



**HAL**  
open science

# Propagation de fronts et p-laplacien normalisé sur graphes : algorithmes et applications au traitement d'images et de données.

Xavier Desquesnes

► **To cite this version:**

Xavier Desquesnes. Propagation de fronts et p-laplacien normalisé sur graphes : algorithmes et applications au traitement d'images et de données.. Traitement des images [eess.IV]. Université de Caen, 2012. Français. NNT : . tel-00773434

**HAL Id: tel-00773434**

**<https://theses.hal.science/tel-00773434>**

Submitted on 14 Jan 2013

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

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

# UNIVERSITÉ de CAEN BASSE-NORMANDIE

U.F.R. Sciences

ÉCOLE DOCTORALE SIMEM

## THÈSE

*Présentée par*

**M. Xavier Desquesnes**

*Et soutenue*

Le 7 décembre 2012

*En vue de l'obtention du*

**DOCTORAT de L'UNIVERSITÉ de CAEN**

Spécialité informatique

*Arrêté du 07 août 2006*

# Propagation de fronts et $p$ -laplacien normalisé sur graphes : algorithmes et applications au traitement d'images et de données.

Devant le jury composé de

M. Rachid DERICHE	Directeur de recherche	INIRIA Sophia-Antipolis	(rapporteur)
M. Laurent NAJMAN	Professeur	ESIEE	(rapporteur)
M. Mahmoud MELKEMI	Professeur	Université de Haute alsace	(examineur)
M. Régis MONNEAU	Professeur	ENPC	(examineur)
M. Olivier LÉZORAY	Professeur	Université de Caen	(examineur)
M. Abderrahim ELMOATAZ	Professeur	Université de Caen	(directeur de thèse)





## Remerciements

Les travaux de recherche présentés dans ce manuscrit ont été initiés par Abderrahim Elmoataz qui m'a encadré durant les trois années de cette thèse. Je le remercie pour m'avoir fait découvrir le monde de la recherche. Je le remercie, ainsi qu'Olivier Lézoray, pour leur investissement et leurs conseils avisés qui m'ont permis de mener à bien ces trois années de doctorat. J'espère avoir l'opportunité de poursuivre cette collaboration dans le futur et continuer à profiter de leur expérience.

Je tiens à remercier Rachid Deriche, Olivier Lézoray, Mahmoud Melkemi, Régis Monneau et Laurent Najman d'avoir accepté de faire partie de mon jury. Je remercie tout particulièrement Rachid Deriche et Laurent Najman pour avoir accepté de rapporter ma thèse. Je les remercie également d'avoir consacré de leur temps à la lecture de mon manuscrit.

Je remercie également les membres de l'association Cœur et Cancer ainsi que le conseil général de la région Basse-Normandie pour leur soutien financier, sans lequel cette thèse n'aurait jamais pu commencer.

Je remercie tous les membres du GREYC, permanents, doctorants et l'équipe administrative pour leur accueil au sein du laboratoire durant les 4 années de mon stage de master d'abord puis de ma thèse. Je remercie tout particulièrement les membres du Greyc Poker Tour et les doctorants de l'équipe image pour les moments de détente que nous avons partagé ensemble.

Je remercie également Benoît, Charlotte, Olivier et Sébastien pour leur relecture attentive du manuscrit.

Merci finalement à tous mes proches, mes amis et ma famille qui ont été présents et mon soutenu durant ces trois années.



# Table des matières

<b>Introduction générale</b>	<b>1</b>
<b>I Graphes et opérateurs</b>	<b>11</b>
<b>1 Graphes</b>	<b>13</b>
1.1 Définitions et notations . . . . .	13
1.2 Types de graphes et construction . . . . .	15
1.2.1 Métrique, Similarité et Fonction de poids . . . . .	16
1.2.2 Domaines non organisés . . . . .	17
1.2.3 Domaines organisés . . . . .	19
<b>2 Calcul discret sur graphe</b>	<b>25</b>
2.1 Opérateurs sur graphe . . . . .	26
2.1.1 Fonctions définies sur graphe . . . . .	26
2.1.2 Opérateur de différence . . . . .	27
2.1.3 Opérateur de dérivée . . . . .	28
2.1.4 Opérateur gradient . . . . .	28
2.1.5 Opérateurs laplaciens . . . . .	29
2.2 Opérateurs directionnels . . . . .	31
2.2.1 Opérateurs de différences directionnelles . . . . .	32
2.2.2 Opérateurs gradients directionnels . . . . .	32
2.3 Opérateurs moyenneurs non-locaux . . . . .	34
2.3.1 Régularisation par $p$ -laplacien et filtre moyeneur non-local	34
2.3.2 EdP morphologiques et filtres moyenneurs non-locaux . .	36
2.4 Conclusion . . . . .	39

<b>II Une nouvelle famille d'opérateurs <math>p</math>-laplaciens</b>	<b>41</b>
Introduction	43
<b>3 L'opérateur laplacien infini non-local</b>	<b>45</b>
3.1 Le laplacien infini continu . . . . .	46
3.1.1 Relation avec les jeux de type Tug-of-War . . . . .	47
3.2 Le laplacien infini sur graphe . . . . .	49
3.2.1 Relation avec le laplacien infini continu . . . . .	49
3.2.2 Relation avec la limite du $p$ -laplacien . . . . .	50
3.2.3 Interprétation morphologique . . . . .	51
3.3 Problème de Dirichlet . . . . .	52
3.4 Expérimentations . . . . .	55
3.4.1 Inpainting d'images . . . . .	55
3.4.2 Segmentation semi-supervisée d'images et classification de données . . . . .	57
3.5 Conclusion . . . . .	58
<b>4 L'opérateur <math>p</math>-laplacien normalisé non-local</b>	<b>61</b>
4.1 Définition . . . . .	62
4.2 Problème de Dirichlet . . . . .	64
4.2.1 Étude de l'existence et de l'unicité . . . . .	65
4.3 Diffusion sur graphe par $p$ -laplacien . . . . .	67
4.4 Expérimentations . . . . .	68
4.4.1 Exemple jouet . . . . .	68
4.4.2 Restauration et simplification d'images . . . . .	68
4.4.3 Interpolation . . . . .	71
4.5 Conclusion . . . . .	78
<b>III Équations de propagation de fronts sur graphe</b>	<b>81</b>
Introduction	83
<b>5 Équation de propagation d'un front</b>	<b>87</b>
5.1 Équation du mouvement . . . . .	88
5.2 Approche par ensembles de niveaux . . . . .	89

5.3	Schémas numériques . . . . .	91
5.4	Algorithmes . . . . .	93
<b>6</b>	<b>Formulation discrète : Équation non-stationnaire</b>	<b>99</b>
6.1	Représentation discrète d'un front . . . . .	100
6.2	Analogie avec la méthode des ensembles de niveaux . . . . .	101
6.2.1	Interprétation morphologique . . . . .	102
6.2.2	Adaptation discrète . . . . .	103
6.2.3	Relation avec les schémas numériques classiques . . . . .	104
6.3	Approximation de la solution . . . . .	105
6.3.1	Cas d'une vitesse constante dans le temps . . . . .	106
6.3.2	Extension pour des vitesses dépendant du temps . . . . .	109
6.4	Algorithme . . . . .	110
6.4.1	Optimisation . . . . .	112
6.5	Expérimentations . . . . .	113
6.5.1	Exemple jouet . . . . .	113
6.5.2	Évolution selon la courbure moyenne . . . . .	115
6.5.3	Contours actifs . . . . .	118
6.6	Conclusion . . . . .	118
<b>7</b>	<b>Équation stationnaire restreinte au cas d'une vitesse positive</b>	<b>121</b>
7.1	Équation eikonale discrète . . . . .	122
7.2	Existence et unicité de la solution . . . . .	123
7.3	Schéma de résolution local . . . . .	126
7.4	Schéma de résolution global . . . . .	130
7.5	Algorithme . . . . .	135
7.6	Relation avec les schémas numériques classiques . . . . .	139
7.7	Expérimentations . . . . .	139
7.7.1	Calcul de distances géodésiques . . . . .	140
7.8	Conclusion . . . . .	144
<b>8</b>	<b>Généralisation à la propagation simultanée de plusieurs fronts</b>	<b>147</b>
8.1	Définitions et notations . . . . .	148
8.2	Équation de propagation du front global . . . . .	149
8.3	Schémas de résolution et algorithme . . . . .	151
8.3.1	Schéma de résolution local . . . . .	151

8.3.2	Schéma de résolution global . . . . .	153
8.3.3	Algorithme . . . . .	154
8.4	Expérimentations . . . . .	154
8.4.1	Exemple jouet . . . . .	154
8.4.2	Partitionnement et simplification de graphes . . . . .	156
8.4.3	Segmentation semi-supervisée d'images . . . . .	159
8.4.4	Classification semi-supervisée de données . . . . .	165
8.5	Conclusion . . . . .	166
<b>9</b>	<b>Généralisation sans restrictions sur le signe de la vitesse</b>	<b>169</b>
9.1	Cas d'un front unique . . . . .	170
9.1.1	Relation entre propagation interne et externe . . . . .	170
9.1.2	Représentation d'un front comme deux fronts . . . . .	172
9.2	Extension au cas de fronts multiples . . . . .	173
9.2.1	Définition du front global et des fronts secondaires . . . . .	175
9.2.2	Schémas de résolution et algorithme . . . . .	177
9.3	Expérimentations . . . . .	181
9.3.1	Exemple jouet . . . . .	181
9.3.2	Contours actifs . . . . .	181
9.4	Conclusion . . . . .	186
<b>IV Application à l'aide au diagnostic en cytopathologie</b>		<b>187</b>
<b>Contexte</b>		<b>189</b>
<b>10 Schéma général</b>		<b>193</b>
10.1	Extraction des noyaux . . . . .	195
10.1.1	Prétraitement . . . . .	195
10.1.2	Segmentation des noyaux . . . . .	197
10.1.3	Post-traitement . . . . .	198
10.2	Extraction des caractéristiques . . . . .	200
10.2.1	Caractéristiques de forme . . . . .	200
10.2.2	Caractéristiques de texture . . . . .	201
10.3	Classification des noyaux . . . . .	202

<b>11</b>	<b>Antarctic</b>	<b>205</b>
11.1	Un outil automatique pour l'aide au diagnostic . . . . .	206
11.2	Une base de connaissances interactive et collaborative . . . . .	209
11.3	Conclusion . . . . .	212
	<b>Conclusion générale</b>	<b>213</b>
	Liste des publications	221
	Bibliographie	223
	Liste des figures	235
	Liste des algorithmes	239
	Index	241





# Introduction générale



---

Les récentes avancées technologiques en termes d'acquisitions et de simulations numériques, liées à tous les domaines scientifiques, engendrent une quantité phénoménale de données. Parmi cette multitude de données, nous pouvons citer les images et vidéos, les maillages, les nuages de points, les réseaux, les bases de données, etc. Ces données sont d'une manière générale de grandes dimensions, de tous types et proviennent de différents domaines d'application comme l'imagerie numérique (médicale, artistique ou quotidienne), les réseaux complexes (sociaux, biologiques ou informatiques), ou encore la bio-informatique.

Si une partie non négligeable de ces données sont structurées ou définies comme des fonctions sur des domaines structurés, comme c'est le cas des images et des signaux discrets, de plus en plus d'applications nécessitent aujourd'hui de manipuler et traiter des données définies sur des domaines irréguliers ou complexes (au sens de la topologie). Ces données peuvent être définies sur des variétés (comme une sphère) ou d'autres domaines irréguliers (comme des maillages). Elles peuvent également être définies sur des structures de type réseaux (comme les données issues des réseaux sociaux) ou encore comme des nuages de points plongés dans un espace de grande dimension (comme des collections de vecteurs).

Toutes ces données, qu'elles soient organisées ou non, peuvent être communément représentées par des graphes dont les sommets représentent les données et dont les arêtes représentent les interactions entre ces données. Ces interactions pouvant dépendre de la géométrie des données mais aussi d'autres mesures de similarités, un même ensemble de données peut donc être représenté par plusieurs graphes de constructions distinctes, chacune portant un type d'interaction différent. Notamment dans le cas des images, où selon la construction du graphe, il est possible de mettre en avant les interactions locales ou non-locales des données, ce qui permet de développer ou retrouver aisément des outils pour le traitement local et non-local des images [Buades et al. 08, Gilboa et Osher 08].

L'exploitation de ces données pour en extraire des connaissances est devenu aujourd'hui un défi majeur pour les communautés de traitement d'images et de données. Par conséquent, il est primordial d'adapter des outils existants développés dans le cadre de données définies sur des espaces euclidiens (comme les images et les signaux) et dont l'efficacité n'est plus à prouver, au domaine de représentation commun de ces données que sont les graphes.

Historiquement, les approches classiques pour le traitement des graphes sont principalement issues de la théorie des graphes et peuvent être réparties en

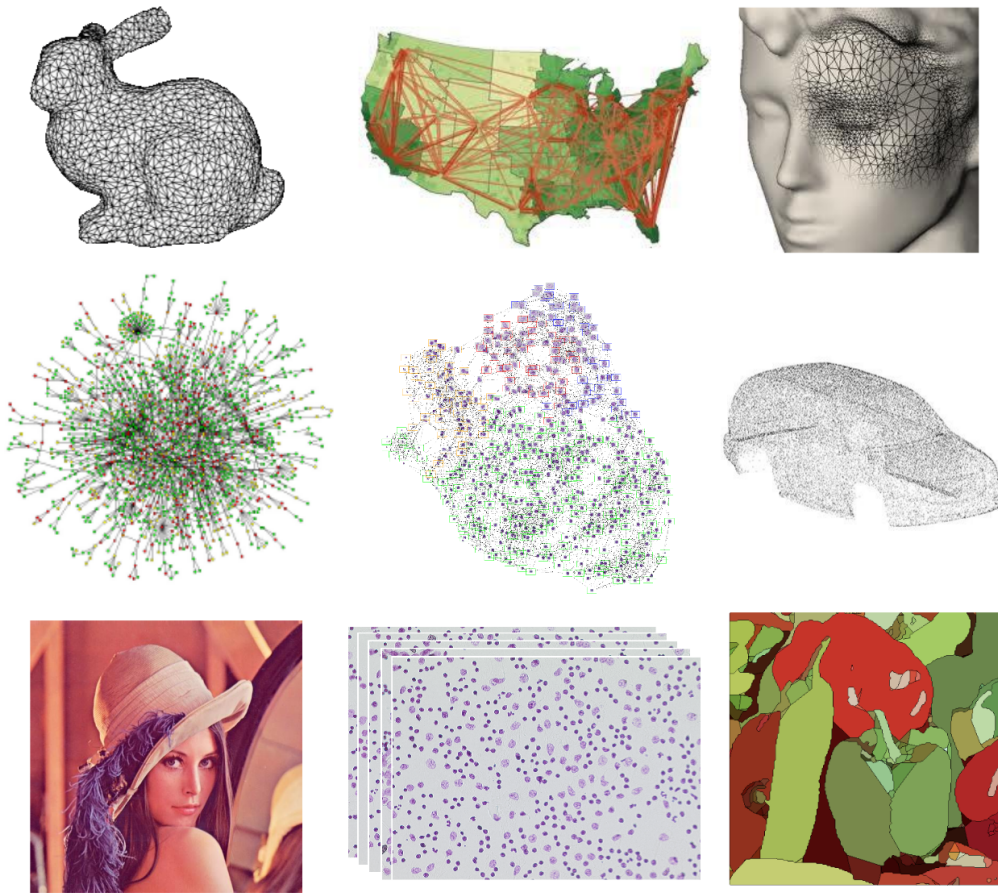


FIGURE 1 – Exemples de données pouvant être représentées par des graphes pondérés. Que ce soit leur représentation naturelle, comme les réseaux, les maillages et les images, ou non, comme les nuages de points ou les données non organisées.

deux catégories. La première regroupe les méthodes basées sur la minimisation d'énergies qui sont principalement appliquées à la segmentation semi-supervisée d'images. Nous pouvons citer les coupures de graphes (graph cut) [Boykov et Jolly 01], les marches aléatoires (random walk) [Grady 06], ou plus récemment le Power Watershed [Couprie et al. 11b]. La seconde catégorie quand à elle regroupe les techniques reposant sur la théorie spectrale des graphes [Chung 97]. Ces techniques ont montré leur efficacité en traitement d'images pour le filtrage [Hein et Maier 06] ou la segmentation [Shi et Malik 00] mais également en classification de données [Luxburg 07] ou pour l'extraction de communautés dans des réseaux complexes [Kurucz et al. 09] et bien d'autres applications.

---

Ces dernières années, deux autres approches, visant à transposer des outils développés pour le traitement du signal et des images au traitement des graphes, ont connu un intérêt croissant : l'une reposant sur la généralisation des ondelettes aux graphes, et l'autre basée sur la transcription des Équations aux Dérivées Partielles (EDP) sur graphes. Concernant l'approche basée sur les ondelettes, nous pouvons citer le travail de [Coifman et Maggioni 6b] sur la généralisation des ondelettes au graphes, ou celui de [Jansen et Oonincx 05] sur les méthodes multi-échelles ou encore celui de [Hammond et al. 11] sur les transformées spectrales.

L'approche basée sur la transcription des EDP consiste quand à elle à utiliser le calcul discret pour reformuler des problèmes d'EDP sur des graphes. Plusieurs formalismes ont été développés pour effectuer ces transcriptions, et nous pouvons citer les travaux de [Hirani 03] ou [Grady et Polimeni 10] parmi plus représentatifs ou encore le formalisme des Équations aux différences Partielles (EdP), qui à notre sens est le plus simple et intuitif d'entre eux.

Conceptuellement, les EdP sont des équations qui miment le comportement des EDP sur le domaine des graphes, en remplaçant l'opérateur dérivée par un opérateur de différence défini sur graphe. Historiquement, ces équations ont tout d'abord été introduites dans l'article de [Courant et al. 28]. Dans leur article, les auteurs introduisent la méthode des différences finies comme un moyen simple pour manipuler numériquement les EDP. Plus récemment, l'étude de ces EdP s'est généralisée jusqu'à ce qu'elles deviennent un sujet d'intérêt en elles mêmes [Bensoussan et Menaldi 03, Neuberger 06, Park et Chung 11]. Si la majorité des formulations pour les EdP ont été proposées à partir de discrétisations d'EDP sur des structures particulières de graphes (comme des grilles), leur utilisation sur des graphes de topologie arbitraire pour le traitement des images et des données suscite elle aussi un intérêt croissant. Nous pouvons citer les travaux de [Elmoataz et al. 08a] ou [Grady et Polimeni 10], ainsi que leurs références connexes. Parmi celles ci, les travaux de [Chan et al. 01] pour le traitement d'image, avec l'introduction de la variation totale pour des filtres digitaux (analogue discrète de la variation totale sur des graphes non pondérés) ou bien [Zhou et Schölkopf 06] qui ont également appliqué la variation totale sur graphe à la classification semi supervisée, présentent un intérêt particulier.

Suivant la même idée, [Bougleux et al. 07, Lézoray et al. 07, Elmoataz et al. 08a] ont introduit un cadre pour le calcul non-local sur graphe qui est le pendant discret du calcul non-local continu, en définissant l'analogue discret de la divergence, une famille de gradients directionnels, ou encore l'opérateur  $p$ -laplacien. Ce cadre bénéficiant simultanément de la structure des graphes et du formalisme des EDP, il permet de transcrire de nombreuses EDP et

méthodes variationnelles sur graphe pour obtenir des EdP dont les coefficients dépendent des données. En particulier, ce cadre a permis de revisiter des problèmes variationnels, comme la régularisation discrète non-locale sur des graphes pondérés qui a conduit à une nouvelle famille discrète et semi-discrète de processus de diffusion basés sur les opérateurs  $p$ -laplaciens. Ces processus, paramétrés par la structure des graphes (topologie et géométrie) et par le degré  $p$  de lissage, ont montré leur intérêt pour diverses applications de filtrage comme le débruitage ou la simplification. De plus, de par leur définition sur des graphes où la différence entre local et non-local tient à la construction du graphe, les processus définis au travers de ce cadre recouvrent les processus de régularisation locaux et non-locaux définis en continu [Gilboa et Osher 08]. Plus récemment, à partir de l'introduction de gradients directionnels, [Ta et al. 08a, Ta et al. 11] ont présentés des EdP décrivant des processus morphologiques sur graphe, interprétés comme des transcriptions des EDP morphologiques continues comme la dilatation et l'érosion.

## Contributions principales

Suivant la ligne de recherche développée par [Bogleux et al. 07, Elmoataz et al. 08a, Ta et al. 11], les contributions de ce manuscrit exploitent le cadre des EdP sur graphe pour transcrire des outils mathématiques continus et étudier leur application au traitement des données discrètes.

Ces contributions sont les suivantes :

- Motivés par la volonté d'étendre les applications du laplacien infini du domaine des images à celui des données non organisées, nous proposons une adaptation de cet opérateur pour les graphes de topologie arbitraire.
- Cette adaptation nous conduit à des EdP à coefficients variables (dépendant des données), qui peuvent être considérées comme des versions non-locales de l'équation du laplacien infini que nous présentons. Nous présentons aussi une preuve de l'existence et de l'unicité des solutions de ces équations.
- À partir de cette adaptation, nous proposons ensuite une adaptation de l'opérateur  $p$ -laplacien normalisé (ou laplacien des jeux) sur graphe.
- Cette adaptation nous permet d'introduire une nouvelle classe d'opérateurs  $p$ -laplacien qui interpolent entre le laplacien et le laplacien infini.
- Nous proposons ensuite une adaptation de l'équation du mouvement d'un front, définie par l'approche des ensembles de niveaux, au cas d'un front évoluant sur un graphe.
- Nous étudions ensuite la version stationnaire de cette adaptation, dans le cas où la vitesse du front est toujours positive. Nous présentons une

---

preuve de l'existence et l'unicité de cette équation, ainsi qu'un algorithme efficace de résolution

- Nous étendons ensuite la formulation et l'algorithme associé de notre équation du mouvement d'un front, pour considérer le cas où plusieurs fronts évoluent simultanément sur un même graphe. Puis nous les étendons une seconde fois pour considérer le cas où les fronts évoluent sans restriction sur le signe de leur vitesse.
- Enfin, nous présentons une application concrète utilisant chacune de nos contributions, dans le contexte de l'aide au diagnostic du cancer en microscopie optique.

## Plan du manuscrit

Ce manuscrit est divisé en quatre parties indépendantes présentant les travaux réalisés pendant cette thèse, hormis la première partie qui rappelle les notations et définitions essentielles à la compréhension de l'ensemble de ce manuscrit.

### Première partie : Graphes et opérateurs

Dans cette première partie, nous rappelons les différentes notions, notations et définitions nécessaires à la bonne compréhension de la suite du manuscrit. Tout d'abord, nous rappelons succinctement les notations et définitions usuelles de la théorie des graphes et présentons les différents types de graphes rencontrés dans ce manuscrit. Puis, nous rappelons les opérateurs discrets définis sur graphe par [Bougleux et al. 07, Elmoataz et al. 08a], incluant les opérateurs différence, dérivée, gradient et  $p$ -laplacien, ainsi que les opérateurs directionnels introduits par [Ta et al. 08a]. L'ensemble de ces opérateurs, basés sur des équations aux différences partielles, seront les constituants principaux des différentes transpositions et approches discrètes que nous introduisons dans les différentes parties de ce manuscrit.

### Seconde partie : Une nouvelle famille d'opérateurs $p$ -laplaciens

Dans cette seconde partie, nous proposons une adaptation des opérateurs laplacien infini et  $p$ -laplacien normalisé sur le domaine des graphes, s'appuyant sur le cadre des EdP sur graphe. Cette adaptation nous permet d'introduire une nouvelle classe d'opérateurs  $p$ -laplaciens définis comme une combinaison de gradients morphologiques sur graphes. Ces opérateurs interpolent entre les opérateurs laplacien et infini laplacien sur graphes et sont obtenus par



discrétisation sur graphe du  $p$ -laplacien normalisé (ou laplacien des jeux). Cette adaptation reposant sur les gradients morphologiques discrets permet de définir des équations aux différences finies sur graphe avec des coefficients variables. L'étude de ces équations, sous l'éclairage des connexions avec les EDP continues correspondantes, est un sujet d'intérêt propre. Nous étudions ensuite le problème de Dirichlet associé à ces équations et prouvons l'existence et l'unicité des solutions de ces problèmes. Récemment, certains jeux aléatoire à somme nulle (Tug-of-War games) ont été utilisés en connexion avec des problèmes d'EDP [Manfredi et al. a]. Nous montrons que les fonctions à valeurs réelles de ces jeux coïncident avec les solutions de ces équations de  $p$ -laplaciens. Finalement, nous proposons d'utiliser ces opérateurs comme un cadre pour résoudre de nombreux problèmes en traitement d'images et de données.

### **Troisième partie : Équations de propagation de fronts sur graphes**

Dans cette troisième partie, nous présentons notre adaptation discrète sur graphe de l'équation du mouvement d'un front, par l'approche des ensembles de niveaux. Cette adaptation donne lieu à une nouvelle équation discrète dont nous étudions les versions non-stationnaires et stationnaires, pour lesquelles nous proposons des schémas numériques, la preuve de l'existence et de l'unicité de la solution, et un algorithme de résolution rapide. Puis, nous étendons la formulation stationnaire de cette équation au cas où plusieurs fronts évoluent simultanément sur un même graphe (selon certaines conditions). Enfin, nous étendons une nouvelle fois cette formulation pour traiter le cas le plus général où plusieurs fronts évoluent sur un graphe sans restrictions sur le signe de leur vitesse.

Les différentes formulations et algorithmes proposés dans cette partie permettent d'étendre une vaste gamme d'applications, du traitement des images définies dans un espace euclidien, aux données représentées par des graphes. Parmi ces applications, nous présentons divers résultats de segmentation d'images reposant sur différentes représentations des images par des graphes (graphes d'adjacence de régions, construction non-locale), mais également des résultats en traitement de maillages et classification de données.

### **Quatrième partie : Application à l'aide au diagnostic**

Dans cette quatrième et dernière partie, nous proposons une application concrète des EdP présentées dans les deux précédentes parties de ce manuscrit pour l'aide au diagnostic en cytopathologie. Nous présentons tout d'abord un schéma général du processus d'aide au diagnostic appliqué à la microscopie quantitative. Ce schéma propose une approche unifiée de l'extraction et de la

---

classification des noyaux cellulaires reposant sur l'utilisation des graphes et du formalisme des EdP.

Enfin, nous présentons le logiciel Antarctic développé au cours de cette thèse et destiné à l'aide au diagnostic en cytopathologie. Ce logiciel doté d'une architecture client-serveur se présente sous la forme d'une application web et regroupe deux outils dont les utilisations sont la plupart du temps interdépendantes. D'une part, un outil automatique dédié à l'aide au diagnostic, et offrant à un pathologiste la possibilité d'obtenir à distance une classification des cellules d'une lame cytologique téléchargée sur le serveur. De l'autre, un outil collaboratif de gestion et de consultation d'une base de connaissances, qui s'étoffe au fur et à mesure des nouvelles lames ajoutées ainsi que des contributions (sous la forme d'expertises de cellules) des pathologistes.



Première partie  
Graphes et opérateurs



# Chapitre 1

## Graphes

### Sommaire

---

<b>1.1 Définitions et notations</b> . . . . .	<b>13</b>
<b>1.2 Types de graphes et construction</b> . . . . .	<b>15</b>
1.2.1 Métrique, Similarité et Fonction de poids . . . . .	16
1.2.2 Domaines non organisés . . . . .	17
1.2.3 Domaines organisés . . . . .	19

---

L'intégralité des notions et travaux présentés dans cette thèse s'appuient sur le concept de graphe pondéré. Dans ce premier chapitre, nous introduisons les différents définitions, notations et concepts inhérents à ces graphes.

### 1.1 Définitions et notations

Un graphe pondéré, noté  $\mathcal{G} = (V, E, w)$ , est constitué d'un ensemble fini de sommets, noté  $V$ , d'un ensemble d'arêtes pondérées, noté  $E$  et d'une fonction de poids, notée  $w$ .

**Remarque.** *Dans ce manuscrit nous ne considérons que des graphes simples (sans arêtes multiples et sans boucles), pondérés [Diestel 05] et dont les arêtes sont toutes symétriques.*

#### Définition d'un sommet

De manière générale, un sommet  $u \in V$  est une abstraction d'un élément de base de la structure de donnée représentée par le graphe. Ce peut-être un

pixel, un point de réseau ou de maillage ou bien encore une entrée dans une base de donnée. La relation entre la structure de données et l'ensemble des sommets est donnée par :

$$V \subseteq \Omega \subset \mathbb{R}^m, \quad (1.1)$$

où  $\Omega$  désigne le domaine discret de la structure de données.

