitié.

Je désire également remercier ma famille tant pour leur présence, leur soutien dans les moments les plus difficiles que pour leurs félicitations lors de mes réussites. Merci à vous trois, maman, Alexandre et Marie-Chantal. Merci aussi à mon copain, Louis-Alexis, qui a su m'écouter dans les moments de doute et de remise en question.

J'adresse un remerciement tout spécial à mon père, décédé pendant la rédaction de ce mémoire. Papa, merci de m'avoir poussée à avoir confiance en mes capacités et d'avoir toujours porté un intérêt à ce que je faisais. Merci aussi de m'avoir guidée de tes conseils chaque fois que j'en ai eu besoin. C'est grâce à toi que j'ai découvert ce qu'est la géomatique et que j'ai entrepris des études dans ce domaine.

Je dédie ce mémoire à mon père, parti trop vite, pour son soutien tout au long de mes études.

Table des matières

Résumé.....	ii
Abstract.....	iii
Avant-Propos	iv
Table des matières	vi
Liste des tableaux.....	viii
Liste des figures.....	ix
Chapitre 1 Introduction.....	1
1.1 Mise en contexte	1
1.1.1 Historique des pêches dans la région de l'Atlantique du nord-ouest.....	1
1.1.2 Agrégations de poissons	3
1.1.3 Systèmes d'information géographique	5
1.1.4 Représentation des agrégations de poissons dans les SIG.....	6
1.2 Problématique et hypothèse.....	8
1.3 Objectifs.....	9
1.4 Méthode.....	9
1.5 Organisation du mémoire.....	12
Chapitre 2 Représentation des phénomènes spatiaux de l'environnement marin dans les SIG.....	13
2.1 Représentation spatiale dans les SIG.....	13
2.1.1 Modélisation de l'espace géographique.....	14
2.1.2 Structures de données géométriques.....	15
2.1.2.1 Structures de données matricielles.....	16
2.1.2.2 Structures de données vectorielles.....	19
2.2 Application des SIG pour la représentation des objets spatiaux liés à l'environnement marin.....	22
2.2.1 Représentation d'objets aux limites floues	24
2.2.2 Représentation 3D.....	27
2.3 Conclusion du chapitre	31
Chapitre 3 Représentation des agrégations de poissons et de leur évolution spatio-temporelle en 2D	32
3.1 Notion d'ensembles flous	33
3.1.1 Théorie classique des ensembles	33
3.1.2 Théorie des ensembles flous.....	34
3.2 Modèles d'objets spatiaux flous	38
3.2.1 Approches basées sur des modèles exacts	38
3.2.2 Approches basées sur la théorie des ensembles flous.....	41
3.3 Approche proposée	43
3.3.1 Interpolation.....	46
3.3.2 Fuzzification	48

3.3.2.1	Définition de la fonction d'appartenance.....	48
3.3.2.2	Calcul des valeurs d'appartenance.....	50
3.3.3	Extraction de limites.....	50
3.4	Application de l'approche aux données de pêche.....	53
3.4.1	Données utilisées.....	54
3.4.2	Résultats obtenus.....	55
3.4.2.1	Fonction d'appartenance définie.....	56
3.4.2.2	Interpolation.....	58
3.4.2.3	Valeurs d'appartenance calculées.....	59
3.4.2.4	Représentation des limites des agrégations.....	59
3.4.3	Analyse des résultats.....	61
3.5	Application de l'approche pour l'étude de l'évolution spatio-temporelle des agrégations de poissons.....	64
3.6	Conclusion du chapitre.....	68
Chapitre 4	Représentation des agrégations de poissons et de leurs dynamiques en 3D.....	70
4.1	Approche intégrée pour la modélisation des agrégations de poissons en 3D.....	70
4.1.1	<i>Clustering</i> 3D.....	71
4.1.2	Modélisation 3D.....	73
4.2	Triangulation Delaunay 3D.....	77
4.2.1	Définition de la triangulation Delaunay 3D.....	77
4.2.2	Construction de la triangulation Delaunay 3D.....	79
4.2.3	Structuration des données.....	81
4.3	<i>Clustering</i> par l'algorithme des α -shapes en 3D.....	83
4.3.1	Description de l' α -shape 3D.....	84
4.3.2	Définition formelle de l' α -shape 3D.....	85
4.3.3	Calcul des α -shapes pour une distribution de points donnée.....	87
4.4	Application du modèle à la modélisation 3D d'agrégations de poissons en milieu marin.....	93
4.4.1	Détection automatique des agrégations de poissons en 3D.....	94
4.4.2	Visualisation et analyse des agrégations de poissons en 3D.....	95
4.4.3	Discussion.....	98
4.4.3.1	Étude du comportement et des dynamiques des poissons en 3D.....	98
4.4.3.2	Limites de l'approche.....	99
4.5	Conclusion du chapitre.....	102
Chapitre 5	Conclusion et perspectives.....	104
5.1	Conclusion.....	104
5.2	Perspectives futures.....	107
Bibliographie	109
Annexe A	Extrait du jeu de données utilisé pour la représentation 2D.....	119
Annexe B	Code implanté pour le <i>clustering</i> 3D (algorithme des α -shapes).....	124
Annexe C	Extrait du jeu de données utilisé pour la représentation 3D.....	143

Liste des tableaux

Tableau 4.1	Tables des intervalles des valeurs α pour lesquelles un simplexe appartient à l' α -complexe.	90
Tableau A.1	Extrait des relevés scientifiques du MPO, morue de l'Atlantique, 1990, région de Terre-Neuve.	119
Tableau A.2	Métadonnées.	120
Tableau C.1	Extrait des données acoustiques.	143

Liste des figures

Figure 1.1	Région d'étude : Atlantique du nord-ouest.....	1
Figure 1.2	Agrégation de poissons.....	4
Figure 1.3	(a) Carte thématique où l'abondance des poissons est montrée à l'aide de variables visuelles proportionnelles, (b) carte interpolée produite avec des méthodes géostatistiques, (c) représentation des agrégations de poissons à l'aide d'ellipses.....	8
Figure 1.4	Méthode proposée pour la réalisation du projet de recherche.....	10
Figure 2.1	Structure de données géométriques pour la représentation d'objets spatiaux.....	16
Figure 2.2	Structure de données matricielle utilisée pour représenter (a) un point, (b) une ligne, (c) un polygone.....	17
Figure 2.3	Attribution de la valeur des pixels dans une représentation matricielle : (a) en considérant qu'un pixel possède la valeur la plus répandue dans son intérieur ou (b) en considérant qu'un pixel possède la valeur qui se trouve à son point central.....	17
Figure 2.4	(a) Méthode de stockage <i>quadtree</i> et (b) méthode de stockage <i>run-length</i>	19
Figure 2.5	Primitives géométriques principales utilisées par la structure de données vectorielle : (a) point, (b) ligne, (c) polygone.....	19
Figure 2.6	Exemple de table pour un type d'objet ponctuel correspondant à une classe <i>poisson</i> ; chacune des lignes représente un poisson se trouvant dans une région donnée et chacune des colonnes correspond à l'une des propriétés liées aux poissons.....	21
Figure 2.7	Structure de données vectorielle topologique où les relations entre les différents objets, ici des polygones A et B, sont stockées dans une table.....	22
Figure 2.8	Différentes interprétations possibles de la limite d'une forêt sur une photographie aérienne.....	26
Figure 2.9	(a) Modélisation du fond océanique dans le logiciel Echoview à l'aide de données multifaisceaux; (b) visualisation de données multifaisceaux dans le logiciel Fledermaus et (c) visualisation 3D de paramètres océanographiques par krigeage 3D dans le logiciel EVS.....	29
Figure 2.10	Représentation d'une section verticale 2D d'un phénomène marin 3D.....	30
Figure 3.1	Comparaison entre (a) la fonction caractéristique d'un ensemble classique et (b) la fonction d'appartenance à un ensemble flou.....	35
Figure 3.2	Comparaison d'un ensemble classique et d'un ensemble flou extraite du Cahier Technique Schneider n°191.....	36
Figure 3.3	(a) Fonction d'appartenance et ses principaux paramètres (<i>noyau</i> , <i>support</i> et <i>limite</i>), (b) représentation graphique des paramètres.....	37
Figure 3.4	Illustration du modèle Egg-Yolk.....	39
Figure 3.5	Illustration du modèle de Clementini et Di Felice.....	40

Figure 3.6	Illustration du modèle de Zhan basé sur la théorie des ensembles flous	42
Figure 3.7	Agrégation de poissons montrée comme (a) un objet spatial discret, (b) un objet flou ayant un intérieur, une limite large et un extérieur, (c) un objet matriciel compris dans une surface continue qui détermine la possibilité d'appartenance de chaque pixel à l'agrégation.....	43
Figure 3.8	Principales étapes de traitement des données de l'approche proposée.....	44
Figure 3.9	(a) Fonction d'appartenance trapézoïdale et ses paramètres associés (a, b, c et d, quatre nombres réels tels que $a \leq b \leq c \leq d$), (b) forme mathématique de cette fonction, (c) fonction d'appartenance triangulaire qui constitue un cas particulier de la fonction d'appartenance trapézoïdale pour $b = c$	49
Figure 3.10	Illustration des différentes composantes et zones d'un objet flou	52
Figure 3.11	(a) Exemple d'objets flous (<i>vague region</i>) dont les <i>noyaux (flat)</i> et les <i>supports (area)</i> sont représentés par des polygones, (b) relations topologiques créées pour ces objets	53
Figure 3.12	Localisation des sondages scientifiques menés par Pêches et Océans Canada pour la morue de l'Atlantique pour l'année 1990 dans le sud de la région de Terre-Neuve.....	54
Figure 3.13	Zones de pêche de l'OPANO.....	55
Figure 3.14	(a) Forme de la fonction d'appartenance et (b) fonction d'appartenance exprimée en termes mathématiques.....	56
Figure 3.15	Résultat de l'interpolation par la méthode IDW pour les données d'observation dans la région d'étude; des artefacts créés par la méthode d'interpolation employée sont visibles dans le <i>support</i> des agrégations de poissons (points verts dans les sections orangées).....	58
Figure 3.16	Carte illustrant la position des <i>noyaux</i> et la dégradation des valeurs d'appartenance.....	59
Figure 3.17	Carte illustrant la position des limites de l'étendue minimale (bleu) et de l'étendue maximale (rouge) des agrégations de poissons.....	60
Figure 3.18	Carte illustrant la position des agrégations de poissons de morue en 1990....	61
Figure 3.19	Représentation hybride d'une agrégation de poissons comprenant une partie matricielle et une partie vectorielle.....	62
Figure 3.20	Représentation de l'évolution spatio-temporelle des morues de l'Atlantique (<i>Gadus morhua</i>) dans les divisions 2J et 3KL de l'OPANO pour la saison d'automne des années 1983 à 1992	66
Figure 4.1	Représentations basées sur des primitives volumiques : (a) structure de données régulière faite de voxels (<i>3D array</i>), (b) représentation <i>octree</i> , (c) modélisation géométrique CSG (<i>Constructive Solid Geometry</i>) représentant un objet 3D en combinant des objets solides simples (ex. cylindre, sphère) à l'aide d'opérateurs géométriques booléens (ex. union, intersection), (d) représentation par des tessellations irrégulières comme la triangulation Delaunay 3D (<i>3D TIN</i>).....	74
Figure 4.2	Exemple de représentation <i>octree</i>	75
Figure 4.3	Exemples (a) d'un ensemble convexe et (b) d'un ensemble non convexe.	77

Figure 4.4	Test géométrique <i>InSphere</i> (a) quatre sommets a, b, c et d d'un tétraèdre sur leur sphère circonscrite et point p à tester (b) calcul du déterminant correspondant.....	78
Figure 4.5	Tétraèdre issu de la triangulation Delaunay 3D de sept points.....	78
Figure 4.6	Simplexes de la triangulation Delaunay 3D.	79
Figure 4.7	Étapes de la construction de la triangulation Delaunay 3D par la méthode incrémentale.....	80
Figure 4.8	Différents cas de flips tridimensionnels.....	81
Figure 4.9	Structure de données basée sur les tétraèdres.	82
Figure 4.10	Test géométrique <i>Orient</i>	83
Figure 4.11	Exemple d'un segment (1-simplexe) α -exposé et non α -exposé.	86
Figure 4.12	Famille d' α -shapes 3D pour un certain jeu de données (a) enveloppe convexe de l'ensemble de points pour $\alpha = \infty$, (b) et (c) α -shape pour $0 < \alpha < \infty$, (d) ensemble de points pour $\alpha = 0$	87
Figure 4.13	(a) Exemples de complexe simplicial en 3D et (b) de structure qui n'est pas un complexe.....	88
Figure 4.14	Formation d'un α -complexe 3D par des simplexes de différentes dimensions	88
Figure 4.15	Exemple de segments singuliers et de segments réguliers dans l' α -complexe	89
Figure 4.16	Étapes pour extraire les agrégations de poissons dans un jeu de données. (a) Jeu de données, (b) triangulation Delaunay 3D, (c) raffinement du modèle avec un α quelconque, (d) résultat final.	95
Figure 4.17	Échogramme de données acoustiques montrant des poissons en milieu marin avec le temps sur l'axe x et la distance en mètres sur l'axe y.	97
Figure 4.18	Modèle 3D obtenu pour un jeu de données acoustiques réel.....	98
Figure B.1	Algorithme de <i>clustering</i> des α -shapes en 3D.	124

Chapitre 1 Introduction

1.1 Mise en contexte

Cette section présente le contexte général dans lequel ce projet de recherche s'inscrit. Elle propose d'abord une description de la situation passée et actuelle des pêches commerciales dans la région de l'Atlantique du nord-ouest. Puis, l'importance des agrégations de poissons et de leurs dynamiques, sujet de cette étude, est montrée. Ensuite, l'applicabilité des systèmes d'information géographique (SIG) pour la représentation de données marines de pêche et des agrégations de poissons est discutée.

1.1.1 Historique des pêches dans la région de l'Atlantique du nord-ouest

La colonisation du Canada, faite à partir de sa côte Est (voir région d'étude à la figure 1.1), a été intimement liée à l'industrie de la pêche. Dès le 16^e siècle, les navires européens sillonnaient les eaux du nord-ouest de l'Atlantique, du printemps à l'automne, pour y pêcher la morue et d'autres espèces de poissons. Il s'agissait de larges flottes de petits bateaux équipés de filets, de lignes à main, ainsi que d'autres équipements du même type. Les stocks de poissons étaient alors incroyablement abondants.



Figure 1.1 - Région d'étude : Atlantique du nord-ouest.

Au cours du 20^e siècle, la pêche a subi de grands bouleversements. Les progrès techniques ont permis la construction de nouveaux types de navires, tels que les chalutiers de fond, beaucoup plus gros et performants que les anciens. De plus, des équipements de plus en plus perfectionnés, comme des sondeurs acoustiques permettant de localiser les poissons ou des capteurs installés sur les filets, permettant de placer ces filets dans des environnements propices aux poissons, ont fait leur apparition à bord des bateaux. Cela a permis d'accroître l'efficacité de la pêche et ainsi d'augmenter le volume de poissons extraits des océans. Les navires de pêche se sont aussi transformés en usines flottantes permettant la transformation et la congélation automatisée des prises directement en mer, éliminant de cette façon une grande part des délais et améliorant la productivité de l'industrie. Par exemple, le chalutier irlandais *Atlantic Dawn*, le plus gros au monde, peut stocker près de 7000 tonnes de poisson à son bord (Marine Marchande, 2008).

Toutes les innovations ont fait que, dans les années 1960, les prises annuelles de morue ont grimpé à presque 2 000 000 tonnes en 1967-1968 alors qu'elles étaient d'approximativement 900 000 tonnes une dizaine d'années auparavant (Pêches et Océans Canada, 2004). Cet accroissement résulte principalement de l'augmentation de la pêche faite par les bateaux non-canadiens pendant cette période. Elles ont ensuite diminué de manière importante après le milieu des années 1970 pour tomber à moins de 500 000 tonnes en 1977 (Pêches et Océans Canada, 2004).

En 1977, le Canada a étendu sa zone de juridiction des pêches à 200 milles marins de ses côtes. Cela permettait au pays de gérer ses stocks de poissons et d'imposer des quotas de pêche aux navires étrangers. Les prises de poissons en territoire canadien se sont alors remises à augmenter graduellement jusqu'à la fin des années 1980 (Atkinson *et al.*, 1997). Cette reconstruction apparente des stocks de poissons a fait croire aux gens qu'ils pouvaient en modifier la taille facilement en augmentant ou en réduisant l'effort de pêche dans certaines zones (Rice *et al.*, 2003).

Toutefois, malgré la création de la zone de pêche de 200 milles marins, le Canada n'a pas réussi à assurer, à long terme, la reconstitution de ses stocks de morues. En effet, vers 1990,

les stocks de morues se sont effondrés à nouveau (Atkinson *et al.*, 1997; Pêches et Océans Canada, 2004). Face à cette situation, un moratoire interdisant officiellement la pêche à la morue a été annoncé par le gouvernement canadien en 1992. Cela s'est traduit par la mise au chômage de près de 40 000 pêcheurs et travailleurs du secteur de la transformation de l'est du Canada (Radio-Canada, 2008).

Malgré le moratoire imposé, peu de signes de reconstruction des stocks de morues ont été observés (Atkinson *et al.*, 1997; Rose, 2004; Fletcher, 2008) et l'industrie s'est tournée vers de nouvelles espèces telles que la crevette, le crabe et le homard. Ces autres espèces sont toutefois intensivement pêchées à leur tour.

Le déclin des stocks de morues dans la région de l'Atlantique du nord-ouest est expliqué en grande partie par la surpêche (Hutchings, 1996; Rose et Kulka, 1999), mais peut également être lié à d'autres facteurs tels que des modifications dans les conditions environnementales et dans la chaîne trophique (Rose *et al.*, 1994; Drinkwater, 2002; Rose, 2004; Pêches et Océans Canada, 2004). En effet, certains changements ont été observés dans l'écosystème de la région nord-ouest de l'Atlantique depuis 1990 (Canada/Newfoundland and Labrador Action Team for Cod Recovery, 2005). Par exemple, il y a eu des changements de température et de concentration en oxygène de l'eau ainsi que des modifications dans les relations entre les prédateurs et les proies.

Les chercheurs en biologie marine et en halieutique (i.e. en rapport avec la pêche) étudient depuis plusieurs années ce problème afin de trouver des pistes d'explication et de développer des stratégies pour le rétablissement des stocks de morues et d'autres types de poissons. Une compréhension des dynamiques de ces stocks est toutefois essentielle pour réaliser un tel objectif.

1.1.2 Agrégations de poissons

La dynamique des stocks de poissons inclut, entre autres, les phénomènes d'agrégation et de désagrégation des poissons à différentes échelles spatiales et temporelles. Le phénomène d'agrégation est très complexe et peut dépendre de plusieurs facteurs dont l'âge des

poissons (Bennett, 2008). Il est, pour plusieurs espèces, une propriété bioécologique fondamentale typiquement reliée à la reproduction, à l'alimentation et à la défense contre les prédateurs.

Une agrégation peut être décrite comme un certain nombre de poissons qui se sont agglomérés et qui forment un groupe, faisant ainsi augmenter radicalement la densité en poissons d'un certain secteur par rapport à d'autres endroits dans une région d'étude donnée (Rose, 2006). À titre d'exemple, dans la période de frai, Domeier et Colin (1997) définissent une agrégation comme un groupe de poissons de la même espèce regroupés dans le but de frayer, avec une densité et un nombre significativement plus élevé que pendant les périodes de non reproduction de l'espèce dans le même secteur. La figure 1.2 montre une agrégation de poisson. Toutefois, dans cette recherche, le terme agrégation n'implique pas nécessairement que les poissons se sont agglomérés pour des raisons sociales et qu'ils sont liés les uns aux autres. En effet, les poissons se regroupent socialement dans de nombreux cas, mais ils peuvent également former une agrégation dans une région précise uniquement parce qu'il y a une forte concentration de proies ou un environnement physique favorable, sans forcément vouloir interagir avec les autres. Le terme agrégation est donc utilisé ici dans un sens général et décrit seulement le fait que les poissons sont ensemble, sans supposer qu'ils le sont selon une certaine structure ou pour une certaine fonction ou cause (Rose, 2006).



Figure 1.2 - Agrégation de poissons (Florida Museum of Natural History, 2008).

La taille d'une agrégation de poissons peut varier grandement. En effet, certaines agrégations ne mesurent que quelques mètres alors que d'autres peuvent atteindre une taille de l'ordre de plusieurs dizaines de kilomètres (Mello et Rose, 2005). Par exemple, Rose (2007, p. 470, traduction libre) décrit une agrégation de morues observée en juin 1990 dans le corridor de Bonavista, à l'est de Terre-Neuve: « L'agrégation couvrait une superficie d'environ 30 kilomètres par 20 kilomètres, avait une profondeur moyenne d'environ 25 mètres et contenait environ 450 000 tonnes de morues (environ 570 000 000 poissons), ce qui représentait approximativement 80% du stock. » Les agrégations de poissons peuvent donc avoir lieu tant à une échelle locale qu'à une échelle plus régionale.

1.1.3 Systèmes d'information géographique

Des changements observés dans le phénomène d'agrégation des poissons décrit dans la section précédente peuvent être instructifs au sujet de la condition des stocks (Rose, 1993). Afin d'analyser et de visualiser ce phénomène, les SIG peuvent être employés. En effet, les SIG constituent un outil intéressant pour manipuler l'information spatiale et sont fréquemment utilisés dans diverses disciplines allant de l'ingénierie aux sciences sociales.

Les applications de SIG à des données liées à l'environnement marin n'ont connu un essor que relativement récemment. Les zones côtières ont été étudiées plus en détails puisqu'elles sont le lieu de nombreuses activités économiques et sont, de manière générale, des systèmes densément peuplés présentant des caractéristiques complexes. Toutefois, la majeure partie des océans et les processus qui s'y déroulent restent largement inexplorés à ce jour, malgré le fait que beaucoup de scientifiques tels que des biologistes, des géologues ou des chimistes s'intéressent au milieu marin depuis de nombreuses années. Il existe donc, à l'heure actuelle, un nombre assez important de données d'observation et de modèles statistiques portant sur les océans. Cependant, ces informations ne sont pas exploitées dans leur totalité. L'utilité d'un SIG pour le traitement de ces données est donc indéniable puisque ce type de système peut rendre ces jeux de données compatibles et offre la possibilité de les combiner afin d'en tirer de nouvelles informations ou interprétations.

Bien que les SIG soient encore rarement employés pour la gestion des pêches et pour la recherche dans ce domaine, un certain nombre d'organismes les emploient pour représenter et analyser des données marines. Dans différents pays, des initiatives pour l'utilisation des SIG pour soutenir la gestion complexe des pêches ont été lancées dans la dernière décennie (Valavanis, 2002). Les SIG sont principalement employés pour intégrer des données de diverses sources et pour les convertir en information significative. Par exemple, les SIG servent à produire des cartes montrant l'abondance des poissons dans certaines régions basées sur des relevés et échantillonnages de pêche, des cartes de frayères ou encore des cartes d'habitats saisonniers. Les SIG sont également utilisés pour combiner les données de pêche avec des données environnementales, telles que la bathymétrie, les caractéristiques du fond marin et la température de l'eau, afin de fournir des outils intégrés pour supporter la gestion des pêches (Meaden, 2000; Wright et Bartlett, 2000; Valavanis, 2002; Fisher et Rahel, 2004). Un exemple d'une grande initiative de production d'un SIG pour la pêche est le Système mondial d'information sur les pêches (FIGIS)¹ développé par l'Organisation des Nations Unies pour l'alimentation et l'agriculture (FAO)². Ce système, basé sur Internet, a pour objectif de fournir de l'information sur le statut des pêches et sur les tendances à une échelle globale. Plusieurs autres portails Web, tels que OBIS (système d'information biogéographique sur l'océan)³, permettent d'accéder à des données de pêche géoréférencées avec quelques fonctionnalités cartographiques de base. Donc, les outils de représentation, de visualisation et d'analyse spatiale des données de pêches deviennent de plus en plus communs et nécessaires dans le domaine marin (Kulka *et al.*, 2005).

1.1.4 Représentation des agrégations de poissons dans les SIG

Malgré le fait que plusieurs études (Rose et Kulka, 1999; Mello et Rose, 2005; Bennett, 2008) aient déjà été réalisées sur les agrégations de poissons, la dynamique spatiale et temporelle de ces agrégations, ainsi que leur relation avec l'environnement physique, restent généralement inexpliqués. Une des raisons est le manque d'outils permettant la représentation spatiale des agrégations de poissons et de leur évolution dans l'espace et dans le temps. En effet, il existe quelques méthodes statistiques pour quantifier les

¹ <http://www.fao.org/fishery/figis/2/en>

² <http://www.fao.org/>

³ <http://www.iobis.org/>

agrégations (ex. indices en géostatistique, coefficient de Gini, indices de variance spatiale), mais très peu de techniques pour leur représentation.

Pourtant, la compréhension du phénomène d'agrégation des poissons revêt une très grande importance d'un point de vue écologique. En effet, sans connaissance de l'écologie liée à cette dynamique majeure des poissons, il est difficile de mettre en place des politiques dans le but de préserver ou reconstruire les stocks.

Par ailleurs, l'aspect géographique des agrégations de poissons est très important pour leur compréhension. En effet, des représentations telles que des cartes de distribution permettent de présenter des informations que les indices statistiques seuls n'exposent pas (Bennett, 2008).

Traditionnellement, la cartographie et la représentation des agrégations de poissons dans les SIG ont été limitées à des cartes thématiques horizontales 2D (pour les agrégations régionales) ou à des profils acoustiques verticaux 2D (pour les agrégations locales). Les données provenant d'un espace tridimensionnel (océan) sont alors projetées sur des surfaces en deux dimensions. La méthode de représentation des agrégations de poissons la plus répandue consiste à cartographier l'abondance en poissons dans une région donnée en utilisant des variables visuelles proportionnelles (figure 1.3a) (Atkinson *et al.*, 1997) ou en créant des surfaces à l'aide de techniques d'interpolation appliquées sur des données d'échantillonnage de la densité ou de l'abondance en poissons (figure 1.3b) (Kulka *et al.*, 2005). Les techniques géostatistiques sont d'ailleurs maintenant fréquemment employées dans le domaine des pêches afin de créer ce type de surfaces. Les géostatistiques servent également à déduire des relations spatiales à partir des données de pêche ou à prédire des phénomènes spatio-temporels (Valavanis, 2002). Les agrégations de poissons peuvent aussi être représentées à l'aide d'autres techniques comme des ellipses statistiques dessinées sur des données d'abondance (figure 1.3c) (Atkinson *et al.*, 1997). Ces ellipses sont utilisées pour décrire la tendance centrale dans les données de relevés scientifiques et leur surface représente 90% de l'abondance observée (Bennett, 2008). Cependant, parmi ces techniques, aucune ne permet de quantifier les agrégations de poissons en termes d'étendue spatiale.

Aussi, la plupart des cartes basées sur des données de pêche sont réalisées à des échelles régionales qui ne sont pas toujours adéquates pour représenter avec exactitude et précision les limites des agrégations, qui sont un phénomène plutôt localisé. De plus, la réalisation de cartes à l'aide des techniques décrites précédemment rend généralement l'étude de l'évolution des agrégations de poissons difficile.

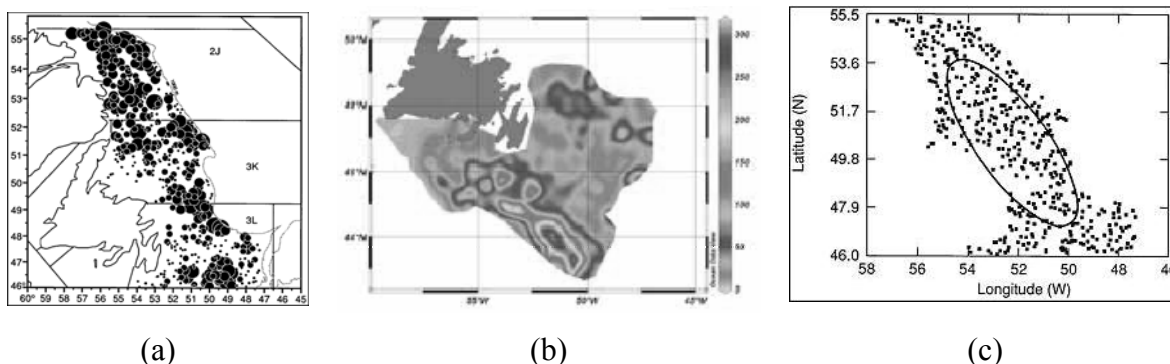


Figure 1.3 - (a) Carte thématique où l'abondance des poissons est montrée à l'aide de variables visuelles proportionnelles (Atkinson *et al.*, 1997), (b) carte interpolée produite avec des méthodes géostatistiques (Kulka *et al.*, 2005), (c) représentation des agrégations de poissons à l'aide d'ellipses (Atkinson *et al.*, 1997).

1.2 Problématique et hypothèse

Bien que la plupart des chercheurs en géomatique marine reconnaissent que les SIG commerciaux peuvent apporter des bénéfices importants à leur discipline, ils s'accordent néanmoins pour dire que ces outils sont limités pour une représentation réaliste des phénomènes qu'ils étudient (Wright et Bartlett, 2000). En effet, le caractère inhérent aux données de pêche pose quelques problèmes uniques (Meaden et Do Chi, 1996; Shyue et Tsai, 1996; Valavanis, 2002) :

- les données de pêche sont liées à des phénomènes et des objets dont les limites ne peuvent pas être précisément définies (limites floues);
- les données de pêche sont tridimensionnelles et dynamiques.

La représentation du phénomène d'agrégation des poissons dans les SIG présente donc certaines difficultés. À la lumière de ces observations, il apparaît nécessaire que les modes de représentation offerts dans les SIG actuels soient améliorés afin de pouvoir traiter de nouveaux types d'information tels que les données de pêche et les phénomènes qui leur sont liés. Ainsi, l'hypothèse principale posée dans ce mémoire est que des approches

présentant l'aspect 3D et dynamique des agrégations de poissons ainsi que leurs limites floues permettraient de représenter plus fidèlement ces agrégations que les modes de représentations existants.

1.3 Objectifs

L'objectif général de cette recherche est de développer des méthodes pouvant permettre d'améliorer la représentation spatiale des agrégations de poissons tant en 2D qu'en 3D afin de pouvoir éventuellement étudier leurs dynamiques spatio-temporelles. Ce mémoire vise ainsi à améliorer les procédés de représentation des agrégations de poissons disponibles dans les SIG et autres outils commerciaux afin que ceux-ci soient systématiques et que les résultats obtenus soit plus faciles à interpréter, de façon quantitative et en termes d'étendue spatiale, lors de leur visualisation.

De manière à atteindre le but établi, l'objectif principal se compose des objectifs spécifiques suivants:

- à une échelle régionale, représenter des agrégations de poissons en 2D en se basant sur une approche reposant sur les modèles d'objets spatiaux flous;
- à une échelle locale, représenter des agrégations de poissons en 3D selon une approche basée sur une structure de données dynamique;
- appliquer et valider les approches proposées pour la représentation des agrégations de poissons en utilisant des données halieutiques.

1.4 Méthode

Une méthode de recherche a été proposée afin d'atteindre les objectifs identifiés dans la section précédente. Cette méthode comporte plusieurs étapes, certaines étant préalables à d'autres et certaines pouvant être effectuées en parallèle, comme l'illustre la figure 1.4.

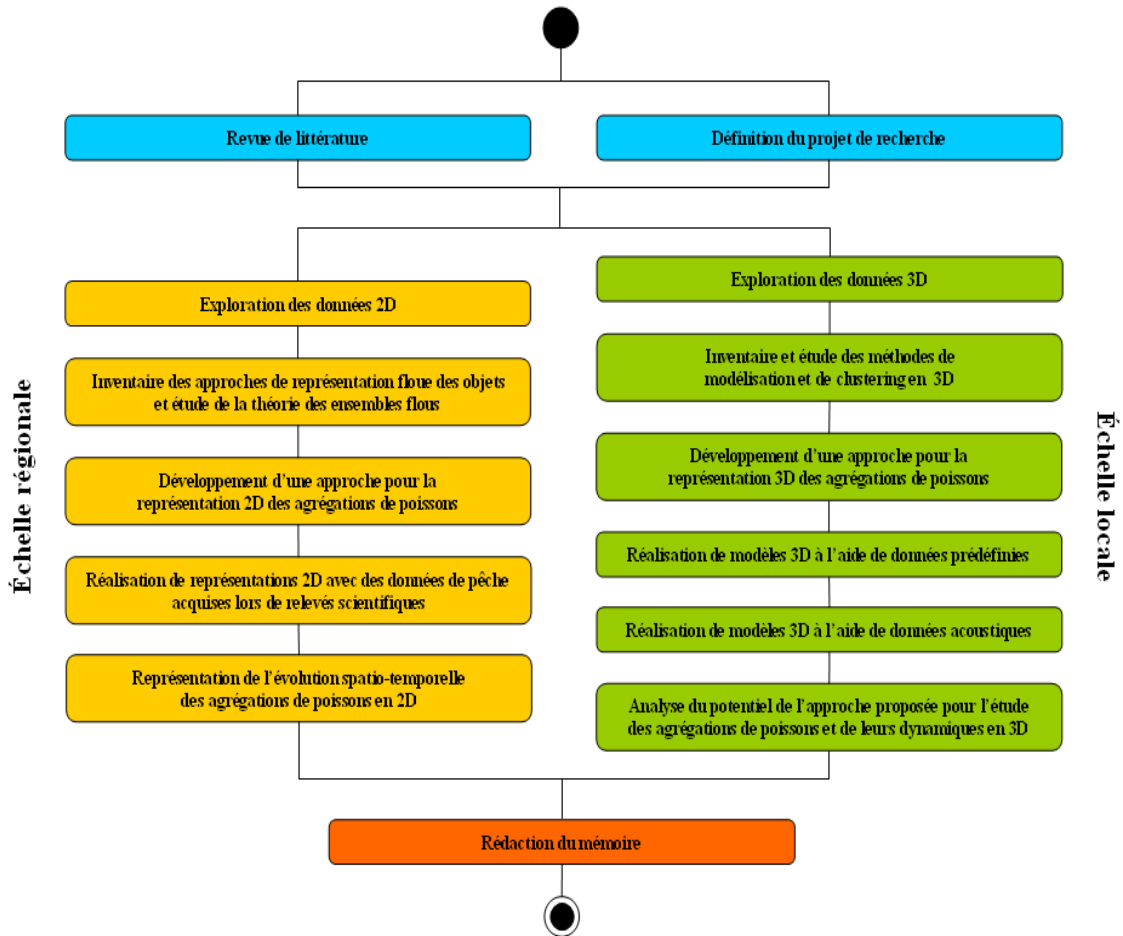


Figure 1.4 - Méthode proposée pour la réalisation du projet de recherche.

La démarche proposée a débuté par une approche théorique servant à circonscrire le projet de recherche (cf. boîtes bleues sur la figure 1.4). Une revue de littérature a été réalisée, orientée principalement sur l'application des SIG aux données marines, afin de mieux comprendre l'intérêt de leur utilisation dans le domaine des pêches et les défis y étant reliés. Une revue du contexte général des pêches en Atlantique du nord-ouest a aussi été faite. À l'aide de ces informations, la problématique à résoudre dans le projet de recherche ainsi que les objectifs ont pu être définis.

Suite à l'établissement des objectifs de recherche, une exploration des méthodes possibles de représentation des agrégations de poissons a été mise en œuvre. Tel que brièvement mentionné dans la section 1.1.2, les agrégations de poissons peuvent être perçues à différentes échelles : l'échelle régionale (~10-100 km) et l'échelle locale (~1-100 m). Les

termes « échelle régionale » et « échelle locale » ont été préférés dans ce mémoire à « petite échelle » et « grande échelle » étant donné la confusion existant entre la définition cartographique de ces termes et l'usage commun qui en est fait. Dans chacun des cas, les données et les méthodes de représentation sont distinctes.

À une échelle régionale, les agrégations de poissons ont été représentées sous la forme de cartes en 2D (cf. boîtes jaunes sur la figure 1.4). Pour ce faire, les données de pêche provenant de relevés scientifiques ont été explorées afin d'en découvrir les principaux paramètres (ex. date, latitude, longitude, âge, poids et nombre de poissons pêchés). Les concepts liés à la théorie des ensembles flous ont été étudiés lors d'une revue de littérature et les approches permettant de produire des représentations floues ont été inventoriées. En se basant sur ces approches, une méthode pour la représentation 2D des agrégations de poissons a été développée. Suivant cette méthode, plusieurs représentations des agrégations ont été obtenues à l'aide des données disponibles pour la région de l'Atlantique du nord-ouest. Les résultats obtenus ont été validés et l'évolution spatio-temporelle des agrégations de poissons a été visualisée en 2D à l'aide de ces représentations.

À une échelle locale, les agrégations de poissons ont été représentées en 3D (cf. boîtes vertes sur la figure 1.4). Pour y arriver, des données de pêche 3D acquises par des méthodes acoustiques ont été explorées afin de connaître leurs principaux attributs (ex. latitude, longitude, profondeur, force du signal). Puis, les notions théoriques relatives aux structures de données disponibles dans les SIG et à leur généralisation en 3D ont été révisées. Les algorithmes de *clustering* dans un espace 3D ont également été étudiés. À l'aide de ces connaissances, une approche intégrée a été proposée pour la représentation des agrégations de poissons en 3D. La méthode trouvée a d'abord été appliquée à des jeux de données artificiels et a ensuite été utilisée pour créer des modèles 3D à l'aide des jeux de données de pêche obtenus par des méthodes acoustiques. Enfin, le potentiel de l'approche proposée pour l'étude des agrégations de poissons et de leurs dynamiques en 3D a été analysé.

La dernière étape de la méthode proposée pour la réalisation du projet de recherche est la rédaction du mémoire (cf. boîte orange à la figure 1.4).

1.5 Organisation du mémoire

Ce mémoire présente les différentes approches développées pour identifier et représenter des agrégations de poissons à diverses échelles, en 2D et en 3D.

Le chapitre 2 propose une revue des concepts théoriques liés à la représentation des phénomènes spatiaux dans les SIG, particulièrement dans le contexte d'utilisation de données marines et halieutiques. Ce chapitre aborde la modélisation de l'espace dans les SIG. Les phénomènes spatiaux peuvent en effet être perçus comme étant continus ou discrets dans l'espace. Les principales structures de données disponibles dans les SIG, soient la structure matricielle et la structure vectorielle, y sont également présentées. Enfin, les limites des SIG pour la manipulation et la modélisation des données marines, liées à des objets tridimensionnels et aux limites floues, sont discutées.

Le chapitre 3 traite de la représentation des agrégations de poissons en 2D, à une échelle régionale. Les notions liées à la logique floue et à son application dans les SIG sont exposées et la méthode de représentation développée est expliquée. Ensuite, des exemples d'application de la méthode sont montrés sur des données de pêche. Puis, un exemple d'étude de l'évolution spatio-temporelle des agrégations de poissons est donné à partir des résultats obtenus auparavant.

Le chapitre 4 aborde la représentation des agrégations de poissons en 3D, à une échelle locale. L'approche proposée est exposée et les notions liées à la structure de données et aux algorithmes employés sont présentées. Les bénéfices et les inconvénients de l'approche proposée sont également discutés et des exemples de représentation en 3D sont donnés.

Le dernier chapitre de ce mémoire, le chapitre 5, présente un résumé général et les perspectives futures de cette recherche.

Chapitre 2 Représentation des phénomènes spatiaux de l'environnement marin dans les SIG

Un système d'information géographique (SIG) peut être défini comme un logiciel capable d'assembler, stocker, manipuler, et afficher l'information à référence spatiale (Longley *et al.*, 2005). Les caractéristiques des SIG reposent donc essentiellement sur leur capacité à gérer des bases de données, à effectuer des analyses spatiales et à produire des cartes numériques. Depuis leur apparition dans les années 1960, les SIG ont surtout été développés pour des besoins de cartographie terrestre, entre autres dans les domaines de l'urbanisme, du cadastre, du transport et de la gestion de l'occupation des sols. Toutefois, les SIG sont de plus en plus utilisés pour d'autres applications comme l'étude de phénomènes biologiques et écologiques se déroulant dans l'environnement marin.

Ce chapitre aborde principalement le sujet de la représentation de phénomènes spatiaux à l'aide des modèles et des primitives géométriques disponibles dans les SIG existants. Il décrit les méthodes traditionnellement employées pour représenter l'information géographique dans les SIG et expose leurs limites. La section 2.1 traite de quelques notions théoriques nécessaires à la représentation spatiale dans les SIG. S'ensuit une discussion sur les limites associées à la représentation des phénomènes spatiaux dans les SIG dans la section 2.2. Celle-ci est basée sur les problématiques liées à l'emploi de données marines dans les SIG.

2.1 Représentation spatiale dans les SIG

Cette section présente les notions théoriques associées à la représentation spatiale dans les SIG. Dans un premier temps, la section 2.1.1 aborde le sujet de la modélisation de l'espace géographique. Ensuite, les deux principales structures de données géométriques pouvant être employées pour intégrer l'information géographique dans un système informatique sont présentées dans la section 2.1.2.

2.1.1 Modélisation de l'espace géographique

Afin de représenter l'espace géographique dans un SIG, il est nécessaire d'en synthétiser la complexité. Pour ce faire, l'espace géographique peut être modélisé dans le SIG suivant deux visions fondamentalement différentes : des champs (*fields*) et des objets (*objects*) (Cova et Goodchild, 2002; Longley *et al.*, 2005).

Ainsi, l'espace géographique peut être perçu comme un *champ* (*field*) pour lequel de l'information descriptive (valeurs d'attributs) est assignée à chaque position spatiale. Suivant cette vision, un phénomène géographique est décrit par un nombre plus ou moins grand de variables pouvant être mesurées en chaque point de l'espace et dont la valeur varie de position en position. Le champ modélise la continuité observée dans l'espace géographique (Burrough, 1996). Cette manière de concevoir l'espace géographique est donc généralement bien adaptée pour la représentation de phénomènes géographiques continus, tels que la variation de la température de l'air ou l'altitude dans une région donnée. Cependant, les entités spatiales renfermées dans un champ peuvent aussi être retrouvées en regroupant les éléments de la représentation (pixels) possédant des propriétés semblables.

D'un autre côté, l'espace géographique peut être vu comme étant vide, à l'exception des endroits où il est peuplé d'*objets* spatiaux discrets. Ces objets spatiaux possèdent des limites bien définies et une étendue spatiale finie. Ils peuvent donc être comptés. Les propriétés de l'espace à l'intérieur d'un objet spatial sont données par des attributs descriptifs dont les valeurs sont assumées constantes pour toute son étendue (Erwig et Schneider, 1997; Schneider, 1999; Pauly et Schneider, 2004). De plus, une géométrie (forme et position spatiale) lui est associée. Par exemple, un polygone représentant une maison pourra être formé de quatre points reliés entre eux pour former un rectangle et avoir les attributs adresse civique, superficie et valeur associés à l'ensemble de l'objet. En fait, l'objet spatial partitionne l'espace relié à un objet donné en un intérieur, une limite et un extérieur (sauf pour les objets ponctuels qui n'ont pas de limite) qui sont mutuellement exclusifs et qui remplissent tout cet espace (Egenhofer, 1989). Les objets spatiaux individuels peuvent interagir entre eux en étant superposés ou contenus les uns dans les

autres par exemple. Chaque objet spatial peut aussi être localisé par rapport à un autre objet du même modèle. Il existe plusieurs phénomènes géographiques qui peuvent être modélisés par des objets spatiaux disséminés dans l'espace. Cette vision de l'espace géographique est généralement adoptée pour la représentation d'entités existant dans la réalité comme des lacs, des routes ou des édifices, ou encore pour la modélisation d'entités immatérielles telles que des pays ou des lots cadastraux.