### Définition d'une arête

Une arête est une connexion entre deux sommets  $u$  et  $v$  d'un graphe, symbolisant une interaction entre les éléments représentés par ces sommets. Les deux sommets sont alors dit adjacents et on note  $u \sim v$ . Une arête reliant deux sommets  $u$  et  $v$  est notée  $(u, v)$ , elle est dite incidente aux sommets  $u$  et  $v$ .

L'ensemble  $E$  des arêtes de  $\mathcal{G}$  est un sous ensemble de  $V \times V$  et est défini par :

$$E = \{(u, v) \in V \times V \mid u \sim v \text{ et } u \neq v\}. \quad (1.2)$$

La propriété de symétrie implique que si  $(u, v) \in E$ , alors  $(v, u) \in E$ .

Par définition, un sommet ne peut être connecté à lui même. Le graphe  $\mathcal{G}$  est donc considéré sans boucles.

### Voisinage d'un sommet

Le voisinage d'un sommet  $u$ , noté  $\mathcal{N}(u)$  désigne l'ensemble des sommets adjacents à  $u$  et est défini par :

$$\mathcal{N}(u) = \{v \in V \mid (u, v) \in E\}. \quad (1.3)$$

Dans le cas d'un graphe complet, on a  $\mathcal{N}(u) = V - \{u\}$ ,  $\forall u \in V$ .

### Fonction de poids

La fonction de poids  $w : V \times V \rightarrow [0, 1]$ , associe à chaque couple de sommets  $\{u, v\} \in V \times V$  une valeur de similarité représentant la quantité d'interaction entre les sommets  $u$  et  $v$ . Par définition, la fonction est symétrique et  $\forall \{u, v\} \in V \times V$ , nous avons la relation  $w(u, v) = w(v, u)$ . Dans la suite de ce manuscrit, nous utiliserons la notation condensée  $w_{uv}$  pour désigner le poids  $w(u, v)$ . Par convention, la fonction de poids vérifie les propriétés suivantes :

$$w_{uv} \begin{cases} \in [0, 1] & \forall (u, v) \in E \\ = 0 & \forall (u, v) \notin E \\ = w_{vu} & (\text{symétrie}). \end{cases} \quad (1.4)$$

### Degré d'un sommet

Le degré d'un sommet, noté  $\delta(u)$ , désigne le nombre de sommets adjacents au sommet  $u$ . Le degré  $\delta : V \rightarrow \mathbb{N}$  est défini par :

$$\delta(u) = \text{card}(\mathcal{N}(u)). \quad (1.5)$$

Le degré pondéré d'un sommet, noté  $\delta_w(u)$ , représente la quantité totale d'interaction d'un sommet avec ses sommets adjacents. Le degré pondéré  $\delta_w : V \rightarrow \mathbb{R}^+$  est défini par :

$$\delta_w(u) = \sum_{v \in \mathcal{N}(u)} w_{uv}. \quad (1.6)$$

Dans le cas où le graphe n'est pas pondéré (c'est à dire que la fonction de poids est toujours égale à 1 sur les arêtes), on a  $\delta(u) = \delta_w(u)$ ,  $\forall u$ .

Le degré normalisé d'un sommet, noté  $\delta^N(u)$ , représente l'interaction globale d'un sommet du graphe, ou interaction moyenne. Le degré normalisé  $\delta^N : V \rightarrow \mathbb{R}^+$  est défini par :

$$\delta^N(u) = \frac{\delta_w(u)}{\delta(u)}. \quad (1.7)$$

### Bordures d'un ensemble de sommets

Nous définissons deux bordures distinctes d'un ensemble sommets : la bordure interne et la bordure externe.

Soit  $\mathcal{A}$  un ensemble de sommets tel que  $\mathcal{A} \subset V$ . La bordure interne de  $\mathcal{A}$ , notée  $\partial^- \mathcal{A}$ , est définie comme l'ensemble des sommets appartenant à  $\mathcal{A}$  et adjacents à au moins un sommet appartenant au complémentaire de  $\mathcal{A}$  (noté  $\mathcal{A}^c$ ).

$$\partial^- \mathcal{A} = \{u \in \mathcal{A} \mid \exists v \in \mathcal{A}^c, v \sim u\} \quad (1.8)$$

La bordure externe de  $\mathcal{A}$ , notée  $\partial^+ \mathcal{A}$ , est définie comme l'ensemble des sommets appartenant au complémentaire de  $\mathcal{A}$  et adjacents à au moins un sommet de  $\mathcal{A}$ .

$$\partial^+ \mathcal{A} = \{u \in \mathcal{A}^c \mid \exists v \in \mathcal{A}, v \sim u\} \quad (1.9)$$

## 1.2 Types de graphes et construction

Dans cette section, nous présentons les différents types de graphes utilisés dans ce manuscrit, ainsi que différentes constructions et fonctions de poids.



La construction de ces graphes est cruciale pour obtenir un bon traitement des données à représenter et dépend fortement de la nature de ces données, qu'elles soient ou non organisées et qu'elles possèdent ou non une représentation naturelle sous forme de graphe.

### 1.2.1 Métrique, Similarité et Fonction de poids

Dans cette section, nous considérons un domaine  $\Omega \subset \mathbb{R}^m$ , et une fonction caractéristique (ou fonction d'attributs)  $\mathcal{F} : \Omega \rightarrow \mathbb{R}^c$ , qui associe un vecteur caractéristique à chaque élément de  $\Omega$ . Il existe différents choix possibles pour définir la fonction caractéristique, ces choix étant généralement dépendants de l'application. Parmi les fonctions caractéristiques les plus usuelles nous pouvons citer celles qui associent à chaque point de  $\Omega$  ses coordonnées sur  $\Omega$ , ainsi que celles qui dépendent d'une fonction initiale  $f^0 : \Omega \rightarrow \mathbb{R}^c$  définissant les données.

La métrique est une notion essentielle dans la construction d'un graphe, que ce soit pour l'établissement de l'ensemble des arêtes (tout particulièrement dans le cas de domaines non organisés), ou pour la définition de la fonction de poids. Une métrique est une application  $\mu : \Omega \times \Omega \rightarrow \mathbb{R}^+$ , qui à chaque paire de points  $u$  et  $v$  ( $u, v \in V \subseteq \Omega$ ) associe une distance.

La métrique la plus usuelle est la distance euclidienne (ou distance  $\mathcal{L}_2$ ) donnée par :

$$\mu_2(u, v) = \sqrt{\sum_{i=1}^c (\mathcal{F}_i(u) - \mathcal{F}_i(v))^2}. \quad (1.10)$$

Dans ce manuscrit, nous serons amenés à utiliser également les distances de Manhattan (ou distance  $\mathcal{L}_1$ ) et de Chebyshev (ou distance  $\mathcal{L}_\infty$ ), respectivement données par :

$$\mu_1(u, v) = \sum_{i=1}^c |\mathcal{F}_i(u) - \mathcal{F}_i(v)| \quad (1.11)$$

et

$$\mu_\infty(u, v) = \max_{i=1}^c |\mathcal{F}_i(u) - \mathcal{F}_i(v)| \quad (1.12)$$

Le choix de la métrique utilisée dépend fortement du type de domaine et d'application considérés, il n'y a pas de règle générale [Hein et al. 08].

### Mesure de similarité

Au delà de la métrique, il est nécessaire de définir la notion de similarité qui est la base de toute fonction de poids. Cette mesure rend compte des affinités

entre les éléments du domaine initial et, plus précisément qu'une métrique, permet de placer l'ensemble des similarités du domaine sur une même échelle. Tout comme la métrique, cette mesure dépend de l'application considérée et du type de données. Il n'existe pas de règle générale permettant de déterminer la mesure de similarité la plus judicieuse pour une application, mais nous verrons plus loin dans ce manuscrit que pour des cas particuliers certaines mesures s'imposent.

De manière générale, une mesure de similarité est une application  $g : \Omega \times \Omega \rightarrow [0, 1]$  qui attribue à chaque paire d'éléments  $u, v \in \Omega$  un indice de similarité. Plus cet indice est proche de 1, plus les éléments sont similaires.

Dans ce manuscrit, et sans être exhaustifs, nous considérerons les mesures de similarité suivantes :

$$g_1(u, v) = 1 \quad (1.13)$$

$$g_2(u, v) = (\mu(u, v) + \varepsilon)^{-1} \quad (1.14)$$

$$g_3(u, v) = 1 - \frac{\mu(u, v)}{\max_{i, j \in \Omega, i \neq j} \mu(i, j)} \quad (1.15)$$

$$g_4(u, v) = \exp\left(-\frac{\mu(u, v)^2}{\sigma^2}\right) \quad (1.16)$$

La fonction de poids  $w : V \times V \rightarrow [0, 1]$  peut donc être définie comme

$$w(u, v) = \begin{cases} g(u, v) & (u, v) \in E \\ 0 & (u, v) \notin E \end{cases} \quad (1.17)$$

Nous allons maintenant présenter les différentes structures de graphes associées aux différents types de domaines. La construction de ces structures dépend également d'une métrique et d'une fonction de poids.

### 1.2.2 Domaines non organisés

Considérons tout d'abord le cas le plus général, correspondant aux domaines ne possédant aucune organisation (ou du moins aucune organisation suffisante pour induire directement une structure de graphe).

Soient  $\Omega \subset \mathbb{R}^m$  un domaine discret non organisé, et  $\mathcal{G} = (V, E, w)$ , un graphe représentant le domaine  $\Omega$ . Par définition, chaque sommet du graphe représente un élément de  $\Omega$ .

## Graphe complet

Le graphe complet est un graphe pondéré tel qu'il existe une arête entre toutes les paires de sommets. Il possède la structure la plus simple à mettre en œuvre, à partir d'un domaine non organisé. Ainsi, l'ensemble des arêtes est défini tel que

$$E = \{V \times V\} \quad (1.18)$$

## Graphe des $k$ plus proches voisins

Soient  $k \in \mathbb{N}$  et  $\mu$  une métrique pour le domaine discret  $\Omega$ . Le graphe des  $k$  plus proches voisins ( $k$ -ppv) est un graphe pondéré dont chaque sommet est relié à ses  $k$  plus proches voisins, selon la métrique  $\mu$ . L'ensemble des  $k$  plus proches voisins d'un sommet  $u$  est noté  $\mathcal{K}(u)$

Par nature, les relations de voisinage d'un graphe des  $k$ -ppv ne sont pas symétriques. Afin de conserver la propriété de symétrie des arêtes, nous utiliserons dans ce manuscrit une version symétrique (ou réciproque) du graphe des  $k$ -ppv. Dans cette version, l'ensemble  $E$  des arêtes est défini par

$$E = \{(u, v) \mid u \in \mathcal{K}(v) \text{ ou } v \in \mathcal{K}(u)\} \quad (1.19)$$

Cette construction garantit que chaque sommet du graphe est connecté à au moins  $k$  autres sommets. En contrepartie, un sommet peut posséder jusqu'à  $N - 1$  voisins (où  $N$  est le nombre total de sommets).

**Remarque.** *Il n'existe pas de règle générale pour déterminer le nombre  $k$  de voisins. Ce choix est fortement dépendant de l'application considérée.*

## Graphe de $\varepsilon$ -voisinage

Le graphe de  $\varepsilon$ -voisinage est un graphe pondéré dont l'ensemble des arêtes est défini par un seuillage sur une métrique  $\mu$ . Un sommet  $v$  appartient au voisinage d'un sommet  $u$  si la distance entre  $u$  et  $v$  est inférieure au seuil  $\varepsilon$ .

D'un autre point de vue, si l'on se place dans l'espace de la fonction caractéristique  $\mathcal{F}$ , sur lequel est définie la métrique  $\mu$ , le  $\varepsilon$ -voisinage d'un sommet  $u$ , noté  $\mathcal{N}_\varepsilon(u)$ , est constitué de l'ensemble des sommets dont l'image par  $\mathcal{F}$  appartient à la boule de rayon  $\varepsilon$  et centrée en  $\mathcal{F}(u)$ . Le  $\varepsilon$ -voisinage pour un sommet  $u$  est alors

$$\mathcal{N}_\varepsilon(u) = \{v \mid \mu(u, v) < \varepsilon\}, \quad (1.20)$$

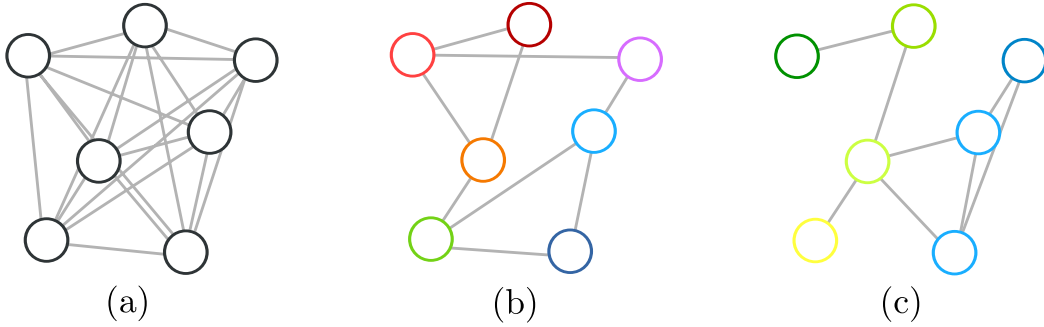


FIGURE 1.1 – Construction de graphes sur un domaine non organisé. (a) Un graphe complet, il existe une arête reliant chaque paire de sommets. (b) Un graphe des  $k$  plus proches voisins (ici 2), où chaque sommet est connecté aux deux sommets qui lui sont le plus proche (ici au sens de la couleur). (c) Un graphe de  $\varepsilon$ -voisinage, où chaque sommet est connecté à l'ensemble des sommets dont la distance (ici au sens de la couleur) n'excède pas une valeur  $\varepsilon$ .

et l'ensemble des arêtes est donc défini par

$$E = \{(u, v) \mid v \in \mathcal{N}_\varepsilon(u)\}. \quad (1.21)$$

Ces trois types de constructions sont illustrées à la figure 1.1. Pour une revue plus complète des graphes de voisinage, le lecteur intéressé pourra se référer à [Jaromsky et Toussaint 92].

### 1.2.3 Domaines organisés

Dans cette section, nous considérons le cas des domaines naturellement organisés, et donc possédant au moins une structure de graphe intrinsèque.

#### Cas des images et autres signaux discrets

Un signal discret peut-être vu comme une fonction  $f : \mathbb{Z}^q \rightarrow \mathbb{R}^c$  où  $q = 1, 2$  ou 3, correspondant respectivement à un signal 1D, une image (2D), ou une image volumique (3D).

Quelle que soit la dimension du signal, l'ensemble des sommets  $V$  est défini comme un sous-ensemble du domaine de définition du signal. Ainsi, chaque sommet représente un échantillon du signal et on a  $V \subset \mathbb{Z}^q$ .

Le voisinage d'un sommet, à partir d'un signal discret, est généralement défini en fonction d'une mesure de distance sur les coordonnées spatiales

associées à chaque sommet. La distance la plus usuelle et correspondant à la structure intrinsèque de l'espace des entiers est la distance de Manhattan (norme  $\mathcal{L}_1$ ), où chaque sommet  $u$  est connecté à l'ensemble des sommets adjacents à  $u$  sur les différentes dimensions du signal. D'autres constructions mettent également en avant la distance de Chebyshev (norme  $\mathcal{L}_\infty$ ), pour un voisinage plus étendu.

Considérant une image, et en imposant une construction d' $\varepsilon$ -voisinage avec  $\varepsilon = 1$ , nous retrouvons les tessellations standards permettant de former les grilles 2D représentant les images.

- dans le cas d'une distance euclidienne  $\mu_2$ , seuls les 4 sommets adjacents sont ajoutés au voisinage et le graphe est un graphe dit de 4-connexité.
- dans le cas d'une distance de Chebyshev  $\mu_1$ , les 4 sommets sur les diagonales sont également ajoutés au voisinage et le graphe est un graphe dit de 8-connexité.

Ces deux constructions standards sont illustrées sur la figure 1.2.

**Remarque.** *Un signal discret, peut également être considéré comme un domaine non organisé. Dans ce cas la structure intrinsèque des données n'est pas prise en compte et le graphe est construit de la même manière que pour un domaine non organisé. Le graphe peut également être construit en combinant les deux approches et en utilisant les arêtes générées par les deux types de constructions (considérant un domaine organisé, puis non organisé).*

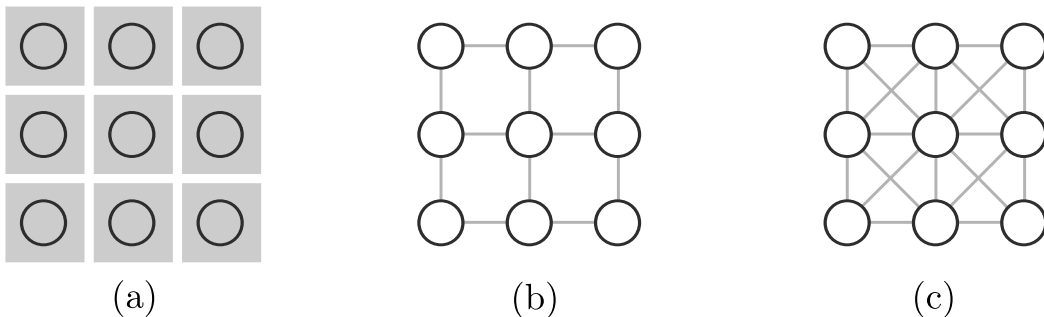


FIGURE 1.2 – Construction de graphes sur une image. (a) Une grille régulière de pixels avec les sommets superposés. (b) Construction d'un graphe de 4-voisinage avec une norme  $\mathcal{L}_2$ . (c) Construction d'un graphe de 8-voisinage avec une norme  $\mathcal{L}_1$ .

Dans le cas des images simples, la fonction de poids est généralement basée sur une comparaison entre l'intensité (ou le vecteur couleur) des pixels. Ainsi, la fonction caractéristique  $\mathcal{F}$  est simplement définie telle que  $\mathcal{F} = f$ .

### Cas des images texturées et représentation non-locale

Dans le contexte des images texturées, la caractérisation d'un pixel par son seul vecteur couleur est généralement insuffisante pour porter l'information de texture (les structures fines et répétitives) associée à ce pixel.

Dans ce cas, il est nécessaire de définir une fonction caractéristique plus large, reposant sur les *patches* de l'image. Ainsi, considérant des patches de dimension  $d \times d$ , la fonction caractéristique  $\mathcal{F}_d : V \rightarrow \mathbb{R}^{c \times d \times d}$  est définie par

$$\mathcal{F}_d(u) = \left( f_i(v), i \in [1, c], v \in \mathcal{N}_\varepsilon \cup \{u\} \right)^T, \forall u \in V \quad (1.22)$$

où  $c$  est la dimension du vecteur couleur défini en chaque sommet, et où la relation entre la dimension  $d$  du patch et la dimension  $\varepsilon$  du voisinage est donnée par  $\varepsilon = (d - 1)/2$ . La figure 1.3 illustre la définition de patches.

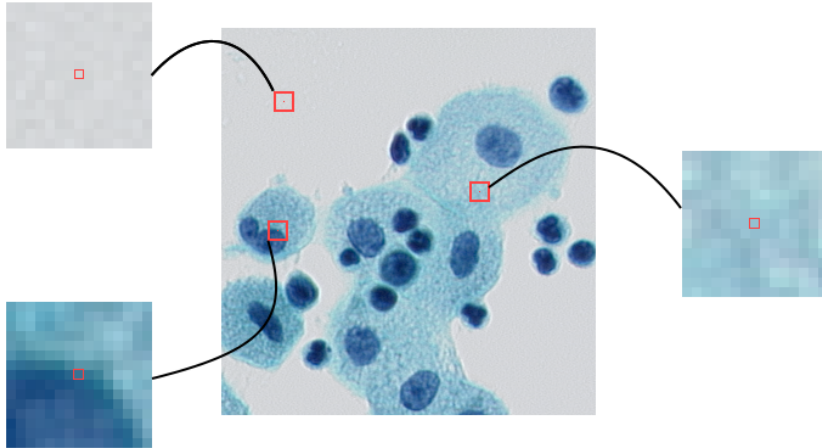


FIGURE 1.3 – Exemples de patches. Trois exemples de patches sur une image couleur. Ici, les pixels (en rouge) sont caractérisés par une fonction  $\mathcal{F}_{13} : V \rightarrow \mathbb{R}^{3 \times 13 \times 13}$ .

L'utilisation de patches pour la caractérisation d'images texturées a été initialement proposée dans le cadre de la synthèse de texture [Efros et Leung 99] et reprise par la suite dans de nombreux travaux sur le filtrage et la restauration d'image et de maillage [Gilboa et Osher 07, Buades et al. 08, Elmoataz et al. 08a, Bougleux et al. 09].

Au delà d'une simple modification de la fonction caractéristique, et pour que les comparaisons entre patches aient un sens, il est également nécessaire d'étendre le voisinage de chaque sommet. C'est ce qui est proposé par les

méthodes dites non-locales, introduites par [Buades et al. 08], où chaque pixel (ou chaque patch) est susceptible d'être comparé à chaque pixel de l'image. En pratique, et en raison de la complexité algorithmique d'une telle méthode, la notion de non-localité est usuellement remplacée par celle de semi non-localité où le nombre de pixels à comparer est réduit à la taille d'une fenêtre de recherche.

Ainsi, le graphe construit à partir d'une image texturée, désigné par le terme graphe non-local est construit comme un graphe de  $D$ -voisinage, où  $D \gg d$ . En pratique la distance utilisée est une distance de Chebyshev, ce qui permet d'avoir des fenêtres de recherche carrées et de dimension  $(2D + 1) \times (2D + 1)$ .

### Cas des RAGs ou superpixels

Une autre construction possible à partir d'un signal discret, permettant de construire un graphe sur une représentation simplifiée du signal, est le graphe d'adjacence de région (RAG). Cette construction se base sur une partition du domaine du signal en régions connexes (par exemple sous la forme d'un diagramme de Voronoï), et associe à chaque région un unique sommet. Avec cette construction, deux sommets  $u$  et  $v$  sont adjacents si les régions associées à ces sommets sont adjacentes, et l'arête associée est ajoutée à l'ensemble  $E$ . Dans le cas des images, chaque région est habituellement appelée un superpixel.

Un exemple de RAG construit à partir d'une image 2D, ainsi que la partition de l'image initiale sont illustrés sur la figure 1.4.

**Remarque.** Ici encore, le graphe peut-être étendu en ajoutant les arêtes induites par d'autres constructions que celle impliquant l'adjacence des régions.

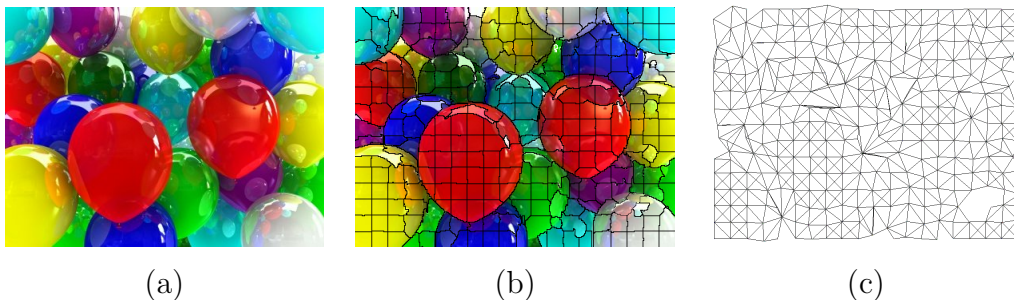


FIGURE 1.4 – Construction d'un graphe à partir d'une carte de régions. (a) L'image originale. (b) La carte de régions correspondant à une partition de l'image. (c) Le graphe d'adjacence de régions associé à cette partition.

### Cas des maillages 3D

Un maillage 3D (communément appelé un *mesh*) est une représentation discrète de la surface d'un objet volumique. Il est composé d'un ensemble de points contenus dans  $\mathbb{R}^3$ , ces points étant reliés entre eux par des arêtes pour former des faces (approximant la surface de l'objet). Dans le cas d'un maillage triangulaire, les faces sont des triangles dont chaque sommet correspond à un point du maillage.

Dans ce cas, la construction du graphe est implicite et se limite généralement à considérer le maillage comme un graphe et d'en conserver les sommets et les arêtes. Néanmoins, il est possible d'utiliser tout les types de construction précédemment définis, y compris d'utiliser un RAG (en utilisant une carte de région spécifique à la structure du maillage).

Ce type de construction est valable pour tout type de maillage, qu'il soit triangulaire ou non, et quelque soit la dimension de l'espace le contenant. Il est illustré à la figure 1.5

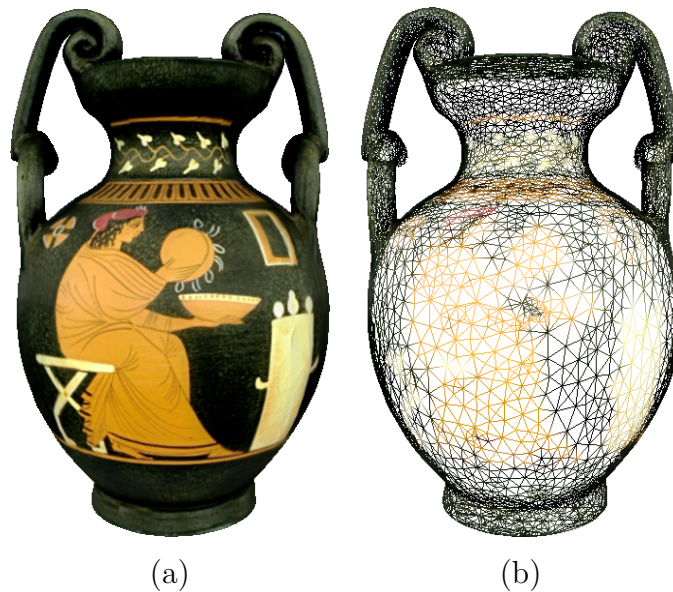


FIGURE 1.5 – Construction d'un graphe à partir d'un maillage 3D. (a) Le maillage 3D texturé. (b) Le graphe induit par ce maillage.





# Chapitre 2

## Calcul discret sur graphe

### Sommaire

---

<b>2.1 Opérateurs sur graphe</b>	<b>26</b>
2.1.1 Fonctions définies sur graphe	26
2.1.2 Opérateur de différence	27
2.1.3 Opérateur de dérivée	28
2.1.4 Opérateur gradient	28
2.1.5 Opérateurs laplaciens	29
<b>2.2 Opérateurs directionnels</b>	<b>31</b>
2.2.1 Opérateurs de différences directionnelles	32
2.2.2 Opérateurs gradients directionnels	32
<b>2.3 Opérateurs moyenneurs non-locaux</b>	<b>34</b>
2.3.1 Régularisation par $p$ -laplacien et filtre moyeneur non-local	34
2.3.2 EdP morphologiques et filtres moyenneurs non-locaux	36
<b>2.4 Conclusion</b>	<b>39</b>

---

Dans ce chapitre, nous présentons les notions de calcul discret sur graphe nécessaires à la bonne compréhension de la suite du manuscrit.

Dans une première partie, nous rappelons les définitions des opérateurs définis sur des graphes proposés par [Bougleux et al. 07, Elmoataz et al. 08a], incluant les opérateurs différence, dérivée, gradient et laplaciens. Puis, dans une seconde partie, nous rappelons les définitions des opérateurs directionnels introduit par [Elmoataz et al. 08b, Ta et al. 09a, Ta et al. 11]. Enfin, dans une troisième partie, nous présentons les opérateurs moyenneurs non-locaux correspondant à une itération des filtres moyenneurs non-locaux et définis à

partir de l'équation du laplacien et des EdP morphologiques discrètes.

L'ensemble de ces opérateurs, basés sur des équations aux différences partielles, sont les constituants principaux des différentes transpositions et approches discrètes que nous introduisons dans les différentes parties de ce manuscrit.