La nature de certains phénomènes géographiques est représentée de manière plus adéquate par des champs continus plutôt que par des objets discrets et *vice-versa*. Toutefois, certains phénomènes géographiques peuvent être modélisés à la fois de manière continue et discrète. Par exemple, un phénomène géographique tel que l'élévation d'un terrain peut être représenté par une surface continue ou à l'aide de courbes de niveaux (ensemble d'objets discrets) selon le contexte (Cross, 1999). Aussi, pour la représentation de phénomènes géographiques dynamiques, l'usage intégré des deux approches de modélisation est souvent souhaitable (Yuan, 2001; Cova et Goodchild, 2002). Par exemple, à première vue, un feu de forêt est perçu comme un objet discret, son front étant bien défini. Cependant, des variations spatiales et temporelles sont observables à l'intérieur de celui-ci. La planification des opérations d'extinction nécessite donc autant une représentation de type *objet* que de type *champ* des feux de forêts (Yuan, 2001). Les infestations d'insectes, les précipitations et les ouragans sont d'autres exemples de phénomènes géographiques dynamiques requérant une représentation combinant les deux visions de l'espace.

2.1.2 Structures de données géométriques

Les champs (continus) et les objets (discrets) sont deux vues conceptuelles (modélisations) de l'espace géographique. Les ordinateurs étant limités, ces modélisations ne peuvent pas être implantées directement (Longley *et al.*, 2005). Ainsi, pour représenter l'espace géographique sous une forme digitale, des structures de données géométriques doivent être utilisées.

Les deux structures de données géométriques les plus employées dans le domaine des SIG sont les structures dites matricielles et vectorielles. Bien que ces structures soient tout à fait

différentes, elles sont toutes deux attrayantes entre autres parce qu'elles permettent une manipulation des données spatiales relativement aisée, qu'elles se basent sur un raisonnement mathématique conventionnel et qu'elles peuvent être facilement implantées dans un SIG (Tang, 2004; Longley *et al.*, 2005).

Dans les SIG, les objets sont généralement représentés à l'aide de structures de données géométriques vectorielles, alors que les champs ont tendance à être représentés à l'aide de structures de données géométriques matricielles. Toutefois, les structures de données géométriques ne sont pas nécessairement liées au type de modélisation employé pour décrire l'espace géographique. Ainsi, pour la représentation d'un ensemble d'objets discrets, l'une ou l'autre des structures de données géométriques peut être utilisée, tel qu'exprimé à la figure 2.1. Le choix de la structure de données géométrique se fait alors de manière à faciliter l'analyse spatiale et selon des critères tels que le mode d'acquisition des données, la complexité des traitements à effectuer ou encore le niveau de performance requis (Abdul-Rahman et Pilouk, 2007).

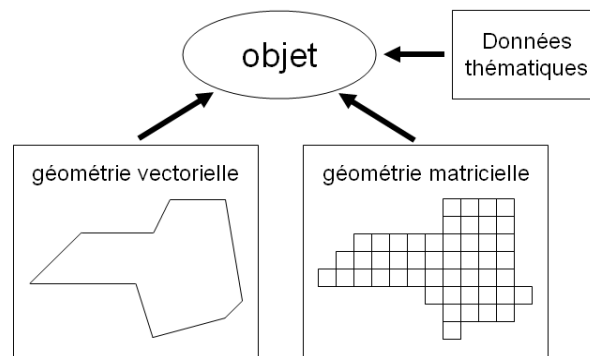


Figure 2.1 - Structure de données géométriques pour la représentation d'objets spatiaux (traduit de Molenaar, 1998).

2.1.2.1 Structures de données matricielles

La structure de données matricielle est constituée d'une collection de points ou de cellules qui couvrent l'espace géographique et forment une grille régulière composée de lignes et de colonnes. Les cellules d'une structure de données matricielle sont habituellement désignées par le terme pixel. L'emplacement de chaque pixel est défini par le numéro de sa ligne et de

sa colonne. À chaque pixel de la grille est rattachée de l'information descriptive (valeurs d'attributs). Toutes ces informations, soient les coordonnées de la position des pixels dans la grille ainsi que les valeurs d'attributs, sont stockées dans le SIG. Une illustration de la représentation d'objets (point, ligne, polygone) suivant une structure de données matricielle est présentée à la figure 2.2.

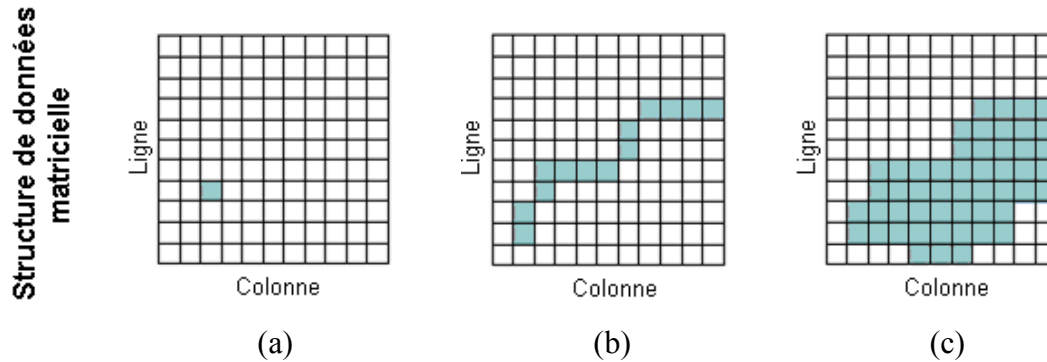


Figure 2.2 - Structure de données matricielle utilisée pour représenter (a) un point, (b) une ligne, (c) un polygone (traduite de UNESCO Ocean Teacher, 2007).

Lorsque l'information géographique est représentée sous forme matricielle, les détails de la variation observée à l'intérieur des pixels sont généralement perdus puisqu'un pixel ne peut normalement posséder qu'une seule valeur pour un même attribut (Longley *et al.*, 2005). Les méthodes couramment admises pour attribuer une valeur à un pixel sont de considérer que ce pixel possède la valeur qui se trouve à son point central ou encore la valeur la plus répandue dans son intérieur (Longley *et al.*, 2005), tel que montré à la figure 2.3.

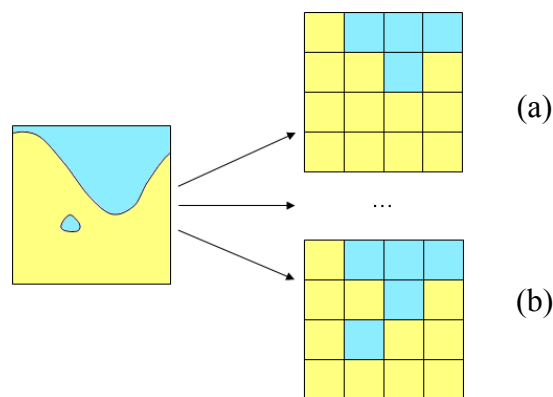


Figure 2.3 - Attribution de la valeur des pixels dans une représentation matricielle : (a) en considérant qu'un pixel possède la valeur la plus répandue dans son intérieur ou (b) en considérant qu'un pixel possède la valeur qui se trouve à son point central.

La topologie d'une structure de données matricielle est basée sur les relations topologiques existant entre les éléments qui la composent, soit l'adjacence entre les pixels qui forment la grille (Molenaar, 1998). Les relations topologiques ne sont donc pas explicitement exprimées. Ainsi, dans une structure de données matricielle, un objet peut être reconstitué en amalgamant les pixels adjacents ayant des valeurs d'attributs semblables. Toutefois, ses limites ne sont pas représentées précisément.

La taille des cellules de la grille définit sa résolution spatiale. Plus les cellules sont petites, plus la résolution de la représentation matricielle est haute. Une représentation matricielle ayant une haute résolution spatiale décrit alors des objets avec un plus grand niveau de détail. Cependant, comme un plus grand nombre de cellules sont nécessaires pour couvrir l'espace géographique, une représentation matricielle à haute résolution géométrique nécessite un espace de stockage très volumineux. Le choix de la résolution géométrique à employer doit donc être jugé selon le niveau de détail requis pour les analyses spatiales souhaitées, par rapport à l'espace de stockage à disposition dans l'outil informatique utilisé. La résolution spatiale de la représentation matricielle doit également être ajustée selon la variabilité des données à disposition, soit la résolution thématique. En effet, une faible variabilité des valeurs d'attributs permet l'emploi d'une résolution géométrique plus faible qu'une grande variabilité de celles-ci (Molenaar, 1998).

Les données matricielles peuvent être stockées selon différentes techniques, notamment dans le but de réduire le volume des fichiers les contenant (compression de données). Ces techniques de stockage des données matricielles sont montrées à la figure 2.4. Par exemple, la méthode de stockage par quadrants arborescents (*quadtree*) peut être utilisée (figure 2.4a). Cette technique hiérarchique découpe l'espace par une règle de quatre. Elle peut également être employée comme méthode d'accès à l'information dans une base de données. La méthode de codage par plages (*run-length*) est aussi utilisée pour le stockage de données matricielles (figure 2.4b). Elle stocke, ligne par ligne, le nombre de pixels de même valeur se suivant plutôt que stocker les données matricielles sous leur forme originale.

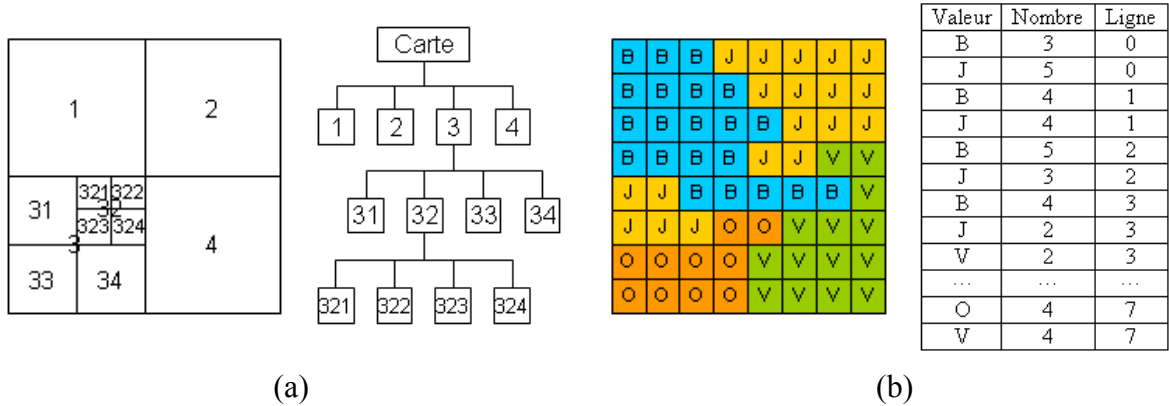


Figure 2.4 - (a) Méthode de stockage *quadtree* et (b) méthode de stockage *run-length*.

Le principal avantage de la structure de données matricielle est sa simplicité. De plus, les représentations matricielles peuvent être facilement superposées et, ainsi, certaines analyses spatiales intéressantes peuvent être effectuées en réalisant des opérations algébriques sur les valeurs des cellules. Aussi, les relations spatiales entre les pixels sont généralement faciles à déduire. Par contre, la structure matricielle est moins appropriée pour représenter des objets spatiaux individuels puisque ceux-ci perdent leur intégrité dans ce mode de représentation.

2.1.2.2 Structures de données vectorielles

La structure de données vectorielle représente individuellement chaque objet spatial à l'aide de primitives géométriques de base telles que des points, des lignes, des polygones et des polyèdres en 3D (Longley *et al.*, 2005). La figure 2.5 illustre les primitives géométriques disponibles pour une représentation vectorielle en deux dimensions.

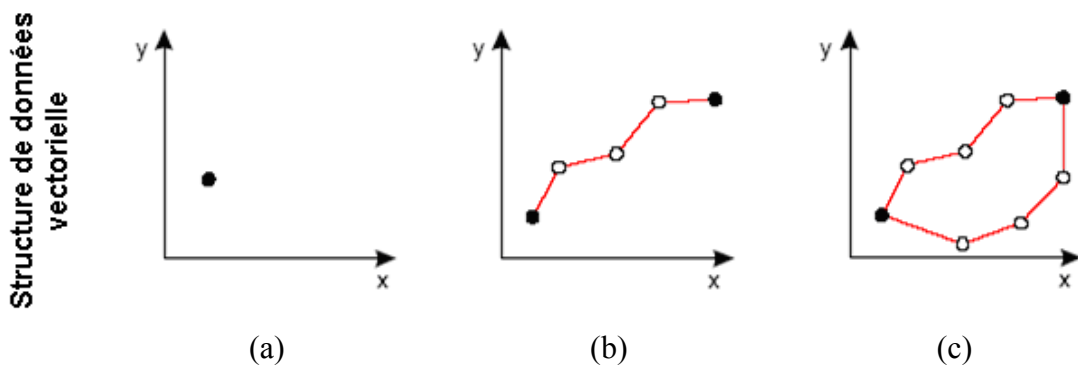


Figure 2.5 - Primitives géométriques principales utilisées par la structure de données vectorielle : (a) point, (b) ligne, (c) polygone (traduite de UNESCO Ocean Teacher, 2007).

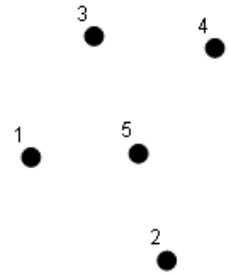
Tel que montré à la figure 2.5, la structure de données vectorielle tend à exprimer les caractéristiques linéaires des objets. Les objets simples sont alors décrits par une série d'une ou de plusieurs coordonnées dans un système de référence spatial donné. Par exemple, un objet ponctuel ne possède pas de dimension spatiale (0D) et sa position est donnée par une coordonnée géographique. Un objet linéaire possède une seule dimension spatiale (1D). Il a une longueur et une forme qui correspond à des paires de points connectées par des lignes. Un objet surfacique possède deux dimensions spatiales (2D) et sa géométrie est donnée par sa limite. Il est formé, comme un objet linéaire, par une série de points connectés par des lignes, mais le dernier point se connecte au premier afin de fermer la géométrie. Les objets plus complexes peuvent tous être représentés en mode vectoriel à partir d'agencements plus ou moins compliqués de ces objets simples.

La résolution de la représentation vectorielle est généralement supérieure à la résolution de la représentation matricielle. Elle est déterminée par la précision des coordonnées décrivant la position des différents objets. Elle dépend aussi du nombre de points (vertices) utilisés pour représenter l'objet. Toutefois, il faut rester prudent car la résolution spatiale donnée par la représentation matricielle correspond parfois plus fidèlement à la qualité réelle des données.

Les objets peuvent être décrits géométriquement de manière plus ou moins réaliste selon l'échelle géographique utilisée pour la représentation. En effet, à certaines échelles, les objets doivent être généralisés afin de ne représenter que les éléments les plus importants et, ainsi, ne pas surcharger la carte avec des détails inutiles. De plus, des primitives géométriques distinctes peuvent être employées pour la représentation d'un même objet à des échelles différentes. Par exemple, une agrégation de poissons peut être décrite à une échelle régionale par un polygone dont les limites correspondent à celles de l'agrégation, tandis qu'à une échelle locale, la même agrégation peut être représentée par un volume.

Dans une représentation vectorielle, les objets de même type sont regroupés dans une même classe qui est représentée sous la forme d'une table dans une base de données (voir

figure 2.6). Chaque objet occupe une rangée et chacune de ses propriétés (attributs) représente une colonne dans la table.



Poisson

No	Poids	Âge	Espèce	Type
1	2,26 kg	3	Morue	Benthique
2	1,45 kg	2	Hareng	Pélagique
3	46,71 kg	1	Thon	Pélagique
4	1,94 kg	1	Morue	Benthique
5	30,56 kg	2	Flétan	Benthique

Figure 2.6 - Exemple de table pour un type d'objet ponctuel correspondant à une classe *poisson*; chacune des lignes représente un poisson se trouvant dans une région donnée à un certain instant et chacune des colonnes correspond à l'une des propriétés liées aux poissons.

Une structure de données vectorielle est généralement dite topologique si elle stocke de manière explicite les relations de topologie entre l'ensemble des objets géométriques d'un même jeu de données (voir figure 2.7). Les relations topologiques des objets subsistent et restent constantes même quand l'espace géographique contenant ces objets est déformé (Longley *et al.*, 2005). Les relations d'inclusion, d'intersection ou d'adjacence par exemple sont des types de relations topologiques qui ne s'altèrent pas avec la déformation des géométries. Une structure vectorielle topologique contient généralement des points ou vertices (coordonnées x, y qui n'ont pas de signification topologique), des nœuds partout où des chaînes se croisent, des chaînes ou arcs représentés par des lignes et composés d'un nœud de départ, d'un nœud d'arrivée et possiblement de plusieurs vertices intermédiaires ainsi que des faces représentées par des polygones orientés (gauche, droite) (Worboys et Duckham, 2004). La triangulation Delaunay est un exemple de structure de données vectorielle topologique. Une structure de données vectorielle non-topologique est appelée « spaghetti », puisque les segments formant les lignes et les polygones peuvent s'entrecroiser ou ne pas se rejoindre complètement. De plus, dans une structure de données vectorielle « spaghetti », aucune relation n'est établie entre les différents objets géométriques et la même information peut être répétée à plusieurs endroits dans la structure. Ainsi, certaines analyses spatiales ne sont pas facilement réalisables. La structure

de données non topologique sert principalement à créer une représentation rapide des données (Lachance, 2005).

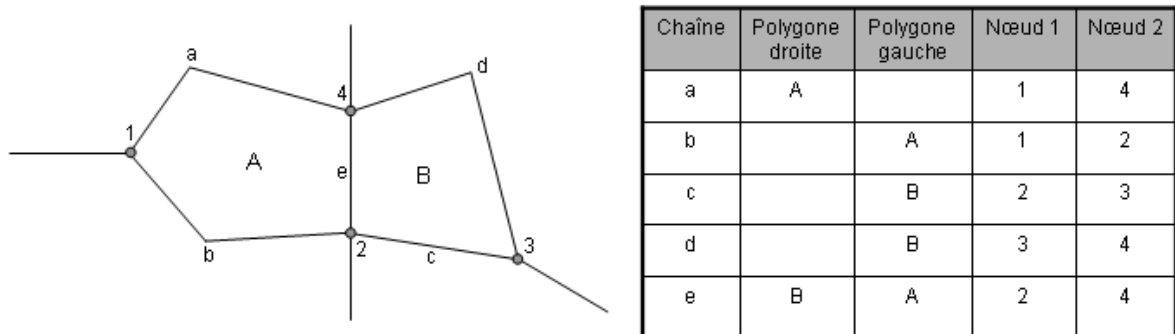


Figure 2.7 - Structure de données vectorielle topologique où les relations entre les différents objets, ici des polygones A et B, sont stockées dans une table (Bédard et Larrivée, 2003).

Les principaux avantages de l'utilisation d'une structure de données vectorielle pour représenter des objets sont que les relations topologiques entre les différents éléments de la représentation peuvent être explicitement décrites et maintenues et que ce type de structure de données nécessite généralement moins d'espace de stockage. Le volume de données à stocker dépend de la densité de points utilisés dans la représentation et le nombre des relations explicites à stocker. De plus, la structure de données vectorielle constitue une représentation explicite des éléments linéaires et elle permet de distinguer facilement la limite des différents objets imbriqués dans l'espace géographique. Elle convient bien pour la représentation de données dispersées dans l'espace, mais est plus complexe que la structure de données matricielle. Sa manipulation requière des algorithmes plus sophistiqués et elle ne permet pas de représenter adéquatement la continuité de certains phénomènes spatiaux comme la température et l'altitude.

2.2 Application des SIG pour la représentation des objets spatiaux liés à l'environnement marin

Les SIG et les représentations spatiales qu'ils supportent peuvent contribuer de manière intéressante à l'océanographie, à la biologie marine et à la gestion des ressources marines (Wright et Goodchild, 1997; St. Martin, 2004). En effet, la représentation de l'aspect géographique des phénomènes marins est importante pour leur compréhension. À l'heure

actuelle, les SIG et leurs structures de données sont employés pour des besoins de représentation dans certaines applications marines tel que mentionné à la section 1.1.3. Ces représentations servent par exemple à sélectionner des sites appropriés pour la mariculture ou l'aquaculture, à visualiser les variations dans les paramètres physiques de l'environnement marin ou montrer la distribution générale de l'abondance des poissons (Meaden, 2000; Wright et Bartlett, 2000; Valavanis, 2002).

Néanmoins, les bénéfices potentiels de l'utilisation de SIG pour traiter et représenter des données marines seraient encore plus considérables si des solutions efficaces étaient trouvées pour palier à certaines de leurs limites. Cette section discute des principaux problèmes liés à l'utilisation des SIG pour la représentation de phénomènes de l'environnement marin. Bien que les SIG constituent une solution intéressante pour la gestion, l'intégration, la manipulation, l'analyse et la diffusion des données en général, les données marines posent certains problèmes (Fonseca *et al.*, 2002). Ces problèmes sont liés à la nature inhérente aux données marines qui incluent, entre autres, des objets tridimensionnels et des objets dont les frontières ne sont pas parfaitement définies.

D'autres problèmes liés aux données marines ne seront pas abordés en détails dans ce mémoire, mais méritent d'être mentionnés. L'un d'eux est l'important espace de stockage nécessaire pour stocker certains types de données marines. La taille imposante des jeux de données marines est liée aux développements récents des technologies et techniques employées pour leur observation. En effet, à l'heure actuelle, la vitesse à laquelle les données marines sont acquises dépasse largement la vitesse à laquelle elles peuvent être interprétées. Par exemple, des levés bathymétriques à haute résolution des fonds marins, réalisés à l'aide de capteurs multifaisceaux, produisent d'énormes volumes de données. De plus, des milliers de capteurs océanographiques collectent des données en continu dans nos océans (ex. température de l'eau et de l'air, direction et vitesse du vent). Le fait que les données marines soient tridimensionnelles ajoute à l'enjeu de la gestion du volume d'information disponible. En effet, le volume de données est beaucoup plus élevé pour des données en 3D que pour des données en 2D. De plus, en raison de la nature dynamique de

l'environnement marin, les jeux de données sont acquis à une fréquence très élevée, ce qui demande une très grande capacité de stockage d'information.

Par ailleurs, certains objets de l'environnement marin, les poissons par exemple, changent de position continuellement dans le temps, ce qui leur confère un aspect dynamique difficile à gérer dans les SIG commerciaux puisque ces derniers sont conçus pour représenter des données statiques. À une échelle régionale, le mouvement des poissons cause peu d'inconvénients puisque ces mouvements sont relatifs, lents et souvent quantifiables sur une base cyclique. Par contre, à une échelle plus détaillée, le mouvement des poissons peut être relativement prononcé, et donc plus important à considérer (Meaden, 2000).

Ainsi, les données marines et les objets spatiaux qui leurs sont associés offrent un énorme défi pour le développement des SIG. Les sections qui suivent abordent les problèmes liés à la représentation d'objets aux limites floues et à la représentation tridimensionnelle qui sont très présents dans l'environnement marin.

2.2.1 Représentation d'objets aux limites floues

Selon Clementini et Di Felice (1997, p.173), « une limite importante des SIG actuels pour la représentation d'objets spatiaux est la disparité entre une réalité géographique complexe (non structurée, avec des frontières incertaines) et la structuration géométrique (objets simples avec une frontière nette) qui lui est associée. » Yanar et Akyürek (2004, p.1) affirment par ailleurs que « puisque la théorie classique des ensembles utilisée dans des SIG conventionnels est insatisfaisante pour exprimer la variabilité naturelle présente dans les phénomènes environnementaux, il existe souvent des anomalies significatives entre la réalité et sa représentation. » En effet, les types de primitives géométriques disponibles dans les SIG tendent à représenter les entités avec une frontière bien définie ainsi qu'une homogénéité des valeurs d'attributs à l'intérieur de cette frontière. De plus, des opérateurs booléens répondant aux critères de la logique classique sont utilisés pour manipuler les objets spatiaux (ex. union, intersection, différence).

Ainsi, les SIG existants requièrent généralement de représenter et de classifier les objets perçus dans la réalité de manière discrète. Toutefois, la complexité des phénomènes naturels, tels que les phénomènes marins, nécessite d'adapter ces représentations. En effet, de nombreux phénomènes et objets appartenant au monde réel, tels que les limites de forêts, de zones humides, des fronts océaniques ou des agrégations de poissons en milieu marin, sont flous. Il a été établi que seuls quelques objets dans l'espace géographique ont des limites naturelles qui sont bien déterminées. La plupart des objets sont donc des abstractions d'entités possédant des limites floues ou qui n'ont pas de limite (Frank, 1996). Selon Wright et Goodchild (1997, p.524), « bien que les données marines ont une position spatiale, ces positions et les frontières des objets qui leurs sont associés tendent à être plus floues que celles trouvées dans un environnement terrestre. » Par exemple, il n'existe pas de points géodésiques ou l'équivalent dans les océans. Ainsi, les données marines sont souvent acquises par rapport à la position d'objets qui sont continuellement en mouvement, comme des navires, et qui ont par conséquent des positions imprécises.

Pour illustrer l'indétermination des limites pour un phénomène naturel, la densité du couvert d'arbres dans une forêt peut être évoquée. Cette densité diminue peu à peu jusqu'à ce qu'on ne retrouve plus ou presque plus d'arbres dans un secteur donné. Pendant le processus de changement, le territoire étudié passe d'une forêt dense à une forêt clairsemée puis à un secteur où il n'y a plus d'arbres. Si cette information est représentée à l'aide de limites nettes, la transition entre la forêt et son extérieur n'est pas montrée et la représentation obtenue ne reflète donc plus adéquatement la réalité. En effet, le changement entre le territoire couvert par la forêt et celui qui est dépourvu d'arbres semble arriver soudainement. Aussi, il peut s'avérer très difficile de placer cette limite nette entre la forêt et son extérieur comme en témoigne la figure 2.8 où trois limites différentes ont pu être identifiées. Ainsi, dans la réalité, certains concepts et objets spatiaux ne peuvent être représentés par des objets aux limites bien déterminées.



Figure 2.8 - Différentes interprétations possibles de la limite d'une forêt sur une photographie aérienne (Caloz, 2005).

Cette analogie peut également être appliquée à des phénomènes spatiaux de l'environnement marin, pour la délimitation des agrégations de poissons par exemple. En effet, tout comme dans la forêt, la densité d'individus observée diminue souvent graduellement à la limite d'une agrégation. Les limites des agrégations de poissons sont alors continues mais plus ou moins étendues. Il est donc laborieux de tracer une ligne délimitant les zones où le phénomène d'agrégation des poissons est présent (zone intérieure) et où il ne l'est pas (extérieur). De plus, la densité au centre de l'agrégation de poissons est généralement plus élevée que la densité observée à sa frontière. Cela signifie qu'il existe une variabilité dans les valeurs d'attributs à l'intérieur de l'objet spatial qui doit être délimité. Conséquemment, une représentation où les limites des agrégations de poissons sont montrées comme des zones de changement graduel plutôt qu'abrupte est nécessaire pour une cartographie plus exacte du phénomène étudié.

Lorsqu'il existe une incertitude sur les limites des entités spatiales à représenter, tel que dans les exemples exposés précédemment, la notion d'objet spatial flou peut être appliquée. Cette notion a été développée à l'aide de la théorie des ensembles flous qui sera abordée dans le chapitre 3 du présent mémoire. Une région floue est une région qui a une limite large, ce qui signifie que cette limite n'est pas nécessairement une ligne nette, mais une zone de transition (Dilo, 2006). Les points se retrouvant dans cette limite large possèdent différents degrés d'appartenance à la région. Toutefois, comme la plupart des SIG

commerciaux ont été conçus pour manipuler des objets spatiaux possédant des frontières précisément déterminées, la manipulation d'objets spatiaux aux frontières indéterminées s'avère souvent difficile. De plus, bien que les limites indéterminées soient relativement simples à montrer dans un environnement matriciel, elles sont beaucoup plus compliquées à représenter dans un environnement vectoriel.

La représentation floue en mode matriciel apporte une solution partielle au problème de perte des détails résultant de la variation qui peut avoir lieu à l'intérieur des pixels. Ainsi, contrairement à une représentation matricielle classique, une représentation floue admet plusieurs valeurs d'un même attribut, à des degrés divers, pour un seul pixel. En effet, dans une représentation floue, une valeur d'appartenance à un objet spatial donné est attribuée à chacun des pixels. Par exemple, un pixel peut appartenir à 30% (0.3) à une agrégation de poissons, et donc ne pas en faire partie à 70% (0.7). Cette valeur d'appartenance exprime alors le degré d'adéquation entre le pixel et l'objet spatial. Plus la valeur d'appartenance est élevée pour un pixel, plus ce pixel contribue à la formation de l'objet spatial. Ces principes seront explicités dans le chapitre 3.

2.2.2 Représentation 3D

L'environnement terrestre est un espace 3D mais sa cartographie en trois dimensions est typiquement faite sous la forme de modèles numériques de terrain (i.e. 2.5D, pas un 3D réel). L'environnement marin est lui un espace géographique demandant une vraie modélisation tridimensionnelle. Les données liées à ce type d'environnement dépeignent l'information décrivant la colonne d'eau, soit l'espace décrit entre la surface de la mer et le fond marin (Fonseca *et al.*, 2002). Des paramètres tels que la température, la salinité, la vitesse du son ou l'abondance des poissons dans l'eau sont mesurés à différentes profondeurs, ce qui produit un cube de données dans un espace 3D. De plus, les nouveaux équipements d'acquisition de données marines, tels que les sonars multifaisceaux, peuvent désormais détecter des cibles acoustiques (ex. poissons) dans la colonne d'eau. Il s'agit d'un exemple typique de phénomène tridimensionnel qui aurait avantage à être représenté, visualisé et analysé en 3D. Ainsi, actuellement, l'un des principaux défis dans le domaine des SIG concerne la représentation d'espaces géographiques tridimensionnels ainsi que des

objets qui en font partie. Cette représentation est utile pour le milieu marin, mais aussi pour représenter l'atmosphère et les couches géologiques.

Il importe de définir ce qu'est une représentation 3D. Dans ce mémoire, une représentation est dite 3D si l'ensemble des objets qui la composent, peu importe leur dimension, se trouvent dans un espace 3D (Pouliot *et al.*, 2003; Lachance, 2005). Ces objets peuvent être des points (0D), des lignes (1D), des polygones (2D) ou des volumes (3D) qui possèdent une coordonnée en z en plus des coordonnées en x et en y.

La plupart des données marines sont situées dans un espace 3D. Elles ont une dimension horizontale (coordonnées en latitude et en longitude) et verticale (coordonnée en profondeur) et peuvent former des objets tridimensionnels complexes (ex. courants marins). Les données marines, telles que les données collectées par des techniques acoustiques, sont rarement distribuées de manière homogène. Elles sont très souvent dispersées dans la dimension horizontale tandis qu'elles sont très abondantes dans la dimension verticale en raison des techniques utilisées pour leur acquisition (Wright et Goodchild, 1997). C'est le cas des données océanographiques captées à l'aide de CTD (*Conductivity Temperature Depth*) qui sont collectées lorsqu'un bateau est arrêté (x et y fixe) tout le long de la colonne d'eau, à un intervalle d'échantillonnage très dense en z.

Certains logiciels commerciaux tels que EchoView de SonarData (SonarData Pty Ltd), Fledermaus de IVS3D (IVS) ou EVS de C Tech's Environmental Visualization System (C Tech EVS) permettent la présentation graphique et la visualisation d'objets de l'environnement marin (3D). Ces outils sont toutefois très limités en termes de fonctionnalités et de capacités d'analyse spatiale, se limitant principalement à la visualisation en 3D des objets. La figure 2.9 illustre des résultats pouvant être obtenus pour la visualisation de données marines à l'aide de ces logiciels.

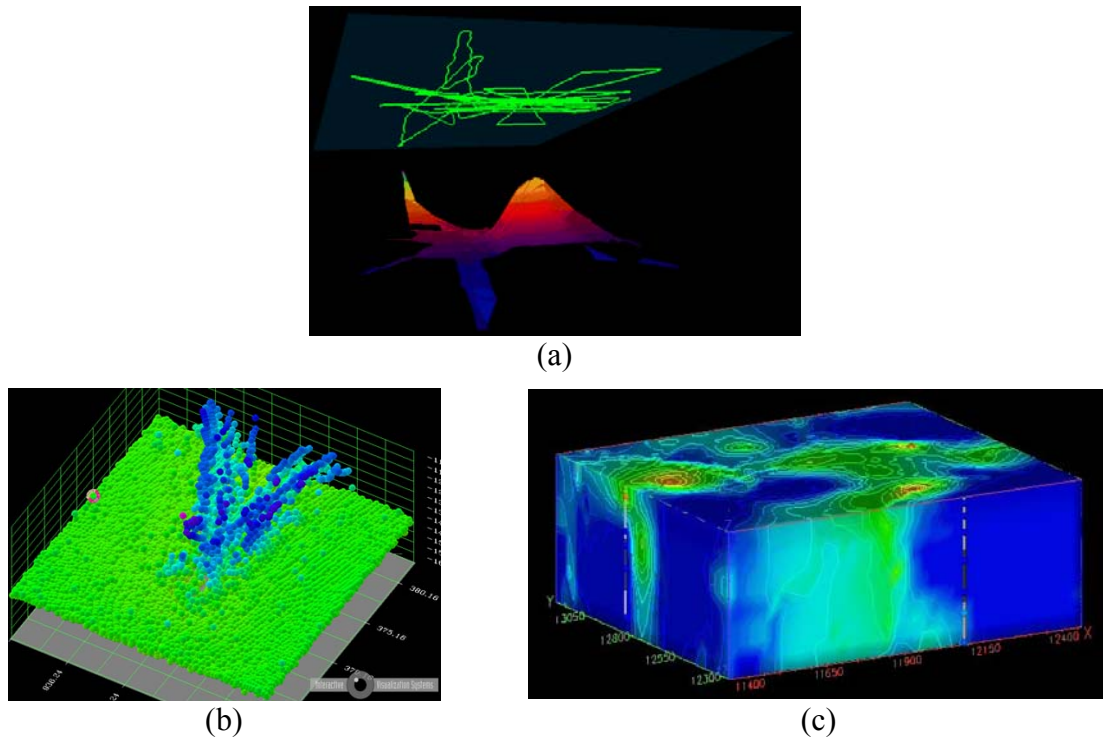


Figure 2.9 - (a) Modélisation du fond océanique dans le logiciel Echoview (SonarData Pty Ltd) à l'aide de données multifaisceaux; la trajectoire suivie par le bateau sur lequel le capteur est embarqué est montrée en vert, (b) visualisation de données multifaisceaux dans le logiciel Fledermaus (IVS) et (c) visualisation 3D de paramètres océanographiques par krigeage 3D dans le logiciel EVS (C Tech EVS).

Les objets observés dans l'environnement marin peuvent également être représentés dans les SIG. Toutefois, la représentation d'un phénomène 3D dans un SIG commercial qui ne gère que 2D ou 2.5D engendre une perte de précision et d'information sur la dimension verticale des données, ce qui altère la compréhension du phénomène analysé. En effet, les objets perceptibles par les humains dans la réalité sont tridimensionnels et leurs représentations dans des dimensions inférieures, dans un SIG par exemple, ne peuvent être que des approximations (Longley *et al.*, 2005). Le développement de la présente génération de SIG s'est effectué en grande partie lors du passage des cartes conventionnelles à un environnement numérique dans les années 1960. Les systèmes actuellement disponibles sur le marché ont donc été développés sur les bases des premiers SIG créés pour numériser et informatiser des cartes papier 2D. Par conséquent, plusieurs limites liées aux techniques traditionnelles de cartographie se retrouvent couramment dans ces systèmes. C'est également pourquoi la plupart des SIG ne travaillent qu'en 2D ou en 2.5D (Raper, 2000). Abdul-Rahman et Pilouk (2007) signalent qu'à l'heure actuelle, un SIG capable de fournir des capacités 3D étendues n'est pas disponible sur le marché. La plupart des SIG sont donc

limités, pour la représentation 3D, à créer un modèle numérique d'élévation ou à donner une élévation moyenne à chaque objet ponctuel, linéaire ou surfacique, comme utilisé par le processus d'extrusion de polygones dans les SIG commerciaux (Abdul-Rahman et Pilouk, 2007).

Ainsi, pour les représentations liées au domaine marin dans les SIG, la troisième dimension a généralement été oubliée dans les travaux où cela s'avérait possible. En effet, même si une représentation en 3D permet une meilleure interprétation du phénomène, la représentation 2D est malgré tout jugée assez représentative par les scientifiques et est généralement utilisée. Dans d'autres cas, afin d'obtenir de l'information sur la troisième dimension en dépit du fait que le SIG utilisé soit 2D ou 2.5D, des sections verticales de la distribution de données 3D sont prélevées et représentées (voir figure 2.10). Cependant, ce type de représentation n'est pas continu et de l'information est alors perdue entre les différentes sections verticales 2D.

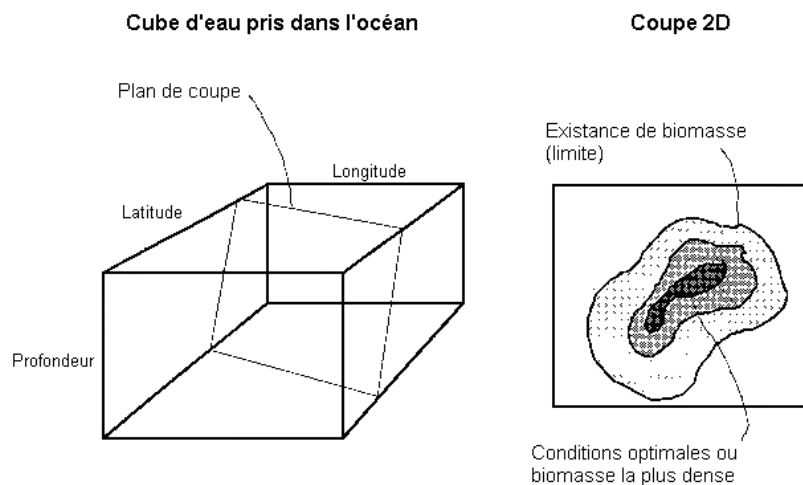


Figure 2.10 – Représentation d'une section verticale 2D d'un phénomène marin 3D (traduite de Meaden et Do Chi, 1996).

Un SIG adapté à la représentation de l'environnement marin (3D) doit entre autres être en mesure de supporter une topologie 3D pour l'interrogation et l'analyse des données à l'aide de requêtes spatiales (Fonseca *et al.*, 2002). Il doit également supporter une visualisation 3D afin de permettre l'examen et la vérification rapide de jeux de données marines, qui sont généralement volumineux, en les présentant d'une manière plus intuitive.

2.3 Conclusion du chapitre

L'emploi de technologies SIG peut s'avérer utile pour la réalisation de certaines opérations telles que l'intégration, la manipulation, l'analyse et la diffusion des données géographiques marines et halieutiques. De plus, les modélisations de l'espace qui sont associées aux SIG permettent de créer une représentation relativement simple d'un environnement marin complexe. En effet, ces modélisations permettent alternativement de représenter cet environnement marin et les processus qui s'y déroulent par un champ continu ou par un espace vide dans lequel sont distribués des objets discrets.

Les SIG actuels se basent sur deux structures de données géométriques, soient les structures vectorielle et matricielle. Ces deux structures de données comportent des caractéristiques très différentes et peuvent être utilisées séparément ou en association pour la description géométrique de phénomènes spatiaux. Dans le cas de la représentation d'objets discrets en 2D par exemple, la structure matricielle décompose ces objets en cellules régulières appelées pixels et stocke les informations descriptives de chacun de ces pixels alors que la structure vectorielle utilise des objets géométriques de types ponctuel, linéaire, polygonal et polyédrique rattachés à une ou des tables d'attributs.

Cependant, certains problèmes associés à la nature des données marines doivent être résolus afin que les SIG actuels puissent réellement décrire les phénomènes spatiaux présents dans l'environnement marin et offrir une solution plus réaliste aux scientifiques et aux gestionnaires. Ces données sont entre autres généralement acquises en 3D, sont dynamiques et la limite des phénomènes qu'elles dépeignent est généralement floue. Dans la suite du présent mémoire, des pistes seront explorées afin de solutionner ces limitations des SIG conventionnels pour la représentation des phénomènes spatiaux liés à l'environnement marin. Une attention particulière sera portée sur le phénomène d'agrégation des poissons. Ces agrégations seront considérées comme des objets spatiaux. Les structures de données géométriques seront quant à elles combinées pour obtenir une représentation des limites floues des agrégations de poissons et généralisées en 3D pour obtenir un modèle tridimensionnel de ce phénomène.

Chapitre 3 Représentation des agrégations de poissons et de leur évolution spatio-temporelle en 2D

Le phénomène d'agrégation revêt une très grande importance pour la survie des stocks de poissons. En effet, tel que mentionné précédemment (voir chapitre 1, section 1.1.2), une agrégation est formée lorsqu'un nombre considérable de poissons se regroupent pour se nourrir, se protéger contre des prédateurs ou encore pour se reproduire. L'étude de ce phénomène et de son évolution spatiale sur une période de temps donnée peut donc conduire à mieux comprendre certains facteurs liés au déclin de certaines espèces de poissons (Bennett, 2008). Par le fait même, des solutions pour freiner ce déclin peuvent être élaborées afin de protéger ces espèces.

Tel que mentionné dans le chapitre précédent (voir chapitre 2, section 2.2.1), de part leur nature, les agrégations de poissons peuvent à certaines échelles être considérées comme des objets flous. En effet, à une échelle régionale, par exemple, les agrégations sont vues comme des régions de forte concentration en poissons pour lesquelles des limites ne peuvent pas être précisément définies. Ces limites sont plutôt des zones de transition vers d'autres régions où la concentration en poissons est moins élevée.

Toutefois, les SIG actuels n'offrent pas d'emblée la possibilité de représenter ces phénomènes ou des objets spatiaux de manière floue. Face à ce problème, ce chapitre propose une approche pour la représentation des agrégations de poissons à une échelle régionale en 2D qui se base sur la théorie des ensembles flous et sur les modèles d'objets spatiaux flous qui en découlent. Dans un premier temps, la section 3.1 présente la notion d'ensembles flous. Ensuite, les principaux modèles d'objets spatiaux flous sont revus à la section 3.2. Puis, la section 3.3 décrit l'approche développée dans la présente recherche pour représenter des objets flous, soient des agrégations de poissons, dans un SIG conventionnel. Cette approche combine les deux structures de données géométriques généralement disponibles dans les SIG afin de présenter un maximum d'information, surtout dans la limite des objets. Finalement, les sections 3.4 et 3.5 présentent des résultats

obtenus suite à l'application de cette approche pour la représentation d'agrégations de poissons de la région de l'Atlantique du nord-ouest et pour la représentation de leur évolution spatio-temporelle.

Il faut mentionner ici que bien que l'agrégation des poissons soit un phénomène tridimensionnel (3D), la plupart des données acquises à l'heure actuelle à l'échelle régionale portent sur l'abondance des poissons en certaines positions spatiales et, par conséquent, elles ne sont principalement acquises qu'en 2D. Dans ce contexte, même si une représentation en 3D telle que celle présentée dans le chapitre 4 pour des données acquises localement permet une meilleure interprétation du phénomène d'agrégation, la représentation 2D est malgré tout généralement utilisée et jugée assez représentative par les experts pour une échelle régionale. En effet, à cette échelle, les spécialistes en biologie marine étudient les changements dans les régions de forte concentration en poissons plutôt que les caractéristiques internes des agrégations par exemple. De plus, comme les données ponctuelles à disposition représentent le nombre ou le poids de poissons pêchés en un site de la région d'étude et non des individus (poissons), l'utilisation de méthodes de *clustering* pour la délimitation des agrégations est difficile, voire impossible.

3.1 Notion d'ensembles flous

Cette section donne les principaux concepts de base associés à théorie classique des ensembles ainsi qu'à la théorie des ensembles flous telle que proposée par Zadeh (1965). Les objets spatiaux peuvent en effet être considérés comme des ensembles du point de vue de la théorie des ensembles (Tang, 2004).

3.1.1 Théorie classique des ensembles

Dans la théorie classique des ensembles, il existe une séparation claire entre un élément qui appartient à un ensemble donné et un élément qui ne lui appartient pas. Effectivement, la théorie classique des ensembles suit une logique booléenne qui n'accepte que des valeurs binaires telles que zéro et un ou vrai et faux. Toutes les mesures d'une certaine variable sont ainsi divisées en deux catégories distinctes avec lesquelles il est facile de composer. La

séparation entre ces deux catégories est souvent effectuée selon un seuil précis. Ce seuil détermine alors à l'intérieur de laquelle des catégories un élément particulier se situe.

Soit X un ensemble classique d'objets. L'appartenance d'un sous-ensemble A , issu de l'ensemble X , peut être décrite par la fonction caractéristique suivante (Schneider, 1999) :

$$\chi_A : X \rightarrow \{0,1\}.$$

Dans cette fonction, pour tout $x \in X$:

$$\chi_A(x) = \begin{cases} 1 & \text{si et seulement si } x \in A \\ 0 & \text{si et seulement si } x \notin A \end{cases}$$

On peut observer, à partir de ces équations, que la théorie classique des ensembles ne permet à un élément qu'une appartenance totale ou nulle à un ensemble donné. Un exemple d'une fonction caractéristique d'un ensemble classique est donné à la figure 3.1a.

La théorie classique des ensembles présente l'avantage d'être simple et facile à implanter d'un point de vue informatique. Toutefois, cette conceptualisation assez rigide ne permet pas de rendre compte des situations plus nuancées rencontrées très souvent dans la réalité où la différenciation entre les éléments n'est pas abrupte, mais plutôt progressive.

3.1.2 Théorie des ensembles flous

Pour palier à cette lacune, la notion d'ensemble flou a été établie. La théorie des ensembles flous permet d'utiliser des éléments du langage naturel et d'introduire des concepts relatifs tels que élevé, grand ou fort, afin de définir des catégories d'objets (Chevrie et Guély, 1998). En fait, la théorie des ensembles flous est une extension de la théorie classique des ensembles réalisée pour incorporer le concept de vérité partielle. Elle se base sur la logique floue qui considère la notion d'appartenance d'un objet à un ensemble non pas comme une fonction booléenne avec deux valeurs possibles, mais plutôt comme une fonction continue pouvant prendre n'importe quelle valeur dans l'intervalle $[0,1]$ (Chevrie et Guély, 1998; Zhan, 1998, Kraipeerapun, 2004; Kremenová, 2004; Dilo, 2006). Ainsi, elle repose sur le principe selon lequel un élément appartient partiellement ou graduellement à un ou des

ensembles flous définis. La fonction caractéristique d'un ensemble classique est donc remplacée par une fonction d'appartenance dans le cas d'un ensemble flou.

La valeur d'appartenance d'un élément à un certain ensemble est attribuée à celui-ci par la fonction d'appartenance dont un exemple est fourni à la figure 3.1b. Elle exprime le degré de compatibilité de l'élément avec cet ensemble. Plus la valeur d'appartenance est élevée, plus un élément possède un fort degré de participation à un ensemble donné. Un élément fait entièrement partie d'un ensemble si sa valeur d'appartenance est égale à 1. Ce même élément appartient partiellement à un ensemble si sa valeur d'appartenance se situe entre 0 et 1. Finalement, l'élément n'appartient pas à un ensemble si sa valeur d'appartenance est de 0. La notion d'appartenance multiple signifie qu'un élément peut avoir une valeur d'appartenance positive pour plusieurs ensembles.

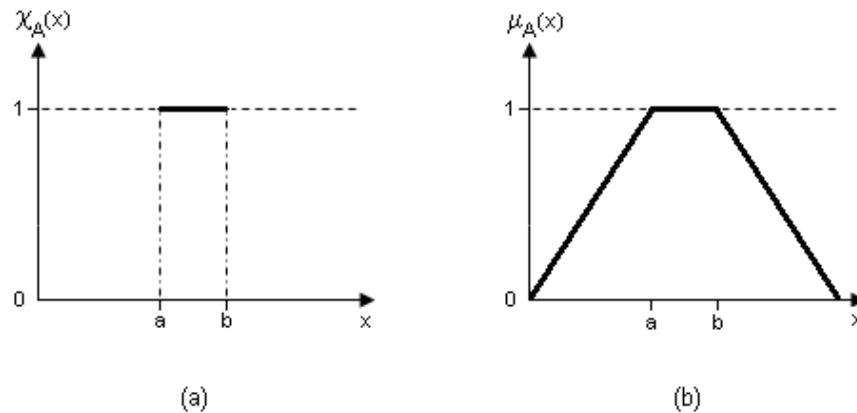


Figure 3.1 - Comparaison entre (a) la fonction caractéristique d'un ensemble classique et (b) la fonction d'appartenance à un ensemble flou.

De façon formelle, un sous-ensemble flou A de l'ensemble X est caractérisé par sa fonction d'appartenance μ_A définie dans l'intervalle $[0,1]$:

$$A = \{(x, \mu_A(x)) \mid x \in X \wedge \mu_A : X \rightarrow [0,1]\}$$

où $\mu_A(x)$ représente le degré ou la valeur d'appartenance d'un élément x à l'ensemble A (Kraipeerapun, 2004). Donc, pour tout $x \in X$:

$$\mu_A(x) = \begin{cases} 1 & \text{si et seulement si } x \in A \\ p; 0 < p < 1 & \text{si et seulement si } x \text{ appartient partiellement à } A \\ 0 & \text{si et seulement si } x \notin A \end{cases}$$

Les contours d'un ensemble flou ne sont pas déterminés et nets mais plutôt flous ou graduels. Ainsi, les ensembles flous peuvent dépeindre certains phénomènes ou objets du monde réel de manière plus réaliste et intuitive que les ensembles classiques. Toutefois, l'implantation de logique floue n'est pas aussi évidente que l'implantation de la logique booléenne. La figure 3.2 présente la différence entre la représentation d'un ensemble classique (A) et d'un ensemble flou (B).

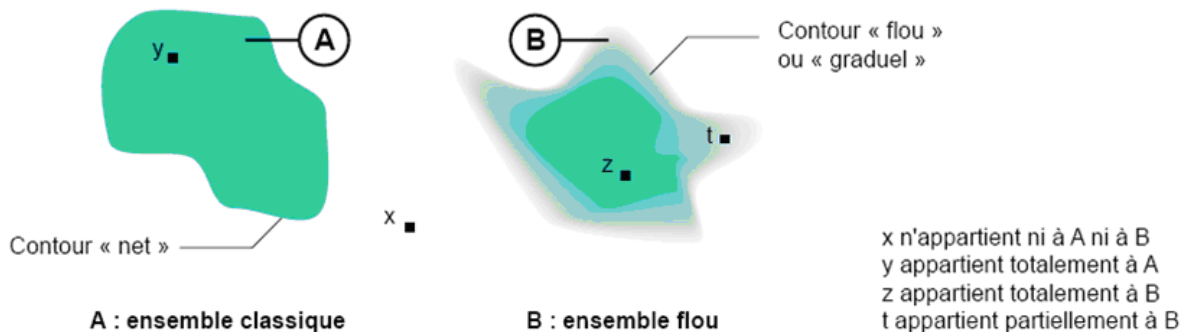


Figure 3.2 - Comparaison d'un ensemble classique et d'un ensemble flou extraite du Cahier Technique Schneider n°191 (Chevrie et Guély, 1998).

Une autre notion utile pour la suite du chapitre est celle de l' α -coupe également employée pour décrire un ensemble flou (Zhan, 1998; Kremenová, 2004). Supposons $\alpha \in \mathbb{R}$. Une α -coupe est un sous-ensemble constitué de l'ensemble des éléments appartenant à l'ensemble flou A qui possèdent une valeur d'appartenance plus grande ou égale à α :

$$A_{\alpha} = \{x \in X \mid \mu_A(x) \geq \alpha\}.$$

De la même façon, l' α -coupe stricte d'un ensemble flou est constitué de l'ensemble des éléments possédant une valeur d'appartenance strictement plus grande que α . L' α -coupe strict d'un ensemble flou A est donc :

$$A_{\alpha}^- = \{x \in X \mid \mu_A(x) > \alpha\}.$$

L' α -coupe et l' α -coupe stricte correspondent toutes deux à des ensembles classiques (Zhan, 1998; Kraipeerapun, 2004). L' α -coupe stricte de A avec un α égal à 0 est nommé le *support*

de l'ensemble flou et représente tous les éléments qui se trouvent dans l'ensemble à des degrés divers tandis que le *noyau* (voir figure 3.3), qui représente les éléments faisant totalement partie de l'ensemble, correspond à l' α -coupe de A pour un α ayant une valeur de 1. Le *noyau* est donc un sous-ensemble du *support*. Il est à noter que pour un ensemble classique A , le *noyau* et le *support* sont confondus avec A . Le *noyau* d'un ensemble flou A est :

$$A_1 = \{x \in X \mid \mu_A(x) \geq 1\}$$

et son *support* est :

$$A_0 = \{x \in X \mid \mu_A(x) > 0\}.$$

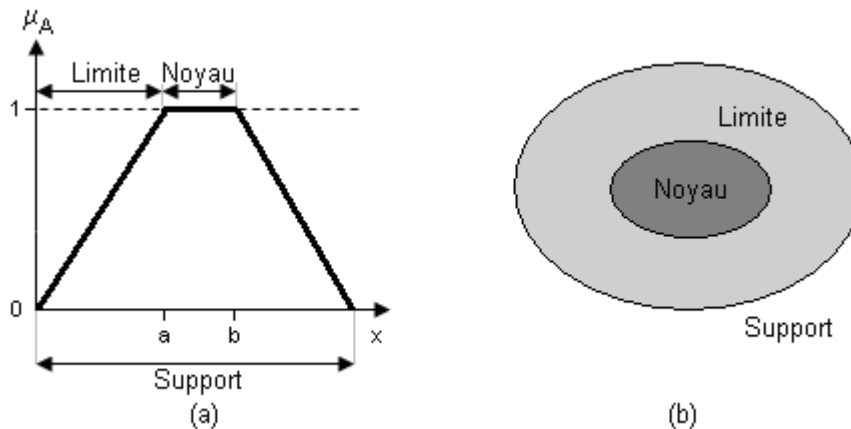


Figure 3.3 - (a) Fonction d'appartenance et ses principaux paramètres (*noyau*, *support* et *limite*),
(b) représentation graphique des paramètres.

La théorie des ensembles flous brièvement exposée dans cette section constitue un outil conceptuel approprié pour la modélisation des objets spatiaux aux limites floues ou indéterminées puisqu'elle remplace la limite nette d'un ensemble classique par une zone de transition graduelle et parce qu'elle admet les notions d'appartenance partielle et multiple (Schneider, 1999). Tel que mentionné dans le chapitre 2, les objets spatiaux sont caractérisés par un attribut et une étendue spatiale représentant la région où cet attribut existe. Dans une représentation floue, les ensembles flous sont utilisés pour modéliser cette étendue spatiale (Fonte et Lodwick, 2005). La théorie des ensembles flous sera donc utilisée dans les sections subséquentes.

3.2 Modèles d'objets spatiaux flous

À une échelle régionale, les agrégations de poissons peuvent être considérées comme des objets spatiaux flous. Un objet flou est un objet possédant une frontière large, c'est-à-dire que sa frontière n'est pas une ligne bien définie, mais est plutôt une zone de transition (Dilo, 2006).

Bien que l'utilité et la validité de la représentation du flou dans les données géographiques soit largement acceptée au sein des spécialistes du domaine des SIG, il existe un consensus sur le fait que les logiciels SIG ne sont pas capables de manipuler des objets spatiaux flous directement. Par exemple, Clementini et Di Felice (1996) affirment que des difficultés se posent dès qu'il est question de dessiner et de représenter des objets géographiques avec des limites indéterminées dans un format numérique. Ils allèguent que ces difficultés viennent du fait que la modélisation dans les SIG tente de simplifier chaque entité géographique et de représenter sa forme en utilisant des objets simples de la géométrie euclidienne soit des points, des lignes et des polygones.

Face à cette situation, plusieurs modèles théoriques ont donc été proposés (Clementini et Di Felice, 1996; Cohn et Gotts, 1996; Erwig et Schneider, 1997; Zhan, 1997; Schneider, 1999; Tang, 2004) pour la représentation de régions possédant des limites floues et quelques-uns parmi ceux-ci sont présentés dans cette section.

3.2.1 Approches basées sur des modèles exacts

Le modèle « Egg-Yolk » a été introduit par Cohn et Gotts (1996) et est illustré à la figure 3.4. Ce modèle présente la limite d'une région floue non pas comme une ligne nette, mais plutôt comme une bande large considérée homogène (Cohn et Gotts, 1996; Dilo *et al.*, 2004). Ce type d'approche vise à considérer les zones déterminées entourant les limites indéterminées et à modéliser l'étendue minimale et l'étendue maximale de la région floue (Schneider, 1996). Une région floue est donc définie par deux zones disjointes dont les limites sont nettes, l'une étant incluse dans l'autre. La limite de la zone interne circonscrit la partie qui appartient certainement à l'objet. Cette limite démarque de la sorte l'étendue minimale de la région. La limite de la zone externe, quant à elle, sépare la partie pour

laquelle il existe une indétermination et qui appartient possiblement à l'objet de la partie qui, de manière certaine, n'appartient pas à l'objet. Elle délimite donc l'étendue maximale de la région. Ainsi, le modèle « Egg-Yolk » suggère l'emploi de deux ou plusieurs sous-régions imbriquées et chacune de ces sous-régions représente un degré d'appartenance ou α -coupe de la région floue. Dans son cas le plus simple, où il n'y a que deux sous-régions, une région floue est comparée à un œuf. Dans cette analogie, le jaune de l'œuf représente la partie certaine de la région, soit son étendue minimale, tandis que le blanc en représente la partie incertaine, soit son étendue maximale. L'union du jaune et du blanc forme l'œuf qui représente toute l'étendue de la région. Le jaune de l'œuf constitue donc le *noyau* de la région alors que l'œuf est le *support*. Le modèle « Egg-Yolk » ne permet pas de représenter un objet exact puisqu'il ne permet pas que l'œuf, le blanc ou le jaune soient vides.

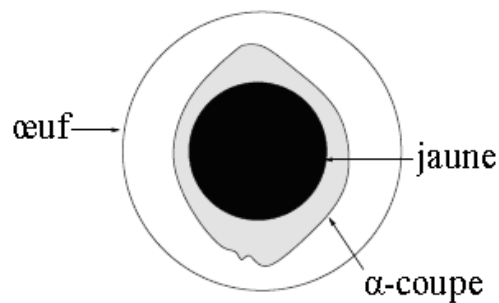


Figure 3.4 - Illustration du modèle Egg-Yolk traduite de Hazarika et Cohn (2001).

Clementini et Di Felice (1996) ont adopté un modèle semblable (voir figure 3.5). Selon leur modèle, les objets flous diffèrent des objets classiques de part la définition de leur limite. Des courbes fermées délimitant une limite extérieure et une limite intérieure comprise dans la première peuvent être définies pour une région floue. Ces deux limites représentent l'indétermination de la région floue et expriment l'étendue maximale et l'étendue minimale possibles. En fait, n'importe quelle courbe comprise entre les deux limites peut constituer la frontière de la région. Une région floue A est donc composée de deux sous-régions aux limites nettes et bien définies A_1 et A_2 telles que A_1 est incluse dans A_2 . δA_1 constitue la limite intérieure de A alors que δA_2 est la limite extérieure de A . La frontière ΔA de la

région floue A comprend donc l'aire comprise entre la limite intérieure δA_1 et la limite extérieure δA_2 de façon à ce que $\Delta A = A_2 - A_1$.

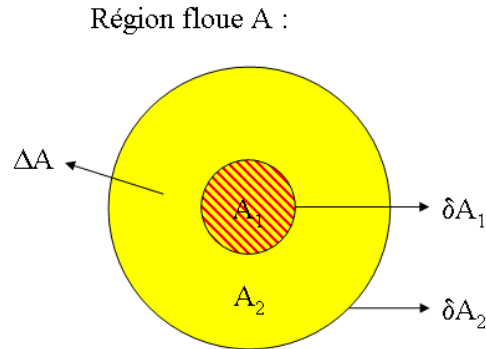


Figure 3.5 - Illustration du modèle de Clementini et Di Felice (1996).

Tang (2004) s'est inspiré de cette dernière approche pour proposer une autre définition des régions floues où il les décompose en quatre composantes, soient l'intérieur, la limite de l'intérieur, la limite de la limite et l'extérieur.

Le modèle d'Erwig et Schneider (1997), diffère légèrement de ceux présentés jusqu'à présent. Il définit une région floue comme une géométrie composée, soit une paire de deux ensembles disjoints. L'un des ensembles est le *noyau*, qui est la partie certaine de la région alors que l'autre ensemble forme la limite qui est la partie incertaine de la région. Ce qui distingue ce modèle des autres est que la limite peut être représentée par une région ou une ligne. Cela signifie que ce modèle permet qu'une région soit partiellement ou totalement déterminée, ce qui n'est pas le cas dans les propositions de Cohn et Gotts (1996), de Clementini et Di Felice (1996) ou de Tang (2004).

Ces approches (Clementini et Di Felice, 1996; Cohn et Gotts, 1996; Erwig et Schneider, 1997) offrent l'avantage d'utiliser des objets spatiaux exacts pour représenter des objets spatiaux flous. Ainsi, les concepts liés aux objets spatiaux exacts sont modifiés et étendus pour la modélisation des objets spatiaux flous. De nouvelles définitions, techniques, structures de données ou encore de nouveaux algorithmes n'ont donc pas à être développés pour traiter le cas des objets flous (Erwig et Schneider, 1997). Par contre, les approches fondées sur des modèles de données exacts comportent certaines limites dont celle de ne

pas représenter la transition graduelle entre l'intérieur de l'objet et son extérieur. En effet, la limite d'un objet est représentée par une bande épaisse et homogène et par conséquent, une distinction plus fine entre les éléments se trouvant dans cette limite ne peut pas être faite. Seules les étendues minimales et maximales sont données.

3.2.2 Approches basées sur la théorie des ensembles flous

Il existe un second groupe de modèles qui sont basés sur des approches mathématiques. Parmi ceux-ci, certains modèles proposent de représenter la transition graduelle dans la limite des objets flous en employant la notion d'ensemble flou qui a été exposée dans la section 3.1 (Leung, 1987; Zhan, 1997; Schneider, 1999). Ainsi, l'application de la théorie des ensembles flous permet l'existence d'une certaine hétérogénéité dans un même objet. Des distinctions plus fines entre les points ou les pixels situés dans la limite large d'une région floue peuvent alors être établies (Zhan, 1997). Cependant, l'utilisation de cette approche requière de définir de nouveaux type d'objets, soient des objets flous (points flous, lignes flous et régions floues), et les relations spatiales entre ces derniers. Par exemple, les principaux travaux de Schneider (1999; 2003) portent sur la définition formelle des différents types d'objets flous à partir de la théorie des ensembles flous et sur la détermination des opérateurs métriques et topologiques pour ces objets flous.

Leung (1987) propose quant à lui un modèle dans lequel les limites sont représentées comme des zones où les valeurs des propriétés associées aux données sont distribuées de manière continue. Dans cette approche, le *noyau* de la région correspond aux positions où les valeurs d'attribut correspondent le plus à la définition de cette région, tandis que la limite de la région représente les points ayant les caractéristiques les moins compatibles avec la définition de la région. Une valeur d'appartenance est alors attribuée à chacun des points de la limite de la région. En connectant tous les points ayant la même valeur α donnée, pour $0 < \alpha < 1$, une α -limite (α -coupe) délimitant une région de valeurs continues est créée (Wang et Hall, 1996). Donc, Leung (1987) montre que même si une région est floue, ses limites et leur étendue peuvent être démarquées précisément.

Le modèle proposé par Zhan (1997), illustré à la figure 3.6, interprète les régions indéterminées comme des ensembles flous. Une région floue peut être décomposée en trois parties, soient le *noyau* A^* , la limite indéterminée δA ainsi que l'extérieur A^- . La limite indéterminée δA est quant à elle formée des trois composantes suivantes : une limite intérieure A° correspondant à l'étendue minimale de la région, une limite extérieure A^\ominus correspondant à l'étendue maximale de la région et la région bordée par ces deux segments. Une valeur d'appartenance entre 0 et 1 est assignée à chacun des points ou pixels se situant dans la région indéterminée δA pour un certain attribut. Il est posé que les valeurs d'appartenance des points ou des pixels d'une région floue A augmentent de la limite extérieure vers la limite intérieure. L'ensemble des points ou des pixels dont la valeur d'appartenance correspond à une valeur α donnée ou est plus élevée forment les α -coupes. Le *noyau* A^* correspond à l' α -coupe pour une valeur α de 1 et ainsi, tous les points ou pixels se trouvant dans cette région possèdent une valeur d'appartenance de 1. De la même manière, l'extérieur A^- est délimité par l' α -coupe trouvé pour une valeur α de 0 et ainsi, tous les points ou pixels se trouvant dans cette région possèdent une valeur d'appartenance de 0.

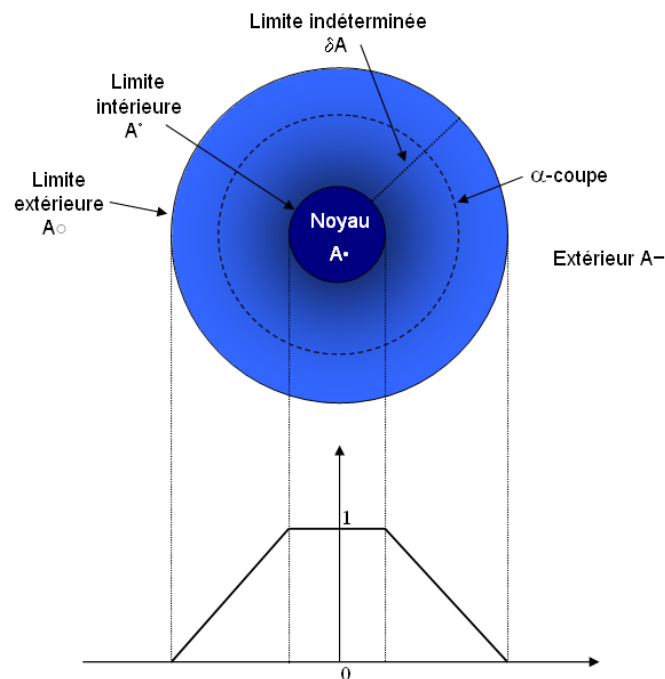


Figure 3.6 - Illustration du modèle de Zhan basé sur la théorie des ensembles flous (Zhan et Lin, 2003).

3.3 Approche proposée

L'approche proposée dans ce chapitre pour la représentation 2D des agrégations de poissons dans un SIG conventionnel est basée sur les modèles décrits à la section 3.2. En fait, ces modèles d'objets flous sont employés pour représenter l'étendue spatiale des entités géographiques que sont les agrégations de poissons. L'approche proposée est inspirée des travaux de Kraiperaapun (2004) et de Dilo (2006). Elle vise à représenter les agrégations de poissons à l'aide d'une structure de données vectorielle en s'inspirant des modèles présentés à la section 3.2.1 et à compléter cette représentation en ajoutant de l'information sur les limites avec une structure de données matricielle en se basant sur les modèles de la section 3.2.2.

L'exemple qui suit illustre l'approche proposée dans ce chapitre. La figure 3.7a montre une agrégation de poissons arbitraire considérée comme un objet spatial discret représenté à l'aide d'une structure vectorielle à une échelle cartographique de 1 : 200 000. Il s'agit d'une généralisation pour laquelle les principaux éléments de la limite de l'agrégation ont été simplifiés ou lissés (Longley *et al.*, 2005). Toutefois, cette représentation n'est pas réaliste puisqu'elle ne prend pas en compte le fait que les limites de l'agrégation de poissons sont incertaines.

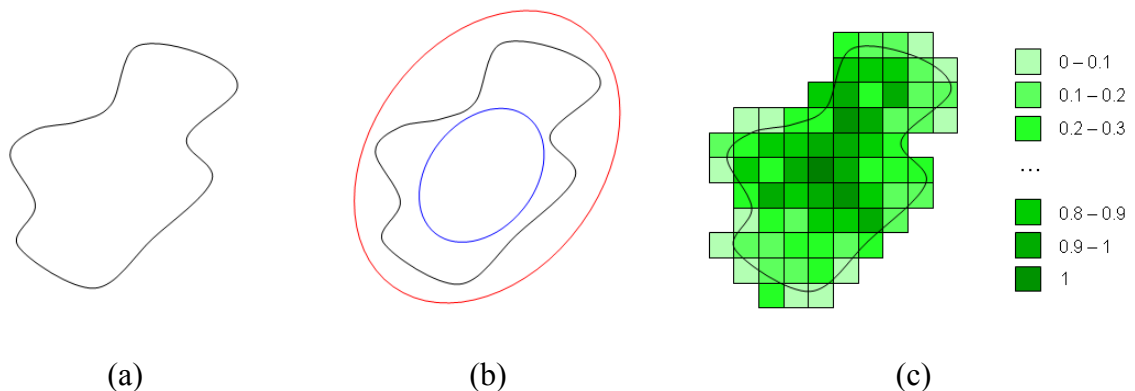


Figure 3.7 - Agrégation de poissons montrée comme (a) un objet spatial discret, (b) un objet flou ayant un intérieur, une limite large et un extérieur, (c) un objet matriciel compris dans une surface continue qui détermine la possibilité d'appartenance de chaque pixel à l'agrégation.

Tel que présenté à la figure 3.7b, une façon de représenter la limite de l'agrégation de poissons ainsi que l'incertitude qui lui est associée est de substituer la ligne nette de la

figure 3.7a par une limite large. Un autre moyen de représenter la limite de l'agrégation de poissons consiste à la modéliser à l'aide d'une surface continue. Cette surface continue est définie pour un attribut qui représente le degré d'appartenance d'une observation donnée à l'agrégation. Cette représentation est illustrée à la figure 3.7c. Ces deux types de représentation peuvent également être employés simultanément pour tirer le maximum d'information sur les limites de l'agrégation de poissons à partir des données à disposition.

Afin d'effectuer la représentation spatiale des agrégations de poissons en 2D selon l'approche suggérée dans ce chapitre, plusieurs étapes, montrées à la figure 3.8, doivent être réalisées.

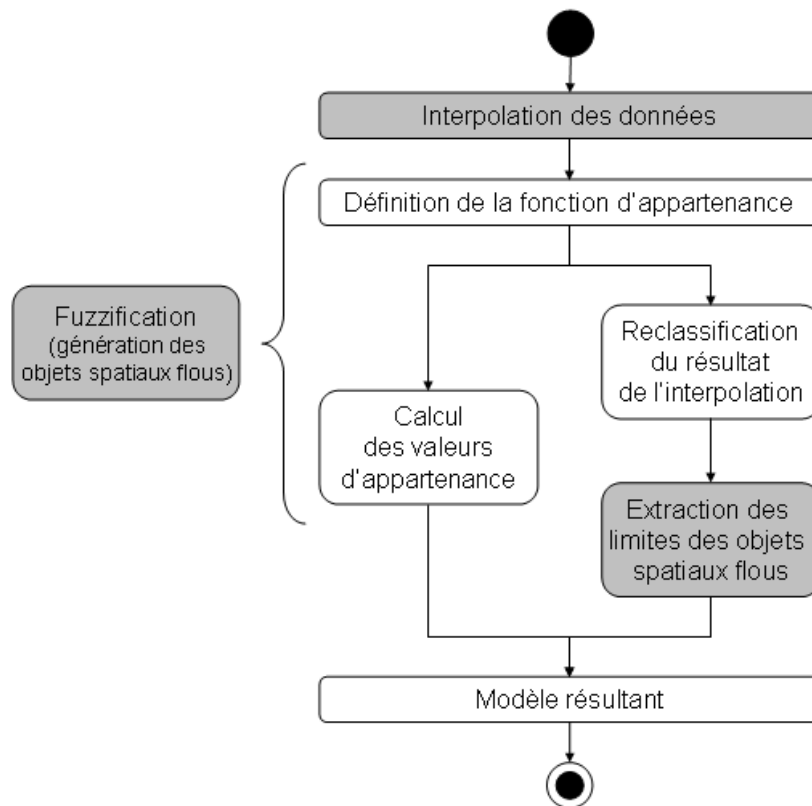


Figure 3.8 - Principales étapes de traitement des données de l'approche proposée.

La première étape utilise une méthode d'interpolation afin de convertir les jeux de données ponctuelles représentant des poids de poissons pêchés pour des positions spatiales aléatoires en des couches de données continues. Ces couches de données continues

correspondent à une représentation de la répartition des poissons partout dans la région d'étude.

La deuxième étape consiste à générer les objets spatiaux flous (fuzzification). Pour ce faire, une fonction d'appartenance est appliquée aux données afin de calculer les valeurs d'appartenance devant être attribuées à chacun des pixels de la surface créée à l'étape précédente. Ainsi, la forme de la fonction et ses principaux paramètres doivent être déterminés. En fait, dans ce chapitre, une agrégation de poissons est assimilée à un ensemble flou dans l'espace 2D possédant des propriétés spécifiques et étant caractérisé par une fonction $\mu : \mathbb{R}^2 \rightarrow [0,1]$. Une agrégation de poissons est donc représentée par un objet spatial dont la limite est une zone de transition graduelle vers l'extérieur de l'objet. Elle est montrée en utilisant des valeurs d'appartenance partielle. Cette limite n'est donc pas homogène et présente plutôt des variations. Cette représentation des agrégations de poissons et, surtout, des variations survenant dans leurs limites est fondée sur le modèle de Zhan (1997), présenté à la section 3.2.2, et se fait sous forme matricielle.

La troisième étape sert à extraire les limites des objets spatiaux flous afin d'obtenir une représentation explicite de celles-ci sous forme vectorielle pour permettre d'autres types d'analyse spatiale. La représentation résultante s'apparente alors aux modèles spatiaux présentés à la section 3.2.1, particulièrement à ceux de Cohn et Gotts (1996) et de Clementini et Di Felice (1996). En effet, la représentation vectorielle cherche uniquement à délimiter l'étendue minimale (*noyau*) et l'étendue maximale (*support*) des agrégations de poissons. Elle ne montre donc pas la transition graduelle observée dans la limite.

Ainsi, à la fin du processus, une représentation vectorielle des agrégations de poissons contenues dans la région d'étude est obtenue d'une part par la création de deux limites pour chacune des agrégations, soit une limite intérieure et une limite extérieure. D'autre part, l'information sur l'imprécision de cette frontière est disponible sous forme matricielle, ce qui complète le modèle et facilite l'interprétation des résultats. Les sections 3.3.1, 3.3.2 et 3.3.3 qui suivent présentent respectivement les notions liées aux trois étapes de

traitement de l'approche proposée, soit l'interpolation des données, la fuzzification et l'extraction des limites des agrégations de poissons.

3.3.1 Interpolation

Dans le cadre de la présente recherche, la première étape de traitement est l'interpolation des données décrivant l'abondance en poissons dans une région d'étude donnée. L'interpolation est la procédure qui permet de prédire les valeurs qu'une caractéristique d'intérêt (attribut) peut prendre à un site non échantillonné à l'aide de mesures effectuées à d'autres endroits de la même zone d'étude (Burrough et McDonnell, 1998). Elle est entre autres utilisée lorsque les données à disposition ne couvrent pas complètement la région d'étude. Dans ce chapitre, l'interpolation sert donc à transformer ces données d'abondance en poissons irrégulièrement réparties dans l'espace en une surface régulière où les valeurs interpolées sont présentées sur une grille. Ainsi, en chaque pixel de cette grille, il existe de l'information sur l'abondance ou la densité de poissons, selon le nombre ou le poids de poissons pêchés. La surface créée décrit donc comment cet attribut varie dans l'espace géographique (Jacquez *et al.*, 2000).

La plupart des techniques d'interpolation se basent sur l'hypothèse que les valeurs d'un attribut à des points proches dans l'espace sont plus susceptibles d'être similaires que les valeurs de cet attribut en des points éloignés (Tobler, 1970; Burrough et McDonnell, 1998). Le calcul des valeurs interpolées pour les endroits non échantillonnés de la zone d'étude est donc effectué en se basant sur les valeurs des points proches.

La génération d'une surface interpolée statistiquement valide est un processus itératif qui nécessite l'application de différentes techniques d'interpolation, suivie de comparaisons qualitatives et quantitatives des résultats (Andrews, 2003). Différentes techniques peuvent être employées afin de procéder à l'interpolation locale de données montrant l'abondance en poissons. Les algorithmes les plus répandus pour ce faire sont disponibles et implantés dans la plupart des logiciels SIG. Dans le cadre de cette recherche, l'interpolation est effectuée dans l'unique objectif d'obtenir une valeur d'attribut en chaque position géographique de la zone d'étude et ce, avec la meilleure exactitude possible. Parmi les

techniques d'interpolation locale les plus courantes, on retrouve entre autres la méthode d'interpolation de pondération par distance inverse ainsi que les méthodes géostatistiques.

L'hypothèse de base de la technique de pondération par distance inverse (*Inverse Distance Weighted* ou IDW) est qu'il existe une certaine corrélation dans les données d'observation et que cette corrélation diminue graduellement en fonction de la distance. Cette technique est communément utilisée en SIG pour créer des couches de données matricielles à partir de données ponctuelles (Burrough et McDonnell, 1998). Elle est simple d'utilisation, requiert peu de temps de traitement et donne généralement des résultats satisfaisants. Cette méthode d'interpolation locale comporte toutefois certaines limitations. Il est par exemple très important de limiter le nombre de points que l'algorithme prend pour le calcul des valeurs prédites. Le choix du poids devant être associé aux points utilisés dans le calcul des valeurs interpolées peut également être délicat. De plus, la méthode par distance inverse considère que la corrélation entre les points est la même dans toutes les directions, ce qui n'est pas nécessairement vrai.

Le but de la géostatistique est d'étudier tout phénomène pouvant être quantifié et qui se développe de façon structurée dans l'espace (Burrough et McDonnell, 1998). Elle vise entre autres l'analyse structurale, l'interprétation, l'estimation, l'interpolation ainsi que la cartographie du phénomène étudié. L'interpolation par des méthodes géostatistiques est intéressante puisqu'elle permet d'obtenir une estimation directe de la qualité des résultats obtenus et des erreurs associées aux valeurs interpolées. La géostatistique utilise des outils tels que le variogramme. Celui-ci mesure l'autocorrélation spatiale et décrit comment les données sont liées entre elles par rapport à leur distance et à leur direction. La géostatistique est utilisée dans de nombreux domaines d'application et est de plus en plus employée pour des études liées au domaine des pêches, particulièrement pour effectuer des analyses spatiales sur la structure des données à partir du variogramme. À titre d'exemple, Mello et Rose (2005) ont exploré l'emploi de méthodes géostatistiques pour quantifier les patrons de répartition des morues de l'Atlantique à différentes échelles à l'aide de données acoustiques. Suite à une interpolation à l'aide de méthodes géostatistiques, une estimation de l'erreur associée à la surface interpolée est donnée, ce qui n'est pas le cas avec la

méthode IDW. Cependant, l'emploi de méthodes géostatistiques implique un temps de traitement accru en comparaison avec d'autres méthodes et celles-ci comportent une certaine difficulté d'application.

3.3.2 Fuzzification

Pour qu'un SIG puisse constituer une aide intéressante pour la prise de décision, un utilisateur doit y retrouver une représentation qui corresponde à sa façon de raisonner. Posons l'hypothèse qu'un utilisateur détermine que les agrégations de poissons se définissent par la proximité dans l'espace de plusieurs points dont la valeur d'abondance (poids de poissons pêchés) est élevée, une valeur élevée correspondant à un poids de poissons pêchés de 60 kg et plus. Dans une représentation faite de façon classique, la valeur de 60 kg est utilisée comme une valeur stricte. Une agrégation de poissons est alors délimitée de manière à ce que seules les observations de poids de poissons pêchés de 60 kg et plus fassent partie de son intérieur. Ainsi, un échantillon présentant un poids de poissons pêchés de 59 kg ne fait strictement pas partie de l'agrégation. Toutefois, dans un processus décisionnel, les valeurs des variables utilisées sont toujours entachées d'une certaine incertitude. De plus, dans l'esprit de l'utilisateur, la valeur de 60 kg n'est qu'une valeur jugée représentative d'une valeur d'abondance élevée et non une valeur absolue (NASCA Géosystèmes, 2006). La représentation floue permet donc de relativiser cette valeur.

L'opération de fuzzification présentée dans cette section sert à passer du domaine réel au domaine flou (Chevrie et Guély, 1998). La présente section vise donc le développement d'une fonction d'appartenance et le calcul des valeurs d'appartenance qualifiant l'ensemble flou correspondant à un objet, une agrégation de poissons dans le cadre de la présente recherche.

3.3.2.1 Définition de la fonction d'appartenance

Tel que mentionné à la section 3.1.2, une fonction d'appartenance doit être déterminée pour caractériser un ensemble flou. La fonction d'appartenance est une représentation graphique de l'importance de la participation d'un élément à un ensemble flou donné. Elle associe donc un poids ou valeur d'appartenance à chacun des éléments, par exemple des pixels, qui

sont traités. Plusieurs ensembles flous peuvent être définis sur la même variable. Par exemple, en considérant le poids de poissons pêchés en différents endroits d'une certaine région, les ensembles flous « faible poids », « poids moyen » et « poids élevé » classés selon une échelle de mesure qualitative ordinaire peuvent être distingués. Chacune de ces notions sera alors explicitée par sa propre fonction d'appartenance (Chevrie et Guély, 1998).

La première étape du calcul des valeurs d'appartenance nécessaires pour représenter les agrégations de poissons est la détermination de cette fonction d'appartenance. Il existe toute une variété de formes de fonctions d'appartenance, chacune étant appropriée pour un jeu de données différent. La détermination de la fonction d'appartenance est une tâche essentielle pour la définition d'un ensemble flou, mais elle constitue également la principale difficulté dans l'application de la théorie des ensembles flous (Schneider, 1999). Certaines règles doivent être respectées. En effet, la fonction d'appartenance doit avoir une forme convexe et ne pas comprendre de zones indéterminées. Deux formes bien connues de fonctions d'appartenance sont les fonctions trapézoïdales (figure 3.9a) et les fonctions triangulaires (figure 3.9c), qui constituent quant à elles un cas particulier des fonctions trapézoïdales. Les fonctions trapézoïdales sont les plus souvent choisies puisqu'elles sont formées de segments de droites et que par conséquent, elles rendent les calculs plus aisés.

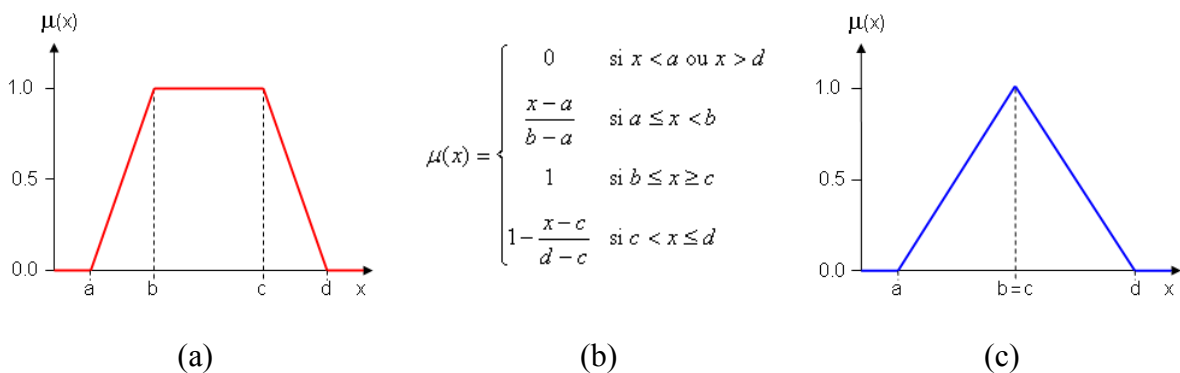


Figure 3.9 - (a) Fonction d'appartenance trapézoïdale et ses paramètres associés (a , b , c et d , quatre nombres réels tels que $a \leq b \leq c \leq d$), (b) forme mathématique de cette fonction, (c) fonction d'appartenance triangulaire qui constitue un cas particulier de la fonction d'appartenance trapézoïdale pour $b = c$.

Suite à la définition de la forme de la fonction d'appartenance, les valeurs des principaux paramètres (*noyau*, *support*) de celle-ci doivent être trouvées (a , b , c et d dans la figure 3.9).

La détermination de ces valeurs peut s'avérer difficile dans certains cas. Par exemple, il est ardu de déterminer les valeurs distinguant une région où l'abondance en poissons est élevée d'une région où elle est faible, étant donné que cela est plutôt subjectif. En général, il existe deux types de méthodes pour déterminer la valeur des paramètres de la fonction d'appartenance : la méthode active et la méthode passive (Tang, 2004). La méthode active propose de dériver la fonction d'appartenance et les valeurs lui étant associées en se basant sur des connaissances empiriques et l'avis des experts. Par ailleurs, la méthode passive calcule la fonction d'appartenance et ses paramètres d'après les données en tant que telles. La détermination des paramètres implique de trouver pour quelles valeurs ou intervalles de valeurs il est certain qu'un objet (un point possédant une certaine valeur d'abondance dans le cas présent) appartient ou n'appartient pas à un ensemble donné.

3.3.2.2 Calcul des valeurs d'appartenance

Une fois la fonction d'appartenance et ses paramètres trouvés, le degré d'appartenance peut être calculé en chaque position, par exemple en chaque pixel, d'un objet spatial. La valeur d'appartenance représente non seulement une indication du degré de participation des éléments de la limite à l'objet, mais décrit également le taux de changement d'un paramètre d'intérêt dans la limite. Elle peut également indiquer s'il est approprié ou plausible de tracer une ligne représentant la limite d'un objet à une certaine position (Wang et Hall, 1996).

La valeur d'appartenance est déterminée mathématiquement ou graphiquement en associant le paramètre de classification à la variable indépendante (axe horizontal) et en projetant sur l'axe de la variable dépendante (axe vertical) les valeurs que la fonction d'appartenance peut prendre.

3.3.3 Extraction de limites

Dans cette étape, une méthode pour la définition des objets, donc de leurs limites, est cherchée. En fait, le but visé est de circonscrire les objets inclus dans les surfaces obtenues lors des étapes de traitement antérieures. Il doit être rappelé ici que Leung (1987) a

démonstré que même si un objet est flou, l'étendue de sa limite floue associée peut quant à elle être identifiée de manière précise.