## 2.1 Opérateurs sur graphe

### 2.1.1 Fonctions définies sur graphe

#### Espaces de fonctions sur graphe

Par convention, l'espace Hilbertien des fonctions à valeurs réelles associant une valeur à chaque sommet  $u \in V$  est noté  $\mathcal{H}(V)$ . Une fonction  $f : V \rightarrow \mathbb{R}$  dans  $\mathcal{H}(V)$  associe un vecteur  $f_v$  à chaque sommet  $v \in V$  et peut-être représentée par un vecteur colonne de  $\mathbb{R}^N$ ,  $f = [f_1, \dots, f_N]^T$ . Par définition, cet espace est muni du produit scalaire tel que pour deux fonctions  $f, g \in \mathcal{H}(V)$  on a

$$\langle f, g \rangle_{\mathcal{H}(V)} = \sum_{u \in V} f(u)g(u) \quad (2.1)$$

De même, l'espace Hilbertien des fonctions à valeurs réelles associant une valeur réelle à chaque arête  $e \in E$  est noté  $\mathcal{H}(E)$ . Par définition, cet espace est muni du produit scalaire tel que pour deux fonctions  $F, G \in \mathcal{H}(E)$  on a

$$\langle F, G \rangle_{\mathcal{H}(E)} = \sum_{(u,v) \in E} F((u,v))G((u,v)) \quad (2.2)$$

#### Intégrale et norme d'une fonction discrète

Contrairement au continu, l'intégrale d'une fonction discrète est triviale. Étant donné que le domaine est fini, intégrer une fonction sur ce domaine revient à sommer cette fonction sur l'intégralité du domaine.

$$\int_{\Omega} f = \sum_{u \in \Omega} f(u) \quad (2.3)$$

et la norme  $\mathcal{L}_p$  est donnée par

$$\|f\|_p = \left( \sum_{u \in \Omega} |f(u)|^p \right)^{1/p}, \quad (2.4)$$

dans le cas où  $1 \leq p < \infty$ , et par

$$\|f\|_\infty = \max_{u \in \Omega} (|f(u)|), \quad (2.5)$$

dans le cas où  $p = \infty$ .

### 2.1.2 Opérateur de différence

L'opérateur de différence d'une fonction  $f \in \mathcal{H}(V)$  est une application  $d : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$ . Il est défini sur une arête  $(u, v) \in E$  par

$$(df)(u, v) \stackrel{\text{déf.}}{=} f(v) - f(u). \quad (2.6)$$

De même, l'opérateur de différence pondérée est une application  $d_w : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$ , définie sur une arête  $(u, v) \in E$  [Bougleux et al. 07, Elmoataz et al. 08a] par

$$(d_w f)(u, v) \stackrel{\text{déf.}}{=} \gamma(w_{uv})(f(v) - f(u)). \quad (2.7)$$

où la fonction  $\gamma : \mathbb{R} \rightarrow \mathbb{R}$  dépend de la fonction de poids  $w$ . Ce facteur peut-être interprété comme l'inverse d'un pas d'échantillonnage  $h_{uv}$  entre  $u$  et  $v$ , qui est l'analogie discret sur graphe d'un taux d'accroissement d'une fonction continue.

**Remarque.** Dans la plupart des cas, et afin de retrouver les versions discrètes et pondérées des opérateurs gradients et laplacien définis sur des graphes, nous utilisons une fonction  $\gamma$  définie telle que  $\gamma(x) = \sqrt{x}$ . Dans le reste du manuscrit et sauf mention contraire nous considérerons cette égalité comme effective.

**Remarque.** Par la suite, nous notons  $\gamma_{uv} = \gamma(w_{uv})$ .

L'opérateur de différence coïncide avec l'opérateur de gradient pondéré, noté  $\nabla_w^E : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$ , et défini sur une arête  $(u, v) \in E$  par

$$(\nabla_w^E f)(u, v) \stackrel{\text{déf.}}{=} \gamma_{uv}(f(v) - f(u)). \quad (2.8)$$

### Opérateur adjoint

L'opérateur adjoint de l'opérateur différence, noté  $d_w^* : \mathcal{H}(E) \rightarrow \mathcal{H}(V)$  est défini par la relation suivante

$$\langle d_w f, F \rangle \stackrel{\text{déf.}}{=} \langle f, d_w^* F \rangle. \quad (2.9)$$

où  $f \in \mathcal{H}(V)$  et  $F \in \mathcal{H}(E)$ . En développant l'expression des produits scalaires selon les définitions (2.1), (2.1), l'opérateur adjoint est ainsi exprimé en un sommet  $u$  par

$$(d_w^* F)(u) \stackrel{\text{déf.}}{=} \sum_{v \in V} \gamma_{uv} (F(v, u) - F(u, v)). \quad (2.10)$$

Cet adjoint est linéaire et mesure le flot d'une fonction de  $\mathcal{H}(E)$  en chaque sommet du graphe. Par analogie avec les opérateurs différentiels continus, l'opérateur de divergence d'une fonction  $F \in \mathcal{H}(E)$  est défini par le relation

$$\operatorname{div}_w F \stackrel{\text{déf.}}{=} -d_w^* F. \quad (2.11)$$

**Remarque.** D'autres définitions de la différence ont été proposées dans la littérature. Nous pouvons citer par exemple [Zhou et Schölkopf 04] :

$$(d_w f)(u, v) = \sqrt{w_{uv}} \frac{f(v)}{\sqrt{\delta(v)}} - \sqrt{w_{uv}} \frac{f(u)}{\sqrt{\delta(u)}}. \quad (2.12)$$

Cette définition ne vérifie cependant pas les mêmes propriétés que la définition (2.7), et n'est pas nulle lorsque la fonction  $f$  est localement constante [Hein et Maier 06].

### 2.1.3 Opérateur de dérivée

Si l'on exprime l'opérateur différence en fonction du pas  $h_{uv}$ , nous obtenons l'expression

$$(d_w f)(u, v) = \frac{f(v) - f(u)}{h_{uv}} \quad (2.13)$$

qui peut être interprétée comme une version discrète définie sur graphe de l'opérateur de dérivée appliqué sur une arête  $(u, v) \in E$  :

$$\left. \frac{\partial f}{\partial(u, v)} \right|_u = \partial_v f(u) = (d_w f)(u, v). \quad (2.14)$$

### 2.1.4 Opérateur gradient

Le gradient pondéré d'une fonction, noté  $\nabla_w : \mathcal{H}(V) \rightarrow \mathcal{H}(V)$ , est défini pour une fonction  $f \in \mathcal{H}(V)$  en un sommet  $u \in V$  par

$$(\nabla_w f)(u) \stackrel{\text{déf.}}{=} \left( (d_w f)(u, v) \right)_{v \in V}^T, \quad (2.15)$$

## 2.1. Opérateurs sur graphe

---

soit le vecteur des différences pondérées calculées sur l'ensemble des arêtes  $(u, v) \in E$ . Cet opérateur est un opérateur de premier ordre, défini localement en un sommet  $u \in V$ .

### Normes

En se basant sur la définition des normes  $\mathcal{L}_p$  (2.4) et  $\mathcal{L}_\infty$  (2.5) nous pouvons exprimer les normes de l'opérateur gradient comme

$$\|(\nabla_w f)(u)\|_p \stackrel{\text{déf.}}{=} \left( \sum_{v \in V} \gamma_{uv}^p |f(v) - f(u)|^p \right)^{\frac{1}{p}}, \quad (2.16)$$

lorsque  $1 \leq p < \infty$ , et

$$\|(\nabla_w f)(u)\|_\infty \stackrel{\text{déf.}}{=} \max_{v \in V} (\gamma_{uv} |f(v) - f(u)|), \quad (2.17)$$

lorsque  $p = \infty$ . Ces normes peuvent être interprétées comme une mesure de la variation locale en un sommet  $u$ , d'une fonction  $f$ .

**Remarque.** Étant donnée la définition de la fonction de poids (1.4), l'opérateur de différence  $(d_w f)(u, v)$  est nul excepté pour les sommets  $v$  adjacents à  $u$ . Les expressions des normes du gradient (2.16) et (2.17) peuvent donc être restreintes au voisinage de  $u$ , et réécrites telles que

$$\|(\nabla_w f)(u)\|_p = \left( \sum_{v \sim u} \gamma_{uv}^p |f(v) - f(u)|^p \right)^{\frac{1}{p}}, \quad (2.18)$$

lorsque  $1 \leq p < \infty$ , et

$$\|(\nabla_w f)(u)\|_\infty = \max_{v \sim u} (\gamma_{uv} |f(v) - f(u)|), \quad (2.19)$$

lorsque  $p = \infty$ , où la notation  $v \in V$  est remplacée par  $v \sim u$ .

### 2.1.5 Opérateurs laplaciens

#### $p$ -laplacien isotrope

L'opérateur de  $p$ -laplacien pondéré isotrope a été introduit [Elmoataz et al. 08a, Bougleux et al. 09] comme l'analogie discret du  $p$ -laplacien continu [Kawohl 11] à partir des opérateurs de différence (2.7) et de différence adjoint (2.10).

Étant donné un réel  $p$  tel que  $0 < p < \infty$ , le  $p$ -laplacien isotrope  $\Delta_{w,p}^i : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$  d'une fonction  $f \in \mathcal{H}(V)$  est défini par

$$\Delta_{w,p}^i f \stackrel{\text{déf.}}{=} -d_w^* \left( \|\nabla_w f\|_2^{p-2} d_w f \right). \quad (2.20)$$

Et peut être exprimé en un sommet  $u \in V$  par

$$\left( \Delta_{w,p}^i f \right)(u) \stackrel{\text{déf.}}{=} \sum_{v \in V} \gamma_{uv}^2 \left( \|\nabla_w f\|_2^{p-2}(v) + \|\nabla_w f\|_2^{p-2}(u) \right) (f(v) - f(u)). \quad (2.21)$$

Dans le cas où  $p = 2$ , le  $p$ -laplacien isotrope (2.20) devient

$$\Delta_{w,p}^i f = -d_w^* (d_w f) = \frac{1}{2} \Delta_w f. \quad (2.22)$$

qui est l'analogie discret de l'opérateur laplacien. L'opérateur laplacien sur graphe peut être exprimé en un sommet  $u$ , et pour une fonction  $f \in \mathcal{H}(V)$  par

$$\left( \Delta_w f \right)(u) \stackrel{\text{déf.}}{=} 2 \sum_{v \in V} \gamma_{uv}^2 (f(v) - f(u)). \quad (2.23)$$

Ce laplacien est linéaire et correspond au laplacien combinatoire [Chung 97].

Dans le cas où  $p = 1$ , le  $p$ -laplacien isotrope (2.20) devient

$$\Delta_{w,p}^i f = -d_w^* \left( \|\nabla_w f\|_2^{-1} d_w f \right) = \mathcal{K}_w^i f. \quad (2.24)$$

qui est l'analogie discret de l'opérateur de courbure isotrope d'une fonction continue. L'opérateur de courbure sur graphe peut être exprimé en un sommet  $u$ , et pour une fonction  $f \in \mathcal{H}(V)$  par

$$\left( \mathcal{K}_w^i f \right)(u) \stackrel{\text{déf.}}{=} \sum_{v \in V} \gamma_{uv}^2 \left[ \frac{1}{\|\nabla_w f\|_2(v)} + \frac{1}{\|\nabla_w f\|_2(u)} \right] (f(v) - f(u)). \quad (2.25)$$

Dans le cas d'un graphe non pondéré, l'expression de la courbure peut être considérée comme étant la version discrète de l'opérateur courbure utilisé dans le contexte du traitement des images [Osher et Shen 00, Chan et Vese 01].

### **$p$ -laplacien anisotrope**

Le  $p$ -laplacien pondéré anisotrope est une extension du  $p$ -laplacien pondéré isotrope. Étant donné un réel  $p$  tel que  $0 < p < \infty$ , le  $p$ -laplacien anisotrope  $\Delta_{w,p}^a : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$  d'une fonction  $f \in \mathcal{H}(V)$  est défini par

$$\Delta_{w,p}^a f \stackrel{\text{déf.}}{=} -d_w^* (|d_w f|_2^{p-2} d_w f). \quad (2.26)$$

Et peut être exprimé en un sommet  $u \in V$  par

$$(\Delta_{w,p}^a f)(u) \stackrel{\text{déf.}}{=} \sum_{v \in V} \gamma_{uv}^p |f(v) - f(u)|^{p-2} (f(v) - f(u)). \quad (2.27)$$

**Remarque.** Dans le cas où  $p = 2$ , les  $p$ -laplacien isotrope (2.21) et anisotrope (2.27) ont la même expression

$$(\Delta_{w,p}^i f)(u) = (\Delta_{w,p}^a f)(u) = \sum_{v \in V} \gamma_{uv}^2 (f(v) - f(u)). \quad (2.28)$$

**Remarque.** Dans la suite de ce manuscrit, nous ferons référence au  $p$ -laplacien anisotrope comme le  $p$ -laplacien.

### **$p$ -laplacien normalisé**

Le  $p$ -laplacien normalisé est une extension du  $p$ -laplacien pondéré anisotrope. Étant donné un réel  $p$  tel que  $0 < p < \infty$ , le  $p$ -laplacien normalisé  $\Delta_{w,p}^N : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$  d'une fonction  $f \in \mathcal{H}(V)$  s'exprime en sommet par

$$(\Delta_{w,p}^N f)(u) \stackrel{\text{déf.}}{=} \sum_{v \in V} \left( \frac{\gamma_{uv}}{\sum_{v \in V} \gamma_{uv}} \right)^p |f(v) - f(u)|^{p-2} (f(v) - f(u)). \quad (2.29)$$

## **2.2 Opérateurs directionnels**

Dans cette section, nous présentons les opérateurs directionnels introduits dans [Ta et al. 08b, Ta et al. 08a] comme base des approches discrètes de la morphologie mathématique basée sur des graphes. Pour chacun des opérateurs, nous présentons une version interne et externe. Étant donné une fonction  $f \in \mathcal{H}(V)$  et un sommet  $u$ , l'opérateur interne (respectivement externe) correspond à une version restreinte de l'opérateur classique, où seuls les sommets  $v$  pour lesquels la valeur de la fonction  $f(v)$  est inférieure (respectivement supérieure) à  $f(u)$  sont considérés.



### 2.2.1 Opérateurs de différences directionnelles

L'opérateur de différence pondérée externe, basé sur l'opérateur max, noté  $d_w^+ f : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$  est défini selon une arête  $(u, v)$  par

$$(d_w^+ f)(u, v) \stackrel{\text{déf.}}{=} \max\left(0, (d_w f)(u, v)\right). \quad (2.30)$$

L'opérateur de différence pondérée interne, basé sur l'opérateur max, noté  $d_w^- f : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$  est défini selon une arête  $(u, v)$  par

$$(d_w^- f)(u, v) \stackrel{\text{déf.}}{=} -\min\left(0, (d_w f)(u, v)\right). \quad (2.31)$$

Ces opérateurs vérifient la propriété

$$(d_w^+ f)(u, v) = (d_w^- f)(v, u). \quad (2.32)$$

### Opérateurs de dérivées directionnelles

Comme pour l'opérateur de différence pondérée, l'opérateur de dérivée possède lui aussi ses pendants directionnels externe et interne, définis respectivement par

$$\partial_v^+ f(u) \stackrel{\text{déf.}}{=} (d_w^+ f)(u, v) \quad (2.33)$$

$$\partial_v^- f(u) \stackrel{\text{déf.}}{=} (d_w^- f)(u, v) \quad (2.34)$$

### 2.2.2 Opérateurs gradients directionnels

L'opérateur gradient pondéré externe d'une fonction  $f \in \mathcal{H}(V)$ , noté  $\nabla_w^+ : \mathcal{H}(V) \rightarrow \mathcal{H}(V)$ , est défini en un sommet  $u \in V$  par

$$(\nabla_w^+ f)(u) \stackrel{\text{déf.}}{=} \left( (d_w^+ f)(u, v) \right)_{v \in V}^T. \quad (2.35)$$

De même, l'opérateur gradient pondéré interne d'une fonction  $f \in \mathcal{H}(V)$ , noté  $\nabla_w^- : \mathcal{H}(V) \rightarrow \mathcal{H}(V)$ , est défini en un sommet  $u \in V$  par

$$(\nabla_w^- f)(u) \stackrel{\text{déf.}}{=} \left( (d_w^- f)(u, v) \right)_{v \in V}^T. \quad (2.36)$$

Ces deux définitions sont consistantes avec les définitions des gradients d'une fonction continue et peuvent être interprétées comme leurs versions discrètes définies sur des graphes.

## Normes

En se basant sur la définition des normes  $\mathcal{L}_p$  (2.4) et  $\mathcal{L}_\infty$  (2.5) nous pouvons exprimer les normes de l'opérateur gradient comme

$$\|(\nabla_w^+ f)(u)\|_p \stackrel{\text{déf.}}{=} \left( \sum_{v \in V} \gamma_{uv}^p [\max(0, f(v) - f(u))]^p \right)^{\frac{1}{p}}, \quad (2.37)$$

$$\|(\nabla_w^- f)(u)\|_p \stackrel{\text{déf.}}{=} \left( \sum_{v \in V} \gamma_{uv}^p |\min(0, f(v) - f(u))|^p \right)^{\frac{1}{p}}, \quad (2.38)$$

lorsque  $1 \leq p < \infty$ , et

$$\|(\nabla_w^+ f)(u)\|_\infty \stackrel{\text{déf.}}{=} \max_{v \in V} \left( \gamma_{uv} \max(0, f(v) - f(u)) \right), \quad (2.39)$$

$$\|(\nabla_w^- f)(u)\|_\infty \stackrel{\text{déf.}}{=} \max_{v \in V} \left( \gamma_{uv} |\min(0, f(v) - f(u))| \right), \quad (2.40)$$

lorsque  $p = \infty$ . Ces normes peuvent être interprétées comme une mesure de la variation locale en un sommet  $u$ , d'une fonction  $f$ .

## Propriétés et relations des gradients

La relation suivant entre l'opérateur gradient et les opérateurs gradient directionnels, dans le cas d'une norme  $\mathcal{L}_p$ ,  $1 \leq p < \infty$  a été montrée dans [Ta et al. 08b] :

$$\|(\nabla_w f)(u)\|_p^p = \|(\nabla_w^+ f)(u)\|_p^p + \|(\nabla_w^- f)(u)\|_p^p. \quad (2.41)$$

De même, dans le cas d'une norme  $\mathcal{L}_\infty$ , la relation entre l'opérateur gradient et les opérateurs gradient directionnels est donnée par

$$\|(\nabla_w f)(u)\|_\infty = \max \left( \|(\nabla_w^+ f)(u)\|_\infty, \|(\nabla_w^- f)(u)\|_\infty \right). \quad (2.42)$$

## Interprétation morphologique de ces gradients

Inspiré par les processus de dilatation et d'érosion définis par des équations aux différences partielles, [Ta et al. 08a] ont proposé une transposition discrète de ces processus morphologiques sur graphe, impliquant les opérateurs de gradients directionnels. Ainsi, reprenant cette transposition, les équations de

dilatation et respectivement d'érosion discrètes sont définies sur un graphe pondéré par

$$\frac{\partial f(u)}{\partial t} = + \left\| (\nabla_w^+ f)(u) \right\|_p \quad (2.43)$$

$$\frac{\partial f(u)}{\partial t} = - \left\| (\nabla_w^- f)(u) \right\|_p \quad (2.44)$$

Ces équations peuvent aisément être interprétées intuitivement. Considérant un sous-ensemble  $\mathcal{A} \subset V$ , et utilisant la définition des bordures internes et externes, l'équation de dilatation appliquée à l'ensemble  $\mathcal{A}$  peut être interprétée comme un processus de croissance qui déplace des sommets de  $\partial^+ \mathcal{A}$  vers  $\mathcal{A}$ . De même, l'équation d'érosion appliquée à l'ensemble  $\mathcal{A}$  peut être interprétée comme un processus de contraction qui déplace des sommets de  $\mathcal{A}$  vers  $\partial^- \mathcal{A}$ .

## 2.3 Opérateurs moyenneurs non-locaux

Dans cette section, nous rappelons le principe des opérateurs moyenneurs non-locaux, correspondant à une itération de filtres moyenneurs non-locaux, et définis à partir de l'équation du laplacien et des EdP morphologiques discrètes.

### 2.3.1 Régularisation par $p$ -laplacien et filtre moyenneur non-local

Soit  $\mathcal{G} = (V, E, w)$  un graphe pondéré, et  $f^0 \in \mathcal{H}(V)$  une fonction représentant l'observation d'une fonction  $h \in \mathcal{H}(V)$  corrompue par un bruit additif  $n \in \mathcal{H}(V)$ , tel que  $f^0 = h + n$ . La méthode la plus courante pour retrouver la fonction originale  $h$ , débarrassée du bruit  $n$ , est de rechercher une fonction  $f \in \mathcal{H}(V)$  qui soit à la fois régulière et proche de l'observation  $f^0$ . Cette méthode peut être formalisée par la minimisation d'une énergie impliquant un terme de régularisation (ou pénalité) et un terme d'approximation (ou terme d'attache aux données) :

$$h \simeq \arg \min_{f: V \rightarrow \mathbb{R}} \mathcal{R}_{w,p}(f) + \frac{\lambda}{2} \|f - f^0\|_2^2, \quad (2.45)$$

où le terme de régularisation  $\mathcal{R}_{w,p}(f)$  est basé sur l'opérateur gradient, tel que,

$$\mathcal{R}_{w,p}(f) \stackrel{\text{déf.}}{=} \frac{1}{2p} \sum_{u \in V} \left\| (\nabla_w f)(u) \right\|_p^p, \quad (2.46)$$

### 2.3. Opérateurs moyenneurs non-locaux

---

et où  $\lambda \in \mathbb{R}$  est le paramètre de régularisation qui contrôle le rapport entre l'attache aux données et la régularisation.

La solution de cette équation peut être obtenue en considérant l'équation d'Euler-Lagrange proposée par [Elmoataz et al. 08a, Bougleux et al. 09] et définie par

$$\frac{\partial \mathcal{R}_{w,p}(f)}{\partial f(u)} + \lambda(f(u) - f^0(u)) = 0, \quad \forall u \in V, \quad (2.47)$$

où le premier terme correspond à la variation du terme de régularisation en fonction de l'inconnue  $f$ . [Elmoataz et al. 08a, Bougleux et al. 09] ont montré que cette variation peut être interprétée comme un  $p$ -laplacien

$$\frac{\partial \mathcal{R}_{w,p}(f)}{\partial f(u)} = -(\Delta_{w,p}f)(u) \quad (2.48)$$

et l'équation (2.47) peut être réécrite sous la forme du système linéaire

$$\begin{aligned} \lambda f^0(u) &= \left( \lambda + \sum_{v \sim u} \gamma_{uv}^p |f(v) - f(u)|^{p-2} \right) f(u) \\ &+ \sum_{v \sim u} \gamma_{uv}^p |f(v) - f(u)|^{p-2} f(v) \end{aligned} \quad (2.49)$$

Une solution à ce système est donnée par l'algorithme itératif linéaire de Gauss-Jacobi (voir [Bougleux et al. 09] pour plus de détails), dont une itération peut être est décomposée en deux étapes

$$\begin{cases} \beta_{uv}^{p,n} = \sum_{v \sim u} \gamma_{uv}^p |f^n(v) - f^n(u)|^{p-2}, & \forall (u, v) \in E, \\ f^{n+1}(u) = \frac{\lambda f^0(u) + \sum_{v \sim u} \beta_{uv}^{p,n} f^n(v)}{\lambda + \sum_{v \in V} \beta_{uv}^{p,n}}, & \forall u \in V. \end{cases}, \quad (2.50)$$

et correspond à une famille de filtres moyenneurs. En effet, à chaque itération, la nouvelle valeur de  $f$  en un sommet  $u$  dépend uniquement de deux quantités : la valeur initiale  $f^0(u)$  et une moyenne pondérée des valeurs filtrées de  $f$  au voisinage de  $u$ . Le choix du paramètre de régularisation, de même que le choix du graphe permettent de retrouver et d'étendre des filtres communément utilisés pour le lissage ou le débruitage d'images. En particulier, pour  $p = 2$ , une itération de ce filtre moyeneur (2.50) devient

$$f^{n+1}(u) = \frac{\lambda f^0(u) + \sum_{v \sim u} \gamma_{uv}^2 f^n(v)}{\lambda + \sum_{v \sim u} \gamma_{uv}^2}. \quad (2.51)$$

Ce filtre permet de retrouver de nombreux autres modèles de filtres, tels le filtre bilatéral, le filtre de variation totale digitale, les moyenneurs non-locaux, etc. Voir [Elmoataz et al. 08a] pour plus de détails.

### Opérateur moyenneur non-local

Lorsque le paramètre  $\lambda$  est nul (pas d'attache aux données), et que  $\gamma_{uv} = \sqrt{w(u, v)}$ , ce filtre devient un filtre moyenneur non-local :

$$f^{n+1}(u) = NLM(f^n)(u) \quad (2.52)$$

où

$$NLM(f)(u) = \frac{\sum_{v \sim u} w(u, v) f(v)}{\sum_{v \sim u} w(u, v)} \quad (2.53)$$

est un opérateur moyenneur non-local, correspondant à une itération du filtre moyenneur non-local.

### 2.3.2 EdP morphologiques et filtres moyenneurs non-locaux

Dans cette section, nous montrons que les opérateurs morphologiques discrets peuvent être interprétés comme une famille de filtres moyenneurs non-locaux, en fonction de la valeur de  $p$  et de la topologie du graphe. Dans le cas où  $p = \infty$ , nous introduisons deux opérateurs morphologiques non-locaux  $NLE$  et  $NLD$  et rappelons l'interprétation morphologique de ces opérateurs. Puis, dans le cas où  $p = 1$ , nous introduisons deux opérateurs de filtre moyenneur non-locaux conditionnels  $NLM^+$  et  $NLM^-$  et proposons une nouvelle interprétation du processus morphologique associé. Enfin, dans le cas où  $p = 2$ , nous rappelons les liens existant entre nos opérateurs morphologiques et certains schémas de discrétisation.

#### Cas où $p = \infty$

Dans le cas où  $p = \infty$ , le processus de dilatation peut être exprimé comme l'EdP itérative suivante :

$$f^{n+1}(u) = f^n(u) + \Delta t \max_{v \in V} \left( \sqrt{w_{uv}} \max(f^n(v) - f^n(u), 0) \right). \quad (2.54)$$

Dans le cas où  $\Delta t = 1$ , cette équation peut être interprétée comme un processus de dilatation non-local,

$$f^{n+1} = NLD(f^n) \quad (2.55)$$

où  $NLD : \mathcal{H}(V) \rightarrow \mathcal{H}(V)$  est défini pour correspondre à la dilatation de  $f^n$  :

$$NLD(f^n)(u) = f^n(u) + \max_{v \in V} \left( \sqrt{w_{uv}} \max(f^n(v) - f^n(u), 0) \right). \quad (2.56)$$

### 2.3. Opérateurs moyenneurs non-locaux

---

Dans le cas où  $w = 1$ , et pour une construction de graphe particulière permettant à chaque noeud  $u$  d'être auto-incident, l'opérateur non-local de dilatation coïncide avec la formulation de la morphologie mathématique définie sur des graphes [Heijmans et Vincent 92, Heijmans et al. 92]. En effet, l'EdP de dilatation devient :

$$\begin{aligned} f^{n+1} &= f^n(u) + \max_{v \in V} \left( \sqrt{w_{uv}} \max(f^n(v) - f^n(u), 0) \right) \\ &= f^n(u) + \max_{v \in V} (f^n(v) - f^n(u)) \\ &= \max_{v \in V} (f^n(v)) \end{aligned} \quad (2.57)$$

De même, nous pouvons définir l'opérateur non-local d'érosion, noté  $NLE$  et défini similairement par

$$NLE(f^n)(u) = f^n(u) - \max_{v \in V} \left( \sqrt{w_{uv}} \min(f^n(v) - f^n(u), 0) \right), \quad (2.58)$$

qui correspond à une itération d'un processus morphologique d'érosion. Dans le cas où le graphe n'est pas pondéré ( $w = 1$ ), cet opérateur coïncide également avec la formulation de l'érosion telle que définie par la morphologie mathématique sur graphes :

$$f^{n+1} = \min_{v \in V} (f^n(v)). \quad (2.59)$$

#### Cas où $p = 1$

Dans le cas où  $p = 1$ , le processus de dilatation peut être exprimé comme l'EdP itérative suivante :

$$f^{n+1}(u) = f^n(u) + \Delta t \sum_{v \in V} \left( \sqrt{w_{uv}} \max(f^n(v) - f^n(u), 0) \right). \quad (2.60)$$

Nous montrons que cette équation peut-être interprétée comme un filtre moyenneur non-local, noté  $NLM^+$  (signifiant non-local Mean Dilation). Si dans l'équation (2.60) nous restreignons la somme au voisinage de  $u$  tel que le maximum soit non nul, nous pouvons écrire

$$\begin{aligned} f^{n+1}(u) &= f^n(u) + \Delta t \sum_{v \in V_u^{+n}} \left( \sqrt{w_{uv}} (f^n(v) - f^n(u)) \right) \\ &= f^n(u) \left( 1 - \Delta t \sum_{v \in V_u^{+n}} \sqrt{w_{uv}} \right) + \Delta t \sum_{v \in V_u^{+n}} \sqrt{w_{uv}} f^n(v), \end{aligned} \quad (2.61)$$

où  $V_u^{+n} = \{v \in \mathcal{N}(u) \mid f^n(v) > f^n(u)\}$ . Avec cette réécriture, et en définissant le pas temporel  $\Delta t$  tel que

$$\Delta t = \max_{u \in V} \left( \frac{1}{\sum_{v \in V_u^{+n}} \sqrt{w_{uv}}} \right) \quad (2.62)$$

le processus de dilatation peut être exprimé comme un filtre moyennneur non-local conditionnel. On écrit :

$$f^{n+1}(u) = \begin{cases} f^n(u) + \frac{\sum_{v \in V_u^{+n}} \sqrt{w_{uv}} f^n(v)}{\sum_{v \in V_u^{+n}} \sqrt{w_{uv}}} & \text{si } V_u^{+n} \neq \emptyset \\ f^n(u) & \text{sinon.} \end{cases} \quad (2.63)$$

Suivant la même démarche, le processus d'érosion peut lui aussi être interprété comme un filtre moyennneur non-local conditionnel et noté  $NLM^-$  (signifiant non-local Mean Erosion).