Il est généralement utile d'extraire les limites des objets, les agrégations de poissons dans le cas de la présente recherche, bien que plusieurs analyses spatiales puissent être effectuées sur de telles surfaces. En effet, la vectorisation des objets permet par la suite l'usage des fonctions d'analyse vectorielle dans les SIG, facilite la référence à un objet donné, permet de traiter des relations spatiales entre les objets et permet une meilleure estimation de leur étendue spatiale. Les objets et leurs limites identifiés sur les surfaces générées correspondent à des entités du monde réel qui revêtent une grande importance aux niveaux scientifique et pratique (Jacquez *et al.*, 2000). Par exemple, bien que les agrégations de poissons ne soient pas des objets dans la réalité, elles sont traitées en tant que tel d'un point de vue scientifique, dans le cadre d'analyse du comportement des poissons. En effet, les limites ont un intérêt scientifique inhérent parce que leurs positions reflètent les processus biologiques, physiques et/ou sociaux sous-jacents (Jacquez *et al.*, 2000).

Une limite peut être définie comme la bordure d'une région homogène ou encore comme une zone de changement dans les valeurs d'une certaine variable (Jacquez *et al.*, 2000). C'est cette dernière définition qui est employée dans ce chapitre. En effet, sur les bords d'une agrégation de poissons, il existe habituellement une bande de changement dans la densité observée. Une limite peut également être vue comme un ensemble de positions spatiales reliées qui séparent des régions possédant des caractéristiques ou valeurs d'attributs différentes (Biomedware, 2008). Par exemple, une limite peut servir à dissocier les régions dont la densité de poissons est élevée des régions où elle ne l'est pas. Les limites peuvent être nettes (bien définies) ou floues (imprécises). Les limites floues se retrouvent pour les phénomènes où il existe une zone de changement assez large d'un type donné à un autre.

Dans le type de représentation privilégié dans ce chapitre et illustré à la figure 3.10, les agrégations de poissons sont vues comme des objets flous. En se basant sur les modèles d'objets flous de la section 3.2, chaque objet flou est décomposé en trois zones distinctes

soient un *noyau*, un *support* et un extérieur (Zhan et Lin, 2003). Le *noyau* représente alors la surface appartenant certainement à l'objet, l'extérieur est la surface qui n'appartient certainement pas à l'objet et la différence entre le *support* et le *noyau* est la limite de l'objet et la zone de transition entre les deux régions certaines. Afin de pouvoir représenter les agrégations de poissons sous forme vectorielle, la limite séparant l'objet de son extérieur ainsi que la limite entre le *noyau* et le *support* doivent être extraites.

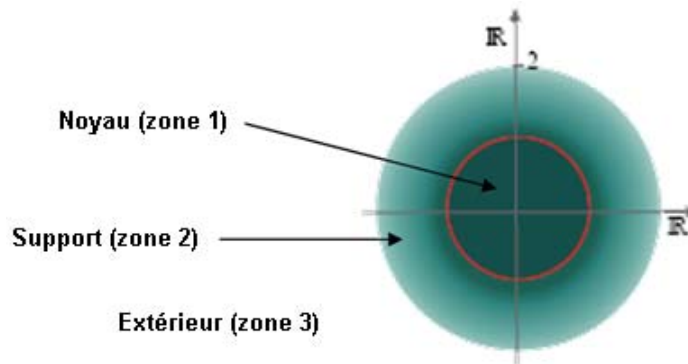


Figure 3.10 - Illustration des différentes composantes et zones d'un objet flou (traduite de Dilo, 2006).

Pour extraire les limites de l'objet flou, les valeurs d'appartenance calculées préalablement sont employées. En effet, un *noyau* correspond à une région pour laquelle les valeurs d'appartenance sont égales à 1, le *support* correspond à la région pour laquelle les valeurs d'appartenance se situent entre 0 et 1 et l'extérieur correspond à la région pour laquelle les valeurs d'appartenance sont de 0. En séparant ces trois régions en des polygones (2D) dont les contours sont bien définis, une représentation vectorielle simple de l'objet flou est créée. Suite à cette opération, les limites de l'étendue maximale (limite extérieure) et de l'étendue minimale (limite intérieure) des objets flous sont également trouvées.

Les limites des *noyaux* trouvés dans l'objet flou, la limite de son *support* (la région où les valeurs d'appartenance sont supérieures à 0) ainsi que leurs relations topologiques doivent être stockées dans le SIG. Tel que mentionné ci-dessus, ces limites sont représentées dans le SIG à l'aide de primitives géométriques vectorielles 2D de type polygone formées de chaînes, de nœuds et de vertices intermédiaires. Pour stocker la topologie liant les polygones décrivant les limites d'un même objet flou, une solution possible consiste à créer

un nouveau type de primitive géométrique vectorielle pour l'objet flou, tel que décrit par Kraipeerapun (2004). La figure 3.11 illustre cette méthode.

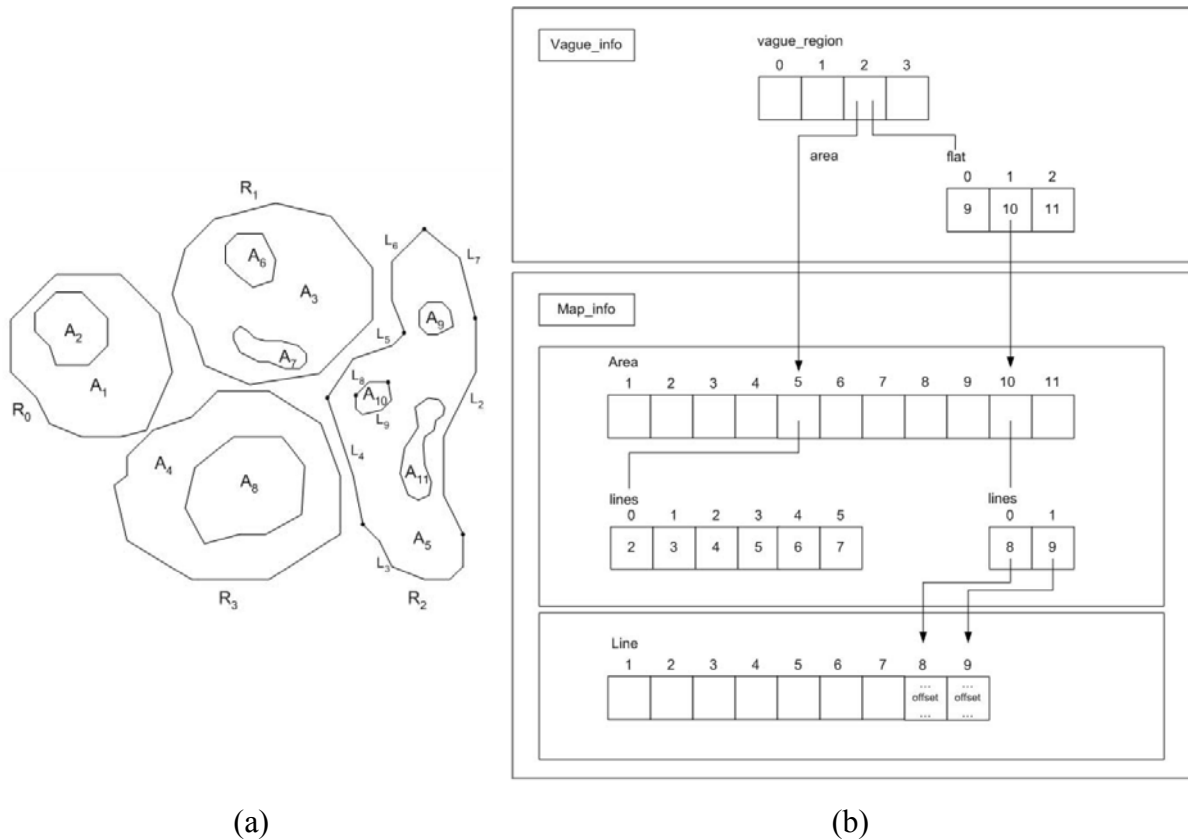


Figure 3.11 - (a) Exemple d'objets flous (*vague region*) dont les *noyaux (flat)* et les *supports (area)* sont représentés par des polygones, (b) relations topologiques créées pour ces objets. Cet exemple est tiré de Kraipeerapun (2004).

3.4 Application de l'approche aux données de pêche

Cette section propose une validation de l'applicabilité de l'approche développée dans la section 3.3 pour la représentation d'agrégations de poissons à une échelle régionale à partir d'observations de poids de poissons pêchés. Dans un premier temps, les jeux de données utilisés sont décrits. Ensuite, la méthode développée est appliquée pour trouver les agrégations de poissons incluses dans ces jeux de données et particulièrement leurs limites. Finalement, les résultats associés à ces expérimentations sont présentés et discutés.

3.4.1 Données utilisées

À l'échelle régionale, les jeux de données biologiques utilisés dans cette recherche proviennent de relevés scientifiques menés par des navires du Ministère des Pêches et Océans Canada (MPO). Ces relevés servent à recueillir des données sur l'abondance et la distribution d'espèces de poissons, particulièrement pour celles commercialement exploitables ou en péril. Ces observations sont acquises par des échantillonnages stratifiés de manière aléatoire dans la région d'étude. Pour la morue de l'Atlantique, par exemple, la zone à échantillonner est divisée en strates en fonction de la profondeur puisque c'est ce paramètre qui est le plus susceptible d'influencer la distribution de cette espèce. Les stations d'échantillonnage sont distribuées aléatoirement au sein de chacune des strates. À chacune des stations, pour une profondeur déterminée, un trait de chalut est effectué pour une durée et une vitesse données. Sur la carte présentée à la figure 3.12, chaque point correspond à une station d'échantillonnage. À chacun des points sont également rattachés plusieurs attributs dont l'espèce de poisson, le nombre d'individus recueillis et le poids total de poissons pêchés. Un extrait de ces jeux de données est présenté à l'annexe A du présent mémoire.

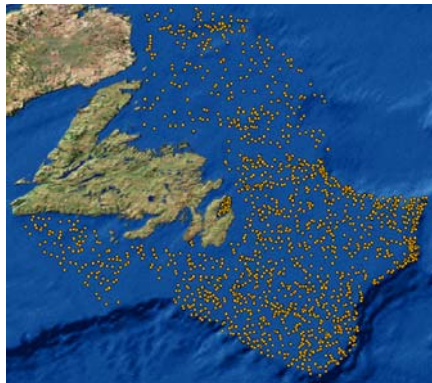


Figure 3.12 - Localisation des sondages scientifiques menés par Pêches et Océans Canada pour la morue de l'Atlantique pour l'année 1990 dans le sud de la région de Terre-Neuve.

De tels jeux de données sont disponibles pour plusieurs espèces de poissons et pour plusieurs années. Par exemple, dans le cas de la morue de l'Atlantique, ces données sont disponibles pour les années 1948 à 2006, avec diverses étendues spatiales et différents nombres de stations, dépendant des années. Les données sont également disponibles pour différentes échelles temporelles. Par exemple, il serait possible d'étudier l'évolution des

agrégations par saison pour voir les migrations dues à la reproduction. En effet, les poissons montrent des changements dans leur distribution et leur processus d'agrégation à diverses échelles temporelles (Mello et Rose, 2005). Par contre, dans le présent projet de recherche, l'intérêt est porté sur l'évolution annuelle des agrégations de poissons.

De plus, dans cette recherche, pour des fins d'illustration de l'approche proposée, seules des agrégations de morues de l'Atlantique (*Gadus morhua*) sont définies. L'attribut que nous utilisons est le poids total de morues prises, exprimé en kilogrammes à deux décimales près. La région étudiée dans cet exemple est montrée sur la figure 3.12 et correspond approximativement aux zones 2J3KL (2J, 3K et 3L) et 3Ps de l'Organisation des Pêches de l'Atlantique du Nord-Ouest (OPANO), telles qu'illustrées sur la carte ci-dessous (figure 3.13).

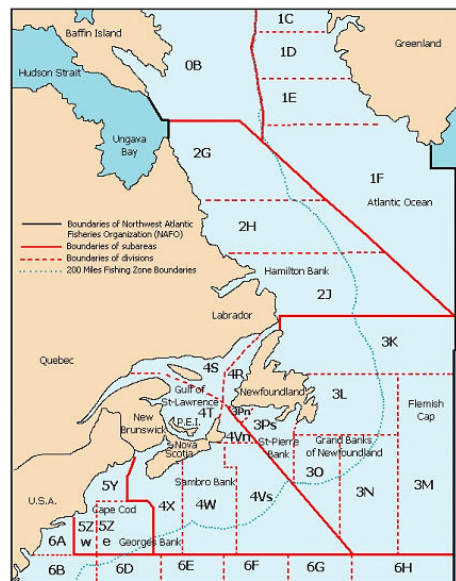


Figure 3.13 - Zones de pêche de l'OPANO (MPO, 2008).

3.4.2 Résultats obtenus

L'application de l'approche proposée sur les jeux de données à disposition nécessite plusieurs étapes de traitement qui ont été décrites dans la section 3.3. Cette section présente les résultats obtenus à chacune de ces étapes de traitement pour le jeu de données montré à la figure 3.12 afin d'illustrer la démarche proposée pour la représentation des

agrégations de poissons en 2D. Il est à noter que l'ensemble des expérimentations ont été effectuées dans le logiciel ArcGIS version 9.2.

3.4.2.1 Fonction d'appartenance définie

Dans le cadre de la présente recherche, l'hypothèse qu'il existe deux classes distinctes pour la définition de la fonction d'appartenance, soient la classe « agrégation » et la classe « non agrégation », est posée. Ces deux classes sont séparées par une zone de dégradation. Une fonction d'appartenance à la classe « agrégation » doit alors être définie puisque c'est sur celle-ci que l'intérêt de la recherche est porté. Cette classe « agrégation » est assimilée à un objet ou ensemble flou pour lequel l'extérieur est la classe « non agrégation » et la zone de dégradation est la différence entre le *support* et le *noyau*.

La fonction d'appartenance est définie sur le paramètre du poids de poissons pêché et elle est linéaire par parties. Sa forme est illustrée à la figure 3.14a.

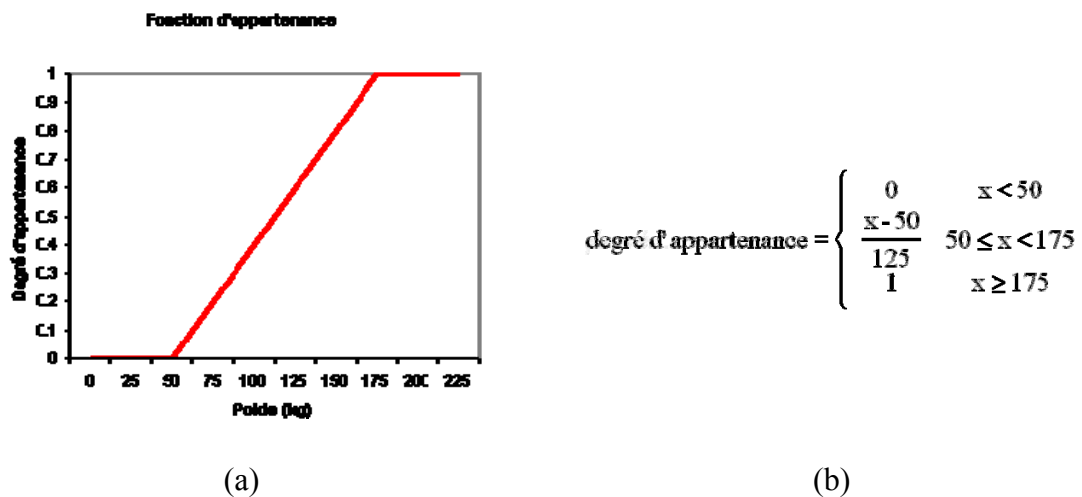


Figure 3.14 - (a) Forme de la fonction d'appartenance et (b) fonction d'appartenance exprimée en termes mathématiques.

En fait, la fonction d'appartenance trouvée est trapézoïdale croissante, composée de deux fonctions équivalentes jointes par une fonction linéaire. Cette fonction linéaire correspond à la limite de l'agrégation et les valeurs à l'intérieur de celle-ci décroissent de l'intérieur vers l'extérieur de l'agrégation. Les points de jointure entre les différentes fonctions constituent les bornes de la fonction d'appartenance.

Les bornes de la fonction d'appartenance doivent être déterminées. Pour ce faire, les trois régions suivantes sont définies pour les agrégations de poissons : une région où le nombre de poissons pêché est élevé correspondant au *noyau* de notre objet « agrégation », une région où le nombre de poissons pêché est moyen correspondant à la limite du *support* et finalement une région où le nombre de poissons pêché est faible ou nul correspondant à l'extérieur de l'objet.

Les bornes définies correspondent aux valeurs pour lesquelles des limites sont tracées. La borne supérieure correspond à l'étendue minimale de l'agrégation de poissons pour laquelle le poids de poissons pêchés correspond totalement à la définition d'une agrégation (elle délimite le *noyau*). Elle représente donc un poids de poissons élevé. La borne inférieure correspond quant à elle à l'étendue maximale de l'agrégation de poissons à partir de laquelle il est tout à fait exclu qu'un poids donné de poissons participe à l'agrégation (elle délimite le *support*). Cette dernière doit correspondre à un poids de poissons faible.

Il faut spécifier ici que, dans le but d'illustrer l'approche proposée, des valeurs arbitraires sont posées pour les valeurs des bornes de la fonction d'appartenance. Cependant, les valeurs de ces bornes sont généralement choisies sur une base statistique et peuvent être modifiées à l'aide des connaissances des spécialistes du domaine des pêches. En effet, la définition des bornes peut changer en fonction de certains critères d'ordre biologique comme l'âge ou l'espèce de poisson étudiée ou la raison pour laquelle l'agrégation s'est formée. En fait, la décision, en logique floue, est basée sur la notion d'expertise qui permet de quantifier le flou à partir de connaissances *a priori* ou acquises antérieurement (Lescieux, 2008). La représentation obtenue suite à un changement dans les bornes de la fonction d'appartenance est par conséquent modifiée.

La fonction d'appartenance exprimée en termes mathématiques est montrée à la figure 3.14b. Les valeurs arbitraires fixées pour cette expérimentation sont également présentées sur cette figure. Il a en effet été déterminé par inspection visuelle des données que les endroits où le poids de morues pêchées est inférieur à 50kg (borne inférieure) se trouvent nécessairement à l'extérieur d'une agrégation de poissons, que les endroits où le poids de

morues pêchées est supérieur à 175kg (borne supérieure) se trouvent certainement dans une agrégation et que la limite entre une agrégation et son extérieur se trouve quelque part entre ces deux valeurs.

3.4.2.2 Interpolation

Les jeux de données doivent être préparés avant l'application des modèles spatiaux issus de la théorie des ensembles flous. Cette préparation implique l'emploi de méthodes d'interpolation spatiale afin d'obtenir une couverture complète de la région d'étude pour l'attribut du poids de poissons pêchés. L'interpolation est réalisée à partir des données d'observation ponctuelles décrites à la section 3.4.1. Pour des fins d'illustration de l'approche proposée et de la démarche suivie, la méthode d'interpolation locale retenue est la méthode IDW étant donnée sa simplicité et sa rapidité d'exécution. Toutefois, il est à noter que certains artefacts peuvent être créés pendant l'interpolation par IDW, dépendant entre autres du poids utilisé pour les points les plus proches ainsi que du nombre de points utilisés. La résolution spatiale de la carte interpolée finale a été fixée à 5km. Suite à l'interpolation, la carte a été reclassifiée afin que les classes correspondent aux différentes catégories déterminées par les paramètres de la fonction d'appartenance. Sur la carte résultante présentée à la figure 3.15, on retrouve donc trois classes distinctes correspondant aux endroits où le poids de morues pêchées est inférieur à 50kg (vert), aux endroits où le poids de morues pêchées se situe entre 50 et 175kg (orange) et, enfin, aux endroits où le poids de morues pêchées est supérieur à 175kg (blanc).

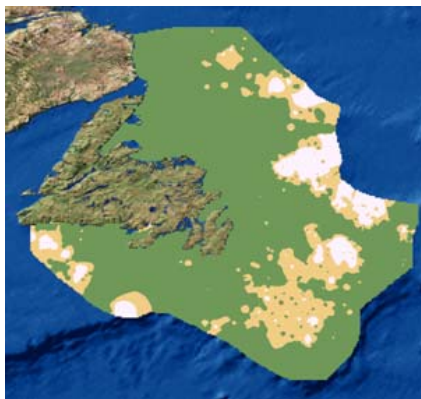


Figure 3.15 - Résultat de l'interpolation par la méthode IDW pour les données d'observation dans la région d'étude; des artefacts créés par la méthode d'interpolation employée sont visibles dans le *support* des agrégations de poissons (points verts dans les sections orangées).

3.4.2.3 Valeurs d'appartenance calculées

Suite à la définition de la fonction d'appartenance et à l'obtention d'une surface matricielle comprenant les valeurs interpolées de poids de poissons pêchés, les valeurs d'appartenance peuvent être calculées. En fait, la fonction d'appartenance définie en 3.4.2.1 permet de déterminer le degré ou la valeur d'appartenance à la classe « agrégation » de chacun des pixels de la carte interpolée à partir des observations de pêche réalisées dans la région d'étude obtenue à la section précédente.

Le résultat de l'application de la fonction d'appartenance sur les valeurs des pixels de la carte issue de l'interpolation nous donne une nouvelle carte qui illustre la position des *noyaux* d'appartenance des données à la classe « agrégation » ainsi que la dégradation des valeurs d'appartenance pour cette même classe. Cette nouvelle carte est présentée à la figure 3.16.

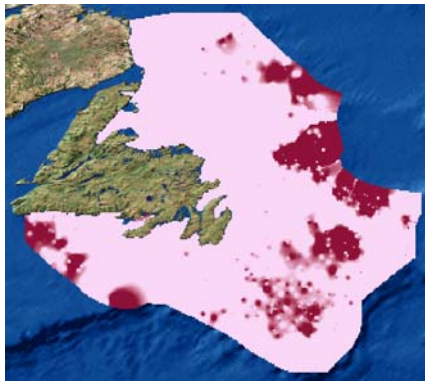


Figure 3.16 - Carte illustrant la position des *noyaux* et la dégradation des valeurs d'appartenance.

3.4.2.4 Représentation des limites des agrégations

La représentation des limites des agrégations de façon vectorielle se base sur la carte obtenue suite à l'interpolation et à la reclassification des valeurs de poids de poissons pêchés en trois catégories. Ces trois catégories ont été déterminées à l'aide des bornes de la fonction d'appartenance. Elles correspondent à l'étendue minimale (*noyau*) et à l'étendue maximale (*noyau* et *support*) de l'agrégation ainsi qu'à son extérieur. En effet, il est supposé ici que les régions de biomasse élevée sont entourées par des régions de biomasse

moyenne, qui forment une zone de transition avec une région où la biomasse faible comme dans l'étude de Hutchings (1996).

Une fois que les données sont présentées sur une grille régulière, des isolignes peuvent être tracées afin d'obtenir une représentation vectorielle de ces catégories. En fait, ces lignes de contours sont couramment utilisées en SIG (ex. courbes de niveaux) pour définir des lignes suivant les mêmes valeurs pour une caractéristique commune (Childs, 2004). Les lignes de contours joignent en effet les positions de même valeur pour un attribut donné. Ainsi, en nous référant à la fonction d'appartenance, nous avons tracé les lignes de contour à l'aide de la carte interpolée. Nous avons donc conservé les lignes de contour pour les poids de poissons pêchés interpolés correspondant à 50kg et à 175kg, ce qui correspond aux valeurs de l'étendue minimale et de l'étendue maximale d'une agrégation que nous avons définies auparavant. La figure 3.17 illustre le résultat obtenu pour la définition des limites des agrégations de poissons. Sur la figure, les lignes bleues correspondent à l'étendue minimale des agrégations tandis que les lignes rouges correspondent à leur étendue maximale. Le fond de la carte correspond quant à lui à la carte interpolée de la figure 3.15.

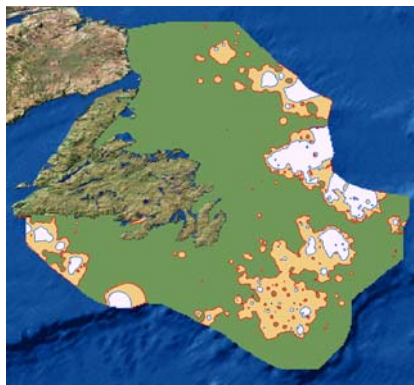


Figure 3.17 - Carte illustrant la position des limites de l'étendue minimale (bleu) et de l'étendue maximale (rouge) des agrégations de poissons.

Suite à l'obtention des limites des agrégations de poissons, il est également possible de découper la carte de la distribution des valeurs d'appartenance obtenue à la section 3.4.2.3. En effet, la limite de l'étendue maximale des agrégations peut être utilisée pour ne garder que les pixels qui appartiennent certainement ou partiellement à l'une ou l'autre des agrégations de poissons de la région d'étude. Ainsi, tel que montré à la figure 3.18, la

représentation ne montre que les agrégations de poissons avec leur *noyau* et la dégradation dans leur limite.



Figure 3.18 - Carte illustrant la position des agrégations de poissons de morue en 1990.

3.4.3 Analyse des résultats

La figure 3.19 donne l'exemple de la représentation d'une des agrégations de poissons obtenues en 2D à une échelle régionale issue des cartes présentées tout au long de ce chapitre. Cette figure illustre parfaitement le résultat attendu de l'approche proposée et de la démarche suggérée dans le présent chapitre. Il s'agit d'une représentation hybride où les limites minimum et maximum de l'agrégation sont montrées de manière vectorielle tandis que l'information sur l'imprécision de cette limite est montrée de manière matricielle entre ces deux limites. La représentation vectorielle visualisée seule permet de voir l'étendue minimale et l'étendue maximale de l'agrégation de poissons, utilisant les bornes fixées pour la fonction d'appartenance. La limite de cette agrégation se situe dans la région comprise entre les étendues minimales et maximales. La représentation matricielle floue visualisée seule permet quant à elle de déceler les *noyaux* ou les endroits qui sont certainement situés à l'intérieur de l'agrégation de poissons. Les limites vectorielles de l'étendue minimale de l'agrégation correspondent d'ailleurs à ces *noyaux*. Les limites vectorielles de l'étendue maximale de l'agrégation correspondent quant à elles au *support* de l'agrégation. La dégradation, ou la variation de degrés d'appartenance, observée sur la représentation matricielle floue peut quant à elle être employée dans un cas où une limite exacte devrait être dessinée, pour évaluer les endroits où cette limite est la plus susceptible de se situer.

Les α -coupes décrites à la section 3.1.2 peuvent être employées à cette fin. Les deux types de représentation sont donc complémentaires.

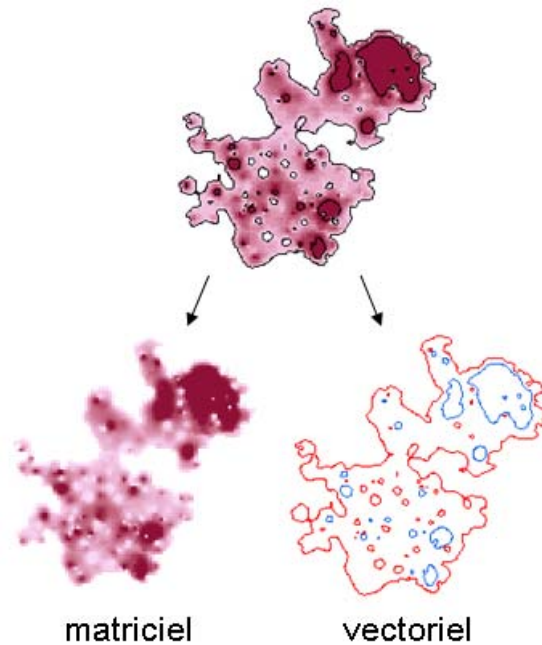


Figure 3.19 - Représentation hybride d'une agrégation de poissons comprenant une partie matricielle et une partie vectorielle.

À la lumière de ces observations, la représentation hybride des agrégations de poissons de la figure 3.19 et basée sur des modèles d'objets flous comporte des gains certains. Elle permet entre autres de combiner les avantages d'une représentation vectorielle avec ceux d'une représentation matricielle et, ainsi, de réaliser différents types d'analyses spatiales. En effet, la représentation vectorielle des objets flous (agrégations de poissons) produite, si elle est employée seule, présente l'avantage de simplifier certaines analyses spatiales sur les limites puisque les opérateurs spatiaux et métriques développés pour les objets exacts peuvent alors être employés directement. La représentation des limites de façon vectorielle facilite également l'étude de l'évolution spatio-temporelle des agrégations de poissons qui est décrite dans la section 3.5. En effet, l'étendue spatiale minimale et maximale des agrégations de poissons est facilement perçue et quantifiée avec ce type de représentation. Cependant, la représentation vectorielle ne donne aucune information sur ce qui se passe dans la limite. Quant à elle, la représentation matricielle floue employée seule possède également quelques avantages. Ce type de représentation est approprié pour la réalisation

de certaines analyses spatiales, comme l'étude de la distribution des données ou déceler certaines tendances dans les données, mais il rend l'étude de l'étendue spatiale des agrégations de poissons plus difficile. En effet, celles-ci ne peuvent pas être distinguées en un coup d'œil par les utilisateurs sur une surface matricielle. La combinaison des deux types de représentation en une représentation hybride permet donc d'obtenir les limites des agrégations facilement en plus de fournir de l'information sur l'imprécision de la localisation de ces limites.

Ainsi, la représentation 2D des agrégations de poissons à l'échelle régionale à partir de l'approche développée dans ce chapitre comporte quelques améliorations par rapport aux représentations existantes présentées au chapitre 1 (section 1.1.4). En effet, il s'agit d'une représentation beaucoup plus intuitive et réaliste des agrégations de poissons que la représentation de données d'abondance à l'aide d'ellipses statistiques, de symboles proportionnels ou même par une carte interpolée. Elle simplifie et facilite alors l'identification des agrégations de poissons. La nature imprécise des limites des agrégations de poissons y est aussi modélisée mathématiquement (Wang, 1994). Puisque les agrégations de poissons sont délimitées et présentées comme des objets spatiaux, leur étendue spatiale est plus facile à estimer que sur une carte interpolée ou un autre mode de représentation. De plus, tel que mentionné précédemment dans ce chapitre, la représentation produite permet d'utiliser autant les fonctions d'analyse spatiale matricielles que vectorielles des SIG pour étudier les agrégations de poissons. L'approche proposée à partir des modèles d'objets spatiaux flous de la section 3.2 est également intéressante puisque, si la topologie des objets flous est gérée, elle permet de manipuler et de visualiser les limites floues des agrégations de poissons dans les technologies SIG existantes en utilisant les requêtes et les opérations topologiques floues développées pour ces modèles (Clementini et Di Felice, 1996; Cohn et Gotts, 1996; Zhan, 1998; Dilo *et al.*, 2004; Dilo, 2006).

Cependant, bien que l'approche proposée puisse améliorer la représentation spatiale dans les zones où les concentrations de poissons sont élevées, elle comporte également certaines lacunes. Par exemple, à l'échelle régionale, étant donné le faible nombre d'observations par

rapport à la superficie du territoire à couvrir, la résolution spatiale n'est pas suffisante pour permettre la délimitation des agrégations proprement dites. En effet, dans certains cas, la représentation montre plutôt des agrégations d'agrégations de poissons.

3.5 Application de l'approche pour l'étude de l'évolution spatio-temporelle des agrégations de poissons

Actuellement, les biologistes marins ne s'entendent pas sur les causes du déclin de la morue de l'Atlantique au large de Terre-Neuve. Identifier les causes de ce déclin peut s'avérer complexe étant donné le nombre de facteurs pouvant avoir joué un rôle et leur importance respective. Selon Hutchings (1996), ce débat est principalement dû à l'absence de modèles spatio-temporels sur la distribution de la densité de cette espèce. De plus, Hilborn et Walters (1992) affirment qu'une des composantes fondamentales d'un modèle portant sur les stocks de poissons est la distribution de ces stocks dans l'espace et dans le temps. De là vient l'importance de créer une représentation réaliste des agrégations de poissons et, surtout, de leur évolution spatio-temporelle.

Ainsi, l'approche proposée et les objets flous 2D qui en résultent peuvent être utilisés pour l'étude de l'évolution spatio-temporelle des agrégations de morues de l'Atlantique (*Gadus morhua*). Pour l'étude de l'évolution spatio-temporelle des agrégations de morues, les données utilisées proviennent des relevés scientifiques de Pêche et Océans Canada réalisés de 1983 à 1992 pour la saison d'automne dans la région correspondant aux divisions 2J3KL de l'OPANO. Les changements des agrégations de poissons dans l'espace et dans le temps ont donc été déduits des représentations créées à partir de jeux de données acquis dans des conditions similaires, voire identiques, pour des années différentes. Ces données ont également déjà été employées pour la réalisation d'études portant sur l'abondance et la distribution des morues de l'Atlantique (ex. Baird *et al.*, 1992; Bishop *et al.*, 1993; Kulka *et al.*, 1995; Hutchings, 1996; Atkinson *et al.*, 1997).

Dans un premier temps, une interpolation a été réalisée par géostatistique en utilisant la méthode de krigeage afin de créer une couche de données de base. La qualité des résultats obtenus dans la présente section est donc potentiellement supérieure à la qualité des

résultats obtenus à la section 3.4.2.2 puisque le krigeage ne crée généralement pas d'artéfacts. Cependant, la résolution choisie étant plus petite (10km au lieu de 5km), les résultats paraissent moins nets et une discordance est observée entre les représentations vectorielles et matricielles des mêmes agrégations. La résolution choisie correspond toutefois mieux à la densité réelle des données observées.

Par ailleurs, les classes utilisées pour la définition des agrégations correspondent à celles spécifiées pour qualifier la biomasse par Hutchings (1996) et sont les suivantes : biomasse faible (moins de 100 kg), biomasse moyenne (100-500 kg) et biomasse élevée (plus de 500 kg). Dans la présente étude, pour la représentation des agrégations de poissons, les valeurs de biomasse faible correspondent à l'extérieur des agrégations alors que les valeurs de biomasse élevée correspondent aux *noyaux* des agrégations. Les valeurs de biomasse égales ou supérieures à 100 kg représentent donc la zone de *support*. L'utilisation de ces classes plutôt que de celles employées dans la section précédente (section 3.4.2.1), qui étaient fixées de façon arbitraire, peut sensiblement améliorer le réalisme de la représentation.

La figure 3.20 montre les résultats obtenus suite à l'application de l'approche proposée sur les données décrites précédemment. Sur les cartes, les zones les plus foncées représentent les régions appartenant le plus certainement aux agrégations de poissons. Les zones délimitées par des polygones bleus font certainement partie des agrégations de poissons (*noyaux*) puisque la biomasse qui y est observée est très élevée (plus de 500 kg) alors que les zones délimitées par des polygones rouges correspondent au *support* des agrégations. De manière générale, les régions appartenant certainement aux agrégations correspondent aux positions de forte biomasse notées par Hutchings (1996) entre 1983 et 1992. De plus, les supports des agrégations correspondent aux positions de biomasse moyenne également trouvées par Hutchings (1996). Ainsi, les mêmes patrons généraux sont observés pour les agrégations (ex. endroits où elles sont situées, forme) dans cette recherche et dans celle de Hutchings (1996).

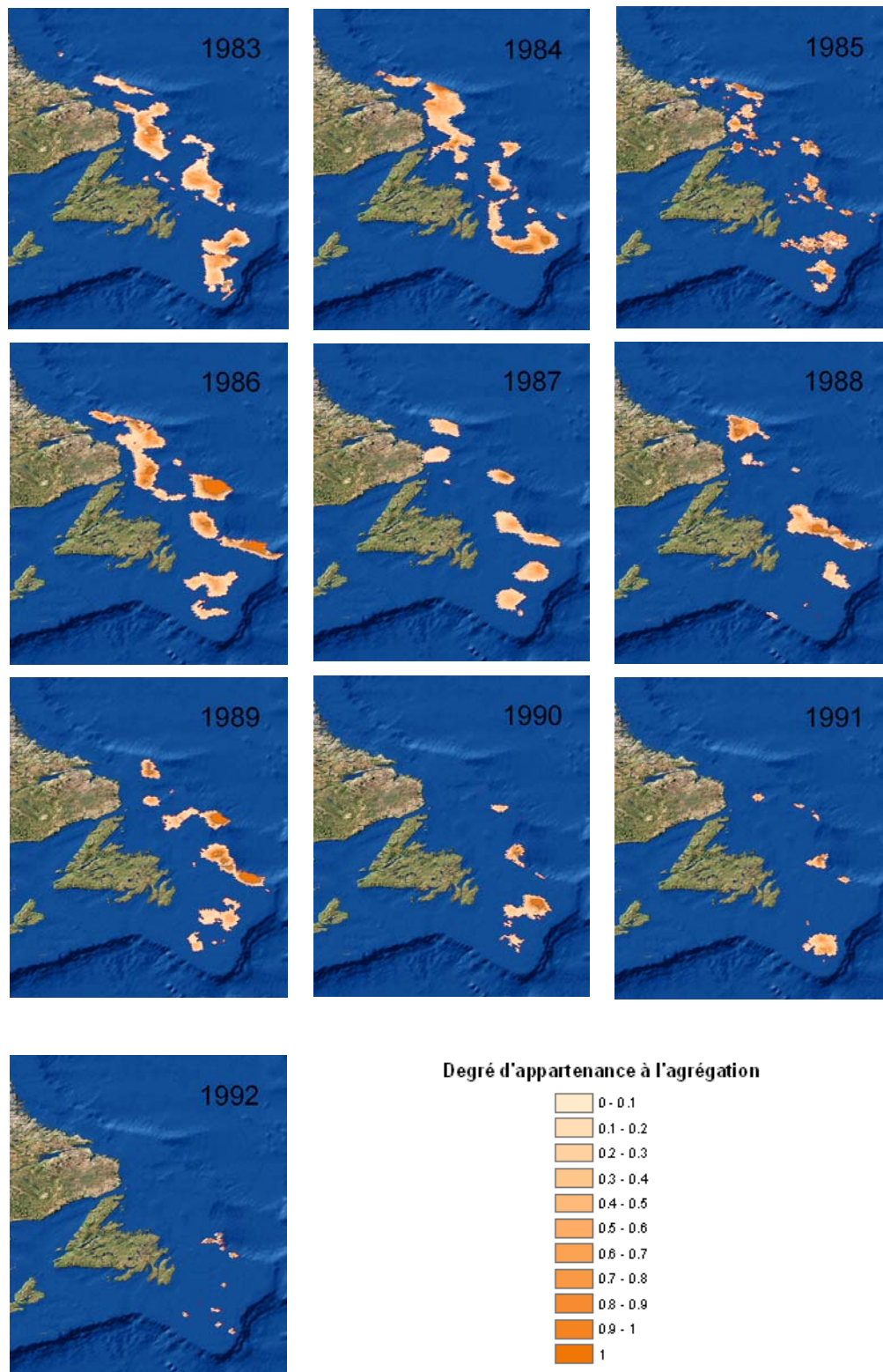


Figure 3.20 - Représentation de l'évolution spatio-temporelle des morues de l'Atlantique (*Gadus morhua*) dans les divisions 2J et 3KL de l'OPANO pour la saison d'automne des années 1983 à 1992.

Les cartes de la figure 3.20 permettent de tirer certaines conclusions d'ordre biologique. Elles permettent entre autres d'observer les fluctuations que le stock correspondant aux divisions 2J3KL de l'OPANO a connues de 1983 à 1992. Par exemple, les étendues maximales des agrégations de poissons apparaissant sur la carte de 1985 semblent plus petites que celles des agrégations montrées sur les cartes des années 1984 et 1986. Par contre, les étendues minimales des agrégations, correspondant aux régions certaines des agrégations, semblent rester constantes pour ces trois années, bien qu'elles ne se situent pas dans les mêmes endroits. Des cartes de la figure 3.20, il est également possible d'observer que la proportion des zones de couleur foncée augmente par rapport à la surface totale des agrégations de poissons, qui elle diminue, entre 1983 et 1992. Ainsi, il semble que l'agrégation des poissons devient plus forte lorsque la population totale du stock diminue. Les mêmes observations ont été réalisées par Rose (2007) ainsi que par Bennett (2008). Il est à noter que ce rapport mathématique support/noyau est un bon indicateur, facile à calculer, de la compaction d'une agrégation de poissons. Un autre exemple de conclusion pouvant être tirée des cartes présentées à la figure 3.20 est qu'un déclin généralisé du stock de morues des divisions 2J3KL de l'OPANO est survenu à partir de 1990. En effet, sur les cartes, la taille des agrégations de poissons diminue de façon constante de 1990 à 1991 alors que l'effort de pêche s'est concentré sur les régions où la morue était hyper-agrégée. Ce résultat peut être confirmé par le fait que plusieurs études ont observé qu'un déclin de la population de morue s'est amorcé vers 1990 et qu'à partir de ce moment, la biomasse a chuté rapidement (Bishop *et al.*, 1993; De Young et Rose, 1993; Hutchings, 1996; Atkinson *et al.*, 1997).

D'un point de vue cartographique, la visualisation des cartes présentées à la figure 3.20 montre que la constatation et l'étude de l'évolution spatio-temporelle des agrégations de poissons est facilitée par le mode de représentation adopté dans ce chapitre. En effet, l'évolution spatiale des agrégations (ex. déplacements, fusion ou division d'agrégations de poissons) dans le temps y est beaucoup plus évidente que sur une carte interpolée ou un autre type de représentation existante. De plus, les changements dans la forme des agrégations de poissons peuvent être caractérisés. La représentation de l'évolution spatio-temporelle réalisée à l'aide de l'approche proposée dans ce chapitre permet également

d'apporter une compréhension spatio-temporelle du phénomène des agrégations de poissons. En effet, en plus de présenter la situation actuelle et la situation passée des agrégations de poissons, elle permet de décrire les différences spatiales entre les différents états de la ressource dans le temps (ex. différences de forme, magnitude et direction du changement). Ainsi, elle permet d'expliquer, entre autres, comment le déclin de la morue observé dans la région de l'Atlantique nord-ouest pour les années étudiées s'est opéré. Pour une compréhension accrue de l'évolution spatio-temporelle des agrégations de poissons, la résolution temporelle pourrait être raffinée (ex. utilisation d'une échelle saisonnière au lieu d'une échelle annuelle). Ainsi, certains mouvements des agrégations de poissons, comme les migrations, pourraient être mises en évidence.

3.6 Conclusion du chapitre

La représentation de l'environnement marin et des données halieutiques en particulier comporte quelques limites dues aux caractéristiques des technologies SIG actuelles. Ces systèmes tendent habituellement à représenter les phénomènes et objets environnementaux tels que les agrégations de poissons à l'aide d'objets géométriques aux limites bien définies, ce qui ne décrit pas adéquatement leurs caractéristiques spatiales.

Ce chapitre présente une utilisation de la théorie des ensembles flous pour modéliser et visualiser géographiquement des agrégations de poissons. En effet, à une échelle régionale, les agrégations de poissons ne constituent pas un objet observable directement et concrètement dans la réalité, mais plutôt une agglomération d'objets spatiaux réels (poissons). Étant donné que la représentation basée sur la théorie des ensembles flous expose les frontières des agrégations de poissons comme des zones de transition et en montre les variations locales, elle semble plus réaliste que les représentations obtenues de manière conventionnelle dans les SIG. Ces limites sont représentées à la fois avec une structure de données vectorielle et avec une structure de données matricielle, ces deux représentations de la même réalité observée se complétant.

Les résultats obtenus dans ce chapitre pour la représentation des agrégations de poissons offrent des perspectives intéressantes pour des besoins d'analyse spatiale et pour l'aide à la

décision. Par exemple, ils peuvent être utilisés pour l'étude de l'évolution spatiale des agrégations de poissons à diverses échelles temporelles. En effet, les résultats obtenus de l'approche proposée permettent de percevoir des changements tels que la modification de la taille, de la forme ou de la répartition des agrégations dans le temps. Cela entraîne ainsi une meilleure compréhension de ce phénomène. Toutefois, pour valider la fiabilité des résultats obtenus pour la représentation des agrégations de poissons, d'autres expérimentations devront être menées à l'aide des données de relevés scientifiques et des comparaisons avec d'autres travaux devront être effectuées.

La topologie des objets flous représentant les agrégations de poissons devrait être stockée, dans des travaux futurs, afin de permettre l'emploi des opérateurs topologiques flous (ex. inclusion, intersection, union) pour leur analyse. Par ailleurs, il pourrait aussi être intéressant, lors de travaux futurs, de superposer les représentations produites dans ce chapitre à des données sur les paramètres physiques de l'eau (ex. température, salinité) et d'effectuer certaines analyses statistiques afin d'identifier une relation éventuelle entre les modifications des agrégations de poissons et les changements dans les facteurs environnementaux.

Chapitre 4 Représentation des agrégations de poissons et de leurs dynamiques en 3D

Tel que mentionné dans le chapitre 2, malgré les capacités intéressantes présentées par les SIG actuels pour la représentation, l'analyse et la gestion de données de toutes sortes, l'utilisation de ces systèmes pour l'étude des données marines, comme des données de pêche acquises à une échelle locale, conduit dans la plupart des cas à des problèmes dus à la nature 3D et dynamique de ces données.

Le présent chapitre propose une nouvelle approche dynamique et interactive pour la modélisation et la visualisation des agrégations de poissons à une échelle locale, dans un espace 3D. La méthode de représentation proposée est basée sur une approche intégrée utilisant la triangulation Delaunay 3D dynamique, une structure de données vectorielle, ainsi que l'algorithme de *clustering* 3D des *α -shapes* (section 4.1). D'abord, les principaux concepts liés à la construction du modèle dynamique 3D sont décrits (sections 4.2 et 4.3). Ensuite, l'approche proposée est validée en utilisant plusieurs jeux de données expérimentaux (section 4.4.1). Elle est par la suite appliquée à des données de pêche acquises par des méthodes et technologies acoustiques afin d'extraire, de visualiser et d'analyser les agrégations de poissons (section 4.4.2). Ce chapitre discute également du potentiel d'utilisation de l'approche pour des applications telles que le suivi et l'étude de la dynamique et du comportement des poissons (section 4.4.3). Finalement, quelques résultats sont présentés et les avantages et inconvénients du modèle 3D suggéré sont discutés.

4.1 Approche intégrée pour la modélisation des agrégations de poissons en 3D

L'approche que nous proposons pour la représentation 3D des agrégations de poissons se base sur la triangulation Delaunay appliquée à un espace 3D afin d'en tirer une structure qui est ensuite raffinée à l'aide d'une technique de *clustering* tridimensionnel.

Rappelons brièvement que la triangulation Delaunay est une structure de données vectorielle fréquemment employée pour des applications en 2D/2.5D, en particulier dans les SIG, pour la création de modèles numériques de terrain et pour réaliser des interpolations (Li *et al.*, 2005). Elle a également été utilisée pour des applications tridimensionnelles dans plusieurs domaines comme la géologie et la modélisation géoscientifique (Gable *et al.*, 1996; Lattuada et Raper, 1996; Courrioux *et al.*, 2001; Xue *et al.*, 2004; Hashemi *et al.*, 2008; Ledoux et Gold, 2008). La triangulation Delaunay possède plusieurs avantages par rapport aux autres structures de données spatiales pour la gestion de données ponctuelles distribuées irrégulièrement (voir section 4.1.2). Elle convient donc bien pour la représentation d'objets complexes tels que des agrégations de poissons.

À l'échelle régionale, les données disponibles pour représenter le phénomène des agrégations de poissons sont limitées à deux dimensions. Ce n'est pas le cas à l'échelle locale pour laquelle il existe de plus en plus de données en 3D grâce aux améliorations des techniques d'acquisition, à l'introduction de nouveaux capteurs acoustiques, tels que l'échosondeur multifaisceaux 3D SIMRAD's MS 70, et au développement de logiciels spécialisés dans le traitement de ces données. Les technologies acoustiques deviennent de plus en plus précises tant pour l'observation des agrégations de poissons dans la colonne d'eau (Gerlotto *et al.*, 1999) que pour le suivi de chacun des individus faisant partie de ces agrégations (Ehrenberg et Torkelson, 1996). Ces dernières observations fournissent une position des poissons en latitude, longitude et profondeur. Cette caractéristique permet l'emploi de techniques de *clustering* pour regrouper les individus appartenant à une même agrégation (objet).

4.1.1 Clustering 3D

Le *clustering* peut être décrit comme la classification d'objets en différents groupes, appelés *clusters*, de façon à ce que les données de chacun d'entre eux partagent des caractéristiques communes comme la proximité dans l'espace. Les regroupements naturels peuvent donc être trouvés dans une distribution de données. Plusieurs algorithmes et techniques ont été développés pour effectuer le *clustering* de jeux de données. Certains sont plus intéressants que d'autres pour la modélisation des agrégations de poissons, notamment

l'algorithme DBSCAN (*Density Based Spatial Clustering of Applications with Noise*) et l'algorithme des α -shapes.

DBSCAN est un algorithme de *clustering* qui s'ajuste aux différentes densités locales de points rencontrées. Par exemple, dans le cas de données de biologie marine 3D, un *cluster* regrouperait l'ensemble des individus trouvés dans une région de forte densité, correspondant alors à une agrégation de poissons. Ensuite, toutes les observations qui ne seraient pas classées dans un *cluster* seraient considérées comme du bruit dans le jeu de données. Suite à un *clustering* des données de biologie marine 3D réalisé par l'algorithme DBSCAN, la triangulation Delaunay pourrait être appliquée à l'intérieur de chaque *cluster* pour créer les objets ou agrégations et en faciliter la visualisation (Buelens *et al.*, 2005).

L'algorithme des α -shapes peut également être appliqué à un jeu de données de pêche 3D afin de créer des *clusters*. Tout comme l'algorithme DBSCAN, l'algorithme des α -shapes comporte l'avantage de ne pas demander que le nombre de *clusters* soit déterminé *a priori* comme c'est le cas avec certains algorithmes de *clustering* conventionnels (ex. *k-means*). L'algorithme des α -shapes est utilisé dans plusieurs applications scientifiques pour identifier les formes qui sont incluses dans des ensembles de points irrégulièrement répartis dans l'espace (Buelens *et al.*, 2004). L'algorithme des α -shapes se base sur la triangulation Delaunay, la structure que nous avons retenue pour la modélisation 3D de nos données (voir section 4.1.2). En effet, un maillage tridimensionnel composé de tétraèdres doit d'abord être obtenu afin d'appliquer l'algorithme des α -shapes. Celui-ci ne conserve alors que des groupes cohérents de tétraèdres qui sont considérés comme des objets. Comme l'algorithme des α -shapes se base sur la distance entre les données, les *clusters* ainsi définis rassemblent les observations ayant une proximité dans l'espace. Donc, comme DBSCAN, l'algorithme des α -shapes est approprié pour la modélisation des agrégations de poissons puisqu'il permet de regrouper les individus trouvés dans les régions de densité élevée. Le résultat final donné par l'application de l'algorithme des α -shapes peut également être raffiné en rejetant les plus petits *clusters* ou ceux qui ne sont pas conformes au modèle attendu par exemple (Buelens *et al.*, 2004). L'avantage principal offert par l'algorithme des α -shapes, en comparaison avec les autres algorithmes tels que DBSCAN, est qu'il définit

immédiatement la limite de chacun des *clusters* inclus dans un nuage de points et, de cette façon, les objets sont directement détectés et formés. De plus, comme l'algorithme des *α -shapes* est basé sur la triangulation Delaunay et sa structure de données dynamique, il permet au modèle 3D obtenu d'être raffiné dynamiquement. Les propriétés de cet algorithme en font donc une méthode de choix pour la détection et la modélisation des agrégations de poissons.

4.1.2 Modélisation 3D

Préalablement à l'application d'une technique de *clustering*, l'approche proposée nécessite le choix et la construction d'un modèle 3D à l'aide d'une structure de données géométrique. L'environnement marin peut être conceptualisé en un espace continu rempli d'objets discrets. Puisque ces objets sont positionnés dans un espace 3D (latitude, longitude, profondeur) dans la réalité, un modèle qui supporte ce type de données est nécessaire. De plus, le type et la quantité d'informations devant être gérées dans le SIG ont une influence sur le type de représentation 3D qui doit être utilisé.

Pour représenter des objets tridimensionnels, une extension à un espace 3D des structures de données présentées dans le chapitre 2 est nécessaire. En effet, les deux structures de données les plus répandues pour représenter l'espace géographique, tel que l'environnement marin, dans un SIG 3D sont basées sur les *rasters* (structures matricielles) et les vecteurs (Shyue et Tsai, 1996). Rappelons que la représentation vectorielle définit les objets discrets à l'aide d'un ensemble de points, de lignes, de polygones et de polyèdres (volumes) dans le cas de données 3D avec leurs relations topologiques tandis que la représentation matricielle décrit les objets à l'aide d'un ensemble de cellules 3D régulièrement distribuées. Plusieurs méthodes peuvent être employées pour représenter des données tridimensionnelles. Li (1994) classe ces méthodes en deux catégories de représentation pour les objets tridimensionnels. La première catégorie regroupe les types de représentation où seule la surface des objets est modélisée. La seconde catégorie comprend, quant à elle, les types de représentation basés sur des volumes où autant l'extérieur que l'intérieur des objets est décrit. Un bref survol de quelques-unes des méthodes comprises dans la deuxième catégorie (voir figure 4.1) est proposé dans cette section.

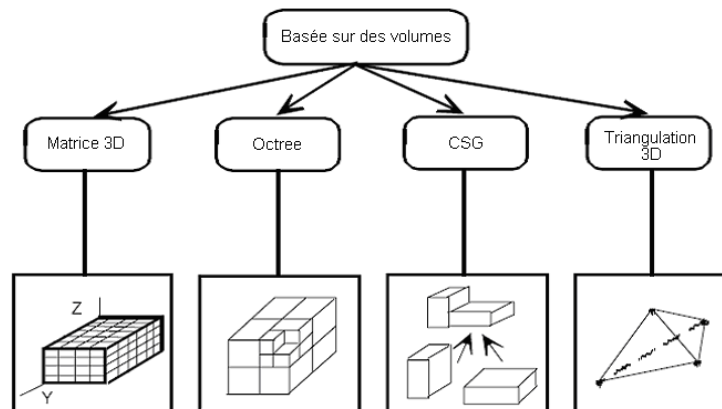


Figure 4.1 - Représentations basées sur des primitives volumiques tirées de Abdul-Rahman et Pilouk (2007) :
 (a) structure de données régulière faite de voxels (*3D array*), (b) représentation *octree*, (c) modélisation géométrique CSG (*Constructive Solid Geometry*) représentant un objet 3D en combinant des objets solides simples (ex. cylindre, sphère) à l'aide d'opérateurs géométriques booléens (ex. union, intersection),
 (d) représentation par des tessellations irrégulières comme la triangulation Delaunay 3D (*3D TIN*).

Dans la plupart des SIG 3D, une structure de données régulière, telle que la structure matricielle, est utilisée pour partitionner l'espace en régions uniformes ou éléments volumiques tels que des voxels (équivalent 3D des pixels dont le nom vient de *VOLUME ELements*). L'avantage principal de cette technique est sa simplicité. En effet, elle est relativement facile à générer, elle est aisément gérée dans un ordinateur puisqu'elle représente une matrice de cellules et sa topologie est définie implicitement (Ledoux et Gold, 2008). Cette structure de données est cependant moins bien adaptée à des données irrégulièrement distribuées, particulièrement si une résolution fine est employée. Dans le cas de données marines, cette structure de données peut également devenir très lourde et volumineuse. De plus, dans une structure de données matricielle 3D, un objet est trouvé en regroupant les voxels possédant les mêmes valeurs d'attributs. Toutefois, cet objet et ses frontières ne sont pas représentés explicitement. À propos des structures de données matricielle employées pour la modélisation et la représentation 3D, Ledoux et Gold (2008, p. 549) affirment qu'« elles ont été sévèrement critiquées par plusieurs en raison de leurs imperfections au niveau théorique, entre autres le fait qu'elles discrétisent l'espace de façon non naturelle, ainsi qu'au niveau pratique comme la quantité de mémoire requise. »

Une autre manière de modéliser les phénomènes 3D est d'employer des tessellations hiérarchiques telles que des structures de données *octree* (Samet, 1984). Celles-ci peuvent également être utilisées pour réduire la quantité de mémoire nécessaire pour stocker une grille 3D possédant une structure de données régulière. Cette structure de données crée un arbre pour décrire comment les objets 3D sont distribués dans l'espace tel que montré à la figure 4.2. L'espace est alors divisé en cubes de grandeurs variées et la position de ces cubes, soit à l'intérieur ou à l'extérieur des objets, est donnée (Abdul-Rahman et Pilouk, 2007). Un des problèmes de cette structure est que, lorsque la distribution de données est irrégulière, l'arbre peut être déséquilibré (De Berg *et al.*, 2000). Dans ce cas, les régions qui sont densément peuplées par des objets ont plus de subdivisions que les régions qui sont faiblement peuplées. Ainsi, la structure est moins efficace pour répondre à des requêtes et pour l'analyse spatiale. De plus, cette structure de données est très sensible à la position des objets. De petits changements ou une donnée aberrante peuvent produire un *octree* complètement différent du même espace.

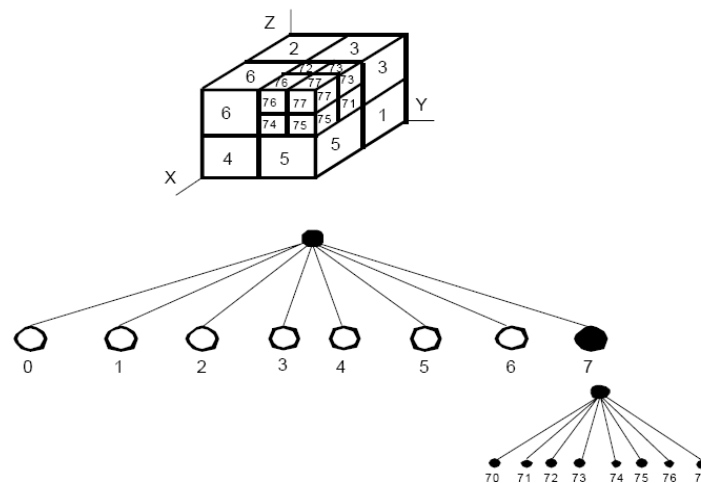


Figure 4.2 - Exemple de représentation *octree* tiré de Abdul-Rahman et Pilouk (2007).

Des tessellations irrégulières comme la triangulation de Delaunay généralisée à un espace 3D peuvent également être appropriées pour modéliser un phénomène ou des objets 3D constitués de points irrégulièrement distribués (Joe, 1991; Ledoux et Gold, 2006; Hashemi *et al.*, 2007; Ledoux et Gold, 2008). Ce type de représentation est basé sur des volumes et

utilise une structure de données composée de tétraèdres. L'avantage principal de ces tessellations est qu'elles suivent habituellement le contour des ensembles de points observés et, ainsi, elles sont adaptées à la distribution de points. Elles offrent donc la possibilité de modéliser n'importe quel volume arbitraire. De plus, la partition de l'espace obtenue à l'aide de tessellations irrégulières est potentiellement meilleure que celle qui résulte de tessellations régulières ou de techniques de type *octree* parce qu'elle n'impose aucune contrainte pour la subdivision de l'espace. Alors, la représentation obtenue est adaptée à la complexité du phénomène étudié et n'est pas comprimée dans une structure rigide telle qu'une grille 3D (Ledoux et Gold, 2008). En regard de ces propriétés, la triangulation Delaunay 3D présente une alternative très intéressante pour la représentation d'agrégations de poissons en 3D.

Afin de réaliser la modélisation spatiale des agrégations de poissons dans un espace 3D, deux principales étapes sont donc exécutées. Premièrement, les points du jeu de données 3D représentant des observations de pêche sont utilisés pour la construction de la triangulation Delaunay. La triangulation Delaunay 3D du jeu de données est construite selon un algorithme incrémental qui définit la proximité spatiale entre les points. Suite à la création du modèle composé de tétraèdres, l'algorithme des α -shapes 3D (Edelsbrunner et Mücke, 1994) est appliqué au jeu de données afin de détecter et d'extraire les agrégations de poissons (Carette *et al.*, 2007). En effet, l'emploi de l'algorithme des α -shapes 3D assure que les points qui appartiennent au même objet sont regroupés et les agrégations de poissons peuvent alors être reconstruites et visualisées (Buelens *et al.*, 2004). Les relations topologiques entre les objets sont stockées explicitement et peuvent être retrouvées en utilisant la structure de données sur laquelle la triangulation Delaunay 3D est basée. Cette structure de données possède la propriété d'être dynamique et cinétique (Mostafavi *et al.*, 2003; Ledoux et Gold, 2006; Hashemi *et al.*, 2007). Cela signifie que les objets, c'est-à-dire les points représentant les poissons, peuvent être ajoutés, supprimés et déplacés. Cette propriété permet donc à l'utilisateur d'avoir différents types d'interactions avec le modèle 3D. Les concepts relatifs à la triangulation Delaunay 3D, à l'algorithme des α -shapes en 3D ainsi qu'à la structure de données employée sont décrits dans les sections suivantes.

4.2 Triangulation Delaunay 3D

Cette section propose quelques notions théoriques sur la triangulation Delaunay dans un espace 3D et la structure de données choisie pour la stocker puisqu'elle constitue la base de notre modèle. Cette structure de données est dynamique et permet donc au modèle d'être interactif. La présente section passe également brièvement en revue les tests géométriques ainsi que l'algorithme incrémental avec flips utilisés pour construire la triangulation Delaunay 3D.

4.2.1 Définition de la triangulation Delaunay 3D

Un ensemble E est dit convexe lorsque pour chaque paire de points a et b contenus dans cet ensemble, le segment $[a, b]$ les joignant est également entièrement contenu dans E (voir figure 4.3).



Figure 4.3 - Exemples (a) d'un ensemble convexe et (b) d'un ensemble non convexe.

L'enveloppe convexe H peut quant à elle être définie comme étant le plus petit ensemble convexe qui contient complètement un ensemble de points S .

La triangulation Delaunay, pour un ensemble de points S distribués dans un espace euclidien 3D, notée $DT(S)$, divise l'espace sous-tendu par l'enveloppe convexe de S en tétraèdres qui admettent tous une sphère circonscrite vide, c'est-à-dire qui ne contient aucun autre sommet de $DT(S)$. Cette dernière propriété est connue sous le nom de critère de Delaunay. Ainsi, parmi toutes les triangulations possibles de l'ensemble de points S , la triangulation Delaunay 3D est unique en vertu de la propriété de Delaunay quand les points sont en position générale (il n'existe pas cinq points ou plus qui soient sur une même

sphère). Le test géométrique *InSphere* permettant de déterminer si la sphère circonscrite à un tétraèdre est vide est présenté à la figure 4.4. Il s'agit du calcul d'un déterminant qui sert à vérifier si un point p se trouve à l'intérieur, à l'extérieur ou encore sur une sphère définie par quatre points (a, b, c, d), les quatre sommets d'un tétraèdre de la triangulation.

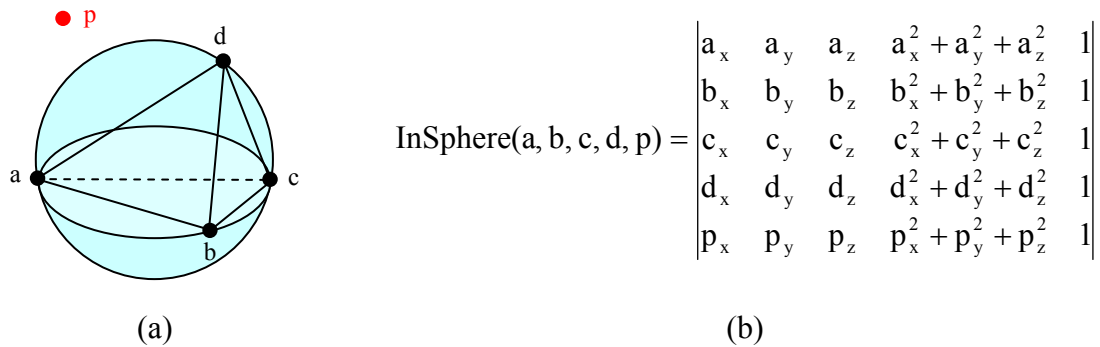


Figure 4.4 - Test géométrique *InSphere* (a) quatre sommets a, b, c et d d'un tétraèdre sur leur sphère circonscrite et point p à tester (b) calcul du déterminant correspondant.

Chacun des sommets de la triangulation Delaunay est en fait un point du jeu de données à partir duquel elle est bâtie. Tel que montré sur la figure 4.5, chaque tétraèdre de la triangulation Delaunay possède exactement quatre sommets et est adjacent à quatre tétraèdres, excepté si ses sommets se trouvent à la limite de la triangulation.

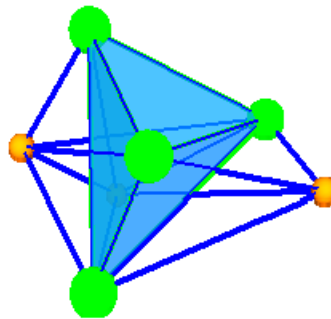


Figure 4.5 - Tétraèdre issu de la triangulation Delaunay 3D de sept points.

La triangulation Delaunay 3D de S peut aussi être considérée comme étant le 3-simplexe défini par les tétraèdres qui remplissent complètement l'espace sans se superposer. Un simplexe est l'élément le plus simple que l'on retrouve dans une dimension donnée. Plus spécifiquement, un simplexe de dimension n est appelé un n -simplexe et forme l'enveloppe

convexe d'un ensemble de $(n + 1)$ points linéairement indépendants dans un espace euclidien de dimension n (R^n) (Okabe *et al.*, 1999). Par exemple, tel que présenté à la figure 4.6, un 0-simplexe correspond à un nœud (un point du jeu de données), un 1-simplexe est un segment, un 2-simplexe est un triangle et enfin, un 3-simplexe est un tétraèdre. Il est à noter qu'un n -simplexe peut être construit par des simplexes de dimension inférieure. Par exemple, le triangle $[a, b, c]$, un 2-simplexe, est une face du tétraèdre $[a, b, c, d]$ qui est un 3-simplexe.

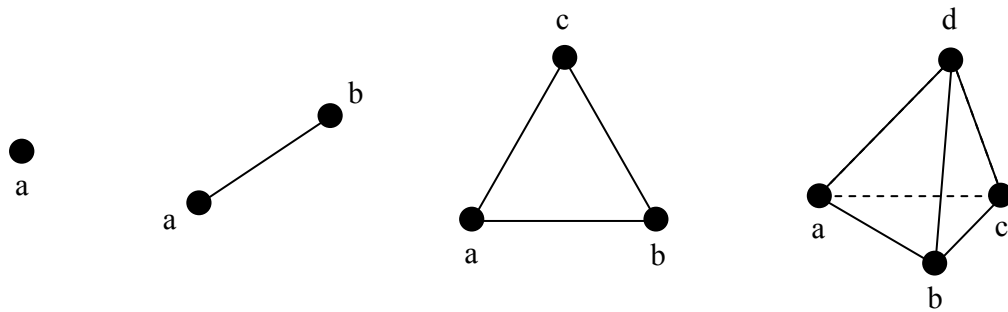


Figure 4.6 - Simplexes de la triangulation Delaunay 3D : nœud $[a]$, segment $[a,b]$, triangle $[a,b,c]$ et tétraèdre $[a,b,c,d]$.

4.2.2 Construction de la triangulation Delaunay 3D

Pour construire la triangulation Delaunay 3D, plusieurs méthodes telles que la méthode « *divide and conquer* », la méthode « *sweep line* » ou encore la méthode incrémentale peuvent être employées. Cependant, elles ne permettent pas toutes d'effectuer des modifications locales dans le modèle. La méthode incrémentale (Joe, 1991; Mostafavi *et al.*, 2003; Ledoux et Gold, 2006) est une méthode dynamique où des modifications locales comme l'insertion et la suppression de points du modèle sont possibles. La construction du modèle 3D par la méthode incrémentale consiste à incorporer les points un à un dans le modèle et à mettre à jour localement la triangulation après chaque insertion pour s'assurer que chacun des tétraèdres du maillage respecte le critère de Delaunay. La complexité de la méthode incrémentale dépend du nombre de tétraèdres construits, ce qui correspond à $O(n^2)$ dans le pire des cas (Ledoux, 2006; Hashemi *et al.*, 2007). Cependant, si aucune modification locale du modèle 3D n'est nécessaire, les autres méthodes de construction, la méthode « *divide and conquer* » par exemple, sont beaucoup plus efficaces avec une complexité de $O(n \log n)$ dans le pire des cas.

L'algorithme utilisé pour la construction de la triangulation Delaunay 3D par la méthode incrémentale comporte plusieurs étapes (voir figure 4.7).

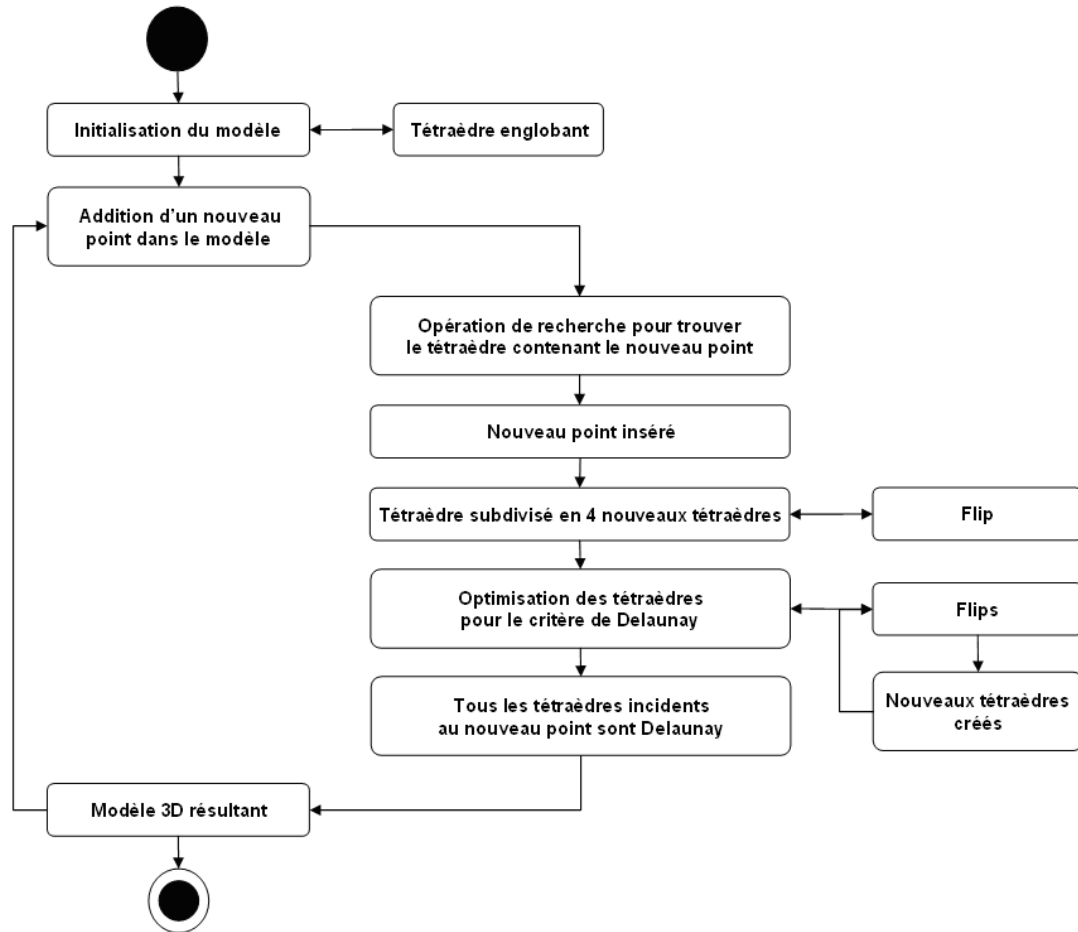


Figure 4.7 - Étapes de la construction de la triangulation Delaunay 3D par la méthode incrémentale.

La première étape consiste à initialiser le modèle en créant un tétraèdre englobant qui doit être assez grand pour contenir tous les points du jeu de données à traiter (tétraèdre universel). Puis, un premier point est introduit dans le modèle. Une recherche est alors effectuée en utilisant les relations d'adjacence entre les tétraèdres du modèle afin d'identifier lequel de ces tétraèdres contient le nouveau point ajouté. Ensuite, le point est inséré dans le tétraèdre trouvé, ce qui subdivise ce dernier en quatre nouveaux tétraèdres. Tous les nouveaux tétraèdres créés doivent alors être testés afin de s'assurer qu'ils respectent le critère de Delaunay. Si ce n'est pas le cas, les tétraèdres doivent être optimisés par des opérations topologiques appelées flips (Joe, 1991; Edelsbrunner et Shah, 1996;

Shewchuk, 2005). Un flip tridimensionnel est une opération topologique qui modifie de manière locale la configuration des tétraèdres adjacents dans une triangulation. Il existe plusieurs façons de trianguler un ensemble de cinq points. En effet, ces cinq points peuvent former soit deux, trois ou encore quatre tétraèdres différents. Ainsi, comme le montre la figure 4.8, divers flips tridimensionnels peuvent être définis. La séquence des flips est contrôlée par une structure « pile » qui contient tous les tétraèdres qui n'ont pas encore été testés. La pile commence avec les quatre nouveaux tétraèdres créés par l'insertion du point et chaque fois qu'un flip est réalisé, les nouveaux tétraèdres sont ajoutés à la pile. Lorsque la pile est vide, cela signifie que tous les tétraèdres formant la triangulation sont Delaunay. Cette procédure est répétée tant que de nouveaux points sont ajoutés à la triangulation.

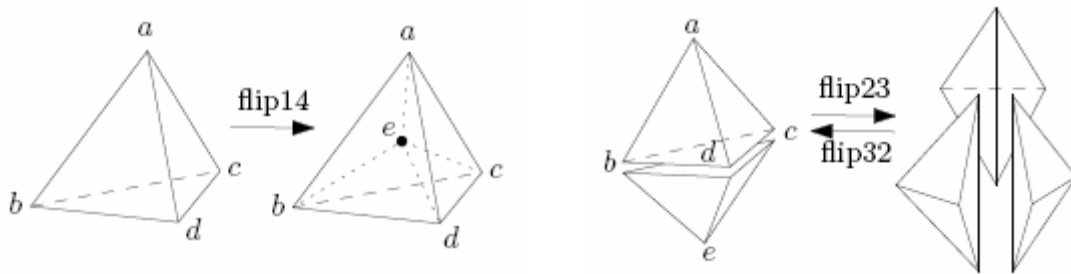


Figure 4.8 - Différents cas de flips tridimensionnels tirés de Ledoux (2006).

4.2.3 Structuration des données

La triangulation Delaunay 3D et les informations topologiques sur les relations entre les tétraèdres peuvent être stockées en utilisant différentes structures de données incluant la structure de données basée sur les tétraèdres et la structure de données « *augmented quad-edge* » (Boguslawski et Gold, 2007). Le présent travail est développé à l'aide du code source fourni par Ledoux (2006) pour la construction de la triangulation Delaunay 3D, lequel est basé sur une structure de données tétraédrique. Cette structure de données stocke chaque tétraèdre en utilisant quatre pointeurs pour les nœuds constituant ses sommets et quatre pointeurs vers les quatre tétraèdres qui lui sont adjacents.

La figure 4.9 montre par exemple que le tétraèdre 2 de la triangulation Delaunay 3D est stocké à l'aide de quatre pointeurs vers ses sommets (a, b, c, d) ainsi qu'à l'aide de quatre

pointeurs vers ses tétraèdres voisins (4, 11, 6, 9). Ces relations topologiques sont employées pour la détermination des agrégations de poissons en 3D à l'aide de l'algorithme des α -shapes qui est présenté dans la prochaine section.

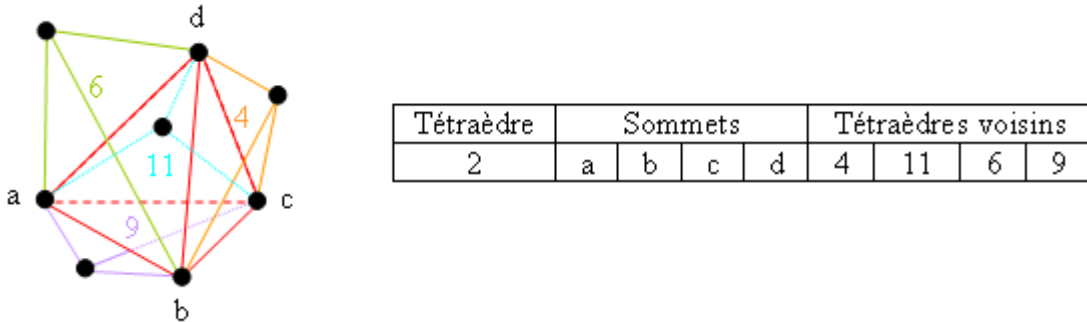


Figure 4.9 - Structure de données basée sur les tétraèdres.

Ainsi, la triangulation Delaunay est reconstruite à chaque fois qu'elle est demandée, cette opération étant très rapide (Ledoux, 2006). Cette structure de données très simple comporte l'avantage de solliciter peu de mémoire pour son stockage dans un ordinateur. Elle est efficace pour parcourir les sommets de la triangulation et elle peut également supporter l'insertion et la suppression de points. Par contre, une telle structure de donnée implique que lorsque certaines manipulations doivent être effectuées sur la triangulation Delaunay, plus d'opérations et de fonctions sont nécessaires en comparaison avec une structure où la triangulation serait complètement conservée (Guo *et al.*, 1997).

Il est à noter que pour naviguer dans la structure de données et pour la manipuler, les nœuds et les triangles (faces) de chacun des tétraèdres doivent être parcourus de manière constante. En fait, la structure de données est cohérente avec la règle de la main gauche. Cela signifie qu'un déterminant calculé à partir des trois sommets d'une face (a, b, c) ainsi qu'avec un autre point (p) retourne une valeur positive si ce point (p) se situe au-dessus du plan défini par les trois sommets de la face (a, b, c), une valeur négative si le point (p) se trouve sous le plan et une valeur égale à 0 si le point (p) est directement sur le plan. Le calcul de ce déterminant correspond au test géométrique *Orient* tel que défini à la figure 4.10. Ainsi, si les sommets (a, b et c) définissant une face d'un tétraèdre de la

triangulation Delaunay 3D sont observés dans le sens horaire à partir d'un certain point, ce point définit la face positive du triangle observé (Ledoux, 2006).

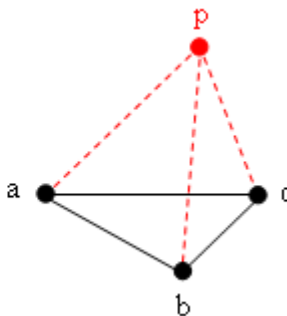
$$\text{Orient}(a, b, c, p) = \begin{vmatrix} a_x & a_y & a_z & 1 \\ b_x & b_y & b_z & 1 \\ c_x & c_y & c_z & 1 \\ p_x & p_y & p_z & 1 \end{vmatrix}$$


Figure 4.10 - Test géométrique *Orient*.

L'utilisation de la triangulation Delaunay 3D pour la représentation de surfaces continues fournit un modèle unique de l'espace qui peut être modifié localement. Cela permet notamment d'avoir des interactions locales avec l'environnement lorsque nécessaire. Un autre aspect important de la structure de données proposée est qu'il est possible d'étendre ses capacités afin de pouvoir considérer les mouvements et suivre les objets incorporés à l'intérieur du modèle (Mostafavi et Gold, 2004; Ledoux, 2006; Hashemi *et al.*, 2007). Cela implique que la structure proposée pourrait être utilisée dans des applications marines complexes, particulièrement dans le domaine de la biologie marine où les biologistes sont intéressés à étudier le comportement des différentes espèces de poissons ainsi que leurs interactions avec leur environnement ou leurs prédateurs par exemple. Le modèle développé à l'aide de cette structure pourrait également être appliqué pour étudier le mouvement et la trajectoire des poissons dans une période de temps donnée afin d'en arriver à mieux comprendre le cycle de vie d'un poisson et sa relation avec son habitat.

4.3 Clustering par l'algorithme des α -shapes en 3D

Après la construction de la triangulation Delaunay associée au jeu de données marines à disposition, le modèle 3D continu obtenu doit être filtré afin d'en extraire les agrégations de poissons. Pour ce faire, les primitives géométriques du modèle (poissons individuels) appartenant à chaque agrégation doivent être trouvées. Cette étape est réalisée par l'application d'un algorithme de *clustering*. Tel que mentionné précédemment dans ce chapitre, les *clusters* sont défini par des mesures de distance, regroupant ainsi les nuages de

points qui sont proches les uns des autres dans l'espace. Plusieurs techniques peuvent être employées pour effectuer le *clustering* d'un ensemble de points dans un espace 3D. L'approche proposée dans ce chapitre utilise l'algorithme des α -shapes 3D dont le choix a été justifié précédemment. Cette section définit et détaille le concept d' α -shape d'un jeu de données et décrit l'algorithme utilisé pour son calcul.

4.3.1 Description de l' α -shape 3D

Pour comprendre la notion de l' α -shape d'une distribution de points, plusieurs comparaisons intuitives ont été proposées. Parmi ces dernières, celle de Fischer (2005) suggère d'assimiler le concept des α -shapes à un espace 3D rempli d'une importante masse de crème glacée et contenant des morceaux de chocolat, un matériel plus solide, à la même position que les points d'un jeu de données. En utilisant une cuillère de forme sphérique possédant un rayon α , nous creusons et retirons toute la crème glacée que nous pouvons atteindre en ne prenant aucun morceau de chocolat. Posons ici qu'il est également possible de creuser dans la crème glacée à partir de l'intérieur. Nous pouvons alors constater que plus le rayon de la cuillère diminue, plus cette cuillère peut atteindre des endroits étroits et plus de crème glacée est enlevée. Pour une cuillère ayant un rayon infiniment petit, la crème glacée est entièrement ôtée et il ne reste que les morceaux de chocolat. Suite à cette opération, l'objet résultant est délimité par des arcs, des surfaces bombées et des points. De plus, il n'est pas garanti que ses différentes parties soient connectées entre elles. Ainsi, l' α -shape des points du jeu de données représentés par des morceaux de chocolat est obtenu en substituant les arcs en segments droits et en aplatissant les surfaces arrondies pour obtenir des triangles plats.

De la même manière, nous pouvons également concevoir que l' α -shape est trouvé lorsque nous faisons passer une balle de rayon α à travers le modèle continu 3D obtenu par la triangulation Delaunay. Partout où la sphère peut passer sans toucher les nœuds de la triangulation Delaunay 3D, les segments, triangles et tétraèdres sont effacés. Ainsi, nous creusons dans le modèle et nous y enlevons toutes les sphères de rayon α vides, c'est-à-dire ne contenant aucun nœud. L' α -complexe est la partie de la triangulation Delaunay qui reste

subséquemment à cette opération. L' α -*shape*, quant à elle, est la limite de l' α -complexe. Nous reviendrons sur ces deux termes dans la suite du présent chapitre.

4.3.2 Définition formelle de l' α -*shape* 3D

L' α -*shape* peut également être définie de manière formelle. Le concept d' α -*shape* a été introduit pour la première fois en 2D par Edelsbrunner *et al.* (1983), puis généralisé et défini en 3D par Edelsbrunner et Mücke (1994). La définition de l' α -*shape* se base sur l'hypothèse de la position générale telle que posée pour la création de la triangulation Delaunay 3D. Ainsi, comme il a été précisé précédemment, l'hypothèse de la position générale stipule qu'il n'y a pas, dans un jeu de données, quatre points qui se trouvent sur un même plan ou cinq points qui soient sur une même sphère. De plus, pour n'importe quelle valeur de α , la position générale implique que la plus petite sphère circonscrite à deux, trois ou quatre points du jeu de données doit avoir un rayon différent de α (Edelsbrunner et Mücke, 1994).

L' α -*shape* est un objet géométrique concret qui est défini de manière unique, selon la valeur d'un paramètre α , pour un ensemble de points particulier. Le paramètre α contrôle le niveau de détail de l' α -*shape*. En fait, les α -*shapes* constituent la famille des formes dérivées de la triangulation Delaunay paramétrées par α . Pour une certaine valeur de α , où $0 < \alpha < \infty$, l' α -*shape* est composée de tous les simplexes de la triangulation Delaunay possédant une sphère circonscrite de rayon inférieur à α qui ne contient aucun autre point du jeu de données. Ainsi, toutes les faces de l' α -*shape* sont des simplexes de la triangulation Delaunay et son intérieur est triangulé par les tétraèdres de cette même triangulation (Edelsbrunner et Mücke, 1994). Cependant, contrairement à la triangulation Delaunay, l' α -*shape* d'un ensemble de points n'est pas nécessairement convexe ni en un seul morceau.

Comme il a été mentionné précédemment, la formation d'une α -*shape* est déterminée par la proximité, plus précisément la distance euclidienne, entre les points d'un ensemble de points. Plus les points sont rapprochés, plus une petite valeur α est nécessaire pour relier ces points et plus l' α -*shape* concrétise le volume contenu à cet endroit rapidement.

Deux autres concepts peuvent être introduits afin de définir ce qu'est l' α -*shape* 3D d'un nuage de points. Il s'agit des concepts d' α -balle et de simplexe α -exposé. D'une part, une α -balle est définie par une sphère de rayon α , où $0 < \alpha < \infty$. Soit S un ensemble fini de points. L' α -balle est vide dans le cas où elle ne contient aucun point de S . L' α -*shape* de la distribution de points est alors formée par le complément de l'union de toutes les α -balles vides, en éliminant les surfaces et les lignes courbes. D'autre part, un simplexe est dit α -exposé s'il s'inscrit dans une α -balle vide, c'est-à-dire si le simplexe est circonscrit par une α -balle qui ne contient aucun autre point de S . La figure 4.11 donne l'exemple d'un segment α -exposé. Pour chacun des simplexes de la triangulation Delaunay 3D, il existe un $\alpha \geq 0$ tel que ce simplexe est α -exposé. Ainsi, la limite de l' α -*shape* de la distribution de points S est constituée de tous les simplexes qui sont α -exposés.

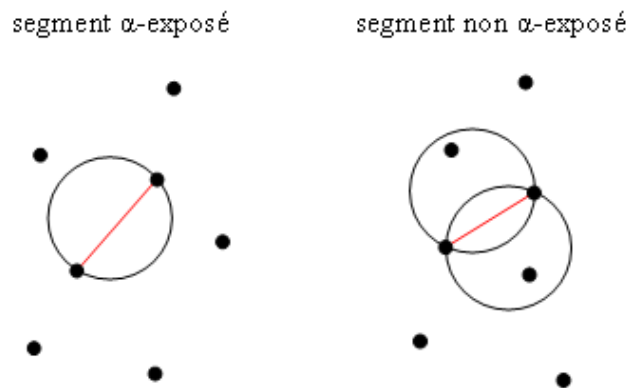


Figure 4.11 - Exemple d'un segment (1-simplexe) α -exposé et non α -exposé.