Finalement, nous avons introduit deux opérateurs de filtre moyennneur non-locaux conditionnels. Ces deux opérateurs sont dits conditionnels car leur fonctionnement en un sommet  $u$  dépend de la relation entre  $f^n(u)$  et  $f^n(v)$  où  $v$  appartient au voisinage de  $u$ . Si la fonction  $f^n$  possède un maximum local en  $u$  (respectivement un minimum local), alors l'opérateur  $NLM^+$  (respectivement  $NLM^-$ ) est constant. Dans le cas contraire, il fonctionne comme un filtre moyennneur ne considérant que les sommets  $v$  pour lesquels la fonction  $f^n$  est supérieure (respectivement inférieure) en  $v$  qu'en  $u$ . Ces opérateurs peuvent être écrits :

$$NLM^+(f)(u) = \begin{cases} f(u) + \frac{\sum_{v \in V_u^+} \sqrt{w_{uv}} f(v)}{\sum_{v \in V_u^+} \sqrt{w_{uv}}} & \text{si } V_u^+ \neq \emptyset \\ f(u) & \text{sinon.} \end{cases} \quad (2.64)$$

et

$$NLM^-(f)(u) = \begin{cases} f(u) + \frac{\sum_{v \in V_u^-} \sqrt{w_{uv}} f(v)}{\sum_{v \in V_u^-} \sqrt{w_{uv}}} & \text{si } V_u^- \neq \emptyset \\ f(u) & \text{sinon.} \end{cases} \quad (2.65)$$

où  $V_u^+ = \{v \in \mathcal{N}(u) \mid f(v) > f(u)\}$  et  $V_u^- = \{v \in \mathcal{N}(u) \mid f(v) < f(u)\}$ .

### Cas où $p = 2$

Dans le cas où  $p = 2$ , le processus de dilatation peut être exprimé comme l'EdP itérative suivante :

$$f^{n+1}(u) = f^n(u) + \Delta t \sqrt{\sum_{v \in V} \left( w_{uv} \max(f^n(v) - f^n(u), 0) \right)^2}. \quad (2.66)$$

Si le graphe représente une grille régulière de  $\mathbb{R}^m$  de dimension  $m$ , alors [Ta et al. 08b] ont montré que cette équation coïncide avec la discrétisation de premier ordre des EDP morphologiques, proposée par [Osher et Sethian 88].

## 2.4 Conclusion

Dans ce chapitre, nous avons rappelé les définitions des opérateurs différentiels et morphologiques définis sur des graphes pondérés et nécessaires à la compréhension de la suite de ce manuscrit.

Ces opérateurs, définis comme les analogues discrets des opérateurs différentiels et morphologiques continus, sont la clé de notre approche visant à transcrire des modèles définis dans le domaine continu au domaine des graphes. Ils sont ainsi les constituants principaux des approches et schémas discrets que nous introduisons dans les différentes parties de ce manuscrit.





## Deuxième partie

### Une nouvelle famille d'opérateurs $p$ -laplaciens



# Introduction

L'étude du  $p$ -laplacien, dans ses versions locales et non-locales, sur des espaces euclidiens ou des graphes, suscite depuis quelques années un grand intérêt. En effet, cet opérateur joue un rôle important en géométrie et apparaît dans la formulation par EDP de nombreux problèmes non linéaires de physique, mécanique, traitement d'images ou vision par ordinateur. Pour une introduction et une revue complète de cet opérateur, voir [Drábek 07, Lindquist 06, Kawohl 11] et les références connexes. Lorsque  $p$  tend vers l'infini, cet opérateur a été défini sous le nom de laplacien infini.

Des version non-locales de l'opérateur  $p$ -laplacien continu apparaissent naturellement dans de nombreux modèles mathématiques, aussi bien en biologie qu'en mathématique ou en traitement d'images. On pourra consulter à ce sujet le récent livre de [Andreu-Vaillio et al. 10]. La version discrète de cet opérateur a également été étudiée dans le cadre de la théorie de l'apprentissage et utilisée dans de nombreuses applications, notamment en classification de données [Luxburg 07] et réduction de dimensions [Belkin et Niyogi 03].

Le  $p$ -laplacien sur graphe (qui est la généralisation du laplacien standard) suscite lui aussi un intérêt croissant de la part des communauté mathématiques, de traitement d'images et de données définies sur des variétés. Nous pouvons par exemple citer la relation entre le  $p$ -laplacien sur graphe et le Cheeger Cut récemment mise en avant par [Hein et Bühler 10], la régularisation par  $p$ -laplacien pour la classification semi supervisée [Bougleux et al. 09] ou bien encore le cadre proposé par [Elmoataz et al. 08a] pour la régularisation non-locale sur graphe, basé sur l'opérateur  $p$ -laplacien, qui unifie le traitement des images, maillages et autres variétés. Ce cadre comprend notamment les versions isotropes et anisotropes du  $p$ -laplacien, exprimées sous la forme de divergences et en lien avec leurs analogues continus. L'étude des EDP sur graphes impliquant le  $p$ -laplacien discret, notamment sur le comportement de ces équations ainsi qu'au sujet de l'existence et de l'unicité de leurs solutions, a fait de ces équations un sujet d'intérêt en elles-mêmes.

Que ce soit sous sa forme discrète ou continue, la plupart des expressions de l'opérateur  $p$ -laplacien sont définies sous la forme d'une divergence. Néanmoins,

il existe dans le cadre continu une expression de cet opérateur définie sous la forme d'une non-divergence, introduite par [Peres et al. 09] lors de l'étude des jeux de type Tug-of-War bruités. Cet opérateur, appelé  $p$ -laplacien normalisé ou laplacien des jeux, s'exprime sous la forme d'une combinaison linéaire du laplacien infini et du laplacien.

Dans cette seconde partie, nous proposons une adaptation de l'opérateur  $p$ -laplacien normalisé sur le domaine des graphes reposant sur le cadre des EdP sur graphes. Cette adaptation nous permet d'introduire une nouvelle classe d'opérateurs  $p$ -laplaciens définis comme des combinaisons de gradients morphologiques sur graphes et d'un filtre moyennneur non-local. Ces opérateurs interpolent entre les opérateur laplacien et laplacien infini sur graphes et sont obtenus par discrétisation sur graphe du  $p$ -laplacien normalisé. Cette adaptation reposant sur les gradients morphologiques discrets, permet de définir des équations aux différences finies sur graphe avec des coefficients variables.

L'étude de ces équations, sous l'éclairage des connexions avec les EDP continues correspondantes est un sujet d'intérêt propre. Nous étudions le problème de Dirichlet associé à ces équations et prouvons l'existence et l'unicité des solutions de ces problèmes. Récemment, certains jeux aléatoires à somme nulle (Tug-of-War games) ont été utilisés en connexion avec des problèmes d'EDP [Manfredi et al. b, Manfredi et al. a]. Nous montrons que les fonctions à valeurs de ces jeux coïncident avec les solutions de ces équations de  $p$ -laplaciens.

Finalement, nous proposons d'utiliser ces opérateurs dans le cadre de la résolution de nombreux problèmes en traitement d'images et de données.

Tout d'abord, dans le chapitre suivant, nous proposons une discrétisation du laplacien infini sur graphes qui est à la base de notre transcription du  $p$ -laplacien normalisé.

# Chapitre 3

## L'opérateur laplacien infini non-local

### Sommaire

---

<b>3.1</b>	<b>Le laplacien infini continu</b>	<b>46</b>
3.1.1	Relation avec les jeux de type Tug-of-War	47
<b>3.2</b>	<b>Le laplacien infini sur graphe</b>	<b>49</b>
3.2.1	Relation avec le laplacien infini continu	49
3.2.2	Relation avec la limite du $p$ -laplacien	50
3.2.3	Interprétation morphologique	51
<b>3.3</b>	<b>Problème de Dirichlet</b>	<b>52</b>
<b>3.4</b>	<b>Expérimentations</b>	<b>55</b>
3.4.1	Inpainting d'images	55
3.4.2	Segmentation semi-supervisée d'images et classification de données	57
<b>3.5</b>	<b>Conclusion</b>	<b>58</b>

---

L'équation du laplacien infini se situe à l'intersection de nombreux champs des mathématiques. Elle est également utilisée dans plusieurs applications, notamment pour le transport optimal, la théorie des jeux, le traitement d'images, la vision par ordinateur ou la reconstruction de surfaces. Elle a été tout d'abord étudiée par Aronsson, motivé par l'analyse classique de la construction d'extensions de Lipschitz pour une fonction donnée.

Ces dernières décennies, les théoriciens des EDP ont établi l'existence et l'unicité des solutions de ces équations, et démontré leur régularité. Récemment, ces équations (ainsi que le  $p$ -laplacien) ont été reliées aux fonctions à valeurs continues des jeux de type Tug-of-War. Nous recommandons aux

lecteurs intéressés par plus de détails et d'exemples d'applications, la lecture des références suivantes [Aronsson 67, Aronsson et al. 04, Crandall et al. 07, Crandall 08, Juutinen et Kawohl 06, Juutinen et al. 99, Manfredi et al. b, Manfredi et al. a].

Cette équation a trouvé son application concrète pour de nombreux problèmes de traitement d'images. Nous pouvons ainsi citer les travaux de [Torre et Poggio 86, Yuille et Poggio 86] qui introduisent l'utilisation du laplacien infini pour le rehaussement de contours et qui ont servi de base pour la méthode de détection de [Canny 86]. Plus tard, [Caselles et al. 98] ont étudié les extensions de Lipschitz de la minimisation absolue du laplacien infini pour l'interpolation de données et l'inpainting d'images. [Cong et al. 04] ont également proposé un schéma d'interpolation pour la déformation de formes. De même, [Elion et Vese 07] ont proposé un modèle pour la décomposition d'images utilisant la variation totale et le laplacien infini. Enfin, [Guillot et Le Guyader 09] ont récemment proposé d'extrapoler le champ des vecteurs définis sur le domaine d'une image dans un cadre variationnel incluant le laplacien infini.

Dans ce chapitre, motivés par la volonté d'étendre les applications du laplacien infini du domaine des images à celui des données non organisées, nous proposons une adaptation de cet opérateur sur le domaine des graphes de topologie arbitraire.

Cette adaptation nous conduit à des EdP à coefficients variables (dépendant des données) qui peuvent être considérées comme des versions non-locales de l'équation du laplacien infini et nous présentons une preuve de l'existence et de l'unicité de la solutions de ces équations.

### 3.1 Le laplacien infini continu

Le laplacien infini est intrinsèquement lié à l'EDP du laplacien infini qui peut être exprimée par

$$\Delta_\infty f = 0, \tag{3.1}$$

où

$$\Delta_\infty f = \|\nabla f\|^{-2} \sum_{i,j}^n \frac{\partial f}{\partial x_i} \frac{\partial f}{\partial x_j} \frac{\partial^2 f}{\partial x_i \partial x_j} \tag{3.2}$$

désigne la version 1-homogène du laplacien infini sur des domaines euclidiens pour des fonctions lisses définies sur un ensemble ouvert  $\Omega \subset \mathbb{R}^n$ .

[Aronsson 67] a interprété formellement l'équation  $\Delta_\infty f = 0$  comme la limite lorsque  $p$  tend vers l'infini de l'équation d'Euler-Lagrange

$$\Delta_p f = \operatorname{div}\left(\|\nabla f\|_2^{p-2} \nabla f\right) = 0, \quad (3.3)$$

associée au problème de Dirichlet

$$\inf_{f \in W^{1,p}(\Omega), f=b \text{ sur } \partial\Omega} \|\nabla f\|_{L,p(\Omega)}, \quad (3.4)$$

lorsque  $1 < p < +\infty$ .

Récemment, [Chambolle et al. 12] ont proposé un laplacien infini de Hölder qui peut être considéré comme une version non-locale du laplacien infini. Pour cela, ils considèrent la minimisation de la fonctionnelle non-locale

$$\int_{\Omega \times \Omega} \frac{|f(x) - f(y)|^p}{|x - y|^{\alpha p}} dx dy, \text{ pour } \alpha \in [0, 1]. \quad (3.5)$$

L'équation d'Euler-Lagrange associée à cette fonctionnelle est

$$\int_{\Omega} \frac{|f(x) - f(y)|^{p-1} \operatorname{sign}(f(x) - f(y))}{|x - y|^\alpha} dy = 0. \quad (3.6)$$

Ils ont montré que lorsque  $p$  tend vers l'infini, ce minimiseur converge vers l'équation non linéaire et non-locale  $L(f) = 0$  sur le domaine  $\Omega$  avec

$$L(f) = \max_{y \in \Omega, y \neq x} \left( \frac{f(y) - f(x)}{|y - x|^\alpha} \right) + \min_{y \in \Omega, y \neq x} \left( \frac{f(y) - f(x)}{|y - x|^\alpha} \right), \text{ for } x \in \Omega, \quad (3.7)$$

Cet opérateur  $L(f)$  est appelé laplacien infini de Hölder. Dans la suite de ce chapitre, nous montrons que cet opérateur est équivalent au laplacien infini non-local pour un graphe particulier.

### 3.1.1 Relation avec les jeux de type Tug-of-War

Le jeu du Tug-of-War (ou jeu du tir à la corde, en français) est un jeu qui oppose deux équipes situées de part et d'autre d'une frontière définie au sol et reliées par une corde. L'objectif du jeu étant de faire franchir la frontière à l'équipe opposée en tirant sur la corde.

Formalisé mathématiquement, le jeu du Tug-of-War est un jeu à somme nulle, où les gains (en terme de distance à la frontière) d'une équipe sont équivalents à la perte de l'autre et dont l'étude est reliée à l'étude du laplacien infini [Peres et al. 09].



Le jeu se déroule sur un domaine borné  $\Omega \subset \mathbb{R}^n$  et est représenté par une fonction continue de Lipschitz (la fonction de répartition des gains entre les joueurs). Chaque équipe est symbolisée par un joueur (respectivement joueur I et joueur II) dont les actions sont modélisées par une loi de probabilité équitable. À l'instant initial, un jeton (symbolisant le milieu de la corde) est placé en un point  $x_0 \in \Omega$ . Puis, un joueur est tiré au sort (de façon équitable) et choisit une nouvelle position  $x_1 \in B_\varepsilon(x_0)$  (où  $B_\varepsilon(x) = \{y \mid |x - y| \leq \varepsilon\}$ ) pour le jeton, dans le but de maximiser son gain (ou de minimiser sa perte). À chaque tour, un joueur est tiré au sort et déplace le jeton vers une nouvelle position  $x_k \in B_\varepsilon(x_{k-1})$ . La partie se termine lorsque le jeton atteint un point  $x_\tau \in \partial\Omega$ , et le premier joueur gagne  $g(x_\tau)$  (tandis que le second gagne  $-g(x_\tau)$ ).

Étant donné un point de départ  $x_0$ , chaque joueur possède une stratégie lui permettant de maximiser ses gains (elles sont notées  $S_I$  et  $S_{II}$ ). [Peres et al. 09] ont montré que les gains minimum espérés par chacun des joueurs suivant ces stratégies (définis comme les valeurs de deux fonctions  $f_I(x_0)$  et  $f_{II}(x_0)$ ) sont égaux et coïncident avec le gain attendu pour le jeu, donné par la valeur d'une fonction  $f_\varepsilon(x_0)$ . Ces trois fonctions  $f_I$ ,  $f_{II}$  et  $f_\varepsilon$  sont des fonctions infini-harmoniques et sont définies pour tout  $x \in \Omega$  par

$$f_\varepsilon(x) = \frac{1}{2} \left( \max_{y \in B_\varepsilon(x)} f_\varepsilon(y) + \min_{y \in B_\varepsilon(x)} f_\varepsilon(y) \right). \quad (3.8)$$

La valeur  $f(x_0)$  est appelée valeur du jeu et la relation entre la fonction de gain et la fonction de jeu est donnée par la relation

$$f_\varepsilon(x) = g(x), \quad \forall x \in \partial\Omega. \quad (3.9)$$

La formulation de la fonction de jeu peut être intuitivement interprétée en termes du jeu. En effet, le premier joueur cherche à maximiser son gain (avec une probabilité de  $\frac{1}{2}$  de pouvoir le faire) tandis que le second joueur va lui chercher à minimiser le gain du premier joueur (également avec une probabilité de  $\frac{1}{2}$  de pouvoir le faire).

Les auteurs ont également montré que la valeur- $\varepsilon$  de ce jeu est une fonction de Lipschitz convergeant uniformément lorsque  $\varepsilon$  tend vers zéro, nommée la valeur continue du jeu, et qui est solution de l'équation du laplacien infini :

$$\begin{cases} \Delta_\infty f(x) = 0 & \text{on } \Omega \\ f(x) = g(x) & \text{on } \partial\Omega. \end{cases} \quad (3.10)$$

Nous allons montrer dans la section suivante, que sur un graphe particulier, la fonction  $f_\varepsilon$  coïncide avec le laplacien infini non-local.

## 3.2 Le laplacien infini sur graphe

Dans cette section, nous proposons une définition de l'opérateur laplacien infini non-local sur graphe, obtenue par transcription de l'équation du laplacien infini continu.

**Définition 1.** *L'opérateur laplacien infini sur graphe est défini comme*

$$(\Delta_{w,\infty} f)(u) \stackrel{\text{déf.}}{=} \frac{1}{2} \left[ \|(\nabla_w^+ f)(u)\|_\infty - \|(\nabla_w^- f)(u)\|_\infty \right]. \quad (3.11)$$

Nous montrons que, similairement au cas continu, cette définition correspond à la limite quand  $p$  tend vers l'infini de l'opérateur  $p$ -laplacien. Nous montrons également que cette définition permet d'interpréter l'opérateur laplacien infini comme une combinaison d'opérateurs morphologiques discrets non-locaux.

### 3.2.1 Relation avec le laplacien infini continu

Nous montrons tout d'abord que pour des graphes particuliers, la définition de l'opérateur laplacien infini non-local (3.11) coïncide avec la formulation du laplacien infini continu. Pour cela, nous montrons que les équations (3.7) et (3.8) sont deux formes distinctes d'une même équation

$$\begin{aligned} L(f)(u) = & \max_{v \sim u} \left( \sqrt{w(u,v)} \max(f(v) - f(u), 0) \right) \\ & + \min_{v \sim u} \left( \sqrt{w(u,v)} \min(f(v) - f(u), 0) \right) = 0, \end{aligned} \quad (3.12)$$

qui peut être interprétée comme l'équation du laplacien infini sur graphe.

En effet, considérant  $\mathcal{G} = (V, E, w)$  un graphe pondéré, défini tel que  $V = \Omega \subset \mathbb{R}^n$ ,  $E = \{(x, y) \in \Omega \times \Omega : w(x, y) > 0\}$ , il peut aisément être montré que

- dans le cas où  $w(x, y) = \frac{1}{|x-y|^{2\alpha}}$ , l'équation (3.12) est strictement équivalente à l'équation (3.7).
- De même, dans le cas où

$$w(x, y) = \begin{cases} 1 & \text{si } |x - y| \leq \varepsilon \\ 0 & \text{sinon,} \end{cases} \quad (3.13)$$

et en remplaçant l'opérateur min par inf et max par sup, l'équation (3.12) est strictement équivalente à l'équation (3.8).

Ainsi, l'équation (3.12) peut être interprétée comme une transcription de l'équation du laplacien infini sur graphe. En remplaçant la définition des opérateurs gradients directionnels, cette équation devient

$$L(f)(u) = \|\nabla_w^- f(u)\|_\infty - \|\nabla_w^+ f(u)\|_\infty = 2(\Delta_{w,\infty} f)(u) = 0, \quad (3.14)$$

qui coïncide avec notre définition de l'opérateur laplacien infini non-local (3.11).

### 3.2.2 Relation avec la limite du $p$ -laplacien

Nous montrons finalement que notre définition du laplacien infini non-local est consistante avec l'interprétation du laplacien infini continu comme la limite lorsque  $p$  tend vers l'infini du  $p$ -laplacien anisotrope.

**Proposition 1.** *Pour tout  $p \in [1, \infty[$ , et tout sommet  $u$ , nous avons la relation suivante,*

$$(\Delta_{w,p+1} f)(u) = \|(\nabla_{w'}^+ f)(u)\|_p^p - \|(\nabla_{w'}^- f)(u)\|_p^p \quad (3.15)$$

où  $w'_{uv} = w_{uv}^{(p+1)/p}$ .

*Démonstration.* D'après les définitions du  $p$ -laplacien anisotrope (2.27) et des gradients directionnels (2.37) et (2.38) et en posant  $A = f(v) - f(u)$ , le  $p$ -laplacien anisotrope s'écrit

$$(\Delta_{w,p} f)(u) = \sum_{v \in V} w_{uv}^{\frac{p}{2}} |A|^{p-2} (A). \quad (3.16)$$

En remplaçant  $A$  et la norme de  $A$  par les combinaisons de min et max suivantes,

$$\begin{aligned} |A| &= A^+ + A^- \\ A &= A^+ - A^-, \end{aligned} \quad (3.17)$$

où  $A^+$  signifie  $\max(A, 0)^+$  et  $A^-$  signifie  $-\min(A, 0)^-$ , nous obtenons

$$\begin{aligned} (\Delta_{w,p} f)(u) &= \sum_{v \in V} w_{uv}^{\frac{p}{2}} (A^- - A^+) (A^+ + A^-)^{p-2}, \\ (\Delta_{w,p+1} f)(u) &= \sum_{v \in V} w_{uv}^{\frac{p+1}{2}} (A^- - A^+) (A^+ + A^-)^{p-1}. \end{aligned} \quad (3.18)$$

Puis, en développant grâce à la relation  $A^- A^+ = 0$ , l'équation devient

$$(\Delta_{w,p+1} f)(u) = \sum_{v \in V} w_{uv}^{\frac{p+1}{2}} \left( (A^+)^p - (A^-)^p \right), \quad (3.19)$$

et enfin,

$$\left(\Delta_{w,p+1}f\right)(u) = \left\|\left(\nabla_{w'}^+f\right)(u)\right\|_p^p - \left\|\left(\nabla_{w'}^-f\right)(u)\right\|_p^p, \quad (3.20)$$

□

Ainsi, d'après cette relation, le laplacien combinatoire classique peut être exprimé sous la forme d'une différence de deux gradients morphologiques externes et internes. En effet, pour  $p = 1$ , nous avons

$$\left(\Delta_{w,2}f\right)(u) = \left\|\left(\nabla_{w^2}^+f\right)(u)\right\|_1 - \left\|\left(\nabla_{w^2}^-f\right)(u)\right\|_1 \quad (3.21)$$

En se basant sur la proposition 1, nous montrons que le laplacien infini peut s'écrire comme une différence de gradients directionnels.

En effet, en reprenant les notations introduites dans cette proposition, l'équation du  $p$ -laplacien s'écrit

$$\left(\Delta_{w,p+1}f\right)(u) = \left\|\left(\nabla_{w'}^+f\right)(u)\right\|_p^p - \left\|\left(\nabla_{w'}^-f\right)(u)\right\|_p^p = 0, \quad (3.22)$$

où  $w'_{uv} = w_{uv}^{(p+1)/p}$ . En négligeant la puissance  $p$  devenue superflue, cette équation peut facilement être réécrite telle que

$$\left\|\left(\nabla_{w'}^+f\right)(u)\right\|_p - \left\|\left(\nabla_{w'}^-f\right)(u)\right\|_p = 0. \quad (3.23)$$

Finalement, quand  $p \rightarrow \infty$ , nous avons  $w' \rightarrow w$  et la limite de l'équation du  $p$ -laplacien est donnée par

$$\left\|\left(\nabla_w^+f\right)(u)\right\|_\infty - \left\|\left(\nabla_w^-f\right)(u)\right\|_\infty = 0, \quad (3.24)$$

et nous obtenons ainsi la relation

$$\left(\Delta_{w,\infty}f\right)(u) = \left[\left\|\left(\nabla_w^+f\right)(u)\right\|_\infty - \left\|\left(\nabla_w^-f\right)(u)\right\|_\infty\right] = 0, \quad (3.25)$$

qui est consistante avec notre définition du laplacien infini non-local (3.11)

### 3.2.3 Interprétation morphologique

Dans ce paragraphe, nous nous intéressons à l'interprétation morphologique de cet opérateur.

Dans le cas particulier où la fonction de poids est constante et égale à 1 sur l'ensemble des arêtes du graphes et dans le cas où chaque sommet est incident à lui même, cette nouvelle définition de l'opérateur laplacien infini coïncide avec l'approximation par différences finies du laplacien infini continu.

En effet, considérant un ensemble d'arêtes  $E$  étendu aux arêtes auto-incidentes et une fonction de poids  $w_{uv} = 1 \forall (u, v) \in E$ , l'opérateur laplacien infini (3.11) devient

$$(\Delta_{1,\infty}f)(u) = \frac{1}{2} \left[ \max_{v \sim u} f(v) + \min_{v \sim u} f(v) \right] - f(u). \quad (3.26)$$

En ajoutant un facteur 2, l'opérateur  $2(\Delta_{1,\infty}f)$  est connu dans la communauté de morphologie mathématique (algébrique) comme étant l'opérateur laplacien morphologique.

De retour dans le domaine discret et considérant un graphe pondéré de fonction de poids arbitraire, l'opérateur laplacien infini non-local peut être réécrit comme un opérateur morphologique combinant les opérateurs d'érosion (2.58) et de dilatation (2.56) non-locaux définis dans la première partie de ce manuscrit (section 2.3).

$$\begin{aligned} (\Delta_{w,\infty}f)(u) &= \frac{1}{2} \left[ 2f(u) + \|(\nabla_w^+ f)(u)\|_\infty - \|(\nabla_w^- f)(u)\|_\infty \right] - f(u) \\ &= \frac{1}{2} \left[ NLD(f)(u) + NLE(f)(u) \right] - f(u). \end{aligned} \quad (3.27)$$

### 3.3 Problème de Dirichlet

Dans cette section, nous nous intéressons au problème de Dirichlet associé à l'équation du laplacien infini  $\Delta_\infty = 0$  et montrons que ce problème possède une solution unique.

Soient  $\mathcal{G} = (V, E, w)$  un graphe pondéré et  $g : V \rightarrow \mathbb{R}$  une fonction associant une valeur réelle à chaque sommet de  $\mathcal{G}$ . Soient  $A \subset V$  le sous-ensemble des sommets pour lesquels la valeur de la fonction est inconnue et  $\partial^+ A$  sa bordure externe. Le problème de Dirichlet associé à l'opérateur laplacien infini s'écrit

$$\begin{cases} (\Delta_{w,\infty}f)(u) = 0 & \forall u \in A \\ f(u) = g(u) & \forall u \in \partial^+ A. \end{cases} \quad (3.28)$$

Nous montrons que ce problème possède une solution unique.

#### Unicité de la solution

Notre preuve repose sur une généralisation des arguments de [Le Gruyer 05] du continu au discret. Afin de démontrer l'unicité de la solution à ce problème nous montrons tout d'abord le lemme suivant

**Lemme 1.** *L'égalité  $(\Delta_{w,\infty}f)(u) = 0$  implique nécessairement que  $f$  n'a pas de maximum local*

*Démonstration.* Nous procédons par l'absurde, en supposant qu'il existe un sommet  $v_0 \in V$  tel que  $v_0$  soit un maximum local de la fonction  $f$ . Ainsi, si  $F \subseteq V$  est un sous-ensemble connexe de  $V$  tel que  $v_0 \in F$ , alors nous pouvons écrire que  $f(v_0) = \max_F f > f(u), \forall u \in \partial^+ F$ .

Soit  $H = \{u \in F, f(u) = f(v_0)\}$ , alors  $\forall u \in \partial^+ H \max_{v \sim u} (f(v) - f(u))^+ = 0 < \max_{v \sim u} (f(u) - f(v))^+$  ce qui implique que  $\|(\nabla_w^+ f)(u)\|_\infty < \|(\nabla_w^- f)(u)\|_\infty$  et revient à écrire  $(\Delta_{w,\infty}f)(u) > 0$ , ce qui est contradictoire avec l'énoncé du lemme.  $\square$

Avec ce résultat, et en se basant sur le principe de comparaison, nous montrons maintenant l'unicité de la solution.