La figure 4.12 montre quelques éléments de la famille des α -*shapes* pour un ensemble de points donné. Remarquons sur la figure 4.12a que pour $\alpha = \infty$, l' α -*shape* est identique à l'enveloppe convexe de la distribution de points. En effet, une α -balle possédant un rayon α supérieur au rayon de la plus grande sphère circonscrite trouvée dans l'ensemble de points (possiblement la sphère contenant tous les points) ne permet pas aux simplexes intérieurs d'être α -exposés (Fischer, 2005). Cependant, plus la valeur de α diminue, plus l' α -*shape* rétrécit et forme graduellement des cavités, des tunnels et même des trous tel que présenté aux figures 4.12b et 4.12c. Ainsi, d'une réduction du paramètre α découle une α -*shape* plus raffinée et possiblement plus proche de l'objet réel. Lorsque $\alpha = 0$, nous pouvons observer sur la figure 4.12d que l' α -*shape* correspondante n'est composée que des points de la

distribution de données. Effectivement, une α -balle ayant un rayon inférieur au rayon de la plus petite sphère circonscrite trouvée dans l'ensemble de points, rayon possiblement infiniment petit, permet à tous les points d'être α -exposés sans le permettre aux simplexes de dimension supérieure (Fischer, 2005).

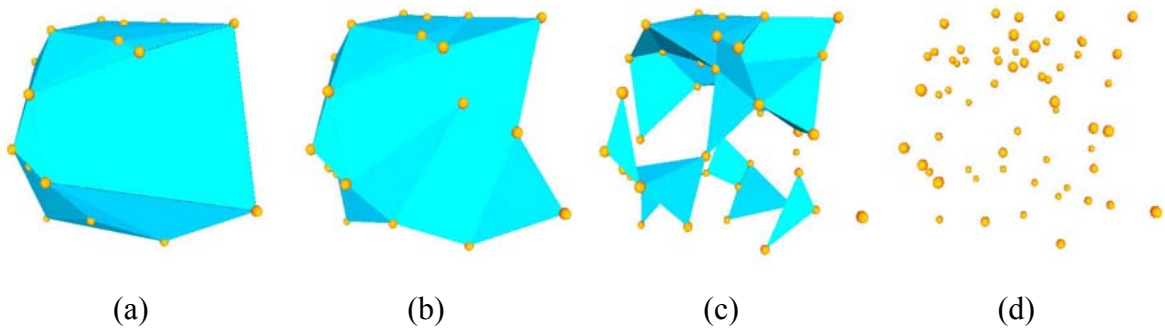


Figure 4.12 - Famille d' α -shapes 3D pour un certain jeu de données (a) enveloppe convexe de l'ensemble de points pour $\alpha = \infty$, (b) et (c) α -shape pour $0 < \alpha < \infty$, (d) ensemble de points pour $\alpha = 0$.

4.3.3 Calcul des α -shapes pour une distribution de points donnée

Afin d'implanter les α -shapes dans la triangulation Delaunay 3D présentée dans la section 4.2, l'algorithme des α -shapes 3D développé et décrit par Edelsbrunner et Mücke (1994) a été utilisé. L'algorithme proposé ne traite cependant pas directement l' α -shape d'un ensemble de points. En effet, il calcule d'abord une structure appelée α -complexe (Fischer, 2005). L' α -complexe, noté C_α , est un complexe simplicial de la triangulation Delaunay. Un complexe simplicial K est un ensemble fini de simplexes dans R^n tel que chacune des faces d'un simplexe de K est dans K et que l'intersection de deux simplexes de K soit une face commune (Munkres, 1993). Un complexe simplicial est donc une structure qui remplit l'espace et un simplexe le constituant ne peut pas transpercer l'intérieur d'autres simplexes (Ledoux, 2006). En 3D, dans le cas d'une triangulation, le complexe simplicial est construit en joignant entre eux des nœuds, des segments, des triangles et des tétraèdres. La figure 4.13 donne les exemples d'un simplexe simplicial en 3D ainsi que d'une structure qui n'est pas un complexe. Un simplexe de la triangulation Delaunay fait partie de l' α -complexe C_α dans deux cas; (1) si sa sphère circonscrite est vide et possède un rayon plus petit que α ou (2) si ce simplexe est une face d'un autre simplexe de dimension supérieure qui fait partie

de C_α . Ces deux conditions constituent le α -test. Il peut être démontré que la frontière de l' α -complexe forme l' α -shape d'un ensemble de points donné (Edelsbrunner et Mücke, 1994). En fait, l' α -shape est la partie de l'espace couvert par les simplexes de C_α .



Figure 4.13 - (a) Exemples de complexe simplicial en 3D et (b) de structure qui n'est pas un complexe (Ledoux, 2006).

Similairement aux observations faites sur la figure 4.12, il peut être remarqué sur la figure 4.14 qu'au fur et à mesure que la valeur du paramètre α augmente, l' α -complexe évolue et que des simplexes s'ajoutent dans sa constitution. En effet, l' α -complexe n'est d'abord formé que de points (pour $\alpha = 0$) comme sur la figure 4.14a (complexe simplicial 0D), pour ensuite être constitué de points et de segments uniquement comme sur la figure 4.14b (complexe simplicial 1D), de points, de segments et de triangles seulement comme sur la figure 4.14c (complexe simplicial 2D) puis, finalement, de points, de segments, de triangles et de tétraèdres soit l'ensemble des types de simplexes trouvés dans la triangulation Delaunay 3D comme sur la figure 4.14d (Lartigue *et al.*, 2000).

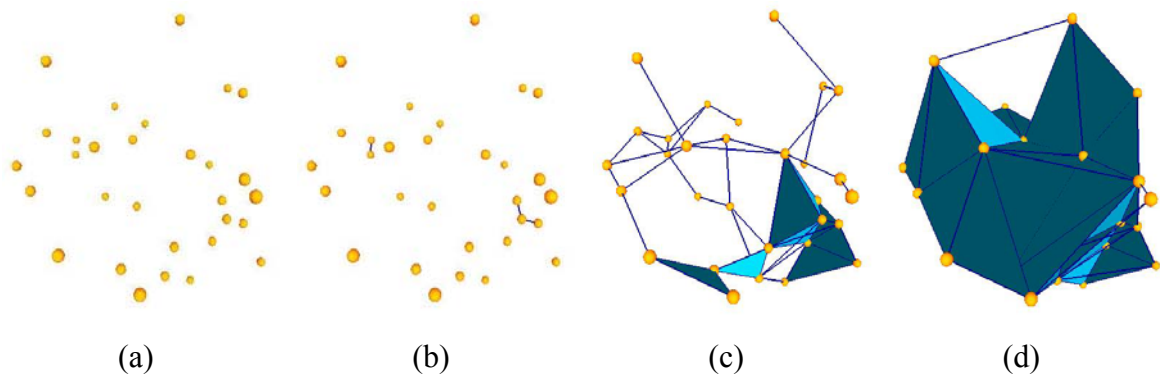


Figure 4.14 - Formation d'un α -complexe 3D par des simplexes de différentes dimensions.

Donc, pour définir l' α -shape d'une distribution de points, les simplexes de la triangulation Delaunay 3D (tétraèdre, triangle, segment, nœud) qui appartiennent à l' α -complexe C_α doivent être identifiés. Chacun des simplexes de la triangulation Delaunay 3D est relié à un intervalle $I = [a, \infty]$, où $0 \leq a \leq \infty$, qui identifie pour quelles valeurs de α un simplexe appartient à C_α (Edelsbrunner et Mücke, 1994; Fischer, 2005). Selon l'algorithme élaboré par Edelsbrunner et Mücke (1994), pour chaque simplexe, l'intervalle $I = [a, \infty]$ est séparé en trois parties. Ces divisions correspondent aux valeurs de α pour lesquelles le simplexe est intérieur, régulier ou singulier. Un simplexe est dit intérieur s'il n'appartient pas à la frontière de l' α -shape, régulier s'il appartient à la frontière de l' α -shape et qu'il fait partie d'un simplexe de dimension supérieure qui appartient à l' α -complexe et singulier s'il appartient à la frontière de l' α -shape mais qu'il ne fait pas partie d'un simplexe de dimension supérieure qui appartient à l' α -complexe. Dans l'exemple montré à la figure 4.15, un segment régulier est un segment qui appartient à la frontière de l' α -shape et qui délimite un triangle ou un tétraèdre et un segment singulier est un segment appartenant à la frontière de l' α -shape mais qui ne délimite pas un triangle ou un tétraèdre.

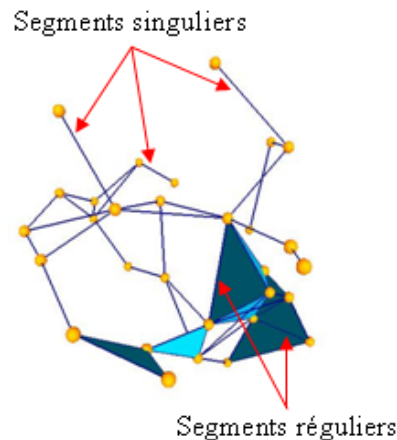


Figure 4.15 - Exemple de segments singuliers et de segments réguliers dans l' α -complexe.

Le maillage intérieur de l' α -shape est fait de tétraèdres issus de la triangulation Delaunay. Ainsi, un tétraèdre est toujours à l'intérieur de l' α -complexe. Pour un triangle ou un segment, un simplexe est appelé non attaché si sa sphère circonscrite est vide et attaché dans le cas contraire. Donc, un simplexe attaché ne peut pas être singulier. L' α -complexe

est formé par l'ensemble des simplexes intérieurs, réguliers et singuliers pour une valeur donnée de α .

Dans le tableau 4.1 traduit de Edelsbrunner et Mücke (1994), 'a' est le rayon de la plus petite sphère circonscrite au simplexe, 'b' est le minimum de tous les 'a' calculés pour les simplexes de dimension supérieure qui contiennent le simplexe et qui sont non attachés tandis que 'c' représente le maximum de tous les 'a' calculés pour les simplexes de dimension supérieure qui contiennent le simplexe.

Tableau 4.1 - Intervalles des valeurs α pour lesquelles un simplexe appartient à l' α -complexe (Edelsbrunner et Mücke, 1994).

		singulier	régulier	intérieur
tétraèdre				$[a, \infty]$
triangle ou segment	pas sur la frontière, non attaché	$[a, b]$	$[b, c]$	$[c, \infty]$
	pas sur la frontière, attaché		$[b, c]$	$[c, \infty]$
	sur la frontière, non attaché	$[a, b]$	$[b, \infty]$	
	sur la frontière, attaché		$[b, \infty]$	
nœud	pas sur la frontière	$[0, b]$	$[b, c]$	$[c, \infty]$
	sur la frontière	$[0, b]$	$[b, \infty]$	

L'implantation de l'algorithme des α -shapes 3D effectuée dans ce chapitre est toutefois quelque peu différente. En fait, elle constitue une version régularisée de l'algorithme de Edelsbrunner et Mücke (1994) où les simplexes singuliers sont éliminés (Da et Yvinec, 2006). De cette façon, quand la valeur de α est augmentée continuellement de 0 à ∞ pour un simplexe, l'intervalle $I = [a, \infty]$ est divisé en deux sous-intervalles $I_1 = [a, b]$ et $I_2 = [b, \infty]$ avec $0 \leq a \leq b \leq \infty$. Ces sous-intervalles sont définis de façon à ce qu'un simplexe de la triangulation Delaunay 3D ne soit pas dans l' α -complexe pour $\alpha < a$, qu'il soit sur la frontière de l' α -complexe pour $\alpha \in [a, b]$ et qu'il soit à l'intérieur de l' α -complexe pour $\alpha \in [b, \infty]$ (voir Fischer, 2005). I_1 est donc l'ensemble des valeurs de α pour lesquelles le simplexe appartient à la frontière de l' α -complexe tandis que I_2 représente l'ensemble des valeurs de α pour lesquelles le simplexe appartient à l'intérieur de l' α -complexe (voir implantation de l'algorithme à l'annexe B).

Dans la procédure suivie pour le calcul des deux intervalles pour chacun des simplexes de la triangulation Delaunay 3D, quatre listes différentes sont d'abord créées : une liste de tétraèdres, une liste de triangles, une liste de segments et une liste de nœuds. Ensuite, l'ensemble des tétraèdres contenus dans la liste correspondante sont traités. Le rayon de leur sphère circonscrite est calculé et correspond au paramètre 'a'. Comme pour un tétraèdre, 'a' = 'b', le calcul s'arrête pour ce type de simplexe. Les autres simplexes (triangles, segments et nœuds) sont alors analysés. D'une part, une vérification est faite afin de déterminer si chaque simplexe, triangle ou segment, est attaché ou non. Pour savoir si un simplexe est attaché, les tests géométriques présentés aux équations 4.1 à 4.3 sont employés.

$$M_{j_1, \dots, j_n}^{i_1, \dots, i_n} = \det \begin{pmatrix} \pi_{i_1, j_1} & \dots & \pi_{i_1, j_n} \\ \vdots & \ddots & \vdots \\ \pi_{i_n, j_1} & \dots & \pi_{i_n, j_n} \end{pmatrix} \quad (4.1)$$

où : π_{i_j} est la j - ème coordonnée du point p_i et $\pi_{i,0} = 1$ et $\pi_{i,d+1} = \sum_{j=1, \dots, d} \pi_{i_j}^2$

$$\text{segment} = \{p_i, p_j\} \text{ attaché} \Leftrightarrow \sum_{l=1}^3 (M_{l,0}^{i,j})^2 - \left(\sum_{l=1}^3 (M_{l,0}^{i,k} + M_{l,0}^{j,k})^2 \right) > 0 \quad (4.2)$$

$$\begin{aligned} \text{triangle} = \{p_i, p_j, p_k\} \text{ attaché} \Leftrightarrow \\ (M_{2,3,4,0}^{i,j,k,m})(M_{2,3,0}^{i,j,k}) + (M_{1,3,4,0}^{i,j,k,m})(M_{1,3,0}^{i,j,k}) + (M_{1,2,4,0}^{i,j,k,m})(M_{1,2,0}^{i,j,k}) - 2(M_{1,2,3,0}^{i,j,k,m})(M_{2,3,0}^{i,j,k}) > 0 \end{aligned} \quad (4.3)$$

Rappelons qu'un simplexe fait partie de l' α -complexe si sa sphère circonscrite est vide et possède un rayon plus petit que α ou si ce simplexe constitue l'une des faces d'un autre simplexe de l' α -complexe de dimension supérieure. Dans le cas où un simplexe, triangle ou segment, est attaché, le rayon de sa sphère circonscrite n'est pas vide. Nous devons alors nous assurer que la deuxième condition est respectée afin de décider s'il appartient à l' α -complexe. Pour ce faire, nous devons vérifier si les simplexes de dimension supérieure qui le contiennent font partie de l' α -complexe. Si leurs sphères circonscrites sont vides et que leur rayon est inférieur à α , la condition est remplie et le simplexe fait partie de l' α -complexe. Ainsi, les rayons des sphères circonscrites de tous les simplexes de

dimension supérieure que le simplexe testé compose doivent être calculés. Le simplexe fait partie de l' α -complexe si α est plus grand que le rayon 'a' de la plus petite sphère circonscrite à un simplexe de dimension supérieure qui le contient. Dans le cas où un simplexe (triangle ou segment) est non attaché, alors ce simplexe fait partie de l' α -complexe si α est plus grand que le rayon de sa sphère circonscrite 'a'. Les relations mathématiques permettant de calculer les sphères circonscrites aux simplexes pour les segments, les triangles et les tétraèdres sont présentées aux équations 4.4 à 4.6.

$$\text{segment} = \{p_i, p_j\}: r_{T_1}^2 = \frac{1}{4} \sum_{l=1}^3 (M_{l,0}^{i,j})^2 \quad (4.4)$$

$$\text{triangle} = \{p_i, p_j, p_k\}: r_{T_2}^2 = \frac{\left(\sum_{l=1}^3 (M_{l,0}^{i,j})^2 \right) \left(\sum_{l=1}^3 (M_{l,0}^{i,k})^2 \right) \left(\sum_{l=1}^3 (M_{l,0}^{k,i})^2 \right)}{4 \left((M_{2,3,0}^{i,j,k})^2 + (M_{1,3,0}^{i,j,k})^2 + (M_{1,2,0}^{i,j,k})^2 \right)} \quad (4.5)$$

$$\begin{aligned} \text{tétraèdre} &= \{p_i, p_j, p_k, p_m\}: \\ r_{T_3}^2 &= \frac{(M_{2,3,4,0}^{i,j,k,m})^2 + (M_{1,3,4,0}^{i,j,k,m})^2 + (M_{1,2,4,0}^{i,j,k,m})^2 + 4(M_{1,2,3,0}^{i,j,k,m})(M_{1,2,3,4}^{i,j,k,m})}{4(M_{1,2,3,0}^{i,j,k,m})^2} \end{aligned} \quad (4.6)$$

D'autre part, un test est réalisé pour vérifier si chacun des simplexes de la triangulation Delaunay 3D se trouve sur l'enveloppe convexe de l'ensemble de points ou non. En effet, un simplexe fait partie de la frontière de l' α -complexe s'il appartient à l'enveloppe convexe de la triangulation Delaunay 3D. Par contre, un simplexe est à l'intérieur de l' α -complexe si tous les tétraèdres qui le contiennent font partie de l' α -complexe. Un tétraèdre fait toujours partie de l'intérieur de l' α -complexe.

Les simplexes qui appartiennent à l'enveloppe convexe de la triangulation Delaunay 3D peuvent être facilement identifiés à partir du résultat de cette triangulation. Pour les simplexes qui sont positionnés sur l'enveloppe convexe, le paramètre 'b' correspond à ∞ . En effet, une fois la valeur 'a' atteinte par α , le simplexe fera toujours partie de l' α -complexe et comme il se trouve sur l'enveloppe convexe, il sera toujours à la frontière de

l' α -complexe. Nous n'avons donc pas à trouver d'intervalle pour lequel le simplexe est intérieur.

Autrement, si un simplexe ne fait pas partie de l'enveloppe convexe de la triangulation Delaunay, il faut vérifier si ce simplexe est intérieur à l' α -complexe. Pour ce faire, nous devons obtenir le rayon des sphères circonscrites à tous les tétraèdres qui contiennent le simplexe et les comparer entre eux. Le simplexe fait partie de l'intérieur de l' α -complexe si α est plus grand que le rayon de la plus grande sphère circonscrite aux tétraèdres qui le contiennent. Le paramètre 'b' équivaut alors à ce rayon.

Le résultat de l'implantation de l'algorithme des α -shape 3D mène à la création de *clusters* tridimensionnels où la valeur de α doit être examinée afin d'obtenir une représentation exacte des formes en 3D. En effet, l'algorithme trouve une représentation implicite qui peut être employée pour obtenir l' α -shape d'une distribution de points pour n'importe quelle valeur de α (Fischer, 2005). Pour des besoins de visualisation, seuls les simplexes de la triangulation Delaunay 3D qui appartiennent à l' α -shape pour une valeur de α déterminée par l'utilisateur sont dessinés dans le modèle. La section suivante décrit l'application des techniques expliquées ci-dessus pour la représentation des agrégations de poissons.

4.4 Application du modèle à la modélisation 3D d'agrégations de poissons en milieu marin

Afin de valider l'approche développée pour la modélisation et la visualisation des agrégations de poissons en 3D, quelques expérimentations ont été menées à l'aide d'ensembles de points représentant plusieurs *clusters* dans un espace tridimensionnel. Premièrement, la triangulation Delaunay 3D couplée à l'algorithme des α -shapes 3D a été appliquée à un jeu de données test. Puis, un ensemble de données acoustiques représentant des observations de poissons individuels a été employé afin d'analyser l'efficacité de la méthode dans le cas d'un jeu de données issu du monde réel. À partir des résultats obtenus de ces expérimentations, les principaux bénéfices et limites de la méthode de représentation proposée ont été identifiés. Les résultats sont présentés et discutés dans cette section.

4.4.1 Détection automatique des agrégations de poissons en 3D

L'une des applications intéressantes de l'approche proposée est la détection automatique de formes contenues dans un ensemble de points 3D. En biologie marine, d'énormes fichiers de données acoustiques peuvent être employés pour trouver les positions individuelles de différents poissons à un instant donné. L'ensemble de données résultant est ainsi considéré comme un instantané de la position des poissons à un moment donné. La triangulation Delaunay 3D combinée à l'algorithme des α -shapes 3D permet de déterminer si un poisson donné appartient à une agrégation ou à une autre. Cette décision est basée sur plusieurs facteurs comprenant la position individuelle dudit poisson ainsi que sa distance par rapport aux autres individus dans l'espace 3D. Il doit être mentionné ici qu'afin de trouver des agrégations de poissons dans un ensemble de données en utilisant l'algorithme des α -shapes 3D, cet ensemble de données doit avoir une résolution relativement haute.

Les expérimentations initiales, faites dans le but de tester l'applicabilité de la méthode proposée pour la détection d'agrégations de poissons, ont été réalisées en utilisant un jeu de données tridimensionnel artificiel. Ce jeu de données inclut deux agrégations de poissons distinctes, le fond de l'océan et un ensemble de poissons individuels (voir figure 4.16). Les poissons sont représentés par des points 3D dont les positions en coordonnées géographiques comprenant la latitude, la longitude et la profondeur ont été transformées en coordonnées métriques x , y et z . La figure 4.16 illustre les résultats obtenus pour l'application de la méthode proposée pour l'extraction et la modélisation 3D des agrégations de poissons contenues dans le jeu de données. La figure 4.16a montre le jeu de données qui contient trois ensembles de points distincts dans l'espace 3D. La figure 4.16b présente le résultat obtenu pour la triangulation Delaunay 3D du jeu de données tandis que la figure 4.16c illustre le raffinement du modèle avec une valeur de α donnée. La figure 4.16d montre quant à elle le résultat trouvé suite à l'application de l'algorithme des α -shapes 3D. Il peut être constaté à partir de ces résultats que l'approche proposée permet de détecter clairement trois objets pouvant être distingués dans le jeu de données initial.

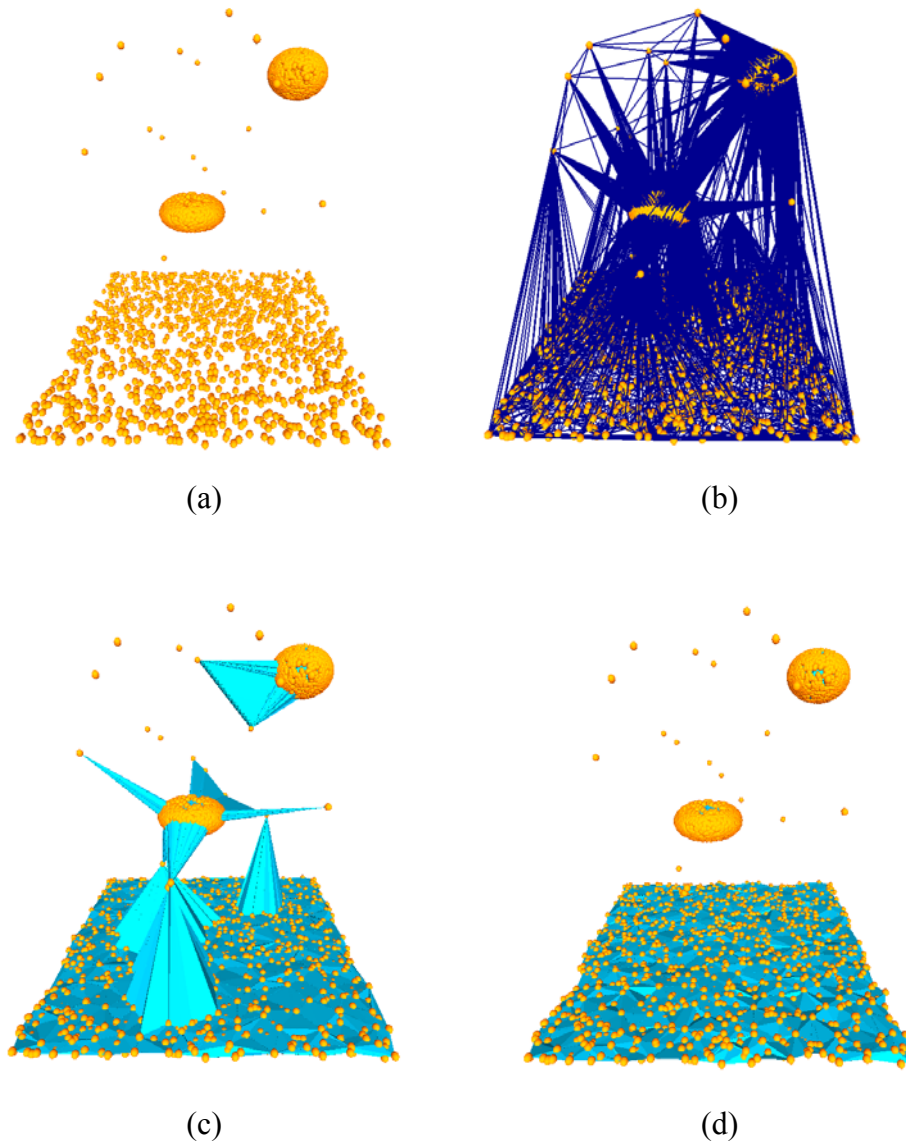


Figure 4.16 - Étapes pour extraire les agrégations de poissons dans un jeu de données. (a) Jeu de données, (b) triangulation Delaunay 3D, (c) raffinement du modèle avec un α quelconque, (d) résultat final.

4.4.2 Visualisation et analyse des agrégations de poissons en 3D

La connaissance des propriétés des agrégations de poissons constitue de l'information utile, voire nécessaire, pour la gestion et l'analyse des ressources halieutiques. La manipulation interactive et l'analyse de modèles représentant fidèlement ces agrégations peuvent être bénéfiques pour une meilleure compréhension des processus biologiques impliquant les poissons. Actuellement, seuls quelques logiciels commerciaux, tels que Fledermaus (IVS) et EchoView (SonarData Pty Ltd), peuvent fournir des fonctionnalités limitées pour

visualiser, effectuer des analyses spatiales et manipuler des données sur la distribution de poissons en 3D. Comme il peut être observé à la figure 4.16, le modèle 3D obtenu par l'application de l'approche proposée dans cette recherche offre de nouvelles possibilités pour visualiser des données acoustiques dans un espace 3D et fournit à l'utilisateur la possibilité d'effectuer des manipulations locales et des raffinements. De plus, la visualisation réaliste des agrégations de poissons en 3D offerte par le modèle développé à partir de l'approche proposée peut assister les gestionnaires des pêches afin que ces derniers aient une meilleure compréhension des phénomènes se déroulant dans l'environnement marin. En effet, les représentations 3D permettent une visualisation plus naturelle et intuitive des données (Mayer *et al.*, 2002). Ainsi, les modèles 3D réalisés peuvent contribuer à améliorer la gestion des ressources halieutiques en fournissant les outils nécessaires pour l'analyse des propriétés morphologiques des agrégations de poissons telles que leur position, leur étendue spatiale, leur forme et leur volume qui est un paramètre important pour les calculs de biomasse.

En plus des expérimentations faites avec le jeu de données test, l'approche proposée a été appliquée à des données acoustiques. Pour ce faire, des jeux de données décrivant les dynamiques d'agrégation et de désagrégation d'un groupe de morues de l'Atlantique (*Gadus morhua*) en période de frai à Smith Sound, sur la côte Est de Terre-Neuve, ont été utilisés. Un extrait de ces jeux de données est présenté à l'annexe C du présent mémoire. Ces données ont été acquises à haute résolution à l'aide d'un échosondeur à faisceau divisé embarqué sur un bateau sillonnant les eaux aléatoirement au-dessus de l'agrégation observée. L'échosondeur à faisceau divisé est un instrument qui envoie des impulsions sonores dans l'eau et qui mesure les échos reçus. Selon Simmonds et MacLennan (2005, p. 82), ce qui le distingue est que cet instrument possède un transducteur partagé en quatre quadrants. Les impulsions sonores sont émises de façon simultanée sur tous les quadrants, mais les signaux reçus (échos) sont analysés distinctement. Les différences de phase observées entre les quadrants étant proportionnelles aux écarts de distance parcourue par les échos, la position d'une cible dans le faisceau acoustique, un poisson par exemple, peut être déduite. Pour ce faire, les données acquises par l'échosondeur à faisceau divisé sont traitées dans des logiciels spécialisés et visualisées sur des échogrammes. La figure 4.17 présente

l'échogramme correspondant à des données acoustiques montrant, à un instant donné, le fond marin, des cibles uniques (poissons individuels) et deux agrégations de morues de l'Atlantique au-dessus du fond marin, dans la colonne d'eau. Comme il peut également être constaté sur la figure 4.17, les patrons de distribution des agrégations de poissons peuvent être détectés visuellement sur les échogrammes.

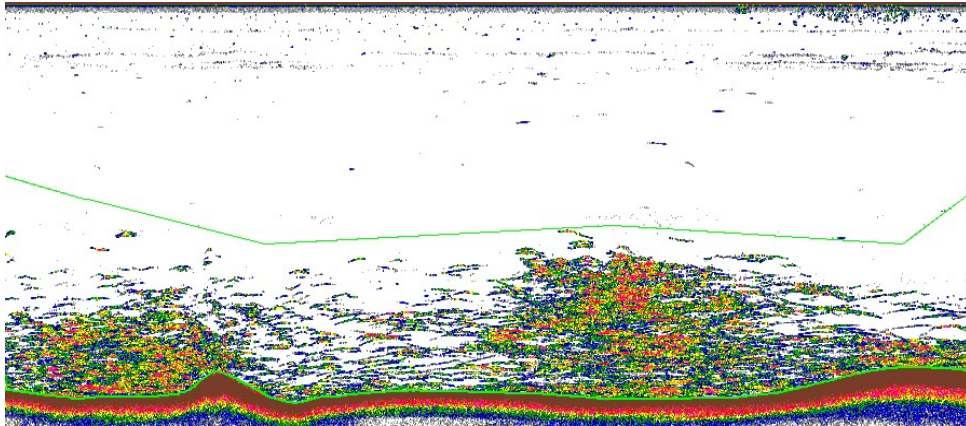


Figure 4.17 - Échogramme de données acoustiques montrant des poissons en milieu marin avec le temps sur l'axe x et la distance en mètres sur l'axe y.

Dans ce travail de recherche, seules les coordonnées spatiales des échos ont été prises en considération bien que plusieurs autres attributs comme la vitesse de la cible ou encore la force du signal reçu soient disponibles et puissent être stockés. Les données échantillonnées forment un ensemble de points irrégulièrement distribués localisés par une latitude, une longitude et une profondeur mesurées en mètres. Dans le jeu de données, les points sont denses dans les régions où il y a des objets comme des agrégations de poissons ou le fond marin et clairsemés partout ailleurs. La figure 4.18a illustre les points, dispersés dans la colonne d'eau, correspondant à une agrégation de poisson simple issue des jeux de données de pêche acoustiques décrits précédemment. Les figures 4.18b et 4.18c montrent respectivement la modélisation 3D de l'agrégation de poissons obtenue ainsi que sa structure interne.

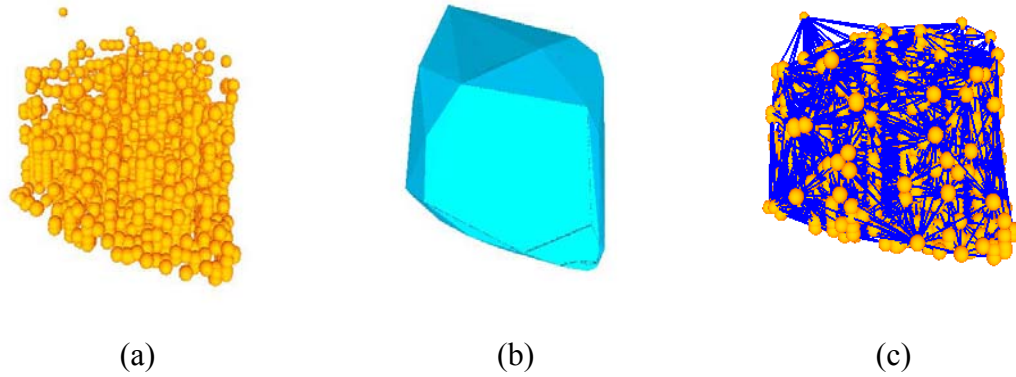


Figure 4.18 - Modèle 3D obtenu pour un jeu de données acoustiques réel.

Le modèle 3D développé et sa structure de données permettent de réaliser des analyses spatiales de base sur les données acoustiques. Comme les valeurs d'attributs peuvent être stockées avec les coordonnées spatiales des données acoustiques, celles-ci peuvent être facilement recherchées en effectuant des requêtes spatiales. Par exemple, la force du signal enregistré pour chacun des points représentant un poisson dans le modèle 3D peut être trouvée et employée afin d'évaluer la biomasse. De plus, le volume d'une agrégation de poissons peut être aisément dérivé du modèle 3D puisque celui-ci est basé sur des tétraèdres. En effet, une fonction simple peut être développée pour additionner le volume de tous les tétraèdres formant une agrégation de poissons. Les relations spatiales entre les différents éléments du modèle 3D peuvent également être exploitées pour en tirer de l'information intéressante telle que le placement des individus (poissons) dans l'espace ou dans une agrégation quelconque, puisqu'elles sont stockées.

4.4.3 Discussion

4.4.3.1 Étude du comportement et des dynamiques des poissons en 3D

En plus de constituer un outil pour la caractérisation des propriétés morphologiques des agrégations de poissons, le modèle 3D proposé peut être appliqué pour la représentation des dynamiques de ces agrégations ainsi que pour la visualisation en 3D des poissons pris individuellement. Tel que mentionné précédemment, l'approche proposée offre la possibilité de déplacer un ou plusieurs points représentant des observations simultanément dans l'espace 3D tout en préservant la validité du modèle géométrique. Cette propriété

unique du modèle proposé peut alors être appliquée pour le suivi individuel des poissons dans une agrégation et pour l'étude de l'évolution de cette agrégation si les données de surveillance des agrégations de poissons sont disponibles en ligne par exemple.

Une agrégation est un processus dans lequel des poissons entrent, bougent et sortent constamment. Par conséquent, étant donné ses propriétés, le modèle 3D permet de décrire les dynamiques spatiales et temporelles d'une agrégation de poissons en visualisant la structure réelle de cette agrégation ainsi que la position des individus à l'intérieur de celle-ci et leurs changements dans le temps. Les changements dans la morphologie de l'agrégation peuvent également être quantifiés à une échelle hebdomadaire ou mensuelle par exemple. Ainsi, les différences observées dans les patrons de distribution des poissons au fil des saisons peuvent être montrées et étudiées en utilisant des techniques de visualisation dynamique pour créer des scénarios. Les changements dans une agrégation de poissons peuvent, par exemple, être montrés par une série d'instantanés où chacun de ces instantanés est une représentation de l'agrégation étudiée à un temps donné.

D'autres analyses intéressantes pourraient être tirées du modèle proposé dans cette recherche dans le but de mieux comprendre le comportement des poissons. Par exemple, des informations relatives aux paramètres environnementaux, telles que la température de l'eau ou la salinité dans la région dans laquelle les agrégations étudiées se trouvent, pourraient éventuellement être ajoutées dans le modèle afin de pouvoir observer comment les modifications spatiales et temporelles des agrégations de poissons sont liées aux changements de l'environnement dans lequel ces poissons vivent. Le modèle pourrait de surcroît être employé afin de permettre de mieux mesurer les aspects physiques tridimensionnels des agrégations de poissons ou la position de nage des individus qui affecte les mesures quantitatives de densité et d'abondance issues des techniques acoustiques.

4.4.3.2 Limites de l'approche

D'une part, du point de vue des analyses et des différents types de visualisation qu'elle offre, la méthode proposée est très innovatrice pour la représentation des agrégations de

poissons et de leurs dynamiques dans l'espace 3D. En effet, à notre connaissance, la représentation et la visualisation de données halieutiques 3D acquises en 3D n'ont pas encore été abondamment explorées. Cette approche ouvre aussi la porte à des techniques d'analyses plus poussées. Bien que la modélisation tridimensionnelle des données réalisée à l'aide de la triangulation Delaunay 3D et de l'algorithme des α -shapes 3D offre de nombreux bénéfices, elle comporte également certaines limitations qui nécessiteront des investigations plus poussées dans des travaux futurs. Quelques-unes de ces limites sont dues à la complexité des données halieutiques en 3D.

Premièrement, il est difficile, voire impossible, de détecter les poissons individuellement dans des agrégations très denses car la différenciation des échos reçus est difficile. En effet, lorsque la densité des observations augmente au-dessus d'un certain niveau par rapport aux propriétés du faisceau acoustique, seuls quelques poissons individuels peuvent être distingués. Pourtant, les agrégations de poissons les plus denses, pour lesquelles peu d'observations sont disponibles, sont biologiquement les plus intéressantes à étudier. Il existe cependant une solution à ce problème qui consiste à extraire un nuage dense de points aléatoires dans l'agrégation apparaissant sur l'échogramme afin de créer un objet 3D et, par la suite, à ajouter les données sur les poissons individuels qui ont pu être distingués.

Un autre problème beaucoup plus important est que les poissons sont des êtres vivants et, en ce sens, bougent constamment. Ils peuvent donc être mesurés plusieurs fois dans le même jeu de données ce qui affecte grandement l'exactitude des mesures aussi bien que la qualité de la modélisation 3D des agrégations de poissons qui en découle. Des recherches sont présentement en cours dans le domaine de la biologie marine dans le but d'explorer différentes méthodes pour améliorer l'analyse des données acoustiques et l'observation des poissons en temps réel.

Des erreurs peuvent également être introduites lorsque les points 3D représentant les poissons sont dérivés des données acoustiques 2D acquises par l'échosondeur à faisceau divisé et montrées sur un échogramme. La position GPS du bateau est utilisée pour estimer la position horizontale d'un poisson alors que l'information sur sa position verticale dans la

colonne d'eau est trouvée sur l'échogramme. Toutes les cibles (poissons) sont supposées se trouver dans l'axe du faisceau de l'échosondeur à faisceau divisé lors de l'observation bien que cela ne soit pratiquement impossible dans la réalité. L'utilisation de données acquises à l'aide d'échosondeurs multifaisceaux et possédant une précision accrue en 3D pourrait permettre d'améliorer la qualité du modèle et de réduire le nombre de problèmes liés aux données.

Par ailleurs, certaines autres limites sont liées à l'utilisation de l'algorithme des α -shapes en 3D pour le raffinement du modèle. Par exemple, il peut parfois être difficile de déterminer une valeur optimale pour le paramètre α lors de la modélisation d'agrégations de poissons. En effet, le paramètre α doit présentement être déterminé par une méthode d'essais et erreurs puisqu'il n'existe aucune règle explicite pour sa détermination pour une certaine configuration de points en 3D. Par contre, une analyse préalable des données et la visualisation du modèle 3D peuvent aider à améliorer l'efficacité de l'assignation de la valeur du paramètre α . Aussi, de nouvelles expérimentations pourraient sans doute permettre de trouver un α optimal de manière automatique.

De plus, tel que décrit précédemment, l'algorithme des α -shapes se base principalement sur les distances entre les points du jeu de données pour déterminer si un point doit être connecté à un objet 3D donné ou non. Dans certains cas, il peut être difficile de trouver une valeur satisfaisante pour le paramètre α pour discriminer les différentes agrégations de poissons si celles-ci ont des densités très distinctes.

Aussi, la position générale telle que supposée dans l'élaboration de la méthode de représentation proposée en 3D ne peut pas toujours être assumée dans des jeux de données issus du monde réel. Alors, quelques modifications infinitésimales doivent être effectuées sur la plupart des jeux de données pour s'assurer que l'algorithme des α -shapes en 3D peut être exécuté normalement sans trouver d'ambiguïtés. Ces modifications sont décrites dans la méthode SoS développée par Edelsbrunner et Mücke (1994).

Finalement, une amélioration de l'efficacité de la structure et des capacités de visualisation du modèle développé est nécessaire afin de considérer complètement la nature dynamique des données halieutiques.

4.5 Conclusion du chapitre

La représentation de l'environnement marin, plus spécifiquement des données halieutiques et de leurs dynamiques, devient très problématique en utilisant les technologies SIG actuelles. L'objectif principal de ce chapitre était d'étendre quelques unes des capacités des SIG en créant des représentations 3D dynamiques et interactives des agrégations de poissons afin de fournir aux biologistes un modèle plus performant permettant l'analyse et la gestion des stocks de poissons. L'approche proposée dans ce chapitre est très innovatrice du fait qu'elle combine la triangulation Delaunay 3D et l'algorithme de *clustering* des α -*shapes* 3D pour modéliser et extraire les agrégations de poissons en 3D, ce qui fournit de plus grandes possibilités d'analyse des données que les représentations traditionnelles 2D (horizontales ou verticales) des agrégations de poissons. L'utilisation de la triangulation Delaunay 3D reposant sur une structure de données dynamique fournit un modèle de l'espace unique qui peut être modifié interactivement sur une base locale. L'utilisation de l'algorithme des α -*shapes* permet quant à lui d'extraire les objets souhaités en conservant et en tirant avantage de cette structure de données. Un autre aspect important du modèle suggéré est qu'il possède le potentiel d'être étendu pour le suivi et la surveillance de la dynamique et du comportement des poissons dans un espace 3D et possiblement dans le temps. De plus, il permet la détection automatique et la visualisation en 3D des agrégations de poissons. L'approche proposée présente donc des améliorations intéressantes en comparaison aux outils existants. Cependant, quelques limites, principalement reliées aux technologies acoustiques utilisées pour acquérir les données de pêche, peuvent affecter la qualité des agrégations de poissons montrées dans le modèle produit. D'autres limites, liées à l'algorithme des α -*shapes* 3D et à la détermination du paramètre α ont été observées pendant les expérimentations réalisées et devraient être corrigées dans des travaux futurs.

De surcroît, de nouvelles expérimentations avec des données acoustiques pourraient être menées afin d'améliorer l'approche proposée et d'explorer le potentiel du modèle pour

l'analyse des données marines ainsi que leur dynamique. Par exemple, il pourrait être intéressant de tester l'approche avec des données acoustiques acquises par un sonar multifaisceaux, une technologie qui est actuellement employée dans le domaine des pêches. De plus, à l'heure actuelle, le modèle 3D ne peut gérer qu'un seul jeu de données à la fois (ex. jeu de données acoustiques). Afin d'analyser l'impact des données environnementales (conductivité, température et profondeur de l'eau) sur les agrégations de poissons, leur dynamique et leur comportement, il doit être possible de gérer plusieurs jeux de données simultanément. Des outils complémentaires de visualisation et d'analyse 3D pourraient aussi être développés pour aider les utilisateurs du modèle à mieux percevoir, comprendre et analyser les données marines et leur environnement.

Chapitre 5 Conclusion et perspectives

5.1 Conclusion

Dans la deuxième moitié du 20^e siècle, le déclin de plusieurs espèces de poissons, dont la morue, a été observé dans la région de l'Atlantique du nord-ouest. Cet affaiblissement est en grande partie attribuable à la surpêche (De Young et Rose, 1993; Rose *et al.*, 1994). L'impact de cette surpêche a été amplifié par l'exploitation très intense de la ressource faite par les chalutiers dans les endroits où les poissons étaient agrégés (Rose, 2007). Les changements environnementaux ont probablement aussi contribué à la surpêche en causant des modifications dans la distribution spatiale des espèces (Rose et Kulka, 1999). En effet, certaines agrégations de poissons semblent se déplacer lorsque des perturbations surviennent dans leur environnement (changements de température et de salinité de l'eau, déplacement de leurs proies, etc.). Un déplacement des agrégations en dehors de la zone de 200 milles nautiques gérée par le Canada a donc entre autres permis aux pêcheurs étrangers d'exploiter la ressource sans restriction dans les eaux internationales.

Une représentation plus réaliste des agrégations de poissons et une compréhension de leurs dynamiques sont donc importantes pour développer des stratégies efficaces d'aide au rétablissement des espèces affaiblies. La géomatique et en particulier les SIG peuvent aider à identifier, représenter et analyser les agrégations de poissons. Toutefois, quelques problèmes liés à la représentation actuelle des agrégations dans les SIG peuvent être cités. En effet, les méthodes existantes de cartographie et de représentation des agrégations de poissons dans les SIG (ex. cartes interpolées, symboles proportionnels, ellipses) sont généralement limitées et ne permettent pas, par exemple, de visualiser directement l'étendue spatiale des agrégations de poissons. De plus, malgré les capacités intéressantes présentées par les SIG actuels, ces outils sont limités en raison de la nature tridimensionnelle, dynamique et floue des agrégations de poissons.

Ce mémoire a permis de développer de nouvelles approches permettant d'améliorer la représentation des agrégations de poissons en 2D ainsi qu'en 3D, ce qui constituait l'objectif principal de cette recherche. Ces approches améliorent les modes de représentation actuels des agrégations de poissons et entraînent une meilleure perception et une compréhension spatiale et temporelle de ce phénomène.

À une échelle régionale, les agrégations sont perçues comme des zones de haute densité en poissons. La représentation de ce phénomène en 2D sous la forme d'objets dans un SIG nécessite alors de considérer que les limites de ces objets ne sont pas nettes, mais plutôt graduelles ou floues dans l'espace. Par contre, à une échelle locale, chacun des poissons composant une agrégation peut parfois être détecté et représenté individuellement étant donné la disponibilité de ces données 3D. Il peut alors être intéressant de représenter les agrégations de poissons sous la forme d'objets 3D correspondant à une vision plus réaliste du phénomène.

Le développement d'approches pour l'identification et la représentation des agrégations de poissons à deux échelles d'observation différentes a nécessité la réalisation de plusieurs étapes. Dans un premier temps, au chapitre 2, une revue des concepts de représentation de l'espace géographique et des phénomènes spatiaux dans les SIG a été réalisée. Cette revue de littérature a permis de constater que, bien que les SIG actuels constituent des outils très performants pour la manipulation et la visualisation de données spatiales terrestres 2D, l'emploi de ces systèmes pour la représentation des données marines ou halieutiques comporte certains problèmes. En effet, les structures de données généralement disponibles dans les SIG ne permettent pas de gérer des données tridimensionnelles, dynamiques ou liées à des phénomènes spatiaux aux frontières floues. Ainsi, puisque les données utilisées pour la représentation des agrégations de poissons possèdent toujours l'une ou l'autre de ces caractéristiques selon l'échelle étudiée, les limites des SIG ont dû être prises en compte et gérées dans les solutions suggérées.

Dans le chapitre 3, une approche se basant sur la théorie des ensembles flous et les modèles d'objets flous qui en découlent a été proposée pour la délimitation et la représentation des

agrégations de poissons en 2D, à une échelle régionale. Cette approche a permis de répondre au premier objectif spécifique de la recherche qui était de **représenter des agrégations de poissons en 2D en se basant sur une approche reposant sur les modèles d'objets spatiaux flous, à une échelle régionale**. L'approche proposée représente les agrégations de poissons à la fois avec une structure de données vectorielle et une structure de données matricielle. Les limites floues des agrégations de poissons perçues à l'échelle régionale sont alors données par la différence entre les étendues spatiales maximales et minimales des agrégations. De plus, la transition graduelle observée à l'intérieur de ces limites est montrée dans la représentation.

Dans le chapitre 4, l'identification et la représentation des agrégations de poissons en 3D, à une échelle locale, ont été explorées. Une approche reposant sur une structure de données dynamique 3D a alors été proposée. Cette approche a permis de répondre au deuxième objectif spécifique de cette recherche qui était de **représenter des agrégations de poissons en 3D selon une approche basée sur une structure de données dynamique, à une échelle locale**. L'approche proposée utilise la triangulation Delaunay pour créer une structure de base 3D pour les données de pêche et l'algorithme de *clustering* 3D des *α -shapes* est ensuite employé pour l'extraction des agrégations de poissons. Le modèle résultant peut alors détecter les agrégations de poissons et les reconstruire à partir de données acoustiques par exemple. L'approche proposée fournit donc un modèle 3D, dynamique et interactif des agrégations de poissons qui peut offrir de nouvelles possibilités d'analyse.

Suite au développement de ces approches, leur applicabilité pour la représentation des agrégations de poissons a été testée et validée à l'aide de jeux de données simulés et de jeux de données provenant de sessions d'observation, ce qui a permis de répondre au troisième objectif spécifique de cette recherche qui était d'**appliquer et valider les approches proposées pour la représentation des agrégations de poissons en utilisant des données halieutiques**. En 2D, les représentations résultant de l'application de l'approche proposée permettent entre autres de simplifier l'identification des agrégations de poissons, de présenter une vision plus réaliste des agrégations avec leurs limites graduelles et de faciliter

l'étude de l'évolution spatio-temporelle des agrégations. En effet, les déplacements ou les changements dans la taille ou la forme des agrégations, par exemple, y apparaissent beaucoup plus clairement et peuvent ainsi être caractérisés. En 3D, les représentations obtenues offrent la possibilité de visualiser les agrégations de poissons en tant que volumes. De ces volumes peuvent être tirées des mesures des propriétés morphologiques des agrégations. De plus, les représentations 3D permettent de voir la position de chacun des poissons à l'intérieur des agrégations afin d'en étudier la structure interne.

Ces résultats permettent de conclure que les objectifs de cette recherche ont été atteints. En effet, les approches proposées ont permis une représentation plus explicite et réaliste des agrégations de poissons que les méthodes généralement utilisées et ouvrent de nouvelles possibilités d'analyse de ce phénomène pour en améliorer la compréhension. L'hypothèse de recherche a également été validée puisque les approches très innovatrices développées dans ce mémoire, basées sur les modèles d'objets flous et la triangulation Delaunay 3D combinée à un algorithme de *clustering*, améliorent la représentation des agrégations de poissons en montrant leurs limites floues et en les représentant en 3D. De plus, de manière plus générale, cette recherche a contribué à étendre les possibilités d'utilisation des SIG à un domaine d'application relativement nouveau, soit celui de la biologie marine.

5.2 Perspectives futures

Plusieurs améliorations pourraient être apportées aux approches proposées dans ce mémoire. Certaines de ces améliorations ont été suggérées dans les chapitres 3 et 4 (sections 3.6, 4.4.3.2 et 4.5). Entre autres, la topologie des objets flous représentant les agrégations de poissons devrait être gérée et une méthode efficace de détermination du paramètre α , pour la représentation des agrégations en 3D, devrait être créée. De plus, les approches développées pourraient être testées avec de nouveaux jeux de données, plus volumineux dans le cas 3D par exemple, afin de pouvoir davantage constater leurs limites et de pouvoir les améliorer. Il pourrait également être intéressant d'utiliser des données multifaisceaux pour créer des représentations 3D. De plus, des expérimentations supplémentaires pourraient être menées afin de valider la fiabilité des résultats issus de l'application des approches proposées et ce, autant en 2D qu'en 3D. Par ailleurs, les

représentations des agrégations de poissons obtenues en 2D et en 3D pourraient être intégrées dans un même système qui permettrait alors la visualisation des agrégations à différentes échelles spatiales (ex. régionale, locale).

Dans des travaux futurs, les approches suggérées dans ce mémoire pourraient être utilisées en association avec des données environnementales (ex. données mesurant les propriétés physiques de l'eau, comme la température ou la salinité) afin d'analyser les effets potentiels des changements environnementaux sur l'évolution spatio-temporelle des agrégations de poissons. Ainsi, les variations des agrégations de poissons dans l'espace (ex. déplacements, changement de forme, de taille ou de volume) et leurs liens avec l'état de l'environnement au même moment pourraient être mis en lumière.

Finalement, les approches développées dans ce mémoire pourraient être appliquées dans d'autres contextes d'étude ou domaines d'application. Par exemple, l'approche proposée en 2D pourrait être utilisée pour l'étude de troupeaux d'animaux et l'approche tridimensionnelle pourrait être employée pour modéliser des objets pouvant être échantillonnés par un ensemble de points en 3D.

Bibliographie

- Abdul-Rahman, A. et Pilouk, M. (2007) *Spatial Data Modelling for 3D GIS*. Springer-Verlag, Berlin, Allemagne, 290 p.
- Andrews, B. (2003) *Techniques for Spatial Analysis and Visualization of Benthic Mapping Data*. NOAA Coastal Services Center, SAIC Report No. 623, 28 p.
- Atkinson, D.B., Rose, G.A., Murphy, E.F. et Bishop, C.A. (1997) Distribution changes and abundance of northern cod (*Gadus morhua*), 1981-1993. *Canadian Journal of Fisheries and Aquatic Science*, 54 (Suppl.1), pp. 132-138.
- Baird, J.W., Bishop, C.A., Brodie, W.B. et Murphy, E.F. (1992) *An assessment of the cod stock in NAFO Divisions 2J3KL*. NAFO SCR Doc. 92/18, Serial No. N2063, 69 p.
- Bédard, Y. et Larrivée, S. (2003) *Composantes spatiales des données. Notes de cours GMT-20843*, Département des sciences géomatiques, Université Laval, Québec.
- Bennett, E.J. (2008) *Annual Changes in Spatial Aggregation of Atlantic cod, Gadus morhua age classes in NAFO Divisions 2J3KL and 4RS (1978-1994)*. Mémoire de maîtrise, Memorial University of Newfoundland, Faculty of Science, 126 p.
- Biomedware. What are boundaries?
<http://www.terraser.com/help/boundaryseer/index.htm>.
(Accédé le : 21-06-2008).
- Bishop, C.A., Murphy, E.F., Davis, M.B., Baird, J.W. et Rose, G.A. (1993) *An assessment of the cod stock in NAFO Divisions 2J + 3KL*. NAFO SCR Doc. 93/86, Serial No. N2271, 51 p.
- Boguslawski, P. et Gold, C.M. (2007) Atomic Operators for Construction and Manipulation of the Augmented Quad-Edge. *CISIM 2007*, 28 au 30 juin 2007, Elk, Pologne, pp. 125-128.
- Buelens, B., Williams, R., Sale, A. et Pauly, T. (2004) A Framework for Scientific Data Mining in Hydro-Acoustic Data Set. *Second International Conference on Artificial Intelligence in Science and Technology (AISAT)*, 21 au 25 novembre 2004, Hobart, Tasmanie, Australie.
- Buelens, B., Williams, R., Sale, A. et Pauly, T. (2005) Model Inversion for Midwater Multibeam Backscatter Data Analysis, *IEEE Ocean'05 Europe*, 20 au 23 juin 2005, Brest, France.
- Burrough, P.A. (1996) Natural Objects with Indeterminate Boundaries. Dans *Geographic Objects with Indeterminate Boundaries*, Taylor & Francis, Londres, UK, pp. 3-28.

- Burrough, P.A. et McDonnell, R.A. (1998) *Principles of Geographic Information Systems, Second Edition*. Oxford University Press, New York, USA, 356 p.
- Caloz, R. (2005) *Réflexions sur les incertitudes et leurs propagations en analyse spatiale*. Présentation, Laboratoire de systèmes d'information géographique, École Polytechnique Fédérale de Lausanne, 25 p.
- Canada/Newfoundland and Labrador Action Team for Cod Recovery (2005) *A Strategy for the Recovery and Management of Cod Stocks in Newfoundland and Labrador*. 69 p.
- Carette, V., Mostafavi, M.A., Devillers, R., Rose, G., Ledoux, H. et Gold, C.M. (2007) Modelling and visualisation of fish aggregations using 3D Delaunay triangulation and alpha shapes. *Proceedings 8th International Symposium on GIS and Computer Mapping for Coastal Zone Management*, 8 au 10 octobre 2007, Santander, Espagne, pp. 403-412.
- Chevrie, F. et Guély, F. (1998) *Cahier technique n° 191 - La logique floue*. Groupe Schneider, pp. 6-13.
- Childs, C. (2004) Interpolating Surfaces in ArcGIS Spatial Analyst. *ArcUser, ESRI Education Service*, pp. 32-35.
- Clementini, E. et Di Felice, P. (1996) An Algebraic Model for Spatial Objects with Indeterminate Boundaries. Dans *Geographic Objects with Indeterminate Boundaries*, Taylor & Francis, Londres, UK, pp. 155-170.
- Clementini, E. et Di Felice, P. (1997) Approximate topological relations. *International Journal of Approximate Reasoning*, 16(2), pp. 173-204.
- Cohn, A.G. et Gotts, N.M. (1996) Representing spatial vagueness: A mereological approach. *KR96: Principles of Knowledge Representation and Reasoning*, Morgan Kaufmann, San Francisco, USA, pp. 230-241.
- Courrioux, G., Nullans, S., Guillen, A., Boissonat, J.D., Repousseau, P., Renaud, X. et Thibaut, M. (2001) Volumetric modelling of Cadomian terranes (Northern Brittany, France): an automatic method using Voronoi diagrams. *Tectonophysics*, 331(1-2), pp. 181-196.
- Cova, T.J. et Goodchild, M.F. (2002) Extending geographical representation to include fields of spatial objects. *International Journal of Geographical Information Science*, 16(6), pp. 509-532.
- Cross, V.V. (1999) Using the fuzzy object model for geographical information systems. *Proceedings of the 18th International Conference of the North American Fuzzy Information Processing Society (NAFIPS)*, 10 au 12 juin 1999, New York, USA, pp. 814-818.

- C Tech EVS. C Tech Development Corporation Website. <http://www.ctech.com>. (Accédé le : 07-08-2008).
- Da, T. K. et Yvinec, M. (2006) CGAL User and Reference Manual, http://www.cgal.org/Manual/3.2/doc_html/cgal_manual/Alpha_shapes_3/Chapter_main.html. (Accédé le : 07-08-2008).
- De Berg, M., van Kreveld, M., Overmars, M. et Schwarzkopf, O. (2000) *Computational Geometry: Algorithms and Applications, Second Edition*. Springer-Verlag, Berlin, 386 p.
- De Young, B. et Rose, G.A. (1993) On recruitment and distribution of Atlantic cod (*Gadus morhua*) off Newfoundland. *Canadian Journal of Fisheries and Aquatic Sciences*, 50(12), pp. 2729-2741.
- Dilo, A., Kraipeerapun, P., Bakker, W. et de By, R.A. (2004) Storing and handling vague spatial objects. *Proceedings of the 15th International Workshop on Database and Expert Systems Applications (DEXA'04)*, IEEE Computer Society, 30 août au 3 septembre 2004, Zaragoza, Espagne, pp. 945-950.
- Dilo, A. (2006) *Representation of and reasoning with vagueness in spatial information – A system for handling vague objects*. Thèse de doctorat, Wageningen University, Pays-Bas, 162 p.
- Domeier, M.L. et Colin, P.L. (1997) Tropical reef fish spawning aggregations: defined and reviewed. *Bulletin of Marine Science*, 60(3), pp. 698-726.
- Drinkwater, K. (2002) A review of the role of climate variability in the decline of northern cod. *American Fisheries Society Symposium*, 32, pp. 113-130.
- Edelsbrunner, H., Kirkpatrick, D.G. et Siedel, R. (1983) On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4), pp.551-559.
- Edelsbrunner, H. et Mücke, E.P. (1994) Three-dimensional alpha shapes. *ACM Transactions on Graphics*, 13(1), pp. 43-72.
- Edelsbrunner, H. et Shah, N.R. (1996) Incremental topological flipping works for regular triangulations. *Algorithmica*, 15(3), pp. 223-241.
- Egenhofer, M.J. (1989) A formal definition of binary topological relationships. *Third International Conference on Foundations of Data Organization and Algorithms (FODO)*, juin 1989, Paris, France, *Lecture Notes in Computer Science*, 367, pp. 457-472.
- Ehrenberg, J.E. et Torkelson, T.C. (1996) Application of dual-beam and split-beam target tracking in fisheries acoustics. *ICES Journal of Marine Science*, 53(2), pp. 329-334.

- Erwig, M. et Schneider, M. (1997) Vague Regions. *5th International Symposium on Advances in Spatial Databases*, 15 au 18 juillet 1997, Berlin, Allemagne, *Lecture Notes in Computer Science*, 1262, Springer-Verlag, pp. 298-320.
- Fischer, K. (2005) Introduction to alpha shapes.
<http://n.ethz.ch/student/fischerk/alphashapes/as/index.html>.
(Accédé le : 12-03-2007).
- Fisher, W.L. et Rahel, F.J. (2004) *Geographic Information Systems in Fisheries*. American Fisheries Society, Bethesda, Maryland, USA, 276 p.
- Fletcher, N. Will Atlantic cod stocks recover? International Council for the Exploration of the Seas (ICES) Website. <http://www.ices.dk/marineworld/recoveryplans.asp>.
(Accédé le : 24-07-2008).
- Florida Museum of Natural History. www.flmnh.ufl.edu. (Accédé le : 02-06-2008).
- Fonseca, L., Mayer, L. et Paton, M. (2002) ArcView Objects in the Fledermaus Interactive 3-D Visualisation System : An Example from the STRATAFORM GIS. Dans *Undersea with GIS*, ESRI Press, Redlands, Californie, USA, pp. 1-21.
- Fonte, C.C. et Lodwick, W.A. (2005) Modelling the Fuzzy Spatial Extent of Geographical Entities. Dans *Fuzzy Modeling with Spatial Information for Geographic Problems*, Springer Berlin Heidelberg, New York, USA, pp. 121-142.
- Frank, A.U. (1996) The Prevalence of Objects with Sharp Boundaries in GIS. Dans *Geographic Objects with Indeterminate Boundaries*, Taylor & Francis, Londres, UK, pp. 29-40.
- Gable, C.W., Trease, H. et Cherry, T. (1996) Automated grid generation from models of complex geologic structure and stratigraphy. *Santa Barbara: National Center for Geographic Information and Analysis, proceedings paper LA-UR-96-1083*.
- Gerlotto, F., Soria, M. et Fréon, P. (1999) From two dimensions to three: The use of multibeam sonar for a new approach in fisheries acoustics. *Canadian Journal of Fisheries and Aquatic Science*, 56(1), pp. 6-12.
- Guo, B., Menon, J. et Willette, B. (1997) Surface Reconstruction Using Alpha Shapes. *Computer Graphics Forum*, 16(4), pp. 177-190.
- Hashemi, L., Mostafavi, M.A. et Gavrilova, M. L. (2007) Moving Objects Management in a 3D Dynamic Environment. *Proceedings Geocomputation 2007 CD-ROM*, 3 au 5 septembre 2007, Maynooth, Irlande.
- Hashemi, L., Blessent, D., Mostafavi, M.A. et Therrien, R. (2008) A 3D Free-Lagrangian Method to Simulate Three-Dimensional Groundwater Flow and Mass Transport.

Dans *Proceedings of the IASTED International Conference on Modelling and Simulation*, 26 au 28 mai 2008, Québec, Canada.

- Hazarika, S.M. et Cohn, A.G. (2001) A Taxonomy for Spatial Vagueness – An Alternative Egg-Yolk Interpretation. *Lecture Notes in Computer Science*, 2205, pp. 92-107.
- Hilborn, R. et Walters, C.J. (1992) *Quantitative Fisheries Stock Assessment: Choice, Dynamics and Uncertainty*. Chapman and Hall, New York, USA, 570 p.
- Hutchings, J.A. (1996) Spatial and temporal variation in the density of northern cod and a review of hypotheses for the stock's collapse. *Canadian Journal of Fisheries and Aquatic Sciences*, 53(5), pp. 943-962.
- IVS. IVS 3D Fledermaus documentation. <http://www.ivs3d.com/products/fledermaus>. (Accédé le : 07-08-2008).
- Jacquez, G.M., Maruca, S.L. et Fortin, M.J. (2000) From fields to objects: A review of geographic boundary analysis. *Journal of Geographical Systems*, 2(3), pp. 221-241.
- Joe, B. (1991) Construction of three-dimensional Delaunay triangulations using local transformations. *Computer Aided Geometric Design*, 8(2), pp. 123-142.
- Kraipeerapun, P. (2004) *Implementation of vague spatial objects*. Mémoire de maîtrise, International Institute for Geo-information Science and Earth Observation, Enschede, Pays-Bas, 60 p.
- Kremenová, O. (2004) *Fuzzy Modeling of Soil Maps*. Mémoire de maîtrise, Helsinki University of Technology, Department of Surveying, Helsinki, Finlande, 74 p.
- Kulka, D.W., Wroblewski, J.S. et Narayanan, S. (1995) Recent changes in the winter distribution and movements of northern Atlantic cod (*Gadus morhua* Linnaeus, 1758) on the Newfoundland-Labrador Shelf. *ICES Journal of Marine Science*, 52(6), pp. 889-902.
- Kulka, D.W., Hendrickson, L., Bez, N., Schlitzer, R., Black, G.A.P. et Simpson, M.R. (2005) Workshop on Mapping and Geostatistical Methods for Fisheries Stock Assessment. *NAFO Scientific Council Studies*, 39, pp. 1-43.
- Lachance, B. (2005) *Développement d'une structure topologique de données 3D pour l'analyse de modèles géologiques*. Mémoire de maîtrise, Université Laval, Département des sciences géomatiques, Québec, Canada, 115 p.
- Lartigue, C., Chevalier, B. et Mathieu, L. (2000) The Concept of Alpha-Shapes Applied to Three-Dimensional Metrology. Dans *Advanced Mathematical and Computational Tools in Metrology IV*, World Scientific Publishing Company, Londres, UK, pp. 133-142.

- Lattuada, R. et Raper, J. (1996) Applications of Delaunay tetrahedralisation Algorithms in Geoscientific Modeling. *The Third International Conference/Workshop on Integrating GIS and Environmental Modeling CD-ROM*, 21 au 25 janvier 1996, Santa Fe, Nouveau-Mexique, USA.
- Ledoux, H. (2006) *Modelling three-dimensional fields in geoscience with the Voronoi diagram and its dual*. Thèse de doctorat, School of Computing, University of Glamorgan, Pontypridd, Wales, UK, 188 p.
- Ledoux, H. et Gold, C.M. (2006) La modélisation de données océanographiques à l'aide du diagramme de Voronoï tridimensionnel. *Revue internationale de géomatique*, 16(1), pp. 151-170.
- Ledoux, H. et Gold, C.M. (2008) Modelling three-dimensional geoscientific fields with the Voronoi diagram and its dual. *International Journal of Geographical Information Science*, 22(5), pp. 547-574.
- Lescieux, M. (2008) *Introduction à la logique floue – Application à la commande floue*. Notes de cours, École Polytechnique de l'Université de Tours, 44 p.
- Leung, Y. (1987) On the imprecision of boundaries. *Geographical Analysis*, 19, pp. 125-151.
- Li, R. (1994) Data Structure and application issue in 3-D Geographic Information Systems. *Geomatica*, 48(3), pp. 209-224.
- Li, Z., Zhu, Q. et Gold, C.M. (2005) *Digital Terrain Modeling: Principles and Methodology*. CRC Press, Floride, USA, 323 p.
- Longley, P.A., Goodchild, M.F., Maguire, D.J. et Rhind, D.W. (2005) *Geographic Information Systems and Science, Second Edition*. John Wiley & Sons, UK, 517 p.
- Marine Marchande. <http://www.marine-marchande.net/Jourlejour/AujourleJour-91.htm>. (Accédé le : 02-07-2008).
- Mayer, L., Li, Y. et Melvin, G. (2002) 3D visualization for pelagic fisheries research and assessment. *ICES Journal of Marine Science*, 59(1), pp. 216-225.
- Meaden, G.J. et Do Chi, T. (1996) *Geographical Information Systems: Applications to marine fisheries*. FAO Fisheries Technical Paper No.356. FAO, Rome, Italie. 335 p.
- Meaden, G.J. (2000) GIS in Fisheries Management. *GeoCoast*, 1(1), pp. 81-102.
- Mello, L.G.S. et Rose, G.A. (2005) Using geostatistics to quantify seasonal distribution and aggregation patterns of fishes: an example of Atlantic cod (*Gadus morhua*). *Canadian Journal of Fisheries and Aquatic Sciences*, 62(3), pp. 659-670.

- Moleenar, M. (1998) *An Introduction to the Theory of Spatial Object Modelling for GIS*. Taylor & Francis, Londres, UK, 246 p.
- Mostafavi, M.A., Gold, C.M. et Dakowicz, M. (2003) Delete and insert operations in Voronoi/Delaunay methods and applications. *Computers & Geosciences*, 29(4), pp. 523-530.
- Mostafavi, MA. et Gold, C.M. (2004) A global dynamic spatial data structure for marine simulation. *International Journal of Geographical Information Science*, 18(3), pp. 211-227.
- MPO. Programme des observateurs en mer. <http://www.qc.dfo-mpo.gc.ca/peches/fr/surveillance/programme.htm>. (Accédé le : 19-06-2008).
- Munkres, J. R. (1993) Simplicial Complexes and Simplicial Maps. Dans *Elements of Algebraic Topology*, Westview Press, USA, pp. 7-14.
- NASCA Géosystèmes (2006) *Un système d'analyse spatiale souple (logique floue), Outil d'aide à la décision*. Développement des Systèmes d'Information Géographique, 12 p.
- Okabe, A., Boots, B., Sugihara, K. et Chiu, S.N. (1999) *Spatial tessellations, concepts and applications of Voronoi Diagrams*. Weinheim : John Wiley, Chichester, New York, USA, 671 p.
- Pauly, A. et Schneider, M. (2004) Vague Spatial Data Types, Set Operations, and Predicates. *Advances in Databases and Information Systems (ADBIS 2004)*, 3255, 22 au 25 septembre 2004, Budapest, Hongrie, pp. 379-392.
- Pêches et Océans Canada (2004) Histoire récente de la pêche à la morue dans le Canada Atlantique. http://www.dfo-mpo.gc.ca/kids-enfants/map-carte/map_f.htm. (Accédé le : 07-08-2008).
- Pouliot, J., Lachance, B., Brisebois, A., Rabaud, O. et Kirkwood, D. (2003) 3D geological modeling: Are GIS or CAD appropriate? *Proceedings of ISPRS Workshop, WG II/5, II/6, IV/1 and IV/2 Joint Workshop on "Spatial, Temporal and Multi-Dimensional Data Modelling and Analysis"*, 2 et 3 octobre 2003, Québec, Canada.
- Radio-Canada. Requiem pour la morue. http://archives.radio-canada.ca/economie_affaires/ressources_naturelles/dossiers/1272/. (Accédé le : 22-07-2008).
- Raper, J. (2000) 2.5- and 3-D GIS for coastal geomorphology. Dans *Marine and Coastal Geographical Information Systems*. Taylor & Francis, Londres, UK, pp. 129-136.

- Rice, J.C., Shelton, P.A., Rivard, D., Chouinard, G.A. et Fréchet, A. (2003) Recovering Canadian Atlantic Cod Stocks : The Shape of Things to Come? *International Council for Exploration of the Sea*, 23 p.
- Rose, G.A. (1993) Cod spawning on a migration highway in the north-west Atlantic. *Nature*, 366(6454), pp. 458-461.
- Rose, G.A., Atkinson, D.B., Baird, J., Bishop, C.A. et Kulka, D.W. (1994) Changes in distribution of Atlantic cod and thermal variations in Newfoundland waters, 1980-1992. *ICES Marine Science Symposia*, 198, pp. 542-552.
- Rose, G.A. et Kulka, D.W. (1999) Hyperaggregation of fish and fisheries: how catch-per-unit-effort increased as the northern cod (*Gadus morhua*) declined. *Canadian Journal of Fisheries and Aquatic Sciences*, 56(Suppl. 1), pp. 118-127.
- Rose, G.A. (2004) Reconciling overfishing and climate change with stock dynamics of Atlantic cod (*Gadus morhua*) over 500 years. *Canadian Journal of Fisheries and Aquatic Sciences*, 61(9), pp. 1553-1557.
- Rose, G.A. (2006) Communication personnelle.
- Rose, G.A. (2007) *Cod: The Ecological History of the North Atlantic Fisheries*. Breakwater Books, St. John's, NL, Canada, 590 p.
- Samet, H. (1984) The quadtree and related hierarchical data structures. *ACM Computing Surveys*, 16(2), pp. 187-260.
- Schneider, M. (1996) Modelling Spatial Objects with Undetermined Boundaries using the Realm/ROSE Approach. Dans *Geographic Objects with Indeterminate Boundaries*, Taylor & Francis, Londres, UK, pp. 141-152.
- Schneider, M. (1999) Uncertainty Management for Spatial Data in Databases: Fuzzy Spatial Data Types. 6th *International Symposium on Advances in Spatial Databases*, 20 au 23 juillet 1999, Hong Kong, Chine, *Lecture Notes in Computer Science*, 1651, Springer-Verlag, pp. 330-351.
- Schneider, M. (2003) Design and Implementation of Finite Resolution Crisp and Fuzzy Spatial Objects. *Data and Knowledge Engineering*, 44(1), pp. 81-108.
- Shewchuk, R. (2005) Star splaying: an algorithm for repairing Delaunay triangulations and convex hulls. *SCG '05: Proceedings of the Twenty-First Annual ACM Symposium on Computational Geometry*, 8 au 10 juin 2005, Pise, Italie, ACM Press, New York, USA, pp. 237-246.
- Shyue, S. et Tsai, P. (1996) A Study on the Dimensional Aspect of the Marine Geographic Information Systems. *OCEANS '96, MTS/IEEE, 'Prospects for the 21st Century'*,

Conference Proceedings, 2, 23 au 26 septembre 1996, Fort Lauderdale, Floride, USA, pp. 674-679.

Simmonds, J. et MacLennan, D. (2005) *Fisheries Acoustics, Theory and Practice, Second Edition*. Blackwell Publishing, Oxford, UK, 437 p.

SonarData Pty Ltd. Echoview on-line documentation.
<http://www.sonardata.com/WebHelp/Echoview.htm>. (Accédé le : 07-08-2008).

St. Martin, K. (2004) GIS in Marine Fisheries Science and Decision Making. Dans *Geographic Information Systems in Fisheries*, W.L Fisher et F.J. Rahel eds. (American Fisheries Society), pp. 237-258.

Tang, X. (2004) Spatial Object Modeling in Fuzzy Topological Spaces with Applications to Land Cover Change. Thèse de doctorat, International Institute for Geo-Information Science and Earth Observation, Enschede, Pays-Bas, 219 p.

Tobler, W.R. (1970) A computer movie simulating urban growth in the Detroit region. *Economic Geography*, 46(2), pp. 234-240.

UNESCO Ocean Teacher. GIS Data Models.
http://www.oceanteacher.org/oceanteacher/index.php?module=resourcekit&action=content&nodeid=gen11Srv32Nme37_529. (Accédé le : 07-08-2008).

Valavanis, V.D. (2002) *Geographic Information Systems in Oceanography and Fisheries*. Taylor & Francis, Londres, UK, 209 p.

Wang, F. (1994) Towards a natural language user interface: An approach of fuzzy query. *International Journal of Geographical Information Systems*, 8(2), pp. 143-162.

Wang, F. et Hall, G.B. (1996) Fuzzy representation of geographical boundaries in GIS. *International Journal of Geographical Information Systems*, 10(5), pp. 573-590.

Worboys, M.F. et Duckham, M. (2004) *GIS : A Computing Perspective, Second Edition*. CRC Press, Floride, USA, 448 p.

Wright, D.J. et Goodchild, M.F. (1997) Data from the deep: Implications for the GIS community. *International Journal of Geographical Information Science*, 11(6), pp. 523-528.

Wright, D.J. et Bartlett, D.J. (2000) *Marine and Coastal Geographical Information Systems*. Taylor & Francis, Londres, UK, 320 p.

Xue, Y., Sun, M. et Ma, A. (2004) On the reconstruction of three-dimensional complex geological objects using Delaunay triangulation. *Future Generation Computer Systems*, 20(7), pp. 1227-1234.

- Yanar, T.A. et Akyürek, Z. (2004) The Enhancement of ArcGIS with Fuzzy Set Theory. *ESRI International User Conference*, 30 juin 2004, San Diego, Californie, USA.
- Yuan, M. (2001) Representing Complex Geographic Phenomena in GIS. *Cartography and Geographic Information Science*, 28(2), pp. 83-96.
- Zadeh, L.A. (1965) Fuzzy Sets. *Information and Control*, 8(3), pp. 338-353.
- Zhan, F.B. (1997) Topological relations between fuzzy regions. *Proceedings of ACM symposium on Applied computing*, 28 février au 2 mars 1997, San Jose, Californie, USA, ACM Press, pp. 192-196.
- Zhan, F.B. (1998) Approximate analysis of binary topological relations between geographic regions with indeterminate boundaries. *Soft Computing*, 2(2), pp. 28-34.
- Zhan, F.B. et Lin, H. (2003) Overlay of Two Simple Polygons with Indeterminate Boundaries. *Transactions in GIS*, 7(1), pp. 67-81.

Annexe A Extrait du jeu de données utilisé pour la représentation 2D

Jeu de données portant sur la morue de l'Atlantique (*Gadus morhua*) provenant de relevés scientifiques menés par des navires du Ministère des Pêches et Océans Canada (MPO). Ces observations sont acquises par des échantillonnages dits stratifiés aléatoires. Ces données sont disponibles pour les années 1948 à 2006, avec diverses étendues spatiales et différents nombres de stations, dépendant des années.

Tableau A.1 - Extrait des relevés scientifiques du MPO, morue de l'Atlantique, 1990, région de Terre-Neuve.

OBJECTID	Shape	ID	RECORD_TYP	VESSEL	TRIP
19	Donnée binaire	35798	6	20	185
20	Donnée binaire	35800	6	20	185
21	Donnée binaire	35802	6	20	185
22	Donnée binaire	35804	6	20	185

OBJECTID	SET_	YEAR_	MONTH_	DAY_	SET_TYPE
19	91	90	7	19	1
20	92	90	7	19	1
21	93	90	7	19	1
22	94	90	7	19	1

OBJECTID	STRATUM	NAFO	UNIT_AREA	LIGHT	WIND_DIREC
19	202	2J	N16	202	4
20	103	2J	N16	177	4
21	103	2J	N16	0	4
22	103	2J	N16	0	5

OBJECTID	WIND_FORCE	SEA	BOTTOM_TYP	TIME_MID	DURATION_S
19	4	3		1826	30
20	4	3		2017	30
21	4	3		2144	30
22	4	4		2328	30

OBJECTID	DISTANCE_T	OPERATION_	MEAN_DEPTH	MIN_DEPTH	MAX_DEPTH
19	1.799999952	2	336	335	336
20	1.799999952	1	376	372	379
21	1.799999952	1	377	376	378
22	1.799999952	1	375	373	377

UNGAVA BAY- SOUTH OF 61N = UB
" " - NORTH OF 61N = HS

3PS = 3P
 3PN = 3Q
 4VS = 4V
 4VN = 4U (4Z)
 5ZE & 5ZW = 5Z

UNIT AREA

NFLD AREA GRID MAP- SQUARE
 EG: L30, K29, P16 ETC

LIGHT

(1) GENERAL USE:

 994.. DARK
 995.. MOONLIGHT
 996.. DUSK & DAWN
 997.. DULL- OVERCAST, RAIN, FOG
 998.. BRIGHT BUT HAZY
 999.. BRIGHT SUNLIGHT

OR (2) | ACTUAL METER READINGS FROM 0 TO 976
 | PRORATING OF READINGS FROM 976-993

OR (3) | CODE ACTUAL LIGHT READINGS AS
 | FOLLOWS:

0.. DARK
 1.. UP TO .016 FOOT CANDLES
 2.. UP TO .032 " "
 3.. UP TO .065 " "
 4.. UP TO .13 " "
 5.. UP TO .26 " "
 6.. UP TO .50 " "
 7.. UP TO 1 " "
 8.. UP TO 2 " "
 9.. UP TO 4 " "
 10.. UP TO 8 " "
 11.. UP TO 16 " "
 12.. UP TO 32 " "
 13.. UP TO 65 " "
 14.. UP TO 130 " "
 15.. UP TO 260 " "
 16.. UP TO 500 " "
 17.. UP TO 1000 " "
 18.. UP TO 2000 " "
 19.. UP TO 4000 " "
 20.. UP TO 8000 " "
 21.. UP TO 16000 " "
 22.. UP TO 32000 " "

WIND DIRECTION

0.. CALM 5.. 158-202(S)
 1.. 338-022(N) 6.. 203-247(SW)
 2.. 023-067(NE) 7.. 248-292(W)
 3.. 068-112(E) 8.. 293-337(NW)
 4.. 113-157(SE) 9.. VARIABLE

WIND FORCE	0 - 9.....BEAUFORT SCALE
SEA	0 - 9.....STANDARD SEA CODE
BOTTOM TYPE	1.. MUD 2.. SANDY MUD 3.. SAND-- SAND & SHELLS 4.. FINE GRAVEL- SAND& GRAVEL 5.. COURSE GRAVEL- ROCK & GRAVEL 6.. BOULDERS 7.. ROCKS 8.. OTHER
TIME- MIDPOINT	NST-- 24 HOUR CLOCK
DURATION OF SET	MINUTES
DISTANCE TOWED	NAUTICAL MILES TO 1 DECIMAL
OPERATION OF GEAR	1.. NORMAL- NO DAMAGE 2.. NORMAL- SOME DAMAGE TO NET, CATCH NOT AFFECTED 3.. UNSUCCESSFUL- GEAR BADLY DAMAGED CATCH AFFECTED 4.. UNSUCCESSFUL- DEPTH RANGE OF SET WAS TOO LARGE 5.. UNSUCCESSFUL- OTHER REASONS, CODEND UNTIED DOORS LOCKED NET NOT ON BOTTOM ETC
DEPTH- MEAN	METERS-- DEPTH FISHED
DEPTH- MINIMUM	METERS-- DEPTH FISHED
DEPTH- MAXIMUM	METERS-- DEPTH FISHED
DEPTH- BOTTOM	METERS-- DEPTH FISHED
TEMPERATURE- SURFACE	DEGREES CELCIUS TO 1 DECIMAL 9 IN POSN 60 DECIGNATES MINUS 1.2 = 12 0.0 = 900 -1.2 = 912
TEMPERATURE-FISHING DEPTH OR BOTTOM	AS PER SURFACE TEMPERATURE
LATITUDE-START	DEGREES AND MINUTES TO 1 DECIMAL(N) AT START OF SET
LONGITUDE-START	AS PER LATITUDE (W)
POSITION METHOD	0.. UNKNOWN 1.. DEAD RECKONING 2.. RADAR 3.. DECCA

4.. LORAN
 5.. SATELLITE NAVIGATION
 6.. GLOBAL POSITIONING SYSYEM (GPS)

GEAR	APPENDIX - GEAR CODE/NAME
SPECIES	COD = 0438 AS PER AKENHEAD/LEGROW DATA REPORT # 309 AS PER LILLY DATA REPORT # 365
NUMBER	ACTUAL TOTAL NO. OF THIS SPECIES CAUGHT
WEIGHT	KGS. TO 2 DECIMALS TOTAL WEIGHT OF THIS SPECIES CAUGHT
LATITUDE- END	DEGREES AND MINUTES TO 1 DECIMAL(N) AT END OF SET
LONGITUDE- END	AS PER LATITUDE (W)
BOTTOM TEMPERATURE COLLECTION DEVICE	1.. CTD-- NET MOUNTED, MODE 2.. CTD-- VERTICAL, END OF TOW 3.. XBT-- " " " " 4.. KNUDSEN REVERSING WATER BOTTLE- VERTICAL, END OF TOW 5.. SCANMAR
GEAR MONITORING DEVICE	1.. SCANMAR 2.. FURUNO 3.. FS3300

Annexe B Code implanté pour le *clustering* 3D (algorithme des α -shapes)

Le programme effectuant les calculs de la triangulation Delaunay 3D a été réalisé dans le langage Delphi et a été fourni par Ledoux (2006). L'algorithme de *clustering* 3D des α -shapes (voir figure B.1) a été intégré dans ce code source.

```

Entrée : Triangulation Delaunay 3D du jeu de données original
Sortie : Intervalles  $I_1$  tel que le simplexe fait partie de l'enveloppe de l' $\alpha$ -shape si et
seulement si  $\alpha$  appartient à  $I_1$ , et  $I_2$ , tel que le simplexe fait partie de l'intérieur
de l' $\alpha$ -shape si et seulement si  $\alpha$  appartient à  $I_2$ 

POUR CHAQUE tétraèdre de la triangulation FAIRE
    a = rT3
    I1 = ∅
    I2 = [a, ∞]
FIN POUR

POUR k = 2 à 0
    POUR CHAQUE k-simplexe FAIRE
        SI k-simplexe est attaché ALORS
            a = min_rT(k+1)
        SINON
            a = rT(k)
        SI k-simplexe appartient à l'enveloppe convexe ALORS
            b = ∞
        SINON
            b = max_rT3
        I1 = [a, b]
        I2 = [b, ∞]
    FIN POUR
FIN POUR

```

Figure B.1 – Algorithme de *clustering* des α -shapes en 3D.

Le programme est essentiellement construit autour d'une classe « Mesh ». Beaucoup de fonctions essentielles au calcul des α -shapes, les fonctions de déterminant, de création

des listes de tétraèdres ou de points par exemple, étaient déjà disponibles dans le code source fourni. Le code a donc été complété en ajoutant les fonctions et classes manquantes au calcul des α -shapes. Voici une description sommaire des ajouts les plus importants :

Classe « Mesh »

Fonctions de création des listes des simplexes de la triangulation Delaunay

- **« CreateTriangleList »**

La fonction « CreateEdgeList » est utile pour créer la liste de tous les triangles de la triangulation Delaunay, soit la liste des 2-simplexes nécessaire au calcul de l' α -shape. Les triangles sont trouvés en identifiant les faces de chacun des tétraèdres de la triangulation Delaunay. En effet, étant donné que l'intersection de deux tétraèdres est leur face commune, il est facile de trouver les triangles à partir de la liste des tétraèdres (cette liste de tétraèdres est mise à jour suite à l'algorithme de construction de la triangulation Delaunay) puisque les tétraèdres adjacents sont stockés. Il faut trouver les trois points qui sont communs au tétraèdre testé et à son tétraèdre adjacent pour ensuite recréer le triangle qui forme l'une des faces de la triangulation Delaunay.

- **« CreateEdgeList »**

La fonction « CreateEdgeList » sert principalement à créer la liste de tous les segments de la triangulation Delaunay, soit la liste des 1-simplexes nécessaire au calcul de l' α -shape. La liste des tétraèdres de la triangulation Delaunay étant connue, les segments sont trouvés en reliant les sommets de chacun des tétraèdres deux à deux.

Fonctions de calcul d'intervalles

- **« IntervalTetra »**
- **« IntervalTriangle »**
- **« IntervalEdge »**
- **« IntervalVertex »**

Ces fonctions servent à trouver les intervalles pour lesquels les différents simplexes de la triangulation de Delaunay appartiennent soit à l'enveloppe de l' α -shape ou encore à son intérieur.