**Proposition 2.** *L'équation (3.28) possède une solution unique.*

*Démonstration.* Nous procédons une nouvelle fois par l'absurde, en supposant qu'il existe deux solutions  $f$  et  $h$ . Nous cherchons à montrer que si  $f \leq h$  sur l'ensemble  $\partial^+ A$ , alors  $f \leq h$  sur  $A$ .

Par l'absurde, nous supposons qu'il existe  $M = \sup(f - h) > 0$ ,  $H = \{u \in V, (f(u) - h(u)) = M\}$  et  $F = \{u \in H, f(u) = \max_H f\}$ . Si  $u \in H$ , alors  $(f - h)$  atteint son maximum en  $u$ . Ceci implique que  $\|(\nabla_w^- f)(u)\|_\infty \geq \|(\nabla_w^- h)(u)\|_\infty$  et  $\|(\nabla_w^+ f)(u)\|_\infty \leq \|(\nabla_w^+ h)(u)\|_\infty$ . Par hypothèse nous savons que  $(\Delta_{w,\infty}f)(u) = (\Delta_{w,\infty}h)(u)$  et, par conséquent,  $\forall u \in H$  nous pouvons écrire

$$\begin{cases} \|(\nabla_w^- f)(u)\|_\infty = \|(\nabla_w^- h)(u)\|_\infty \\ \|(\nabla_w^+ f)(u)\|_\infty = \|(\nabla_w^+ h)(u)\|_\infty \end{cases} \quad (3.29)$$

Nous montrons ensuite que cette configuration implique nécessairement que l'ensemble  $\mathcal{F}$  soit un maximum local.

Par l'absurde, nous choisissons un sommet  $v_0 \in \partial^+ F$  tel que  $f(v_0) \geq \max_F f$  pour tout  $u \in F, v_0 \sim u$ , ce qui implique que  $\|(\nabla_w^+ f)(u)\|_\infty = f(v_0) - f(u)$ . Étant donné que  $v_0 \notin H$ , nous avons la relation  $f(v_0) - h(v_0) < f(u) - h(u)$ , puis  $f(v_0) - f(u) < h(v_0) - h(u)$  et finalement  $\|(\nabla_w^+ h)(u)\|_\infty > \|(\nabla_w^+ f)(u)\|_\infty$ , ce qui entre en contradiction avec (3.29).  $\square$

### Existence de la solution

Afin de démontrer l'existence, nous considérons l'opérateur moyenné non-local suivant :

$$NLA(f)(u) = \frac{1}{2} \left[ NLD(f)(u) + NLE(f)(u) \right]. \quad (3.30)$$

Reformulée avec cet opérateur, l'équation du laplacien infini devient

$$\begin{aligned} (\Delta_{w,\infty} f) &= 0 \\ NLA(f)(u) - f(u) &= 0 \\ NLA(f)(u) &= f(u). \end{aligned} \quad (3.31)$$

**Proposition 3.** *L'équation (3.28) possède une solution.*

*Démonstration.* Pour le besoin de cette démonstration, nous rappelons le théorème du point fixe de Brouwer : une fonction continue définie depuis un sous-ensemble compact et convexe d'un espace euclidien vers lui-même possède un point fixe.

Puis, nous identifions  $\mathcal{H}(V)$  comme  $\mathbb{R}^n$  et considérons l'ensemble  $K = \{f \in \mathcal{H}(V) \mid f(u) = g(u) \forall u \in \partial^+ A, \text{ et } m \leq f(u) \leq M \forall u \in A\}$ , où  $m = \min_{\partial^+ A} (g(u))$  et  $M = \max_{\partial^+ A} (g(u))$ . Par définition,  $K$  est un sous-ensemble compact et convexe de  $\mathbb{R}^n$ .

Il est ensuite aisé de montrer que l'application  $f \rightarrow NLA(f)$  est continue et à valeurs de  $K$  dans  $K$ . Par conséquent, et d'après le théorème de Brouwer, l'application  $NLA$  possède un point fixe qui est solution de  $NLA(f) = f$ . Ceci complète la preuve.  $\square$

### Schéma numérique

Soit  $(f^n)_{n \in \mathbb{N}}$  une suite de fonctions construite telle que

$$\begin{cases} f^0(u) = g(u) & \forall u \in \partial^+ A \\ f^n(u) = g(u) & \forall n, \forall u \in \partial^+ A \\ f^{n+1}(u) = NLA(f^n)(u). & \forall u \in A \end{cases} \quad (3.32)$$

Il est trivial de montrer que si cette suite converge, sa limite quand  $n$  tend vers l'infini satisfait  $f(u) = NLA(f)(u)$  et donc l'équation (3.28)

## 3.4 Expérimentations

Dans cette section, nous proposons d'illustrer le comportement et l'intérêt de notre expression du laplacien infini non-local pour des problèmes de segmentation et classification semi-supervisés, ainsi que des problèmes d'inpainting.

Ces deux types de problèmes peuvent être formulés comme des problèmes d'interpolation. Concrètement, considérant un ensemble de données où certaines données sont manquantes, l'interpolation consiste à remplacer les données manquantes en produisant de nouvelles données qui sont cohérentes avec les données connues du reste de l'ensemble.

Dans notre cas, nous proposons de considérer que le problème d'interpolation peut être réduit au problème de Dirichlet suivant

$$\begin{cases} (\Delta_{w,\infty} f)(u) = 0 & u \in V_0 \\ f(u) = g(u) & u \in V - V_0 = \partial V_0 \end{cases} \quad (3.33)$$

où  $V_0 \subset V$  est le sous-ensemble des sommets pour lesquels l'information à reconstruire est manquante. La fonction  $g$  représente l'information connue aux bords de  $V_0$  et dépend de l'application.

### 3.4.1 Inpainting d'images

L'inpainting d'images est un processus qui consiste à remplir les parties manquantes d'une image (ou d'une vidéo) de la manière la plus cohérente possible, afin d'obtenir une reconstruction harmonieuse et difficile à détecter.

Ces dernières années, de nombreuses méthodes d'interpolation ont été développées pour répondre à ce problème. Parmi ces méthodes, un certain nombre basées sur des EDP ou des méthodes variationnelles, ont montré leur efficacité à fournir une approximation des données manquantes (voir les articles de [Arias et al. 11, Schoenlieb et Bertozzi 11], ainsi que les références connexes pour plus de détails). Avec les travaux de [Buades et al. 08] sur le filtrage non-local, l'utilisation de méthodes non-locales pour l'inpainting d'images a connu un gain d'intérêt considérable. En effet, les performances de ces méthodes dans le cas d'images texturées sont bien supérieures à celles des méthodes locales. Plusieurs travaux, dont ceux de [Gilboa et Osher 07], se sont consacrés à unifier les méthodes locales et non-locales. Nous pouvons citer [Arias et al. 11] qui a présenté un cadre variationnel pour l'inpainting d'images, ou encore [Ghoniem et al. 11] dont le travail repose sur le cadre des EDP et qui a présenté une approche unifiant les méthodes géométriques locales et non-locales pour l'inpainting de vidéos.



Reprenant le même cadre que [Ghoniem et al. 11], nous considérons le problème d'inpainting au travers du problème d'interpolation (3.33). Dans ce cas la fonction  $g$  du problème d'interpolation est définie comme une application qui à tout sommet  $u$  de  $\partial V_0$  associe une couleur (l'information connue), et la fonction  $f$  (la fonction reconstruite) comme une application qui à tout sommet  $u$  de  $V$  associe une couleur reconstruite.

Afin de préserver la texture et les détails fins de l'image, les graphes utilisés pour ces exemples reposent sur une construction non-locale couplée avec une similarité entre les pixels basée sur des patches.

### Illustration

La figure 3.1 présente un exemple d'inpainting d'images sur un exemple de "libération" numérique d'animaux. La photo originale, représentant des ours dans un enclos, est traversée au premier plan par le grillage de l'enclos. Afin de donner l'illusion que les animaux sont en liberté, nous avons marqué manuellement les parties de l'image correspondant à la grille puis reconstruit les parties manquantes dans le masque de la grille avec notre algorithme d'interpolation. Nous pouvons observer que les zones reconstruites ne sont pas détectables à l'oeil nu, la texture ayant été correctement et harmonieusement répliquée grâce à l'utilisation d'un graphe non-local.



FIGURE 3.1 – Illustration d'inpainting d'images. (a) L'image originale et la région à inpainter (en bleu). (b) Le résultat de l'inpainting.

### 3.4.2 Segmentation semi-supervisée d'images et classification de données

Nous présentons ici quelques exemples d'applications de l'opérateur laplacien infini à la segmentation (ou classification) semi-supervisée. Pour ce type d'application, la fonction à interpoler ne définit plus la donnée en elle-même mais une fonction de marqueurs représentant plusieurs classes de la donnée.

Dans le cas d'un partitionnement en deux classes  $A$  et  $B$ , la fonction à interpoler est donc une application de l'ensemble des sommets vers un ensemble de marqueurs correspondant aux deux classes  $A$  ou  $B$ . La fonction initiale  $g$  est elle définie sur les données connues (éventuellement fournies par l'utilisateur) telle que

$$g(u) = \begin{cases} -1 & \text{si } u \in A \\ 1 & \text{si } u \in B \\ 0 & \text{sinon.} \end{cases} \quad (3.34)$$

**Remarque.** Dans le cas où le nombre de classes est supérieur à deux ( $N$  classes), il est nécessaire de résoudre  $N$  systèmes similaires à (3.33), où le marqueur  $A$  correspond à une classe et le marqueur  $B$  à toutes les autres classes (méthode du un contre tous). Finalement, la fonction de marqueur  $L : V \rightarrow \{C_i\}_{i=1,\dots,N}$  (qui associe à chaque sommet une classe) est définie telle que

$$L(u) = C_i \mid f_i(u) = \max_{j=1,\dots,N} f_j(u) \quad (3.35)$$

Ici encore, les exemples sont fournis à titre d'illustration du potentiel et de l'adaptabilité de notre formalisme d'EdP. Pour une revue plus complète des méthodes de segmentation semi-supervisée sur graphe, voir la troisième partie de ce manuscrit.

#### Illustration

La figure 3.2 illustre l'application de notre approche pour la segmentation d'une image. L'image et les labels initiaux proviennent de la base Grabcut. Nous fournirons plus d'exemples et de détails sur la segmentation d'images par des méthodes d'interpolation dans le chapitre suivant.

La figure 3.3 illustre l'adaptabilité de notre approche aux graphes de données. Le graphe est un graphe des  $k$ -ppv construit à partir d'un ensemble de 400 imageries représentant des zéros et des uns manuscrits et qui proviennent de la base des données des services postaux USPS. La classification est réalisée par l'interpolation d'une fonction de marqueurs définie sur les données.

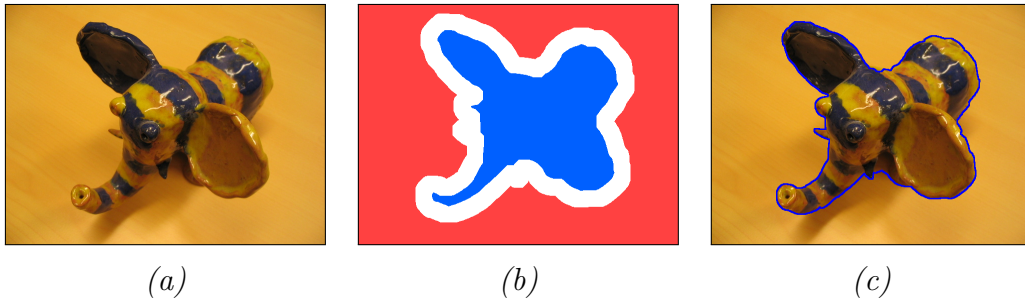


FIGURE 3.2 – Segmentation d'image par interpolation. (a) L'image originale. (b) Les classes A (en bleu) et B (en rouge) à l'instant initial. La partie blanche représente la partie du graphe où l'information sur la fonction de marqueurs est manquante. (c) Le résultat de la segmentation.

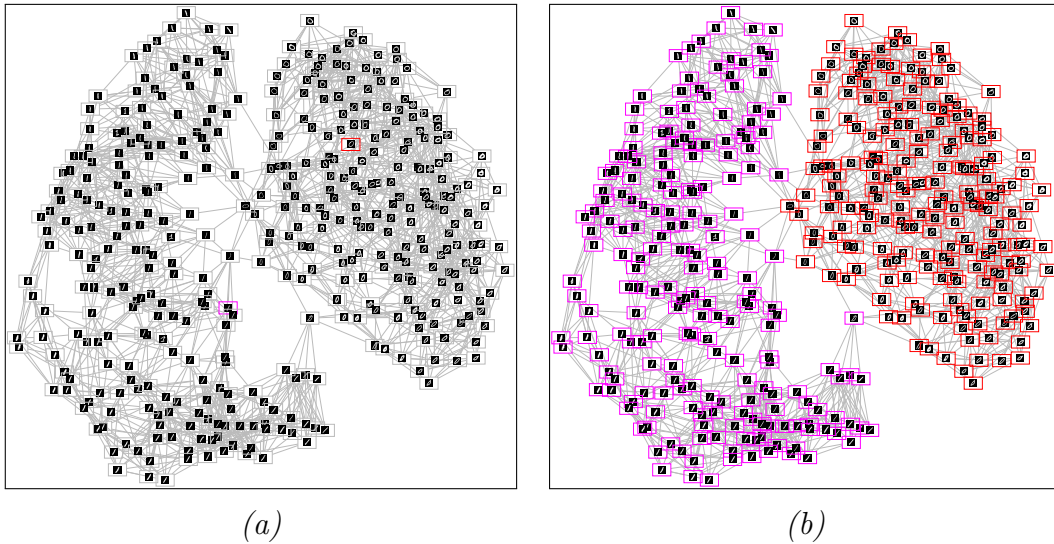


FIGURE 3.3 – Classification de données par interpolation. (a) Le graphe sur lequel sont représentés les marqueurs initiaux. (b) Le résultat de la classification par interpolation de la fonction de marqueur.

### 3.5 Conclusion

Dans ce chapitre, motivés par la volonté d'étendre les applications du laplacien infini du domaine des images à celui des données non organisées, nous avons proposé une adaptation de cet opérateur au domaine des graphes de topologie arbitraire.

### 3.5. Conclusion

---

Nous avons ainsi proposé une définition de l'opérateur laplacien infini non-local sur graphe basées sur une transcription du laplacien infini continu par EdP.

Nous avons montré que pour des graphes euclidiens particuliers cet opérateur coïncide avec la définition du laplacien infini de Hölder ainsi qu'avec les fonctions de certains jeux stochastiques qui sont liés au laplacien infini continu.

Nous avons également montré que l'opérateur laplacien infini non-local peut être interprété comme un opérateur morphologique combinant les opérateurs d'érosion et de dilatation non-locaux définis dans la première partie de ce manuscrit.

Enfin, nous avons proposé une étude du problème de Dirichlet associé à l'équation du laplacien infini et montré que ce problème possède une solution unique.

Finalement, nous avons illustré l'application de cet opérateur à des problèmes d'interpolation pour le traitement des images et des données représentées par des graphes. Les résultats présentés dans ce chapitre le sont à titre qualitatif et non quantitatif afin d'illustrer une large variété d'applications sur une large variété de types de données. Notre objectif étant, nous le rappelons, de montrer le potentiel et l'adaptabilité de notre formalisme sur graphes.

Dans le chapitre suivant et basés sur notre définition du laplacien infini non local, nous introduisons un nouvel opérateur laplacien discret : le  $p$ -laplacien normalisé discret. Cet opérateur est l'analogue discret du  $p$ -laplacien normalisé et peut être interprété comme une nouvelle famille d'opérateurs  $p$ -laplaciens.



# Chapitre 4

## L'opérateur $p$ -laplacien normalisé non-local

### Sommaire

---

<b>4.1 Définition</b>	<b>62</b>
<b>4.2 Problème de Dirichlet</b>	<b>64</b>
4.2.1 Étude de l'existence et de l'unicité	65
<b>4.3 Diffusion sur graphe par <math>p</math>-laplacien</b>	<b>67</b>
<b>4.4 Expérimentations</b>	<b>68</b>
4.4.1 Exemple jouet	68
4.4.2 Restauration et simplification d'images	68
4.4.3 Interpolation	71
<b>4.5 Conclusion</b>	<b>78</b>

---

Dans ce chapitre, nous proposons une discrétisation du  $p$ -laplacien normalisé communément appelé *game  $p$ -laplacien* ( $p$ -laplacien des jeux), et qui s'exprime sous la forme d'une combinaison linéaire du laplacien combinatoire et du laplacien infini.

En mimant son expression continue, nous proposons une version discrète de cet opérateur définie comme une combinaison linéaire entre l'opérateur laplacien infini non-local présenté dans le chapitre précédent et l'opérateur laplacien discret. Cet opérateur à coefficients variables peut être considéré comme une nouvelle famille d'opérateurs  $p$ -laplaciens interpolant entre le laplacien infini et le laplacien normalisé.

À partir de cette définition, nous montrons que cet opérateur peut être également interprété comme un opérateur morphologique combinant un processus d'érosion et de dilatation non-local et un filtre moyenneur non-local. Puis, nous présentons les relations existant entre cet opérateur discret et

certaines EDP non-locales ainsi qu'avec une version bruitée du jeu stochastique du Tug-of-War.

Nous proposons également une étude du problème de Dirichlet associé à l'équation de l'opérateur  $p$ -laplacien normalisé discret et montrons que ce problème possède une solution unique. Enfin, nous présentons une étude du problème de diffusion impliquant notre opérateur et montrons que le processus de diffusion associé peut être exprimé comme un processus de filtrage moyennneur non-local.

Finalement, nous proposons plusieurs illustrations de l'application de notre opérateur à des problèmes de traitement d'images, de maillages et de données.

## 4.1 Définition

Nous rappelons que l'opérateur  $p$ -laplacien d'une fonction  $f : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}$  est défini en continu par

$$\begin{aligned}\Delta_p f &= \operatorname{div}(|\nabla f|^{p-2} \nabla f) \\ &= |\nabla f|^{p-2} [(p-2)\Delta_\infty f + \Delta f]\end{aligned}\tag{4.1}$$

La version normalisée de l'opérateur  $p$ -laplacien, connue sous le nom de laplacien des jeux, est définie par [Kawohl 11] comme

$$\Delta_p^N f(u) = \frac{1}{p|\nabla f|^{p-2}} \Delta_p f.\tag{4.2}$$

Cet opérateur a été récemment introduit pour modéliser le jeu du Tug-of-War dans sa version bruitée. Il s'écrit

$$\begin{aligned}\Delta_p^N f &= \frac{p-2}{p} \Delta_\infty^N f + \frac{1}{p} \Delta f \\ &= \alpha \Delta_\infty^N f + \beta \Delta f,\end{aligned}\tag{4.3}$$

où  $\alpha = (p-2)/p$ ,  $\beta = 1/p$  et  $\alpha + \beta = 1$ , et où

$$\Delta_\infty^N f = \frac{1}{p|\nabla f|^2} \nabla_\infty f.\tag{4.4}$$

Avec une discrétisation de l'équation précédente (4.3) par EdP et en utilisant le  $p$ -laplacien discret, nous proposons une nouvelle formulation, non-locale, du  $p$ -laplacien normalisé sur graphes. Cette nouvelle formulation est donnée par l'équation

$$(\Delta_{w,\alpha,\beta} f)(u) \stackrel{\text{déf.}}{=} \alpha (\Delta_{w,\infty} f)(u) + \beta (\Delta_{w,2}^N f)(u).\tag{4.5}$$

### Une nouvelle famille d'opérateurs $p$ -laplaciens

Cet opérateur peut être considéré comme une nouvelle famille d'opérateurs  $p$ -laplaciens définis en fonction des coefficients  $\alpha$  et  $\beta$ . Cette famille regroupe une classe d'opérateurs qui interpolent entre le laplacien infini non-local et le laplacien normalisé. Pour des valeurs particulières des paramètres  $\alpha$  et  $\beta$ , il est trivial de retrouver les laplaciens usuels (laplacien normalisé et laplacien infini non-local), tels que :

$$\begin{aligned}\Delta_{w,1,0} &= \Delta_{w,\infty} \\ \Delta_{w,0,1} &= \Delta_{w,2}^N\end{aligned}\tag{4.6}$$

**Remarque.** En raison de la relation entre  $\alpha$ ,  $\beta$  et  $p$ , nous ne pouvons retrouver les  $p$ -laplaciens que pour  $p \geq 2$

Finalemnt, en reprenant les définitions des opérateurs NLD, NLE et NLM, notre nouvel opérateur laplacien peut être réécrit comme un opérateur morphologique combinant un processus d'érosion et de dilatation non-local et un filtre moyeneur

$$(\Delta_{w,\alpha,\beta})f = \frac{\alpha}{2} [NLD(f) + NLE(f)] + \beta NLM(f) - f.\tag{4.7}$$

### Relation avec le jeu du Tug-of-War bruité

Considérons maintenant une version modifiée du jeu du Tug-of-War où, en plus des déplacements des joueurs, le jeton subit des déplacements aléatoires provoqués par un phénomène extérieur. Ainsi, dans cette version et à partir d'un point  $x$ , chaque joueur joue avec une probabilité  $\frac{\alpha}{2}$  de choisir le déplacement du jeton et selon une probabilité  $\beta$  (telle que  $\alpha + \beta = 1$ ), le jeton est déplacé vers une nouvelle position aléatoire dans la boule  $B_\varepsilon(x)$ .

Il a été montré dans [Manfredi et al. b] que la fonction de ce jeu, notée  $f_\varepsilon$ , satisfait le principe de programmation dynamique

$$f_\varepsilon(x) = \frac{1}{2} \left( \max_{y \in B_\varepsilon(x)} f_\varepsilon(y) + \min_{y \in B_\varepsilon(x)} f_\varepsilon(y) \right) + \frac{\beta}{|B_\varepsilon(x)|} \int_{B_\varepsilon(x)} f(y) dy.\tag{4.8}$$

De même que pour le jeu sans bruit, la relation entre la fonction de gain et la fonction de jeu est donnée par la relation

$$f_\varepsilon(x) = g(x), \quad \forall x \in \partial\Omega\tag{4.9}$$

Dans le cas général ( $\beta \neq 0$ ), il a été montré par [Manfredi et al. a] que cette fonction approxime la solution de l'équation du  $p$ -laplacien  $\Delta_p f = 0$ .



Tout comme à la section 3.2.1, où nous avons montré que la fonction à valeur du jeu du Tug-of-War simple coïncide avec notre définition du laplacien continu pour un graphe particulier, il est aisé de montrer que la fonction à valeur du Tug-of-War bruité (4.8) coïncide avec notre nouvel opérateur laplacien (4.7) (et ce pour le même graphe particulier).

### Connexion avec les EdP non-locales

De la même manière que dans la section 3.2.1, nous montrons que pour des graphes particuliers, la définition de l'opérateur  $p$ -laplacien normalisé coïncide avec certaines EDP non-locales.

En effet, considérant  $\mathcal{G} = (V, E, w)$  un graphe pondéré, défini tel que  $V = \Omega \subset \mathbb{R}^n$ ,  $E = \{(x, y) \in \Omega \times \Omega : w(x, y) > 0\}$ , et

$$w(x, y) = \begin{cases} \frac{1}{|x-y|^{2s}} & \text{si } x \neq y, s \in [0, 1] \\ 0 & \text{sinon} \end{cases} \quad (4.10)$$

il peut aisément être montré que :

- dans le cas où  $\beta = 0$ , notre formulation correspond au laplacien infini de Hölder (4.11), introduit par [Chambolle et al. 12]

$$L(f) = \max_{y \in \Omega, y \neq x} \left( \frac{f(y) - f(x)}{|y - x|^s} \right) + \min_{y \in \Omega, y \neq x} \left( \frac{f(y) - f(x)}{|y - x|^s} \right), \text{ for } x \in \Omega. \quad (4.11)$$

- dans le cas où  $\alpha = 0$ , cet opérateur recouvre l'opérateur laplacien non-local défini par

$$\frac{1}{\delta_w(x)} \int_{\Omega} w(x, y) (f(y) - f(x)) dy. \quad (4.12)$$

## 4.2 Problème de Dirichlet

Soient  $\mathcal{G} = (V, E, w)$  un graphe connexe,  $A \subset V$  un ensemble de sommet et  $g : \partial^+ A \rightarrow \mathbb{R}$  une fonction définie sur le bord de  $A$ . Nous considérons l'équation suivante correspondant au problème de Dirichlet associé à notre opérateur laplacien non-local

$$\begin{cases} (\Delta_{w, \alpha, \beta} f)(u) = 0 & u \in A \\ f(u) = g(u) & u \in \partial^+ A \end{cases} \quad (4.13)$$

De la même manière que dans le chapitre précédent et afin de simplifier l'écriture des preuves de l'existence et de l'unicité de la solution de l'équation (4.13), nous introduisons l'opérateur suivant :

$$NLA(f)(u) = \frac{\alpha}{2} \left[ NLD(f)(u) + NLE(f)(u) \right] + \beta NLM(f)(u). \quad (4.14)$$

Réécrite avec cet opérateur, l'équation (4.13) du problème de Dirichlet devient

$$f(u) = NLA(f)(u). \quad (4.15)$$

### 4.2.1 Étude de l'existence et de l'unicité

Le principe des preuves de l'existence et de l'unicité de la solution du problème de Dirichlet reprend l'idée présentée pour les preuves de l'existence et de l'unicité de la solution du problème de Dirichlet associé au laplacien infini et présenté dans le chapitre précédent.

#### Existence de la solution

Intéressons nous tout d'abord à montrer l'existence d'une solution pour cette équation.

**Proposition 4.** *L'équation (4.13) admet au moins une solution.*

*Démonstration.* Pour le besoin de cette démonstration, nous rappelons le théorème du point fixe de Brouwer : une fonction continue définie depuis un sous-ensemble compact et convexe d'un espace euclidien vers lui même possède un point fixe.

Puis, nous identifions  $\mathcal{H}(V)$  comme  $\mathbb{R}^n$  et considérons l'ensemble  $K = \{f \in \mathcal{H}(V) \mid f(u) = g(u) \forall u \in \partial^+ A, \text{ et } m \leq f(u) \leq M \forall u \in A\}$ , où  $m = \min_{\partial^+ A} (g(u))$  et  $M = \max_{\partial^+ A} (g(u))$ . Par définition,  $K$  est un sous-ensemble compact et convexe de  $\mathbb{R}^n$ .

Il est ensuite aisé de montrer que l'application  $f \rightarrow NLA(f)$  est continue et à valeurs de  $K$  dans  $K$ . Par conséquent, et d'après le théorème de Brouwer, l'application  $NLA$  possède un point fixe qui est solution de  $NLA(f) = f$ . Ceci complète la preuve.  $\square$

### Unicité de la solution

Nous montrons maintenant l'unicité de cette solution.

**Proposition 5.** *L'équation (4.13) admet une unique solution.*

*Démonstration.* D'après la proposition 4, nous savons que l'équation (4.13) possède au moins une solution. Nous allons prouver que cette solution est unique en procédant par l'absurde.

Soient  $f, h$  deux fonctions distinctes vérifiant l'équation (4.13). Par l'absurde, nous supposons qu'il existe un réel  $M$  vérifiant

$$M = \max_A (f(u) - h(u)) > 0 \quad (4.16)$$

Soit  $B = \{u \in A : f(u) - h(u) = M\}$ . Par construction nous savons que  $B \neq \emptyset$  et  $B \text{cap} \partial^+ A = \emptyset$ . Nous affirmons qu'il existe un sommet  $u_\lambda \in B$  et un sommet  $v_\lambda \in N(u_\lambda)$  tel que  $v_\lambda \notin B$ . Dans le cas contraire, cela impliquerait que  $B \cap \partial^+ A = \emptyset$  ce qui est contradictoire avec la définition du graphe  $\mathcal{G}$ .

Ainsi, d'après la définition de  $M$ , nous pouvons écrire

$$\begin{aligned} f(u_\lambda) - h(u_\lambda) &\geq f(u) - h(u), \quad \forall u \in N(u_\lambda) \\ h(u) - h(u_\lambda) &\geq f(u) - f(u_\lambda), \quad \forall u \in N(u_\lambda) \end{aligned} \quad (4.17)$$

et en particulier, si l'on considère le sommet  $v_\lambda$

$$h(v_\lambda) - h(u_\lambda) > f(v_\lambda) - f(u_\lambda) \quad (4.18)$$

De ces inégalités et en se basant sur les définitions des processus morphologiques non-locaux, nous pouvons écrire les inégalités suivantes

$$NLD(h)(u_\lambda) - h(u_\lambda) \geq NLD(f)(u_\lambda) - f(u_\lambda), \quad (4.19)$$

$$NLE(h)(u_\lambda) - h(u_\lambda) \geq NLE(f)(u_\lambda) - f(u_\lambda) \quad (4.20)$$

et enfin, en se basant sur l'inégalité stricte (4.18), nous pouvons écrire

$$NLM(h)(u_\lambda) - h(u_\lambda) > NLM(f)(u_\lambda) - f(u_\lambda) \quad (4.21)$$

Finalement, d'après les relations (4.19), (4.20) et (4.21), nous pouvons écrire

$$\begin{aligned} NLA(h)(u_\lambda) - h(u_\lambda) &> NLA(f)(u_\lambda) - f(u_\lambda) \\ h(u_\lambda) - h(u_\lambda) &> f(u_\lambda) - f(u_\lambda) \\ 0 &> 0. \end{aligned} \quad (4.22)$$

Ce qui est impossible. Ainsi, l'équation (4.13) ne peut avoir deux solutions distinctes.  $\square$

### Schéma numérique

Soit  $(f^n)_{n \in \mathbb{N}}$  une suite de fonctions construite telle que

$$\begin{cases} f^0(u) = g(u) & \forall u \in \partial^+ A \\ f^n(u) = g(u) & \forall n, \forall u \in \partial^+ A \\ f^{n+1}(u) = NLA(f^n)(u). & \forall u \in A \end{cases} \quad (4.23)$$

Il est trivial de montrer que si cette suite converge, sa limite quand  $n$  tend vers l'infini satisfait  $f(u) = NLA(f)(u)$  et donc l'équation (4.13).

## 4.3 Diffusion sur graphe par $p$ -laplacien

Dans cette section, nous étudions le problème de diffusion non-locale impliquant l'opérateur  $p$ -laplacien normalisé et nous montrons que le processus de diffusion associé peut être exprimé comme un processus de filtrage moyennneur non-local.

Soit  $f : \Omega \times [0, T] \rightarrow \mathbb{R}$ , ( $\Omega \subset \mathbb{R}^m$ ) une fonction. Nous considérons alors l'équation suivante

$$\begin{cases} \frac{\partial f(x,t)}{\partial t} = \Delta_p^N f(x,t) \\ f(x,t=0) = f_0(x), \end{cases} \quad (4.24)$$

et montrons qu'une discrétisation de cette équation par le  $p$ -laplacien normalisé sur graphe et par le temps mène à un système de diffusion sur graphe impliquant des opérateurs de filtrage non-locaux tels que NLE, NLD ou encore NLM.

En discrétisant la fonction  $f$  selon le temps, nous obtenons une fonction  $f : V \times \{0, \dots, N\} \rightarrow \mathbb{R}$ . On note  $f^k(u) = f(u, k), \forall k \in \{0, \dots, N\}$ . La dérivée de  $f$  selon le temps est alors

$$\begin{aligned} \partial_n f(u, n) &= f(u, n+1) - f(u, n) \\ &= f^{n+1}(u) - f^n(u). \end{aligned} \quad (4.25)$$

Puis, en discrétisant par l'opérateur  $p$ -laplacien normalisé sur graphe, l'équation (4.24) devient

$$\begin{aligned} f^{n+1}(u) - f^n(u) &= (\Delta_{w,\alpha,\beta} f^n)(u) \\ &= NLA(f^n)(u) - f^n(u) \\ f^{n+1}(u) &= NLA(f^n)(u). \end{aligned} \quad (4.26)$$

Finalement, nous obtenons le système suivant

$$\begin{cases} f^{n+1}(u) = NLA(f^n)(u) \\ f^0(u) = f_0(u), \end{cases} \quad (4.27)$$

qui décrit un processus de filtrage combinant un processus morphologique et un processus de filtrage moyenné.

## 4.4 Expérimentations

Dans cette section, nous illustrons le comportement de l'opérateur  $p$ -laplacien normalisé  $\Delta_{w,\alpha,\beta}$  appliqué à des problèmes inverses de restauration ou d'interpolation sur graphe. Les expériences présentées ici n'ont pas pour objectif de chercher à résoudre un problème particulier mais d'illustrer le potentiel de notre opérateur. Ainsi, les résultats présentés sont à considérer à titre qualitatif (illustrant une large variété d'applications sur une large variété de données) et non quantitatif ni comparatif.

### 4.4.1 Exemple jouet

Avant de présenter des applications plus concrètes de notre approche, nous illustrons le comportement de l'opérateur  $p$ -laplacien normalisé sur un exemple jouet. La figure 4.1 présente les résultats obtenus sur un graphe grille à partir de la résolution du problème de Dirichlet impliquant une fonction initiale  $g$  nulle sur l'ensemble du graphe, excepté le sommet correspondant au pixel central. Les résultats sont présentés pour plusieurs configurations de la fonction de poids ( $w = 1$  puis  $w = g_4$ ), et différentes combinaisons des paramètres  $\alpha$  et  $\beta$ . Nous remarquerons que la fonction se comporte comme une distance au point central, interpolant sous l'effet des paramètres  $\alpha$  et  $\beta$  entre une distance induite par la norme  $\mathcal{L}_\infty$  et une distance induite par la norme  $\mathcal{L}_2$ .

### 4.4.2 Restauration et simplification d'images

Dans ce paragraphe, nous montrons le comportement et l'efficacité de l'opérateur  $p$ -laplacien normalisé pour la restauration et la simplification de fonctions définies sur des graphes (images et maillages).

Les résultats présentés sont obtenus par l'application du processus de filtrage (4.27) sur différents types de graphes (image et maillage), pour quelques itérations et différentes combinaisons des paramètres  $\alpha$  et  $\beta$ .

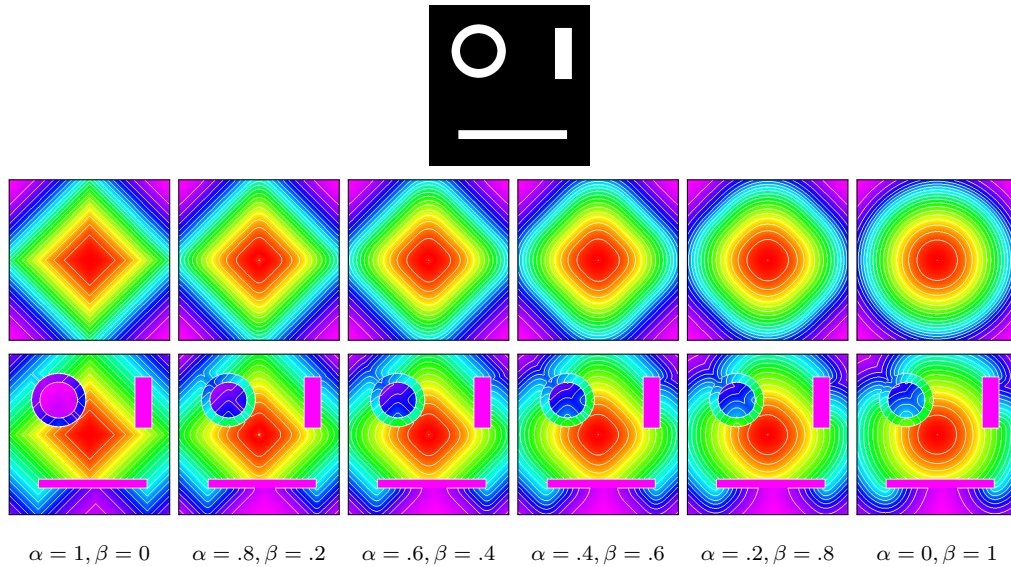


FIGURE 4.1 – Exemple jouet. La première ligne présente les résultats obtenus pour une fonction de poids constante ( $w = 1$ ). La seconde ligne présente les résultats obtenus pour une fonction de poids dépendant de l'image ( $w = g_4$ ). Voir le texte pour plus de détails.

### Simplification d'images scalaires

La figure 4.2 illustre les résultats obtenus sur une image bruitée artificiellement, ce qui nous permet d'observer à la fois l'intérêt pour le débruitage et la simplification. L'image est considérée comme une fonction  $f^0 : \mathbb{Z}^2 \rightarrow \mathbb{R}$  qui définit une application de l'ensemble  $V$  des sommets vers l'intensité des pixels. Les résultats sont présentés après 5 ou 20 itérations de l'algorithme, pour différentes combinaisons des paramètres  $\alpha$  et  $\beta$ , différentes constructions du graphe (locales ou non-locales) et différentes fonctions de poids. Le graphe local est construit comme un graphe de 4-connexité, et le graphe non-local est quant à lui construit à partir d'un voisinage  $15 \times 15$  où les pixels sont caractérisés par un patch de dimension  $5 \times 5$ . La première ligne illustre les résultats obtenus pour  $\alpha = 1$  et  $\beta = 0$ , ce qui correspond au laplacien infini non-local présenté au chapitre précédent. La seconde ligne illustre les résultats obtenus pour  $\alpha = 0$  et  $\beta = 1$ , ce qui correspond au laplacien normalisé. Enfin, la troisième ligne illustre les résultats pour  $\alpha = 0.5$  and  $\beta = 0.5$ , ce qui peut être interprété comme un opérateur filtre intermédiaire entre le laplacien et le laplacien infini non-local. Les colonnes 1 et 4 présentent les résultats obtenus sur le graphe local avec une fonction de poids constante. Les colonnes 2 et 5 présentent les résultats obtenus sur le même graphe mais avec une fonction de

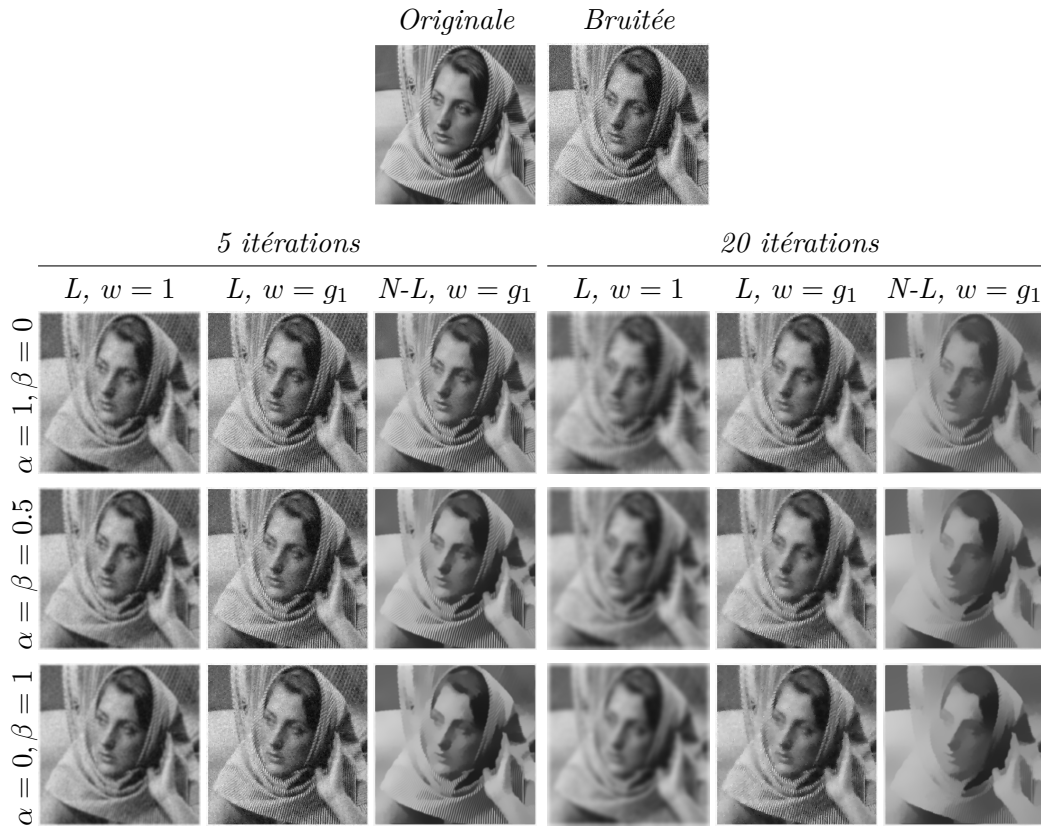


FIGURE 4.2 – Simplification d’images scalaires. Les résultats sont présentés pour 5 ou 20 itérations avec différentes combinaisons des paramètres  $\alpha, \beta$  et pour différentes fonctions de poids et constructions de graphes (local et non-local). Voir le texte pour plus de détails.

poids dépendant de l’intensité des pixels ( $w = g_4$ ). Enfin, les colonnes 3 et 6 présentent les résultats obtenus sur le graphe non-local avec une fonction de poids dépendant de l’intensité des pixels et utilisant des patches.

### Simplification de textures sur un maillage 3D

La figure 4.3 présente une seconde illustration impliquant un maillage 3D irrégulier et texturé. Le graphe utilisé correspond au maillage et la fonction  $f^0$  est une application qui à chaque sommet associe l’intensité moyenne de la texture à proximité de ce sommet. Les résultats sont présentés pour différentes combinaisons des paramètres  $\alpha$  et  $\beta$  et différentes fonctions de poids. La première colonne présente les résultats obtenus pour une unique itération et une fonction de poids constante ( $w = 1$ ). La seconde colonne présente les résultats

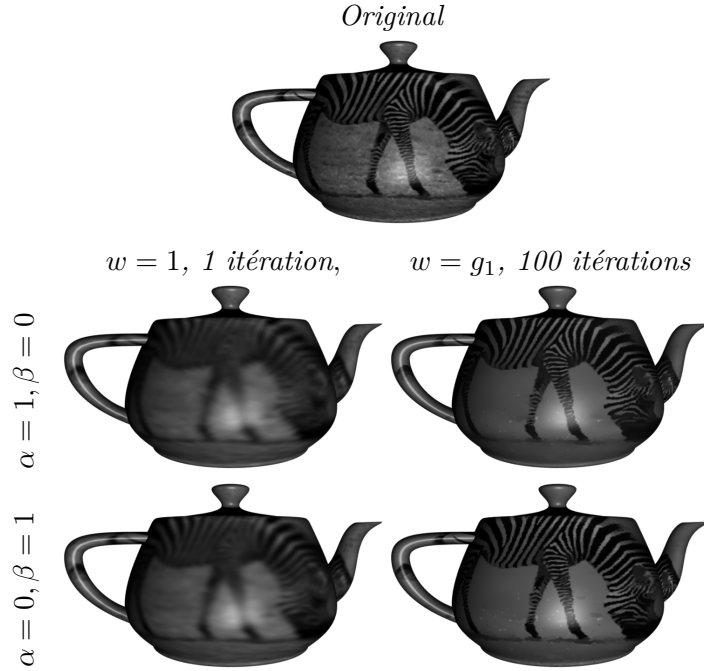


FIGURE 4.3 – Simplification de texture sur un maillage 3D. Les résultats sont présentés pour différentes combinaisons des paramètres  $\alpha$  et  $\beta$  et pour différentes fonctions de poids. Le graphe utilisé correspond à la structure du maillage de l'objet. La première colonne présente les résultats pour une fonction de poids constante. La seconde colonne présente les résultats pour une fonction de poids dépendant de l'intensité de la texture. Voir le texte pour plus de détails.

obtenus pour 100 itérations et une fonction de poids dépendant de l'intensité de la texture ( $w = g_4$ )

### 4.4.3 Interpolation

Dans cette section, nous présentons l'intérêt et le comportement de notre expression du  $p$ -laplacien pour des problèmes d'interpolation, tels qu'ils ont été définis dans le chapitre précédent (section 3.4).

Dans le cas du  $p$ -laplacien normalisé, le problème d'interpolation peut-être réduit au problème de Dirichlet suivant

$$\begin{cases} (\Delta_{w,\alpha,\beta} f)(u) = 0 & u \in V_0 \\ f(u) = g(u) & u \in V - V_0 = \partial V_0 \end{cases} \quad (4.28)$$



où  $V_0 \subset V$  est le sous-ensemble des sommets pour lesquels l'information à reconstruire est manquante. La fonction  $g$  représente l'information connue aux bords de  $V_0$  et dépend de l'application.

### Segmentation d'images naturelles

Dans ce paragraphe, nous illustrons le comportement du processus d'interpolation reposant sur l'opérateur  $p$ -laplacien normalisé pour la segmentation d'images. Nous montrons également l'intérêt d'un schéma non-local (comparé à un schéma local) pour ce type d'application, tout particulièrement lorsque les images sont texturées et présentent des structures fines et répétitives.

La figure 4.4 présente différents résultats de segmentation sur une image naturelle, obtenus pour deux constructions distinctes de graphes (une locale et l'autre non-locale) et différentes combinaisons des paramètres  $\alpha$  et  $\beta$ .

Le graphe non-local est construit en utilisant un voisinage de dimension  $11 \times 11$  et chaque pixel est caractérisé par un patch couleur de dimension  $3 \times 3$ .

### Comparaison avec l'état de l'art des méthodes de segmentation semi-supervisées basées sur des graphes

Dans ce paragraphe, nous présentons des résultats comparatifs de segmentation semi-supervisée entre notre approche et les méthodes les plus usuelles de la littérature et utilisant des graphes, comme les coupures de graphes, les marches aléatoires, le Power Watershed, les chemins minimaux et la ligne de partage des eaux. Les trois images utilisées pour cette expérience proviennent de la base grabcut. Afin d'augmenter la variation entre les résultats des différentes méthodes, nous utilisons une version érodée des marqueurs initiaux fournis avec les images tels qu'utilisés dans l'article de [Couprie et al. 11a]. Les résultats présentés pour les différentes méthodes de la littérature proviennent également de cet article. Afin de montrer l'efficacité de notre formulation, nous présentons le résultat obtenu avec la meilleure combinaison des paramètres  $\alpha$  et  $\beta$  et des configurations locales et non-locales des graphes utilisés. Il apparaît que les résultats obtenus avec notre méthode sont tout à fait compétitifs avec les résultats de la littérature, avec même une meilleure précision dans le cas des images texturées.

#### 4.4. Expérimentations

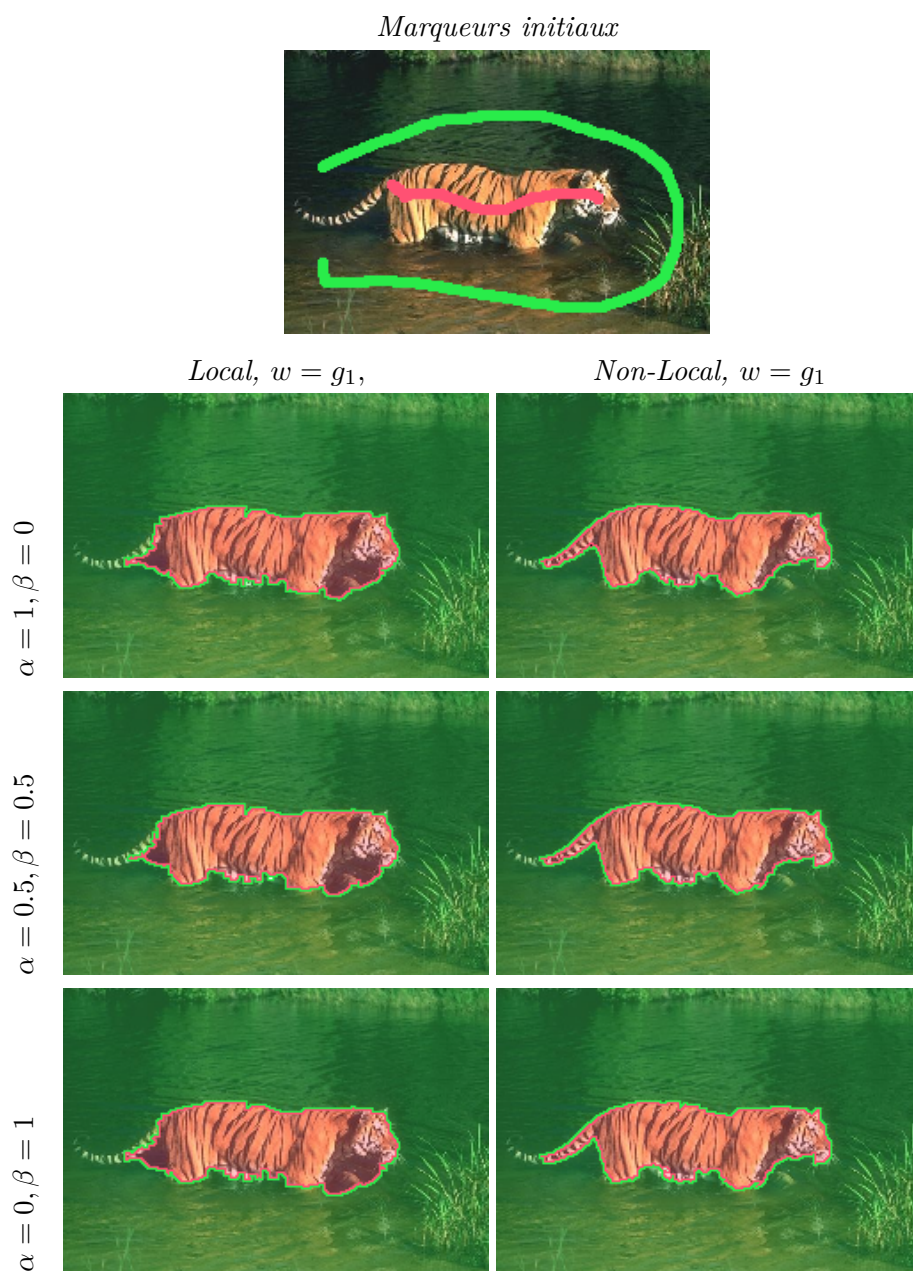


FIGURE 4.4 – Segmentation semi-supervisée d’images naturelles impliquant  $\Delta_{w,\alpha,\beta}$ . La première colonne présente les résultats obtenus pour un graphe local (un graphe grille), et la seconde présente les résultats obtenus pour un graphe non-local reposant sur un voisinage de dimension  $11 \times 11$ , où chaque pixel est représenté par un patch de dimension  $3 \times 3$ .

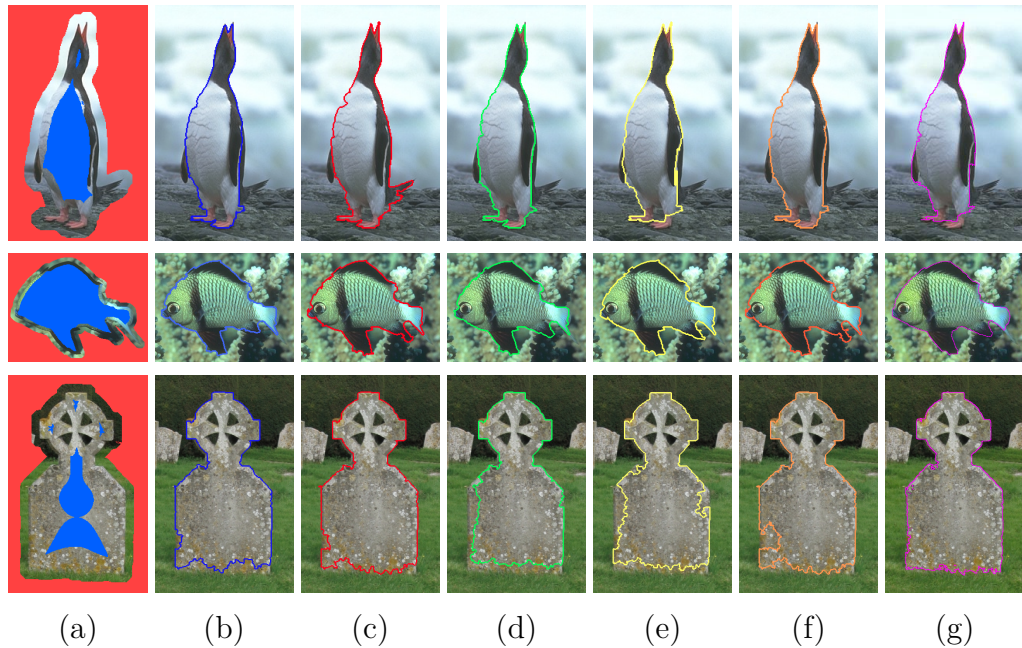


FIGURE 4.5 – Comparaison avec l'état de l'art des méthodes de segmentation semi-supervisées basées sur des graphes. (a) Les marqueurs initiaux. (b) Coupures de graphe. (c) Powerwatershed. (d) Random Walker. (e) Shortest Path. (f) Watershed. (g)  $p$ -laplacien normalisé. Voir le texte pour plus de détails.

### Segmentation d'images naturelles basée sur des RAG

Dans ce paragraphe, nous illustrons l'adaptabilité de notre formulation, au travers d'une application impliquant un RAG (voir la section 1.2.3 pour plus de détails sur ce type de graphes) .

Dans certains cas, en particulier lorsque les objets à extraire sont nombreux et non connexes, la représentation naturelle des images par un graphe grille rend impossible la segmentation. Aussi, nous proposons de remplacer la représentation de l'image via un graphe de 4-connexité, par un graphe plus petit et possédant des arêtes entre les parties similaires et non connectées de l'image. Pour cela, nous utilisons une version simplifiée de l'image (une partition) sur laquelle nous construisons un RAG. Afin de connecter les régions distantes, le graphe est étendu de telle sorte qu'une arête supplémentaire soit ajoutée entre chaque région et ses deux régions les plus similaires (au sens de la couleur moyenne) sur l'ensemble de la partition de l'image. La figure 4.6 illustre l'intérêt d'une telle approche pour l'extraction de noyaux cellulaires (en bleu foncé) depuis un extrait de lame cytologique. Ces noyaux sont nombreux

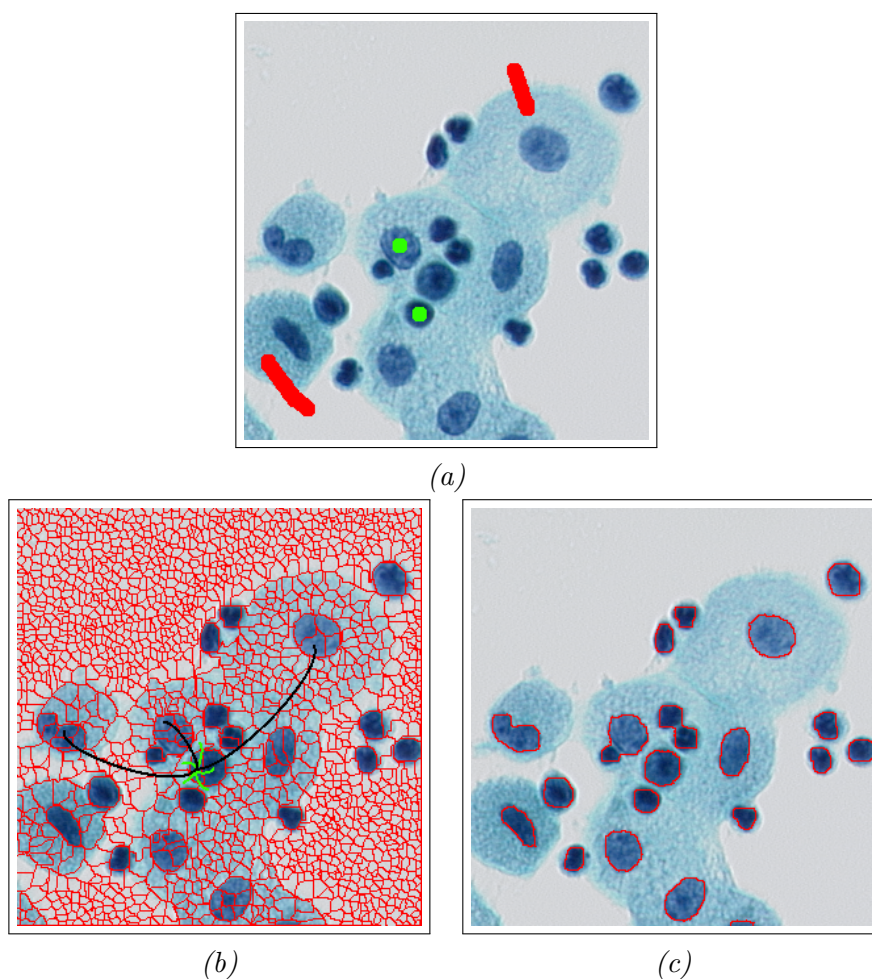


FIGURE 4.6 – Segmentation d’images naturelles basée sur des RAG. (a) L’image originale, avec les marqueurs initiaux des deux classes (fond en rouge et noyaux en vert). (b) La partition de l’image en régions (les frontières entre les régions sont représentées en rouge). Le RAG étendu est illustré pour un sommet. En vert ses arêtes locales (connectant les régions adjacentes) et en noir les arêtes non-locales (connectant les régions les plus similaires). (c) Le résultat de la segmentation. Voir le texte pour plus de détails.

et non connectés spatialement car séparés par le cytoplasme (en bleu clair) et le fond (en blanc). L’image originale est simplifiée et réduite à une partition en régions (respectant les discontinuités internes de l’image) où la couleur de chaque région correspond à la couleur moyenne des pixels à l’intérieur de celle-ci. Le graphe est construit comme un RAG à partir de cette carte de régions, et étendu afin de connecter les régions similaires mais non connectées (deux



arêtes sont ajoutées pour chaque sommet). L'extraction des noyaux en elle-même est effectuée par l'interpolation d'une fonction de marqueurs (initialisée par l'utilisateur) et pour des valeurs de  $\alpha = 1$  et  $\beta = 0$  (ce qui correspond au laplacien infini). Nous pouvons remarquer que grâce aux arêtes additionnelles tous les noyaux sont extraits.