Fonctions de tri des triangles et des segments pour éliminer les doublons

- « **TriTriangleList** »
- « **TriEdgeList** »

Fonctions de traçage de l' α shape

- « **DrawAlphaShapeTetra** »
- « **DrawAlphaShapeTriangle** »
- « **DrawAlphaShapeEdge** »

Ces fonctions servent au traçage de l' α -shape. En premier lieu, elles appellent les fonctions de détermination des intervalles selon le type de simplexes qu'elles dessinent. Par la suite, elles comparent la valeur de α entrée par l'utilisateur aux intervalles pour décider quels simplexes doivent être tracés. Finalement, elles appellent la fonction de dessin appropriée.

Classe « AlphaTools »

La classe « AlphaTools » regroupe toutes les fonctions géométriques employées pour le calcul de l' α -shape en 3D. Il s'agit d'une nouvelle classe créée dans le code source.

Fonctions de calcul du rayon de la sphère ou du cercle circonscrit

- « **RadiusTetra** »
- « **RadiusTriangle** »
- « **RadiusEdge** »

Ces fonctions servent à calculer les rayons des sphères circonscrites à chacun des tétraèdres, des triangles et des segments. Elles appellent les fonctions de déterminants.

Fonctions de test du prédicat attaché

- « **AttachTriangle** »
- « **AttachEdge** »

Ces fonctions servent à déterminer si un segment ou un triangle est attaché ou non. Elles appellent les fonctions de déterminant.

EXTRAIT DU CODE IMPLANTÉ

```

// Les listes de tous les nœuds et de tous les tétraèdres de la triangulation
// Delaunay existent déjà dans le programme

// Création de la liste de tous les segments de la triangulation Delaunay
function TMesh.CreateEdgeList: TList;
var
  te: TTetra;
  pt1, pt2, pt3, pt4: TPoint;
  listEdges: TList;
  i, j: integer;
begin
  listEdges := TList.Create;
  for i := 0 to (m_IstTetra.Count - 1) do
    for j := 1 to 4 do
      begin
        te := m_IstTetra[i];
        pt1 := te.GetP(j);
        if (j = 1) then
          begin
            pt2 := te.GetP(2);
            pt3 := te.GetP(3);
            pt4 := te.GetP(4);
            if (self.IsBigTetra(pt1) = false) and (self.IsBigTetra(pt2) = false) and (self.IsBigTetra(pt3)
              = false) and (self.IsBigTetra(pt4) = false) then
              begin
                listEdges.Add(TEdge.Create(pt1, pt2, te));
                listEdges.Add(TEdge.Create(pt1, pt3, te));
                listEdges.Add(TEdge.Create(pt1, pt4, te));
              end;
            end;
          if (j = 2) then
            begin
              pt2 := te.GetP(3);
              pt3 := te.GetP(4);
              pt4 := te.GetP(1);
              if (self.IsBigTetra(pt1) = false) and (self.IsBigTetra(pt2) = false) and (self.IsBigTetra(pt3)
                = false) and (self.IsBigTetra(pt4) = false) then
                  begin
                    listEdges.Add(TEdge.Create(pt1, pt2, te));
                    listEdges.Add(TEdge.Create(pt1, pt3, te));
                    listEdges.Add(TEdge.Create(pt1, pt4, te));
                  end;
                end;
            if (j = 3) then
              begin
                pt2 := te.GetP(4);
                pt3 := te.GetP(1);
                pt4 := te.GetP(2);
                if (self.IsBigTetra(pt1) = false) and (self.IsBigTetra(pt2) = false) and (self.IsBigTetra(pt3)
                  = false) and (self.IsBigTetra(pt4) = false) then
                    begin

```

```

        listEdges.Add(TEdge.Create(pt1, pt2, te));
        listEdges.Add(TEdge.Create(pt1, pt3, te));
        listEdges.Add(TEdge.Create(pt1, pt4, te));
    end;
end;
if (j = 4) then
    begin
        pt2 := te.GetP(1);
        pt3 := te.GetP(2);
        pt4 := te.GetP(3);
        if (self.IsBigTetra(pt1) = false) and (self.IsBigTetra(pt2) = false) and (self.IsBigTetra(pt3)
            = false) and (self.IsBigTetra(pt4) = false) then
            begin
                listEdges.Add(TEdge.Create(pt1, pt2, te));
                listEdges.Add(TEdge.Create(pt1, pt3, te));
                listEdges.Add(TEdge.Create(pt1, pt4, te));
            end;
        end;
    end;
    Result := listEdges;
end;

```

// Création de la liste de tous les triangles de la triangulation Delaunay

```

function TMesh.CreateTriangleList: TList;
var
    listFace: TList;
    tetra1, tetra2: TTetra;
    point1, point2, point3, point4: TPoint;
    i, p: integer;
begin
    listFace := TList.Create;
    for p := 0 to (m_lstTetra.Count - 1) do
        begin
            for i := 1 to 4 do
                begin
                    tetra1 := m_lstTetra[p];
                    tetra2 := tetra1.GetT(i);
                    point1 := tetra1.GetP(1);
                    point2 := tetra1.GetP(2);
                    point3 := tetra1.GetP(3);
                    point4 := tetra1.GetP(4);
                    if (self.IsBigTetra(point1) = false) and (self.IsBigTetra(point2) = false) and
                        (self.IsBigTetra(point3) = false) and (self.IsBigTetra(point4) = false) then
                        begin
                            if (tetra2.HasFace(point1, point2, point3) = true) then
                                begin
                                    listFace.Add(TFace.Create(point1, point2, point3, tetra1, tetra2));
                                end;
                            if (tetra2.HasFace(point1, point2, point4) = true) then
                                begin
                                    listFace.Add(TFace.Create(point1, point2, point4, tetra1, tetra2));
                                end;
                            if (tetra2.HasFace(point1, point3, point4) = true) then
                                begin
                                    listFace.Add(TFace.Create(point1, point3, point4, tetra1, tetra2));
                                end;
                        end;
                end
            end
        end

```

```

    end;
    if (tetra2.HasFace(point2, point3, point4) = true) then
    begin
        listFace.Add(TFace.Create(point2, point3, point4, tetra1, tetra2));
    end;
    end;
    end;
    end;
    end;
    result := listFace;
end;

//*****
// Description: Calcul de l'alpha shape pour les tétraèdres
// Input:      triangulation Delaunay de l'ensemble de points
// Output:     intervalles pour les tétraèdres
//*****

//Calcul des intervalles pour les tétraèdres
function TMesh.IntervalTetra: TList;
var
    listTetra: TList;
    listAlphaTetra: TList;
    te: TTetra;
    point1, point2, point3, point4: TPoint;
    i: integer;
    aTetra: double;
begin
    listTetra := self.GetListTetra;
    listAlphaTetra := TList.Create;
    for i := 0 to (listTetra.Count - 1) do
    begin
        te := listTetra[i];
        point1 := te.GetP(1);
        point2 := te.GetP(2);
        point3 := te.GetP(3);
        point4 := te.GetP(4);
        aTetra := TAlphaTools.RadiusTetra(point1, point2, point3, point4);
        listAlphaTetra.add(TTetra2.Create(point1, point2, point3, point4, aTetra));
    end;
    result := listAlphaTetra;
end;

//Dessin de l'alpha shape
procedure TMesh.DrawAlphaShapeTetra;
var
    i, j: integer;
    tetra, tetra1: TTetra;
    point1, point2, point3, point4: TPoint;
    tetra2, tetra3: TTetra2;
    listAlphaTetra: TList;
    alpha: double;
begin

```

```

alpha := self.SetAlpha;
listAlphaTetra := self.IntervalTetra;
if (bDrawAlphaShape = true) then
  begin
    for i := 0 to (listAlphaTetra.Count - 1) do
      begin
        tetra3 := listAlphaTetra[i];
        point1 := tetra3.GetP(1);
        point2 := tetra3.GetP(2);
        point3 := tetra3.GetP(3);
        point4 := tetra3.GetP(4);
        tetra1 := TTetra.Create(tetra3.GetP(1),tetra3.GetP(2),tetra3.GetP(3),tetra3.GetP(4));
        if (self.IsOutsideCH(tetra1) = false) then
          begin
            if (tetra3.GetAlpha < alpha) then
              begin
                TDrawTools.DrawTetraFaces(tetra1, 0, 0.4, 0.5, 1);
              end;
            end;
          end;
        end;
      end;
    end;
  end;

```

```

//*****
// Description: Calcul de l'alpha shape pour les triangles
// Input:      triangulation Delaunay de l'ensemble de points
// Output:     intervalles pour les triangles
//*****

```

```

//Calcul des intervalles pour les triangles

```

```

function TMesh.IntervalTriangle: TList;

```

```

var

```

```

  listFace: TList;
  listAlphaTriangle: TList;
  tri: TFace;
  point1, point2, point3: TPoint;
  pt1, pt2, pt3, pt4, pt: TPoint;
  p1, p2, p3, p4: TPoint;
  pTest1, pTest2: TPoint;
  tetra1, tetra2: TTetra;
  i: integer;
  aTriangle, bTriangle: double;
  b_rep1, b_rep2: boolean;
  radT1, radT2: double;
  bTriangleBoundary: boolean;

```

```

begin

```

```

  listFace := self.CreateTriangleList;
  listAlphaTriangle := TList.Create;

```

```

  for i := 0 to (listFace.Count - 1) do

```

```

    begin

```

```

      tri := listFace[i];
      point1 := tri.GetPoint(1);

```

```
point2 := tri.GetPoint(2);
point3 := tri.GetPoint(3);
```

//Triangle attaché ou non attaché

```
tetra1 := tri.GetTetra1;
pt1 := tetra1.GetP(1);
if (((pt1.x <> point1.x) and (pt1.y <> point1.y) and (pt1.z <> point1.z)) and ((pt1.x <>
  point2.x) and (pt1.y <> point2.y) and (pt1.z <> point2.z)) and ((pt1.x <> point3.x) and
  (pt1.y <> point3.y) and (pt1.z <> point3.z))) then
  begin
    pTest1 := pt1;
  end;
pt2 := tetra1.GetP(2);
if (((pt2.x <> point1.x) and (pt2.y <> point1.y) and (pt2.z <> point1.z)) and ((pt2.x <>
  point2.x) and (pt2.y <> point2.y) and (pt2.z <> point2.z)) and ((pt2.x <> point3.x) and
  (pt2.y <> point3.y) and (pt2.z <> point3.z))) then
  begin
    pTest1 := pt2;
  end;
pt3 := tetra1.GetP(3);
if (((pt3.x <> point1.x) and (pt3.y <> point1.y) and (pt3.z <> point1.z)) and ((pt3.x <>
  point2.x) and (pt3.y <> point2.y) and (pt3.z <> point2.z)) and ((pt3.x <> point3.x) and
  (pt3.y <> point3.y) and (pt3.z <> point3.z))) then
  begin
    pTest1 := pt3;
  end;
pt4 := tetra1.GetP(4);
if (((pt4.x <> point1.x) and (pt4.y <> point1.y) and (pt4.z <> point1.z)) and ((pt4.x <>
  point2.x) and (pt4.y <> point2.y) and (pt4.z <> point2.z)) and ((pt4.x <> point3.x) and
  (pt4.y <> point3.y) and (pt4.z <> point3.z))) then
  begin
    pTest1 := pt4;
  end;

tetra2 := tri.GetTetra2;
p1 := tetra2.GetP(1);
if (((p1.x <> point1.x) and (p1.y <> point1.y) and (p1.z <> point1.z)) and ((p1.x <> point2.x)
  and (p1.y <> point2.y) and (p1.z <> point2.z)) and ((p1.x <> point3.x) and (p1.y <>
  point3.y) and (p1.z <> point3.z))) then
  begin
    pTest2 := p1;
  end;
p2 := tetra2.GetP(2);
if (((p2.x <> point1.x) and (p2.y <> point1.y) and (p2.z <> point1.z)) and ((p2.x <> point2.x)
  and (p2.y <> point2.y) and (p2.z <> point2.z)) and ((p2.x <> point3.x) and (p2.y <>
  point3.y) and (p2.z <> point3.z))) then
  begin
    pTest2 := p2;
  end;
p3 := tetra2.GetP(3);
if (((p3.x <> point1.x) and (p3.y <> point1.y) and (p3.z <> point1.z)) and ((p3.x <> point2.x)
  and (p3.y <> point2.y) and (p3.z <> point2.z)) and ((p3.x <> point3.x) and (p3.y <>
  point3.y) and (p3.z <> point3.z))) then
  begin
    pTest2 := p3;
  end;
```

```

p4 := tetra2.GetP(4);
if (((p4.x <> point1.x) and (p4.y <> point1.y) and (p4.z <> point1.z)) and ((p4.x <> point2.x)
and (p4.y <> point2.y) and (p4.z <> point2.z)) and ((p4.x <> point3.x) and (p4.y <>
point3.y) and (p4.z <> point3.z))) then
  begin
    pTest2 := p4;
  end;

b_rep1 := TAlphaTools.AttachTriangle(point1, point2, point3, pTest1);
b_rep2 := TAlphaTools.AttachTriangle(point1, point2, point3, pTest2);

radT1 := TAlphaTools.RadiusTetra(point1, point2, point3, pTest1);
radT2 := TAlphaTools.RadiusTetra(point1, point2, point3, pTest2);

if ((b_rep1 = true) and (b_rep2 = true)) then
  begin
    if (radT1 < radT2) then
      begin
        aTriangle := radT1;
        pt := pTest1;
      end;
    if (radT1 > radT2) then
      begin
        aTriangle := radT2;
        pt := pTest2;
      end;
    if (radT1 = radT2) then
      begin
        aTriangle := radT1;
        pt := pTest1;
      end;
    end;
    if ((b_rep1 = true) and (b_rep2 = false)) then
      begin
        aTriangle := radT1;
        pt := pTest1;
      end;
    if ((b_rep1 = false) and (b_rep2 = true)) then
      begin
        aTriangle := radT2;
        pt := pTest2;
      end;
    if ((b_rep1 = false) and (b_rep2 = false)) then
      begin
        aTriangle := TAlphaTools.RadiusTriangle(point1, point2, point3);
        pt := pTest1;
      end;

//Triangle fait partie de l'enveloppe convexe ou non
bTriangleBoundary := false;
if (self.IsBoundaryCH(point1)= true) and (self.IsBoundaryCH(point2)= true) and
(self.IsBoundaryCH(point3)= true) then
  begin
    bTriangleBoundary := true;
  end;
if (bTriangleBoundary = true) then

```

```

begin
  bTriangle := 1000000000;
end;
if (bTriangleBoundary = false) then
begin
  if (radT1 > radT2) then
begin
  bTriangle := radT1;
end;
  if (radT2 > radT1) then
begin
  bTriangle := radT2;
end;
end;
listAlphaTriangle.add(TFace2.Create(point1, point2, point3, pt, tetra1, tetra2, aTriangle,
bTriangle));
end;
Result := listAlphaTriangle;
end;

```

//Tri de la liste des triangles pour éliminer les doublons

```

function TMesh.TriTriangleList: TList;
var
  listAlphaTriangle: TList;
  listAlphaTriangleTrie: TList;
  triangle1, triangle2, triangle3: TEdge2;
  m, n, p: integer;
  bCompa: boolean;
begin
  listAlphaTriangle := self.IntervalTriangle;
  listAlphaTriangleTrie := TList.Create;
  triangle3 := listAlphaTriangle[0];
  listAlphaTriangleTrie.Add(triangle3);
  n := 1;
  for m := 0 to (listAlphaTriangle.Count - 1) do
  for p := 0 to (n - 1) do
  begin
    triangle1 := listAlphaTriangle[m];
    triangle2 := listAlphaTriangleTrie[p];
    bCompa := triangle1.Compare(triangle2);
    if (bCompa = true) then
    begin
      continue;
    end;
    if (bCompa = false) then
    begin
      listAlphaTriangleTrie.Add(triangle1);
      n := n + 1;
      break;
    end;
  end;
end;
Result := listAlphaTriangleTrie;
end;

```



```

//Dessin de l'alpha shape
procedure TMesh.DrawAlphaShapeTriangle;
var
  i: integer;
  triangle: TFace2;
  listAlphaTriangle: TList;
  alpha: double;
begin
  alpha := self.SetAlpha;
  if (bValid = false) then
    exit;
  listAlphaTriangle := self.TriTriangleList;
  if (bDrawAlphaShape = true) then
    begin
      for i := 0 to (listAlphaTriangle.Count - 1) do
        begin
          triangle := listAlphaTriangle[i];
          if (triangle.GetA < alpha) and (triangle.GetB > alpha) then
            begin
              TDrawTools.DrawTriangleFace_facingPT(triangle.GetPoint(1), triangle.GetPoint(2),
              triangle.GetPoint(3), triangle.GetPt, 0, 0.4, 0.5, 1);
            end;
          end;
        end;
      end;
    end;

```

```

//*****
// Description: Calcul de l'alpha shape pour les segments
// Input:      triangulation Delaunay de l'ensemble de points
// Output:     intervalles pour les segments
//*****

```

```

//Calcul des intervalles pour les segments
function TMesh.IntervalEdge: TList;
var
  listEdge: TList;
  listTetra: TList;
  listAlphaEdge: TList;
  edge: TEdge;
  point1, point2, point3, point4: TPoint;
  pt1, pt2, pt3, pt4: TPoint;
  te, tetra, tetra2: TTetra;
  i, j, k, l: integer;
  aEdge, bEdge: double;
  bEdgeBoundary, bEdgeAttach1, bEdgeAttach2: boolean;
  radTetra, radTri1, radTri2: double;
begin
  listEdge := self.CreateEdgeList;
  listAlphaEdge := TList.Create;
  listTetra := TList.Create;
  bEdgeAttach1 := false;
  bEdgeAttach2 := false;
  for i := 0 to (listEdge.Count - 1) do

```

```

begin
edge := listEdge[i];
point1 := edge.GetOrg;
point2 := edge.GetDest;
te := edge.GetAdjTetra;

//Segment attaché ou non attaché
pt1 := te.GetP(1);
pt2 := te.GetP(2);
pt3 := te.GetP(3);
pt4 := te.GetP(4);
if (((point1.x = pt1.x) and (point1.y = pt1.y) and (point1.z = pt1.z)) and ((point2.x = pt2.x)
and (point2.y = pt2.y) and (point2.z = pt2.z))) or (((point1.x = pt2.x) and (point1.y = pt2.y)
and (point1.z = pt2.z)) and ((point2.x = pt1.x) and (point2.y = pt1.y) and (point2.z =
pt1.z)))) then
begin
point3 := TPoint.Create(pt3.x, pt3.y, pt3.z, pt3.data);
point4 := TPoint.Create(pt4.x, pt4.y, pt4.z, pt4.data);
end;
if (((point1.x = pt1.x) and (point1.y = pt1.y) and (point1.z = pt1.z)) and ((point2.x = pt3.x)
and (point2.y = pt3.y) and (point2.z = pt3.z))) or (((point1.x = pt3.x) and (point1.y = pt3.y)
and (point1.z = pt3.z)) and ((point2.x = pt1.x) and (point2.y = pt1.y) and (point2.z =
pt1.z)))) then
begin
point3 := TPoint.Create(pt2.x, pt2.y, pt2.z, pt2.data);
point4 := TPoint.Create(pt4.x, pt4.y, pt4.z, pt4.data);
end;
if (((point1.x = pt1.x) and (point1.y = pt1.y) and (point1.z = pt1.z)) and ((point2.x = pt4.x)
and (point2.y = pt4.y) and (point2.z = pt4.z))) or (((point1.x = pt4.x) and (point1.y = pt4.y)
and (point1.z = pt4.z)) and ((point2.x = pt1.x) and (point2.y = pt1.y) and (point2.z =
pt1.z)))) then
begin
point3 := TPoint.Create(pt2.x, pt2.y, pt2.z, pt2.data);
point4 := TPoint.Create(pt3.x, pt3.y, pt3.z, pt3.data);
end;
if (((point1.x = pt2.x) and (point1.y = pt2.y) and (point1.z = pt2.z)) and ((point2.x = pt3.x)
and (point2.y = pt3.y) and (point2.z = pt3.z))) or (((point1.x = pt3.x) and (point1.y = pt3.y)
and (point1.z = pt3.z)) and ((point2.x = pt2.x) and (point2.y = pt2.y) and (point2.z =
pt2.z)))) then
begin
point3 := TPoint.Create(pt1.x, pt1.y, pt1.z, pt1.data);
point4 := TPoint.Create(pt4.x, pt4.y, pt4.z, pt4.data);
end;
if (((point1.x = pt2.x) and (point1.y = pt2.y) and (point1.z = pt2.z)) and ((point2.x = pt4.x)
and (point2.y = pt4.y) and (point2.z = pt4.z))) or (((point1.x = pt4.x) and (point1.y = pt4.y)
and (point1.z = pt4.z)) and ((point2.x = pt2.x) and (point2.y = pt2.y) and (point2.z =
pt2.z)))) then
begin
point3 := TPoint.Create(pt1.x, pt1.y, pt1.z, pt1.data);
point4 := TPoint.Create(pt3.x, pt3.y, pt3.z, pt3.data);
end;
if (((point1.x = pt3.x) and (point1.y = pt3.y) and (point1.z = pt3.z)) and ((point2.x = pt4.x)
and (point2.y = pt4.y) and (point2.z = pt4.z))) or (((point1.x = pt4.x) and (point1.y = pt4.y)
and (point1.z = pt4.z)) and ((point2.x = pt3.x) and (point2.y = pt3.y) and (point2.z =
pt3.z)))) then
begin

```

```

    point3 := TPoint.Create(pt1.x, pt1.y, pt1.z, pt1.data);
    point4 := TPoint.Create(pt2.x, pt2.y, pt2.z, pt2.data);
end;

bEdgeAttach1 := TAlphaTools.AttachEdge(point1, point2, point3);
bEdgeAttach2 := TAlphaTools.AttachEdge(point1, point2, point4);

radTri1 := TAlphaTools.RadiusTriangle(point1, point2, point3);
radTri2 := TAlphaTools.RadiusTriangle(point1, point2, point4);

if ((bEdgeAttach1 = true) and (bEdgeAttach2 = true)) then
  begin
    if (radTri1 < radTri2) then
      begin
        aEdge := radTri1;
      end;
    if (radTri1 > radTri2) then
      begin
        aEdge := radTri2;
      end;
    if (radTri1 = radTri2) then
      begin
        aEdge := radTri1;
      end;
    end;
  if ((bEdgeAttach1 = true) and (bEdgeAttach2 = false)) then
    begin
      aEdge := radTri1;
    end;
  if ((bEdgeAttach1 = false) and (bEdgeAttach2 = true)) then
    begin
      aEdge := radTri2;
    end;
  if ((bEdgeAttach1 = false) and (bEdgeAttach2 = false)) then
    begin
      aEdge := TAlphaTools.RadiusEdge(point1, point2);
    end;

//Connaitre tous les tetra qui contiennent le segment
listTetra.Clear;
for k := 0 to (m_lstTetra.Count - 1) do
  begin
    tetra := m_lstTetra[k];
    if ((tetra.HasEdge(point1, point2) = true) or (tetra.HasEdge(point2, point1) = true)) then
      begin
        listTetra.Add(tetra);
      end;
    end;
  end;

//Segment fait partie de l'enveloppe convexe ou non
bEdgeBoundary := false;
if (self.IsBoundaryCH(point1)= true) and (self.IsBoundaryCH(point2)= true) then
  begin
    bEdgeBoundary := true;
  end;
if (bEdgeBoundary = true) then

```

```

begin
  bEdge := 1000000000;
end;
if (bEdgeBoundary = false) then
begin
  bEdge := 0;
  for j := 0 to (listTetra.Count - 1) do
begin
  tetra2 := listTetra[j];
  radTetra := TAlphaTools.RadiusTetra(tetra2.GetP(1), tetra2.GetP(2), tetra2.GetP(3),
    tetra2.GetP(4));
  if (radTetra > bEdge) then
begin
  bEdge := radTetra;
end;
end;
end;
listAlphaEdge.Add(TEdge2.Create(point1, point2, te, aEdge, bEdge));
end;
Result := listAlphaEdge;
end;

```

//Tri de la liste des segments pour éliminer les doublons

```

function TMesh.TriEdgeList: TList;
var
  listAlphaEdge: TList;
  listAlphaEdgeTrie: TList;
  edge1, edge2, edge3: TEdge2;
  m, n, p: integer;
  bCompa: boolean;
begin
  listAlphaEdge := self.IntervalEdge;
  listAlphaEdgeTrie := TList.Create;
  if (listAlphaEdge.Count = 0) then
begin
  exit;
end;
  edge3 := listAlphaEdge[0];
  listAlphaEdgeTrie.add(edge3);
  n := 1;
  for m := 0 to (listAlphaEdge.Count - 1) do
  for p := 0 to (n - 1) do
begin
  edge1 := listAlphaEdge[m];
  edge2 := listAlphaEdgeTrie[p];
  bCompa := edge1.Compare(edge2);
  if (bCompa = true) then
begin
  continue;
end;
  if (bCompa = false) then
begin
  listAlphaEdgeTrie.Add(edge1);
  n := n + 1;
  break;
end;
end;
end;

```

```

    end;
  end;
  Result := listAlphaEdgeTrie;
end;

```

```
//Dessin de l'alpha shape
```

```
procedure TMesh.DrawAlphaShapeEdge;
```

```
var
```

```
  i: integer;
```

```
  edge: TEdge2;
```

```
  te: TTetra;
```

```
  listAlphaEdge: TList;
```

```
  alpha, aEdge, bEdge: double;
```

```
begin
```

```
  alpha := self.SetAlpha;
```

```
  if (bValid = false) then
```

```
    exit;
```

```
  listAlphaEdge := self.TriEdgeList;
```

```
  if (listAlphaEdge.Count = 0) then
```

```
    begin
```

```
      exit;
```

```
    end;
```

```
  if (bDrawAlphaShape = true) then
```

```
    begin
```

```
      for i := 0 to (listAlphaEdge.Count - 1) do
```

```
        begin
```

```
          edge := listAlphaEdge[i];
```

```
          te := edge.GetTetra;
```

```
          if (self.IsOutsideCH(te) = false) then
```

```
            begin
```

```
              if ((edge.GetA < alpha) and (edge.GetB > alpha)) then
```

```
                begin
```

```
                  TDrawTools.DrawLine(edge.GetOrg, edge.GetDest, 0, 0, 0.7, 2);
```

```
                end;
```

```
            end;
```

```
          end;
```

```
        end;
```

```
    end;
```

```

//*****
// Description: Calcul de l'alpha shape pour les points
// Input:      triangulation Delaunay de l'ensemble de points
// Output:     intervalles pour les points
//*****

```

```
//Calcul des intervalles pour les points
```

```
function TMesh.IntervalVertex: TList;
```

```
var
```

```
  listTetra: TList;
```

```
  listVertex: TList;
```

```
  listAlphaVertex: TList;
```

```
  te: TTetra;
```

```

tetra: TTetra2;
point: TPoint;
i, j, k: integer;
aTetra, bVertex: double;
begin
  listVertex := self.GetListPts;
  listTetra := TList.Create;
  listAlphaVertex := TList.Create;
  for i := 0 to (listVertex.Count - 1) do
    begin
      bVertex := 0;
      point := listVertex[i];
      if (self.IsBoundaryCH(point) = true) then
        begin
          bVertex := 1000000;
        end;
      if (self.IsBoundaryCH(point) = false) then
        begin
          //Faire une liste de tous les tetra qui contiennent le point
          for j := 0 to (m_lstTetra.Count - 1) do
            begin
              te := m_lstTetra[j];
              if (te.HasPoint(point) = true) then
                begin
                  aTetra := TAlphaTools.RadiusTetra(te.GetP(1), te.GetP(2), te.GetP(3), te.GetP(4));
                  listTetra.Add(TTetra2.Create(te.GetP(1), te.GetP(2), te.GetP(3), te.GetP(4),
                    aTetra));
                end;
            end;
          end;

          for k := 0 to (listTetra.Count - 1) do
            begin
              tetra := listTetra[k];
              if (tetra.GetAlpha > bVertex) then
                begin
                  bVertex := tetra.GetAlpha;
                end;
            end;
          end;
          listAlphaVertex.add(TPoint2.Create(point.x, point.y, point.z, point.data, bVertex));
        end;
      result := listAlphaVertex;
    end;
  end;

//Dessin de l'alpha shape
procedure TMesh.DrawAlphaShapeVertex;
var
  i, j: integer;
  point: TPoint2;
  point2: TPoint;
  listAlphaVertex: TList;
  alpha: double;
begin
  alpha := self.SetAlpha;
  listAlphaVertex := self.IntervalVertex;

```

```

if (bDrawAlphaShape = true) then
  begin
    for i := 0 to (listAlphaVertex.Count - 1) do
      begin
        point := listAlphaVertex[i];
        if (point.GetInterval < alpha) then
          begin
            point2 := TPoint.Create(point.Getx, point.Gety, point.Getz, point.GetData);
            TDrawTools.DrawNode(point2, 0.7, 0.3, 0.0, 1)
          end;
        end;
      end;
    end;
  end;
end;

//*****

/-- Calcul du rayon de la sphere circonscrite
class function TAlphaTools.RadiusTetra(a, b, c, d: TPoint): double;
var
  a4, b4, c4, d4: double;
  temp1, temp2, temp3, temp4, temp5: double;
  r2: double;
  radius: double;
begin
  a4 := (a.x*a.x) + (a.y*a.y) + (a.z*a.z);
  b4 := (b.x*b.x) + (b.y*b.y) + (b.z*b.z);
  c4 := (c.x*c.x) + (c.y*c.y) + (c.z*c.z);
  d4 := (d.x*d.x) + (d.y*d.y) + (d.z*d.z);
  temp1 := TGeomTools.Det4x4(a.y, a.z, a4, 1, b.y, b.z, b4, 1, c.y, c.z, c4, 1, d.y, d.z, d4, 1);
  temp2 := TGeomTools.Det4x4(a.x, a.z, a4, 1, b.x, b.z, b4, 1, c.x, c.z, c4, 1, d.x, d.z, d4, 1);
  temp3 := TGeomTools.Det4x4(a.x, a.y, a4, 1, b.x, b.y, b4, 1, c.x, c.y, c4, 1, d.x, d.y, d4, 1);
  temp4 := TGeomTools.Det4x4(a.x, a.y, a.z, 1, b.x, b.y, b.z, 1, c.x, c.y, c.z, 1, d.x, d.y, d.z, 1);
  temp5 := TGeomTools.Det4x4(a.x, a.y, a.z, a4, b.x, b.y, b.z, b4, c.x, c.y, c.z, c4, d.x, d.y, d.z,
    d4);
  r2 := ((temp1*temp1)+(temp2*temp2)+(temp3*temp3)+(4*temp4*temp5)) / (4*(temp4*temp4));
  radius := sqrt(r2);
  result := radius;
end;

class function TAlphaTools.RadiusTriangle(a, b, c: TPoint): double;
var
  temp1, temp2, temp3: double;
  temp4, temp5, temp6: double;
  temp7, temp8, temp9: double;
  temp10, temp11, temp12: double;
  r2: double;
  radius: double;
begin
  temp1 := TGeomTools.Det2x2(a.x, 1, b.x, 1);
  temp2 := TGeomTools.Det2x2(a.y, 1, b.y, 1);
  temp3 := TGeomTools.Det2x2(a.z, 1, b.z, 1);
  temp4 := TGeomTools.Det2x2(b.x, 1, c.x, 1);
  temp5 := TGeomTools.Det2x2(b.y, 1, c.y, 1);
  temp6 := TGeomTools.Det2x2(b.z, 1, c.z, 1);
  temp7 := TGeomTools.Det2x2(c.x, 1, a.x, 1);

```

```

temp8 := TGeomTools.Det2x2(c.y, 1, a.y, 1);
temp9 := TGeomTools.Det2x2(c.z, 1, a.z, 1);
temp10 := TGeomTools.Det3x3(a.y, a.z, 1, b.y, b.z, 1, c.y, c.z, 1);
temp11 := TGeomTools.Det3x3(a.x, a.z, 1, b.x, b.z, 1, c.x, c.z, 1);
temp12 := TGeomTools.Det3x3(a.x, a.y, 1, b.x, b.y, 1, c.x, c.y, 1);
r2 := ((sqr(temp1) + sqr(temp2) + sqr(temp3)) * (sqr(temp4) + sqr(temp5) + sqr(temp6)) *
      (sqr(temp7) + sqr(temp8) + sqr(temp9))) / (4 * (sqr(temp10) + sqr(temp11) +
      sqr(temp12)))));
radius := sqrt(r2);
result := radius;
end;

```

```

class function TAlphaTools.RadiusEdge(a, b: TPoint): double;

```

```

var

```

```

temp1: double;
temp2: double;
temp3: double;
r2: double;
radius: double;

```

```

begin

```

```

temp1 := TGeomTools.Det2x2(a.x, 1, b.x, 1);
temp2 := TGeomTools.Det2x2(a.y, 1, b.y, 1);
temp3 := TGeomTools.Det2x2(a.z, 1, b.z, 1);
r2 := (sqr(temp1) + sqr(temp2) + sqr(temp3)) / 4;
radius := sqrt(r2);
result := radius;

```

```

end;

```

```

/-- Calcul attaché/non attaché

```

```

class function TAlphaTools.AttachTriangle(a, b, c, d: TPoint): boolean;

```

```

var

```

```

temp1: double;
temp2: double;
temp3: double;
temp4: double;
temp5: double;
temp6: double;
temp7: double;
temp8: double;
b_attachTriangle: boolean;
a1, a2, a3, a4: double;
b1, b2, b3, b4: double;
c1, c2, c3, c4: double;
d1, d2, d3, d4: double;

```

```

begin

```

```

b_attachTriangle := false;
a4 := (a.x*a.x) + (a.y*a.y) + (a.z*a.z);
b4 := (b.x*b.x) + (b.y*b.y) + (b.z*b.z);
c4 := (c.x*c.x) + (c.y*c.y) + (c.z*c.z);
d4 := (d.x*d.x) + (d.y*d.y) + (d.z*d.z);
temp1 := TGeomTools.Det4x4(a.y, a.z, a4, 1, b.y, b.z, b4, 1, c.y, c.z, c4, 1, d.y, d.z, d4, 1);
temp2 := TGeomTools.Det4x4(a.x, a.z, a4, 1, b.x, b.z, b4, 1, c.x, c.z, c4, 1, d.x, d.z, d4, 1);
temp3 := TGeomTools.Det4x4(a.x, a.y, a4, 1, b.x, b.y, b4, 1, c.x, c.y, c4, 1, d.x, d.y, d4, 1);
temp4 := TGeomTools.Det4x4(a.x, a.y, a.z, 1, b.x, b.y, b.z, 1, c.x, c.y, c.z, 1, d.x, d.y, d.z, 1);
temp5 := TGeomTools.Det3x3(a.y, a.z, 1, b.y, b.z, 1, c.y, c.z, 1);

```



```

temp6 := TGeomTools.Det3x3(a.x, a.z, 1, b.x, b.z, 1, c.x, c.z, 1);
temp7 := TGeomTools.Det3x3(a.x, a.y, 1, b.x, b.y, 1, c.x, c.y, 1);
temp8 := (temp1 * temp5) + (temp2 * temp6) + (temp3 * temp7) - (2*(temp4 * temp5));
if temp8 > 0 then
  begin
    b_attachTriangle := true;
  end;
result := b_attachTriangle;
end;

```

```

class function TAlphaTools.AttachEdge(a, b, c: TPoint): boolean;

```

```

var
  temp1: double;
  temp2: double;
  temp3: double;
  temp4: double;
  temp5: double;
  temp6: double;
  temp7: double;
  temp8: double;
  temp9: double;
  temp10: double;
  temp11: double;
  temp12: double;
  b_attachEdge: boolean;
begin
  b_attachEdge := false;
  temp1 := TGeomTools.Det2x2(a.x, 1, b.x, 1);
  temp2 := TGeomTools.Det2x2(a.y, 1, b.y, 1);
  temp3 := TGeomTools.Det2x2(a.z, 1, b.z, 1);
  temp4 := (temp1*temp1) + (temp2*temp2) + (temp3*temp3);
  temp5 := TGeomTools.Det2x2(a.x, 1, c.x, 1);
  temp6 := TGeomTools.Det2x2(a.y, 1, c.y, 1);
  temp7 := TGeomTools.Det2x2(a.z, 1, c.z, 1);
  temp8 := TGeomTools.Det2x2(b.x, 1, c.x, 1);
  temp9 := TGeomTools.Det2x2(b.y, 1, c.y, 1);
  temp10 := TGeomTools.Det2x2(b.z, 1, c.z, 1);
  temp11 := ((temp5 + temp8)*(temp5 + temp8)) + ((temp6 + temp9)*(temp6 + temp9)) + ((temp7
    + temp10)*(temp7 + temp10));
  temp12 := temp4 - temp11;
  if temp12 > 0 then
    begin
      b_attachEdge := true;
    end;
  result := b_attachEdge;
end;

```

Annexe C Extrait du jeu de données utilisé pour la représentation 3D

Jeu de données acoustiques décrivant les dynamiques d'agrégation et de désagrégation d'un groupe de morues de l'Atlantique (*Gadus morhua*) en avril 1999 à Smith Sound, sur la côte Est de Terre-Neuve. Ces données ont été acquises à haute résolution à l'aide d'un échosondeur à faisceau divisé embarqué sur un bateau (*NGCC Teleost*) sillonnant les eaux aléatoirement au-dessus de l'agrégation observée.

Tableau C.1 - Extrait des données acoustiques.

	Region ID	Ping M	Dist M	Date M	Time M
1	69032	11	15.046638	19990609	50:52.9
2	69033	7	15.046638	19990609	50:44.9
3	69034	13	15.046638	19990609	50:56.9
4	69038	18	15.051886	19990609	51:06.9
5	69040	22	15.053298	19990609	51:14.9
6	69042	25	15.053298	19990609	51:20.9
7	69043	34	15.059958	19990609	51:38.9
8	69044	52	15.076618	19990609	52:14.9
9	69045	62	15.076618	19990609	52:34.9
10	69047	95	15.099937	19990609	53:40.9
11	69049	111	15.123257	19990609	54:12.9
12	69050	119	15.129917	19990609	54:28.9
13	69052	155	15.169897	19990609	55:40.9
14	69054	152	15.169897	19990609	55:34.9
15	69055	159	15.176557	19990609	55:48.9

	Lat M	Lon M	Num targets	TS mean	TS max
1	48.2421667	-53.4926667	3	-41.746315	-41.160484
2	48.2421667	-53.4926667	12	-27.346653	-21.55826
3	48.2421667	-53.4926667	6	-28.855429	-28.213844
4	48.2421667	-53.492798	8	-47.68977	-46.887112
5	48.2421667	-53.4928333	7	-20.954063	-16.713558
6	48.2421667	-53.4928333	10	-25.714802	-20.523468
7	48.2421667	-53.493	9	-24.520398	-21.664091
8	48.242	-53.4931667	3	-22.825641	-21.828716

9	48.242	-53.4931667	5	-30.239964	-26.355925
10	48.2418333	-53.4935	11	-34.175366	-30.683231
11	48.2416667	-53.4938333	6	-24.968044	-22.534254
12	48.2416667	-53.494	18	-21.341101	-18.830175
13	48.2413333	-53.4945	4	-39.892232	-37.315296
14	48.2413333	-53.4945	10	-43.045715	-42.418697
15	48.2413333	-53.4946667	3	-28.857762	-28.272638

	TS min	Target range mean	Target range max	Target range min	Speed 2D mean unsmoothed
1	-42.077686	261.399994	261.399994	261.399994	0
2	-41.089931	267.925013	268.300018	267.300018	0.301636
3	-30.001209	270.78334	271.200012	270.5	0.090002
4	-48.392262	179.012505	180	177.900009	0.70935
5	-37.268261	278.142862	278.399994	278	0.213983
6	-36.809662	280.82001	281.700012	280.200012	0.602078
7	-28.931143	269.355564	269.600006	269.100006	0.545058
8	-23.86302	269.966675	270.200012	269.800018	0.099998
9	-41.501495	269.840002	270.100006	269.399994	0.050001
10	-41.348629	209.509093	211.199997	208.199997	0.739983
11	-30.389256	263.600006	263.800018	263.399994	1.064955
12	-26.579346	269.094452	270.100006	268.800018	1.013664
13	-42.136482	251.475006	251.600006	251.400009	0.066663
14	-43.782738	89.880002	90.400002	89.599998	0.061111
15	-30.22463	270.633331	270.899994	270.399994	0.41108

	Speed 4D mean unsmoothed	Speed 2D max unsmoothed	Speed 4D max unsmoothed	Direction horizontal	Direction vertical
1	0.811	0	1.207738	236.154975	0.494208
2	0.8972	1.710623	1.445596	256.312287	-12.998099
3	0.8666	0.149994	1.819441	237.620471	5.124625
4	0.855	4.877809	3.282901	5.486818	-20.124498
5	0.7979	1.311873	2.373795	74.977955	-2.542856
6	1.1911	1.239847	3.476944	198.615093	2.738659
7	1.3022	1.681973	1.935775	115.32926	5.298378
8	1.1062	0.150009	1.255701	271.374402	35.511815
9	0.9492	0.050003	1.71149	95.824739	5.238224
10	1.2018	4.00438	2.600837	45.004625	26.577548
11	1.474	4.289229	3.105995	251.417976	-0.575349
12	1.3788	6.766769	5.862775	155.541254	1.647712
13	0.8357	0.099998	1.409858	101.291162	-2.649896
14	0.514	0.150002	0.712915	74.875914	-3.778777
15	0.6306	0.722154	1.1612	105.033207	7.883448

	Fish track change in range	Time in beam	Tortuosity 2D	Tortuosity 3D	Distance 2D unsmoothed
1	0	4	9.90E+37	1.028261	0
2	0.899994	26	1.137438	5.04993	7.842524
3	-0.700012	10	1.285727	1.856039	0.900024
4	2.099991	20	1.129921	3.228847	14.187004
5	0	16	1.304903	5.233094	3.423733
6	-1.5	42	1.019533	1.457161	25.287261
7	0.100006	24	1.056729	5.485655	13.08138
8	-0.399994	4	1	6.101919	0.399994
9	-0.700012	14	1	1.149615	0.700012
10	-3	20	1.16192	3.71054	14.799665
11	0.300018	12	1.032059	3.175139	12.779456
12	-0.799988	44	1.441745	2.192013	44.601199
13	0.199997	6	1.999924	1.740342	0.399979
14	0.699997	18	1.571444	1.077072	1.100006
15	-0.5	4	1.097181	1.073761	1.644321

	Distance 3D unsmoothed	Target depth mean	Target depth max	Target depth min
1	3.244	261.399994	261.399994	261.399994
2	23.3265	267.925013	268.300018	267.300018
3	8.6659	270.78334	271.200012	270.5
4	17.1004	179.012505	180	177.900009
5	12.7665	278.142862	278.399994	278
6	50.0248	280.82001	281.700012	280.200012
7	31.2519	269.355564	269.600006	269.100006
8	4.4248	269.966675	270.200012	269.800018
9	13.2895	269.840002	270.100006	269.399994
10	24.035	209.509093	211.199997	208.199997
11	17.6883	263.600006	263.800018	263.399994
12	60.6658	269.094452	270.100006	268.800018
13	5.0144	251.475006	251.600006	251.400009
14	9.2524	89.880002	90.400002	89.599998
15	2.5224	270.633331	270.899994	270.399994