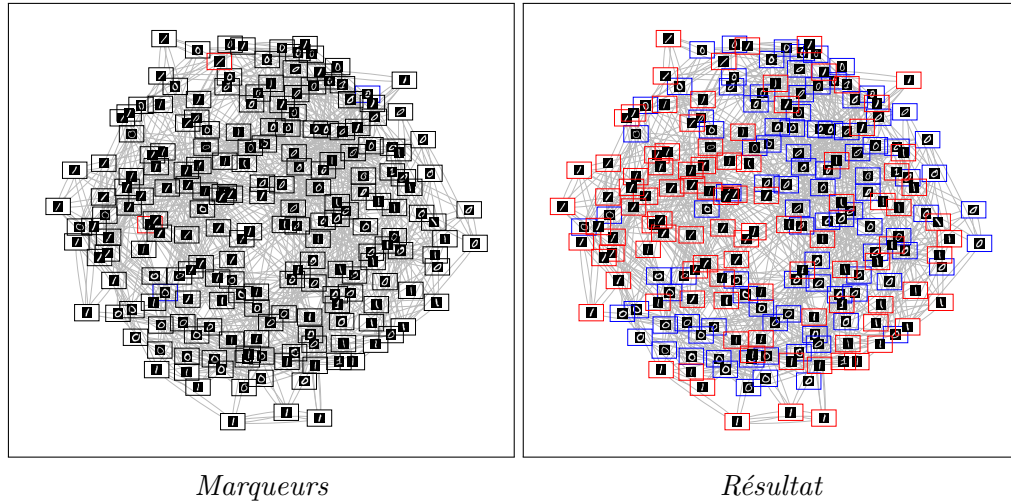


FIGURE 4.7 – Classification semi-supervisée de données réelles.

### Classification semi-supervisée de données réelles

Dans ce paragraphe, nous présentons une illustration de la classification de données réelles par l'interpolation d'une fonction de marqueurs définie sur graphe. Les données sont un ensemble de 200 imagerie représentant des zéros et des uns manuscrits et qui proviennent de la base des données des services postaux USPS. Le graphe est un graphe des  $k$ -ppv (avec  $k = 5$ , voir la section 1.2.2 pour plus de détails sur ce type de graphes), et pour simplifier le calcul de la fonction de poids chaque imagerie est représentée par le vecteur des intensités de ses  $28 \times 28$  pixels (dans ce cas, il n'est pas nécessaire d'utiliser d'autres descripteurs). L'interpolation est effectuée avec  $\alpha = 1$  et  $\beta = 0$ . Le graphe avec les marqueurs initiaux (fournis par l'utilisateur) ainsi que le résultat de la classification sont présentés à la figure 4.7.

### Segmentation de maillage 3D texturé

La figure 4.8 présente enfin une dernière illustration de l'adaptabilité de notre approche, par un exemple d'extraction d'objets d'intérêt depuis une

texture plaquée sur un maillage 3D. Le graphe est exactement le maillage 3D et la fonction de poids dépend de la couleur moyenne à proximité des sommets du maillage ( $w = g_4$ ). Ici encore l'objet est extrait par l'interpolation d'une fonction de marqueurs initialisée par l'utilisateur (avec  $\alpha = 1$  et  $\beta = 0$ ).

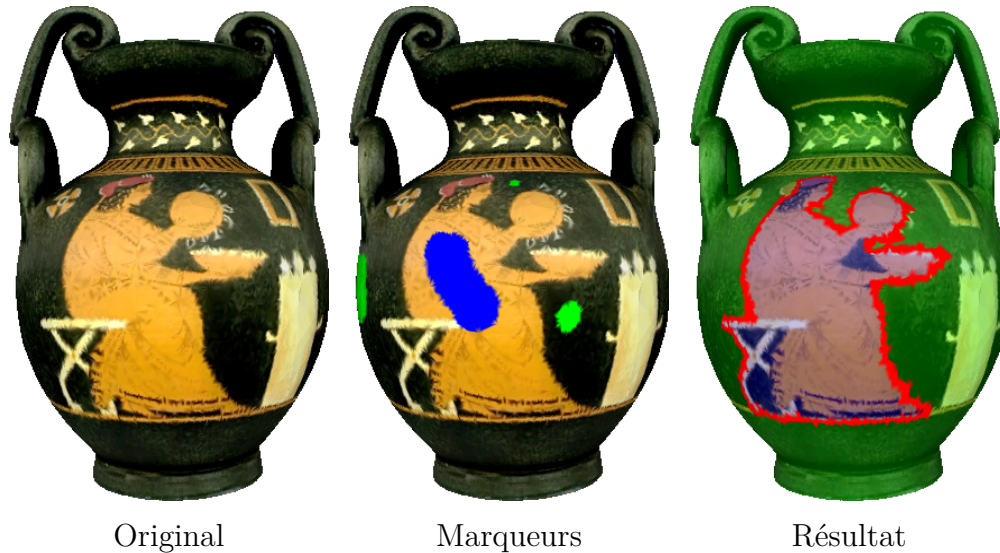


FIGURE 4.8 – Segmentation de maillage 3D texturé à partir de deux marqueurs initiaux placés par un utilisateur. La figure résultat présente l'objet extrait en bleu avec un contour rouge.

### Inpainting

Dans ce paragraphe, nous illustrons l'intérêt et le comportement de notre équation d'interpolation (4.28) appliquée à l'inpainting (voir la section 3.4.1 pour plus de détails concernant l'inpainting).

La figure 4.10 présente différents résultats d'inpainting sur une image naturelles. Ces résultats ont été obtenus avec différentes combinaisons des paramètres  $\alpha$  et  $\beta$  et pour différentes constructions de graphes (locale et non-locale). Le graphe local est un simple graphe grille (4-connexité), et le graphe non-locale est construit à partir d'un voisinage de dimension  $31 \times 31$  pour des patches de dimension  $15 \times 15$ . Nous pouvons remarquer la nette différence de reconstruction entre le graphe local et le graphe non-locale. En effet, de par l'utilisation d'un voisinage étendu et de patches, la reconstruction reposant sur un graphe non-locale respecte bien mieux la texture et produit un résultat plus cohérent (et donc plus difficile à détecter).

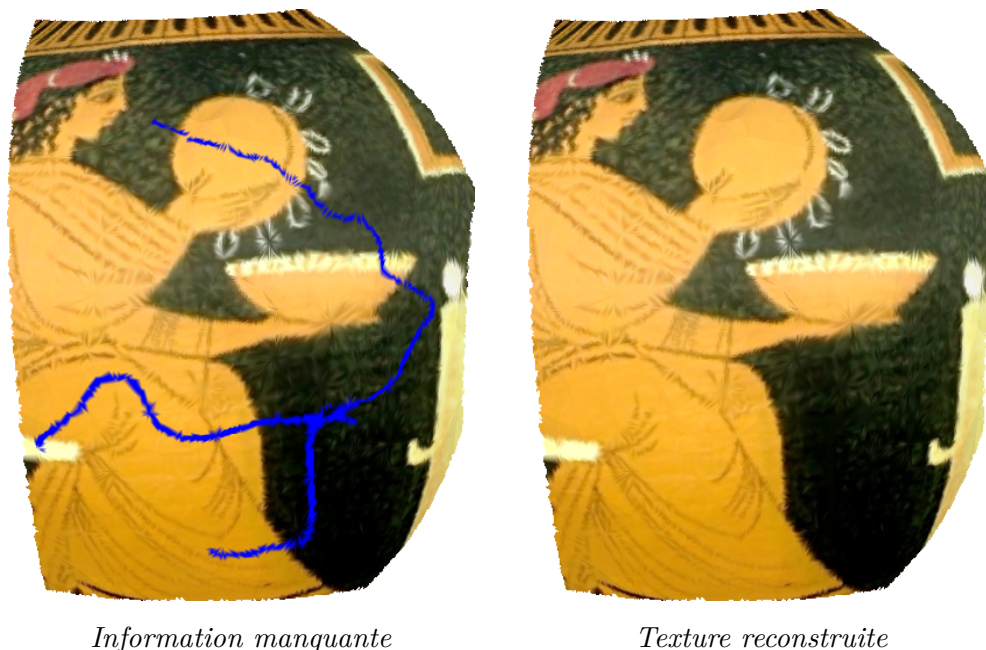


FIGURE 4.9 – Inpainting sur un maillage 3D. La partie à reconstruire est présentée en bleu et simule la présence d'une fissure sur un vase ancien.

La figure 4.9 présente un second résultat d'inpainting, appliqué cette fois à la texture d'un maillage 3D. Le maillage correspond à un fragment du vase présenté à la figure 4.8 sur lequel une partie de la texture a été volontairement masquée (en bleu) pour simuler une fissure. Le résultat présenté a été obtenu pour  $a = 0, \beta = 1$ .

## 4.5 Conclusion

Dans ce chapitre, nous avons proposé une nouvelle formulation, non-locale, du  $p$ -laplacien normalisé sur graphes. Cette formulation peut être considérée comme l'analogue discret du  $p$ -laplacien des jeux défini en continu et s'exprime sous la forme d'une combinaison linéaire du laplacien normalisé et du laplacien infini non-local.

Nous avons montré que pour certaines constructions de graphes cet opérateur coïncide avec certaines EDP non-locales continues et coïncide également avec les fonctions de jeu d'une version bruitée du jeu du Tug-of-War.

Nous avons également montré que l'opérateur  $p$ -laplacien normalisé peut être interprété comme un opérateur morphologique combinant des processus

#### 4.5. Conclusion

---

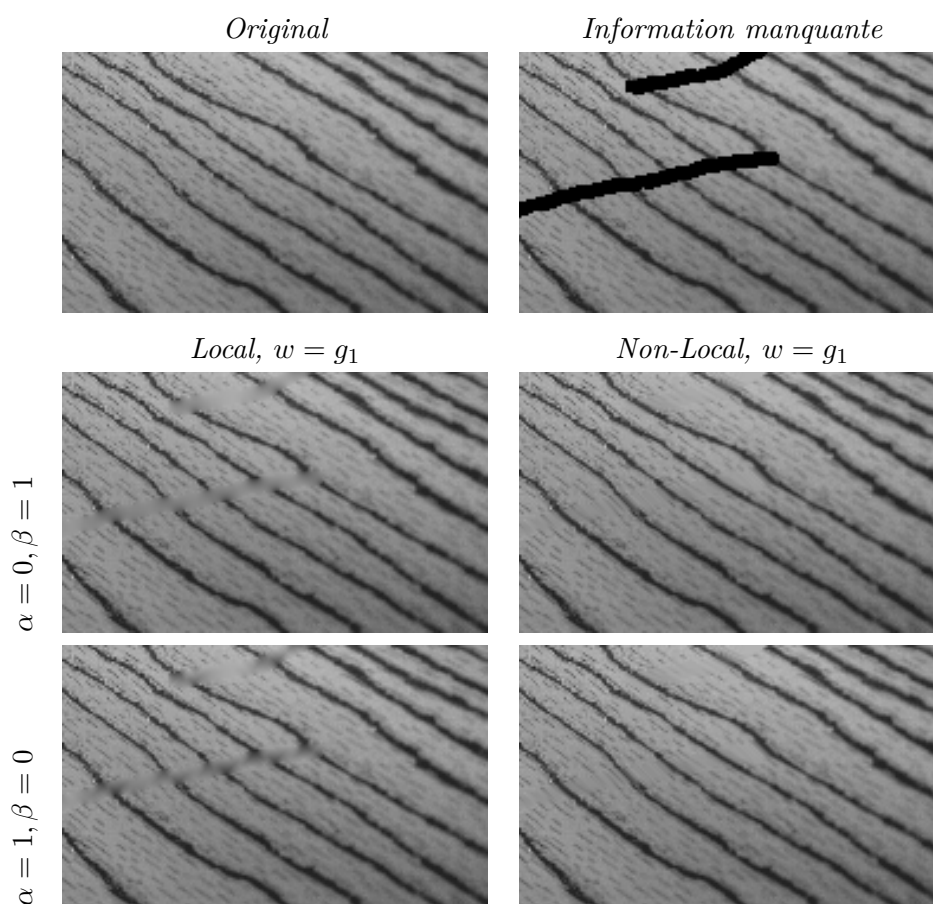


FIGURE 4.10 – Inpainting sur une image naturelle. Les résultats présentés sont obtenus avec différentes combinaisons des paramètres  $\alpha$  et  $\beta$  et pour différentes constructions de graphes (locale et non-locale). La colonne de gauche présente les résultats pour un graphe local (grille graphe). La seconde colonne présente les résultats obtenus avec un graphe non-local utilisant un voisinage de dimension  $31 \times 31$  et des patches de dimension  $15 \times 15$ .

d'érosion et de dilatation non-locaux et un filtre moyenneur non-local. Cette interprétation est basée sur la définition des opérateurs morphologiques non-locaux introduits dans la première partie de ce manuscrit.

Enfin, nous avons proposé une étude du problème de Dirichlet associé à notre opérateur et montré que ce problème possède une solution unique. De même, nous avons proposé une étude du problème de diffusion impliquant l'opérateur  $p$ -laplacien normalisé non-local et montré que le processus de diffusion associé peut être exprimé comme un processus de filtre moyenneur non-local.



Finalement, nous avons illustré l'application de cet opérateur à des problèmes d'interpolation et de diffusion pour le traitement des images, des maillages et des données représentées par des graphes. Ici encore, les résultats présentés le sont à titre qualitatif et non quantitatif; ceci afin d'illustrer l'adaptabilité de notre formalisme à différents types de données et pour des applications variées.

Ce chapitre clôt la partie de ce manuscrit consacrée à l'étude d'une nouvelle famille d'opérateurs  $p$ -laplaciens discrets sur graphes.

## Troisième partie

# Équations de propagation de fronts sur graphe



# Introduction

Les EDP et approches variationnelles par ensembles de niveaux pour la propagation de fronts, introduites par [Osher et Sethian 88], ont trouvé des applications dans de nombreux domaines comme la mécanique des fluides, la géométrie algorithmique, le traitement d'image, la vision par ordinateur, l'infographie, ou encore la science des matériaux.

Les raisons principales de leur succès sont l'adaptativité de ces méthodes pour des problèmes variés, la capacité à gérer efficacement les changements de topologie et la garantie de l'existence de leurs solutions. De plus, des algorithmes efficaces basés sur des équations de Hamilton-Jacobi ont été développés, ainsi que des schémas numériques stables pour leur implémentation.

Ces techniques ont été introduites en vision par ordinateur pour des tâches de segmentation [Caselles et al. 97, Kichenassamy et al. 95, Chan et Vese 01] et étendues pour des tâches de reconstruction stéréo [Faugeras 97], de suivi d'objets [Paragios et Deriche 02] ou encore de reconnaissance de formes [Leventon et al. 00]. Ces méthodes ont également été adaptées sur des maillages et certains graphes particuliers [Kimmel et Sethian 98].

Les solutions de ces équations sont usuellement basées sur une discrétisation de l'équation de Hamilton-Jacobi où l'approximation du gradient est effectuée selon les méthodes de Godunov [Leveque 02] ou Lax-Friedrich [Sethian 99]. Lorsque la propagation du front ne dépend pas de caractéristiques géométriques, ces équations se ramènent à des équations plus simples comme l'équation eikonale. De nombreux schémas numériques ont été proposés et étudiés pour résoudre ces équations. Parmi eux, nous pouvons citer le schéma itératif basé sur la méthode du point fixe proposé par [Rouy et Tourin 92], la méthode du Fast Sweeping introduite par [Zhao 05], l'adaptation de l'algorithme de Dijkstra proposée par [Tsitsiklis 95], le très populaire Fast Marching de [Sethian 99] ou encore son extension, le Generalized Fast Marching [Carlini et al. 08].

D'autres approches ont également été développées pour résoudre ces équations sur certains domaines irréguliers comme les domaines triangulaires des maillages 3D. Nous pouvons citer les adaptations du Fast Marching par

[Kimmel et Sethian 98] et du Fast Sweeping [Qian et al. 07] aux domaines triangulaires. Cependant, à notre connaissance et hormis l'adaptation de l'équation eikonale non-stationnaire proposée par [Ta et al. 09a], ces méthodes n'ont jamais été utilisées pour l'analyse et le traitement de données définies sur des graphes de topologie arbitraire.

Aussi, poursuivant notre objectif de transcription d'EDP définies sur des domaines continus au domaines des graphes pour le traitement des données, nous proposons d'étendre les équations d'ensembles de niveaux pour la propagation d'un front et définies en continu au domaine des graphes pondérés.

Dans le chapitre 5, nous rappelons les équations et schémas numériques de la méthode des ensembles de niveaux utilisés pour simuler la propagation d'un front sur un domaine Euclidien.

Dans le chapitre 6, nous présentons une adaptation discrète de l'équation continue non-stationnaire des ensembles de niveaux au domaine des graphes de topologie arbitraire. Cette adaptation repose sur une interprétation morphologique de l'équation continue et s'écrit comme sous la forme d'un processus morphologique impliquant les opérateurs de dilatation et d'érosion sur graphe. Nous proposons également un schéma numérique de résolution de cette équation ainsi qu'un algorithme optimisé.

Puis, dans le chapitre 7, nous présentons une version stationnaire de l'équation discrète introduite au chapitre précédent en nous restreignant au cas où la vitesse est positive sur l'ensemble du graphe. Cette équation présente une forte analogie avec l'équation eikonale et nous montrons que pour des graphes euclidiens particuliers notre formulation coïncide avec certaines discrétisations de cette équation. Nous proposons également des schémas numériques de résolution de cette équation ainsi qu'un algorithme rapide implémentant ces schémas.

Dans le chapitre 8, nous proposons d'étendre notre formulation discrète pour la propagation d'un front unique au cas où plusieurs fronts évoluent simultanément sur un même graphe. Nous montrons que cette extension repose sur une représentation unifiée des fronts et une version modifiée de l'équation stationnaire proposée au chapitre précédent. Cette extension ouvre la voie à de nombreuses applications, pour le traitement des images des maillages et des données, que nous illustrons.

Finalement, dans le chapitre 9, nous étendons une nouvelle fois notre formulation pour considérer le cas plus général où plusieurs fronts évoluent simultanément sous l'influence de vitesses pouvant être positives ou négatives. Nous étudions tout d'abord le cas d'un front unique puis nous généralisons l'approche proposée aux cas où plusieurs fronts évoluent simultanément. Cette

---

approche générale ouvre également la voie à de nombreuses applications illustrées dans ce chapitre.



# Chapitre 5

## Équation de propagation d'un front

### Sommaire

---

<b>5.1</b>	<b>Équation du mouvement</b>	<b>88</b>
<b>5.2</b>	<b>Approche par ensembles de niveaux</b>	<b>89</b>
<b>5.3</b>	<b>Schémas numériques</b>	<b>91</b>
<b>5.4</b>	<b>Algorithmes</b>	<b>93</b>

---

Dans ce chapitre, nous rappelons l'équation de propagation d'un front selon la direction de la normale et son implémentation par la méthode des ensembles de niveaux.

Nous présentons ainsi l'approche des ensembles de niveaux introduite et popularisée par [Osher et Sethian 88] pour décrire l'évolution d'un front sur un domaine euclidien qui présente de nombreux avantages comme l'insensibilité aux changements de topologie et qui peut être aisément appliqué pour des grilles de toutes dimensions. Nous rappelons l'expression des équations de Hamilton-Jacobi décrivant l'évolution des fonctions d'ensembles de niveaux, ainsi que le lien entre la version stationnaire de ces équations et l'équation eikonale.

Nous présentons ensuite les schémas numériques les plus usuels pour approximer les solution des différentes formulations stationnaires et non-stationnaires de ces équations. Ces schémas reposent sur des différences finies permettant de calculer une valeur approchée de l'opérateur gradient et ont donné lieu à des algorithmes variés pour le traitement des images et la simulation numérique. Nous présentons également ces algorithmes.



## 5.1 Équation du mouvement

Dans cette section, nous rappelons l'équation classique du mouvement d'un front évoluant sur un domaine euclidien continu. Afin de simplifier les écritures, nous prenons l'exemple d'un front se propageant dans le plan, mais une version similaire de cette équation existe pour des domaines de dimensions supérieures.

Nous considérons un front  $\Gamma$  évoluant sur un plan du domaine euclidien  $\Omega \subset \mathbb{R}^2$ . À un instant  $t \in [0, \infty[$ , ce front est représenté par une courbe paramétrique notée  $\gamma(t)$ . Le front  $\gamma$  se propage sur  $\Omega$  sous l'effet d'un champ de vecteurs normaux  $\vec{N}(s, t)$  et d'une fonction vitesse  $\mathcal{F}$ . Généralement, cette vitesse est définie en fonction du domaine considéré et de la courbure. La courbe  $\gamma(t)$  est paramétrée par un vecteur position noté  $\vec{X}(s, t)$  défini par

$$\begin{cases} \vec{X}(s, t) = (x(s, t), y(s, t)) & , \forall s \in [0, S] \\ \vec{X}(0, t) = \vec{X}(S, t) \end{cases} \quad (5.1)$$

La physique Newtonienne décrit l'évolution d'un tel front aux travers de l'équation du mouvement suivante :

$$\vec{N}(s, t) \cdot \frac{\partial \vec{X}(s, t)}{\partial t} = \mathcal{F}(s, t). \quad (5.2)$$

En fonction des coordonnées du plan, l'équation de mouvement peut être réécrite sous la forme des Lagrangiens suivants :

$$\begin{aligned} \frac{\partial x}{\partial t} &= \mathcal{F} \frac{\frac{\partial y}{\partial s}}{\sqrt{\frac{\partial^2 y}{\partial s^2} + \frac{\partial^2 x}{\partial s^2}}} \\ \frac{\partial y}{\partial t} &= - \mathcal{F} \frac{\frac{\partial x}{\partial s}}{\sqrt{\frac{\partial^2 y}{\partial s^2} + \frac{\partial^2 x}{\partial s^2}}} \end{aligned} \quad (5.3)$$

Les premières méthodes de résolution numérique appliquées à la propagation d'un front [Zabusky et Overman 84] sont basées sur la discrétisation de ces équations. Elles utilisent une représentation discrète du front, échantillonné sous forme de marqueurs, dont la position est mise à jour au cours du temps en fonction des approximations de l'équation du mouvement. Malgré une certaine efficacité pour simuler l'évolution de courbes avec peu de singularités, [Sethian 85] a montré que ces techniques présentent de nombreux désavantages, parmi lesquels une trop grande erreur d'approximation, l'apparition de singularités, et la difficulté (voire l'impossibilité) de gérer les changements de topologie (lorsque

la courbe se démultiplie, ou fusionne). De plus, il est nécessaire de développer des méthodes distinctes pour la propagation d'un front dans le plan et l'espace.

Les lacunes de ces méthodes, ainsi que celles d'autres approches comme l'algorithme SLIC [Noh et Woodward 76], ont conduit [Osher et Sethian 88, Sethian 90] à introduire une nouvelle formulation considérant une représentation abstraite et de dimension supérieure de la courbe, permettant de passer outre ces limitations. C'est l'approche dite par ensembles de niveaux, que nous allons détailler dans la section suivante.

## 5.2 Approche par ensembles de niveaux

Cette approche a été introduite et popularisée par [Osher et Sethian 88], et repose sur l'idée de représenter un front  $\Gamma$  par le niveau zéro d'une fonction d'ensembles de niveau  $\phi(x, y, t)$ . Ainsi, considérant un instant  $t$  et en utilisant cette approche, la position du front est donnée par l'ensemble des points  $(x, y)$  de l'espace tels que  $\phi(x, y, t) = 0$ , ou bien

$$\gamma(t) = \{(x, y) \mid \phi(x, y, t) = 0\} \quad (5.4)$$

La fonction d'ensembles de niveaux est une fonction lisse continue  $\phi : \mathbb{R} \times \mathbb{R} \times [0, \infty[ \rightarrow \mathbb{R}$  vérifiant les propriétés suivantes.

$$\phi(x, y, t) \begin{cases} = 0 & \text{si } (x, y) \in \gamma(t) \\ > 0 & \text{si } (x, y) \text{ est à l'intérieur de la courbe } \gamma(t) \\ < 0 & \text{si } (x, y) \text{ est à l'extérieur de la courbe } \gamma(t) \end{cases} \quad (5.5)$$

**Remarque.** Nous considérons toujours ici un front se propageant dans le plan, mais les notations et équations peuvent simplement être étendues à toute dimension supérieure.

L'approche par ensembles de niveaux présente l'avantage de ne pas nécessiter de paramétrer la courbe puisque. Par la suite, seule la fonction d'ensembles de niveaux (dans laquelle est implicitement représentée la courbe paramétrique) sera considérée.

L'équation du mouvement du front  $\Gamma$  devient donc dans cette approche une équation d'évolution de la fonction  $\phi$  au cours du temps. Cette équation non-stationnaire s'écrit sous la forme du Hamiltonien

$$H(x, y, \nabla\phi) = \frac{\partial\phi(x, y, t)}{\partial t} - \mathcal{F}\|\nabla\phi(x, y, t)\| = 0. \quad (5.6)$$

Cette formulation présente plusieurs avantages tels que l'insensibilité aux changements de topologie de la courbe  $\gamma(t)$  ou encore la possibilité de l'utiliser aussi bien pour suivre une courbe évoluant dans le plan, qu'une surface évoluant dans l'espace, et peut être étendue à toute dimension supérieure [Osher et Sethian 88, Malladi et al. 95].

### Lien avec l'équation eikonale

Dans le cas où le signe de la vitesse est constant et ne dépend que de la position, l'équation des ensembles de niveaux peut s'écrire sous la forme d'une équation stationnaire,

$$\mathcal{F}\|\nabla\mathcal{T}\| = 1, \quad (5.7)$$

où  $\mathcal{T} : \Omega \rightarrow \mathbb{R}^+$  est la fonction qui associe à chaque point du domaine  $\Omega$  l'instant d'arrivée du front en ce point [Sethian 99]. Le lien entre les fonctions  $\phi$  et  $\mathcal{T}$ , et de même le lien entre les équations (5.6) et (5.7), sont donnés par les relations suivantes,

$$\mathcal{T}(x, y) = t - \phi(x, y, t). \quad (5.8)$$

En se basant sur cette relation, et puisque la fonction  $\mathcal{T}$  est indépendante par rapport au temps  $t$ , l'équation non-stationnaire (5.6) peut donc être réécrite telle que :

$$\begin{aligned} \frac{\partial\phi(x, y, t)}{\partial t} - \mathcal{F}(x, y)|\nabla\phi(x, y, t)| &= 0 \\ \frac{\partial(t - \mathcal{T}(x, y))}{\partial t} - \mathcal{F}(x, y)|\nabla(t - \mathcal{T}(x, y))| &= 0 \\ \mathcal{F}(x, y)|\nabla\mathcal{T}(x, y)| &= \frac{\partial(t - \mathcal{T}(x, y))}{\partial t}, \end{aligned} \quad (5.9)$$

Enfin, en dérivant le membre de droite, nous retrouvons l'équation stationnaire (5.7). Réécrite sous la forme d'un Hamiltonien, et en introduisant une condition initiale, cette équation devient

$$\begin{cases} H(x, y, \nabla\mathcal{T}) = \mathcal{F}(x, y)\|\nabla\mathcal{T}(x, y)\| - 1 = 0, & (x, y) \in \Omega \setminus S \\ \mathcal{T}(x, y) = \psi(x, y), & (x, y) \in S \subset \Omega \end{cases} \quad (5.10)$$

qui correspond à l'équation eikonale.

L'équation eikonale est une équation issue de l'optique géométrique qui connaît une certaine popularité dans le domaine du traitement d'images [Bruss 82, Malladi et al. 95, Kimmel et Sethian 98, Jeong et Whitaker 08]. De manière

générale, l'équation eikonale est un cas particulier des équations stationnaires de Hamilton-Jacobi définies sur un domaine  $\Omega \subset \mathbb{R}^n$  par

$$\begin{cases} H(x, \nabla f) = 0, & x \in \Omega \setminus S \\ f(x) = \psi(x), & x \in S \subset \Omega, \end{cases} \quad (5.11)$$

où  $\psi$  est une fonction positive définie sur le domaine  $\Omega$  et  $f$  est une fonction de distance depuis la source  $S$ . Exprimée sous la forme d'un Hamiltonien, l'équation eikonale s'écrit

$$H(x, \nabla \mathcal{T}) = \|\nabla \mathcal{T}(x)\| - \frac{1}{\mathcal{F}(x)} = 0. \quad (5.12)$$

Finalement, en remplaçant cette équation dans le système (5.11), la version stationnaire de l'équation eikonale est donnée par le système

$$\begin{cases} \mathcal{F}(x)\|\nabla \mathcal{T}(x)\| = 1, & x \in \Omega \setminus S \\ \mathcal{T}(x) = \psi(x), & x \in S \subset \Omega. \end{cases} \quad (5.13)$$

Ainsi, l'équation du mouvement d'un front par la méthode des ensembles de niveaux, peut-être considérée comme un simple cas particulier des équations stationnaires de Hamilton-Jacobi.

L'équation eikonale (5.13) peut également être écrite sous la forme d'une équation non-stationnaire de Hamilton-Jacobi

$$\begin{cases} \frac{\partial \mathcal{T}(x,t)}{\partial t} = \frac{1}{\mathcal{F}(x)} - \|\nabla \mathcal{T}(x)\| = 1, & x \in \Omega \setminus S \\ \mathcal{T}(x, 0) = \psi_0(x), & x \in S \subset \Omega. \end{cases} \quad (5.14)$$

où  $\psi_0$  est une fonction de valeurs initiales définie sur l'ensemble du domaine. À convergence, la solution de ce système (5.14) satisfait l'équation stationnaire (5.13). Dans la suite de ce chapitre, nous négligerons cette formulation pour ne considérer que l'équation stationnaire.

Dans la section suivante, section 5.3, nous présentons différents schémas de discrétisation de l'équation par ensembles de niveaux (5.6) et de son pendant stationnaire (5.7) définis sur des grilles régulières. Ces schémas permettent d'approximer numériquement la solution de l'équation du mouvement associée.

## 5.3 Schémas numériques

Dans cette section, nous présentons les schémas numériques les plus usuels pour approximer la solution des différentes formulations de l'équation du

mouvement d'un front. Tout particulièrement le schéma de Godunov, dit vers l'avant, et ses adaptations. Ces schémas reposent tous sur des discrétisations différentes de l'opérateur gradient et sont basées sur des différences finies.

### Différences finies

Avant toute chose, nous rappelons les notations standard des différences finies en deux dimensions qui sont utilisées dans ces schémas :

Les différences finies avant (*upwind*)

$$D_{i,j}^{+x} f = \frac{f(i+1, j) - f(i, j)}{\Delta x}, \quad D_{i,j}^{+y} f = \frac{f(i, j+1) - f(i, j)}{\Delta y} \quad (5.15)$$

Les différences finies arrière (*downwind*)

$$D_{i,j}^{-x} f = \frac{f(i, j) - f(i-1, j)}{\Delta x}, \quad D_{i,j}^{-y} f = \frac{f(i, j) - f(i, j-1)}{\Delta y} \quad (5.16)$$

Les différences finies centrées

$$D_{i,j}^x f = \frac{f(i+1, j) - f(i-1, j)}{\Delta x}, \quad D_{i,j}^y f = \frac{f(i, j+1) - f(i, j-1)}{\Delta y} \quad (5.17)$$

### Schéma de Lax-Friedrich

Le schéma de Lax-Friedrich est le schéma numérique le plus simple pour approximer les équations aux dérivées partielles. Il repose sur une approximation de la dérivée par des différences finies centrées (5.17). Ce schéma est monotone et respecte la loi de conservation mais présente l'inconvénient de lisser la courbe du front au fur et à mesure de la propagation. Ceci introduit une erreur d'approximation supplémentaire.

Avec ce schéma (au premier ordre et en dimension 2) l'équation non-stationnaire (5.6), devient

$$\phi(t+1) = \phi(t) - \Delta t \mathcal{F} \frac{1}{2} \left( (D_{i,j}^x \phi(t))^2 + (D_{i,j}^y \phi(t))^2 \right)^{\frac{1}{2}} \quad (5.18)$$

De même, l'équation stationnaire (5.7) devient

$$\frac{1}{2} \left( (D_{i,j}^x \mathcal{T})^2 + (D_{i,j}^y \mathcal{T})^2 \right) = \frac{1}{\mathcal{F}^2}. \quad (5.19)$$

### Schéma de Godunov

Le schéma proposé par Godunov et repris par [Rouy et Tourin 92], repose quand à lui sur une approximation de la dérivée par des différences finies avant et arrière (5.15), (5.16). Ce schéma reprend l'idée de considérer la solution comme une fonction constante par morceaux et de résoudre localement l'équation sur chaque intervalle.

Avec ce schéma (au premier ordre et en dimension 2) l'équation (5.6), devient

$$\phi(t+1) = \phi(t) - \Delta t \mathcal{F} \left[ \max(D_{i,j}^{-x}\phi(t), -D_{i,j}^{+x}\phi(t), 0)^2 + \max(D_{i,j}^{-y}\phi(t), -D_{i,j}^{+y}\phi(t), 0)^2 \right]^{\frac{1}{2}} \quad (5.20)$$

De même, l'équation stationnaire (5.7) devient

$$\max(D_{i,j}^{-x}\mathcal{T}, -D_{i,j}^{+x}\mathcal{T}, 0)^2 + \max(D_{i,j}^{-y}\mathcal{T}, -D_{i,j}^{+y}\mathcal{T}, 0)^2 = \frac{1}{\mathcal{F}^2} \quad (5.21)$$

### Schéma d'Osher-Sethian

Ce schéma proposé par [Osher et Sethian 88] est une version modifiée du schéma de Godunov. Dans ce cas, l'équation (5.6) devient,

$$\phi(t+1) = \phi(t) - \Delta t \mathcal{F} \left[ \max(D_{i,j}^{-x}\mathcal{T}, 0)^2 + \min(D_{i,j}^{+x}\mathcal{T}, 0)^2 + \max(D_{i,j}^{-y}\mathcal{T}, 0)^2 + \min(D_{i,j}^{+y}\mathcal{T}, 0)^2 \right]^{\frac{1}{2}}. \quad (5.22)$$

De même, l'équation stationnaire (5.7) devient

$$\max(D_{i,j}^{-x}\mathcal{T}, 0)^2 + \min(D_{i,j}^{+x}\mathcal{T}, 0)^2 + \max(D_{i,j}^{-y}\mathcal{T}, 0)^2 + \min(D_{i,j}^{+y}\mathcal{T}, 0)^2 = \frac{1}{\mathcal{F}^2} \quad (5.23)$$

À partir de ces schémas, notamment le schéma d'Osher-Sethian, de nombreux algorithmes ont été développés pour résoudre de manière efficace la version stationnaire de l'équation du mouvement. Nous en présentons trois parmi les plus récentes et populaires dans la section suivante.

## 5.4 Algorithmes

Dans cette section, nous présentons trois méthodes parmi les plus populaires pour la résolution de l'équation stationnaire : Le Fast sweeping introduit

par Zhao et Tsai [Tsai et al. 04], le Fast Marching introduit par Sethian [Sethian 99] et plus récemment la Fast Iterative Method proposée par Jeong et Whitaker [Jeong et Whitaker 08]. Nous présentons également une approche récente de résolution de l'équation générale, introduite par [Carlini et al. 06], et généralisant l'algorithme du Fast Marching dans le cas où le signe de la vitesse peut être positif ou négatif.

### Fast Sweeping

L'algorithme du Fast Sweeping [Tsai et al. 04, Zhao 05] repose sur la méthode de résolution itérative de Gauss-Seidel où la valeur de la fonction en chaque noeud de la grille est mise à jour itérativement jusqu'à converger vers une valeur minimum correspondant à la solution. À cette fin, l'algorithme effectue un balayage (*sweeping*) de la grille selon quatre ordres de parcours prédéfinis (en dimension 2). Cet algorithme reprend une idée initiale de [Danielsson 80] qui a proposé un algorithme pour calculer la distance Euclidienne sur une grille 2D en visitant chaque noeud de la grille dans un ordre pré-établi.

### Fast Marching

L'algorithme du Fast Marching se base sur la particularité du front à se propager de manière monotone dans la direction de la normale, de telle sorte que les sommets sont atteints dans l'ordre croissant des valeurs de la fonction  $\mathcal{T}$ . Ainsi, l'algorithme suit l'évolution du front en résolvant à chaque itération l'équation (5.23) à partir des plus petites valeurs de  $\mathcal{T}$ .

Sur une grille régulière, l'algorithme consiste en une liste de nœuds actifs (*active list*) pour lesquels la solution est déjà connue et fixée, ainsi qu'en une bande mince (*narrow band*) de nœuds adjacents aux nœuds actifs et pour lesquels il existe une approximation de la solution. Les nœuds restants sont dits "au loin" (*far away*). La bande mince repose sur une structure de pile ordonnée et croissante de telle manière qu'à chaque itération de l'algorithme le premier nœud de la pile est retiré de la bande mince et ajouté à la liste des nœuds actifs. L'estimation de la solution en ce nœud est garantie comme exacte et une nouvelle approximation est calculée pour l'ensemble des nœuds adjacents. Les nœuds adjacents qui étaient au loin sont ajoutés à la bande mince et la position de ceux qui y étaient déjà présents est mise à jour en fonction de la nouvelle valeur de l'approximation de la solution. Cette suite d'opérations est répétée jusqu'à ce que la bande mince soit vide. Le lecteur intéressé par plus de détails sur cet algorithme peut se référer à [Sethian 99].

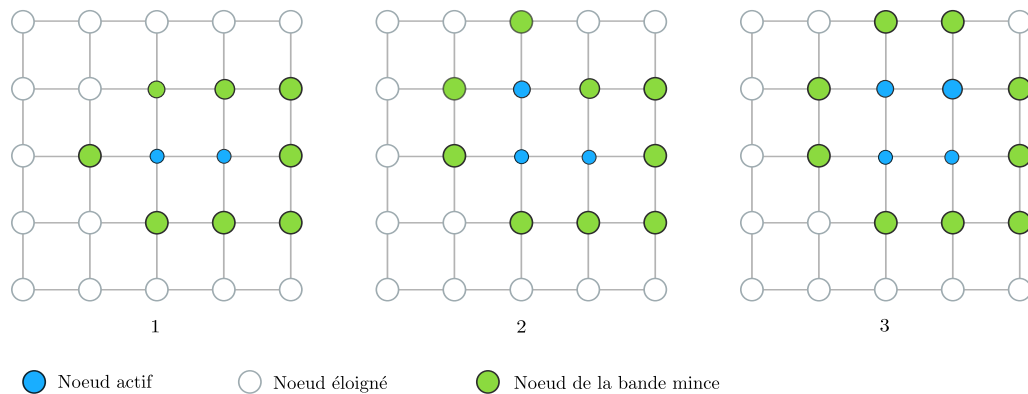


FIGURE 5.1 – Illustration de l’algorithme Fast Marching au travers de quelques étapes. Les sommets actifs (pour lesquels l’instant d’arrivée est connu et fixé) sont représentés en bleu et les sommets de la bande mince sont représentés en vert. À chaque étape, le sommet de la bande mince pour lequel l’instant d’arrivée (symbolisé par le diamètre du cercle) est minimum est retiré de la bande mince et ajouté aux sommets actifs. Ses voisins sont ajoutés à la bande mince.

### Fast Iterative Method

L’algorithme Fast Iterative Method introduit par [Jeong et Whitaker 08] repose lui aussi sur une bande mince de nœuds au voisinage des nœuds pour lesquels la solution est déjà connue, mais sans relation d’ordre entre ces nœuds. Il n’entretient donc pas de structure de données pour ordonner la bande mince, contrairement au Fast Marching. À chaque itération de l’algorithme, une solution approchée est estimée simultanément en chaque nœud de la bande mince et les nœuds pour lesquels la solution converge sont fixés et retirés de la bande. Au contraire, les voisins de ces nœuds, pour lesquels la solution ne converge pas, sont ajoutés à la bande. Contrairement au Fast Marching, cet algorithme n’est pas monotone et un nœud fixé peut être replacé dans la bande mince si les nouvelles solutions estimées à son voisinage le rendent à nouveau candidat. Ceci est dû aux variations locales de la direction de propagation du front.

Le principal avantage de cet algorithme tient au fait qu’il puisse être parallélisé (contrairement au Fast Marching) puisque la mise à jour des sommets situés dans la bande mince peut être effectuée simultanément.



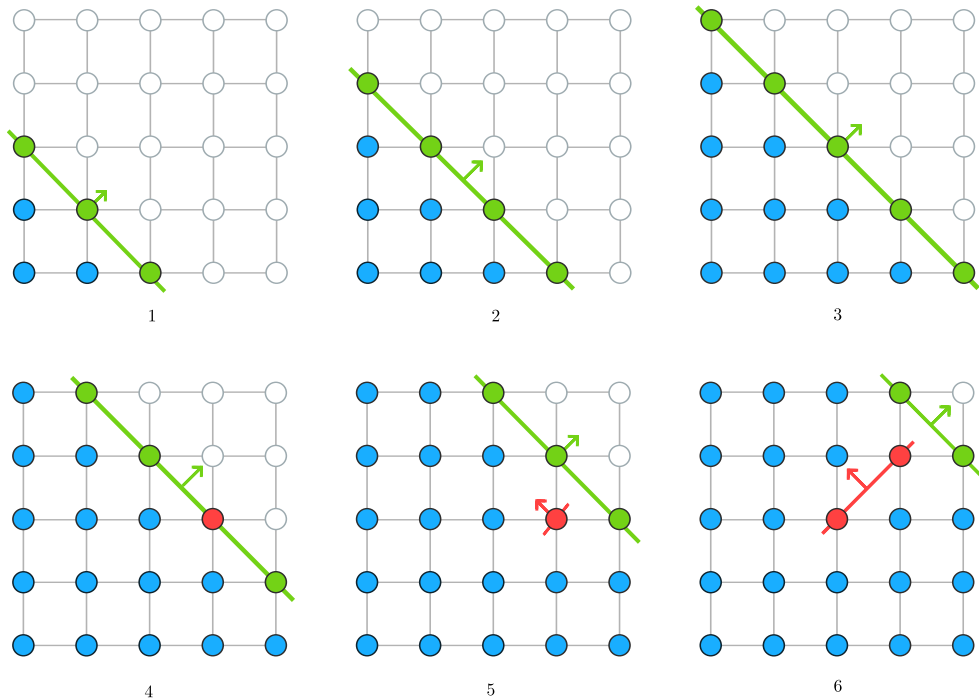


FIGURE 5.2 – Illustration de l’algorithme Fast Iterative Method. En bleu, les sommets fixés. En vert et rouge, les sommets de la bande mince. Lors des trois premières itérations, l’instant d’arrivée est mis à jour pour l’ensemble des sommets de la bande mince, illustrant une propagation du front dans la direction de la flèche verte. À l’étape 4, la solution au sommet rouge ne converge pas et nécessite de conserver ce sommet dans la bande mince. À l’itération 6, et en raison de la nouvelle configuration en ce sommet, les voisins (en rouge) de ce sommet sont réactivés, formant un second pseudo front. Ce second pseudo front correspond à un changement local de la direction de propagation du front.

### Generalized Fast Marching Method

Nous l’avons vu, la plupart des algorithmes proposés pour résoudre l’équation du mouvement d’un front, via la méthode des ensembles de niveaux, se concentrent sur le cas particulier où le signe de la vitesse est constant et strictement positif (ce qui est le cas des trois algorithmes présentés ci-dessus).

L’algorithme du Generalized Fast Marching [Carlini et al. 08, Forcadel et al. 08], supprime les restrictions sur le signe de la vitesse et étend l’algorithme du Fast Marching au cas plus général où le front peut être soumis à une vitesse positive ou négative. Cette approche repose sur une séparation du front en deux parties : la portion du front soumise à une vitesse positive d’une part

## 5.4. Algorithmes

et la portion du front soumise à une vitesse négative d'autre part. Les deux portions sont différenciées par le signe d'une fonction caractéristique  $\theta$  définie sur le domaine. Ainsi, l'algorithme du Fast Marching est étendu et muni de deux ensembles de points actifs. L'un associé au front positif et correspondant à l'ensemble des points dont l'instant d'arrivée est déjà connu et tels que  $\theta$  soit positive. L'autre est au contraire associé au front négatif et correspondant à l'ensemble des points dont l'instant d'arrivée est déjà connu et tels que  $\theta$  soit négative. De même, à chacun de ces ensembles actifs est associé une bande mince. Le déroulement de l'algorithme reste le même que dans la version initiale, à savoir que le candidat possédant le plus petit instant d'arrivée estimé est ajouté à l'un des ensembles actifs et les bandes minces sont mises à jour.

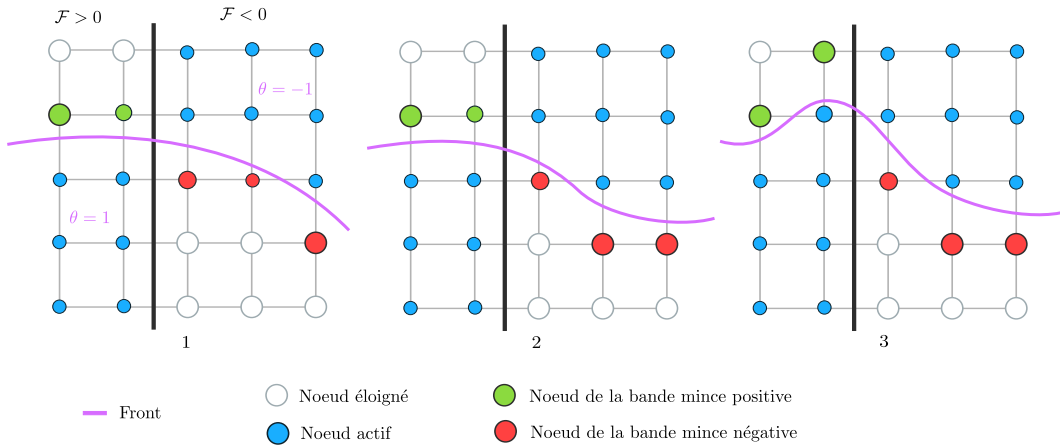


FIGURE 5.3 – Illustration de l'algorithme du Generalized Fast Marching au travers de quelques étapes. Les sommets actifs (pour lesquels l'instant d'arrivée est connu et fixé) sont représentés en bleu, les sommets de la bande mince de la partie de front mue par une vitesse positive sont représentés en vert et les sommets de la bande mince de la partie du front mue par une vitesse négative sont représentés en rouge. À chaque étape, le sommet des bandes minces pour lequel l'instant d'arrivée (symbolisé par le diamètre du cercle) est minimum est retiré de la bande mince et ajouté aux sommets actifs. Ses voisins sont ajoutés à la bande mince correspondante.



# Chapitre 6

## Formulation discrète : Équation non-stationnaire

### Sommaire

---

<b>6.1</b>	<b>Représentation discrète d'un front</b>	<b>100</b>
<b>6.2</b>	<b>Analogie avec la méthode des ensembles de niveaux</b>	<b>101</b>
6.2.1	Interprétation morphologique	102
6.2.2	Adaptation discrète	103
6.2.3	Relation avec les schémas numériques classiques	104
<b>6.3</b>	<b>Approximation de la solution</b>	<b>105</b>
6.3.1	Cas d'une vitesse constante dans le temps	106
6.3.2	Extension pour des vitesses dépendant du temps	109
<b>6.4</b>	<b>Algorithme</b>	<b>110</b>
6.4.1	Optimisation	112
<b>6.5</b>	<b>Expérimentations</b>	<b>113</b>
6.5.1	Exemple jouet	113
6.5.2	Évolution selon la courbure moyenne	115
6.5.3	Contours actifs	118
<b>6.6</b>	<b>Conclusion</b>	<b>118</b>

---

Dans ce chapitre, nous proposons d'adapter l'équation du mouvement d'un front, telle que définie par l'approche des ensembles de niveaux, au cas d'un front évoluant sur un graphe pondéré et de topologie arbitraire. Cette généralisation de l'équation du mouvement d'un front repose sur une interprétation morphologique de l'équation non-stationnaire continue présentée dans le chapitre précédent et qui permet une transcription directe de l'équation au domaine discret des graphes pondérés. Notre approche repose sur la famille

des gradients directionnels dont les liens avec les processus morphologiques discrets ont été montrés.

La transcription de l'équation du mouvement d'un front au domaine des graphes permet de renforcer les approches de résolution connues de l'équation du mouvement sur des domaines discrets particuliers (comme des grilles) en les étendant à tout type de domaine organisé ou non organisé, via un formalisme discret mimant le formalisme continu.

Nous présentons ainsi dans ce chapitre une adaptation discrète et sur graphe de l'équation non-stationnaire des ensembles de niveaux, ainsi qu'un schéma numérique de résolution associé à une preuve de convergence de ce schéma sous certaines conditions. Nous présentons également quelques illustrations de l'application de cette adaptation pour l'évolution d'un front sur plusieurs types de graphes irréguliers.

Ce chapitre est une première étape vers une formulation plus complète des équations de propagation de fronts sur graphe, qui sera étoffée au fil des chapitres de cette partie.

## 6.1 Représentation discrète d'un front

Nous considérons un front évoluant sur un graphe comme une interface séparant ce graphe en deux sous parties : les sommets déjà atteints par le front que nous désignerons par le terme intérieur du front, et les sommets qui ne sont pas encore atteints que nous désignerons par le terme extérieur du front. Soient  $\mathcal{G} = (V, E, w)$  un graphe pondéré et  $\Gamma$  un front évoluant sur  $\mathcal{G}$ . À l'instant  $t$ , le front  $\Gamma$  est donc défini comme l'interface entre les deux sous ensembles  $\Omega_t \subset V$  et  $\overline{\Omega}_t \subset V$ , représentant respectivement les noeuds à l'intérieur et à l'extérieur du front  $\Gamma$ . Par définition (un sommet étant nécessairement soit à l'intérieur, soit à l'extérieur), la relation entre ces deux sous-ensembles est donnée par :

$$\begin{aligned}\Omega_t \cup \overline{\Omega}_t &= V \\ \Omega_t \cap \overline{\Omega}_t &= \emptyset.\end{aligned}\tag{6.1}$$

Étant donné cette relation, nous considérerons par la suite un front  $\Gamma$  comme un sous-ensemble  $\Omega_t$ , et ne ferons référence explicitement à  $\overline{\Omega}_t$  qu'en tant que complémentaire de l'intérieur du front.

À l'instant  $t$ , le front  $\Gamma$  est implicitement représenté par une fonction caractéristique  $\psi_t : V \rightarrow \{-1, 1\}$ , définie par :

$$\psi_t = \mathcal{X}_{\Omega_t} - \mathcal{X}_{\overline{\Omega}_t}\tag{6.2}$$

où la fonction  $\mathcal{X}_\Omega : V \rightarrow \{0, 1\}$  est définie par

$$\mathcal{X}_\Omega(u) = \begin{cases} 0 & \text{si } u \notin \Omega \\ 1 & \text{si } u \in \Omega. \end{cases} \quad (6.3)$$

Ainsi, la fonction  $\psi_t(u)$  vaut 1 si  $u$  est situé à l'intérieur du front, et  $-1$  si  $u$  est situé à l'extérieur du front.

**Remarque.** *Une telle représentation du front, sous la forme d'un ensemble de sommets, permet de manipuler tous types de fronts, sans contrainte sur leur topologie, ni sur celle du domaine (le graphe) sur lequel ils se propagent. Cette représentation présente également l'avantage d'être robuste et particulièrement souple aux phénomènes de changement de topologie du front (lorsque celui-ci se démultiplie ou fusionne).*

L'évolution du front  $\Gamma$  se fait sous l'influence d'une vitesse  $\mathcal{F} : V \times \mathbb{R} \rightarrow \mathbb{R}$ , de telle manière que la direction de propagation du front est contrôlée par le signe de  $\mathcal{F}$ . Le front s'étend (i.e., des sommets sont ajoutés à  $\Omega_t$ ) lorsque ce signe est positif et se rétracte lorsqu'il est négatif (i.e., des sommets sont retirés de  $\Omega_t$ ). Cette représentation discrète d'un front est illustrée à la figure 6.1.

## 6.2 Analogie avec la méthode des ensembles de niveaux

Dans cette Section, nous montrons que l'équation du mouvement, telle qu'écrite dans l'approche par ensembles de niveaux, peut-être interprétée sous la forme d'un processus morphologique d'évolution. Puis, en se basant sur cette interprétation morphologique, nous proposons une adaptation discrète de l'approche par ensembles de niveaux et ainsi de l'équation du mouvement d'un front. Cette adaptation est basée sur les opérateurs morphologiques discrets définis dans la première partie de ce manuscrit. Finalement, nous montrons que sur un graphe particulier cette adaptation coïncide avec certains schémas de discrétisation de l'équation continue présents dans la littérature.

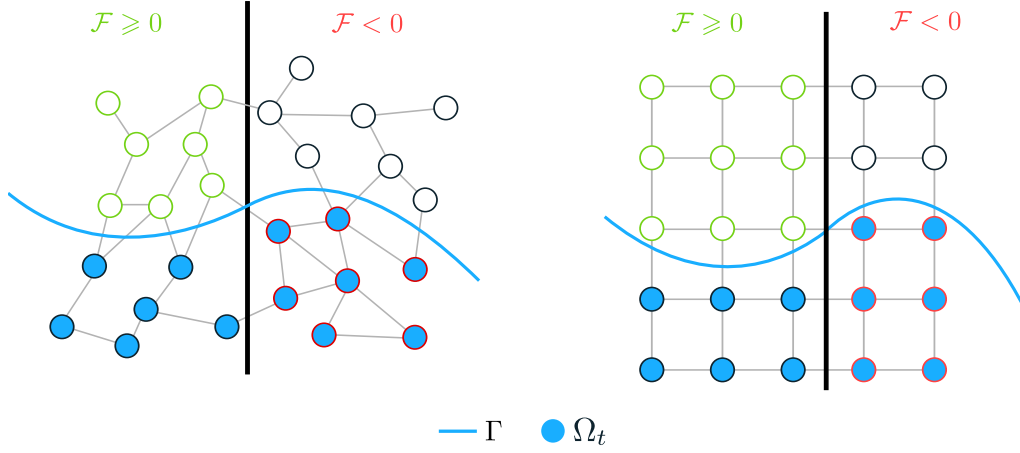


FIGURE 6.1 – Représentation discrète d'un front. À gauche, un graphe arbitraire, à droite un graphe grille. Dans les deux cas, le front et l'ensemble des sommets caractérisant ce front sont représentés en bleu. Les sommets susceptibles d'être atteints par le front et ajoutés à  $\Omega_t$  sous l'effet d'une vitesse positive sont représentés entourés de vert, et les sommets susceptibles d'être quittés par le front et retirés de l'ensemble  $\Omega_t$  sont représentés entourés de rouge.

### 6.2.1 Interprétation morphologique

Nous rappelons que la dilatation et l'érosion planaires définies par des équations aux dérivées partielles, d'une fonction  $f$  par des disques unitaires et induites par la norme  $\mathcal{L}_2$  sont respectivement

$$\frac{\partial \delta(f)}{\partial t} = +\|\nabla f\| \quad \text{et} \quad (6.4)$$

$$\frac{\partial \varepsilon(f)}{\partial t} = -\|\nabla f\|. \quad (6.5)$$

Dans le cas où la valeur absolue de la vitesse est constante et égale à 1, nous pouvons montrer que l'équation continue (5.6), est équivalente à l'un ou l'autre de ces opérateurs (en fonction du signe de  $\mathcal{F}$ ). En effet, étant donné une vitesse  $\mathcal{F}$  constante et  $|\mathcal{F}| = 1$ , et en considérant l'équation (5.6) dans chacune des deux configurations du signe de la vitesse, nous obtenons

$$\frac{\partial\phi(x,t)}{\partial t} = + |\mathcal{F}(x,t)| \|\nabla\phi(x,t)\|, \quad \text{si } \mathcal{F}(x,t) > 0 \quad (6.6)$$

$$\frac{\partial\phi(x,t)}{\partial t} = - |\mathcal{F}(x,t)| \|\nabla\phi(x,t)\|, \quad \text{si } \mathcal{F}(x,t) < 0 \quad (6.7)$$

qui sont respectivement équivalentes à une dilatation (6.4) de la fonction  $\phi$  dans le cas où la vitesse est positive et à une érosion (6.5) de la fonction  $\phi$  dans le cas où la vitesse est négative.

Cette interprétation de la propagation d'un front comme un ensemble de processus morphologiques unitaires, [Arehart et al. 93], coïncide avec le principe de Huygens selon lequel la propagation d'une onde peut être considérée comme la propagation simultanée d'un ensemble d'ondelettes qui se renforcent sur l'onde principale (le front). Dans notre cas, chaque ondelette peut-être réduite à un disque unitaire.

**Remarque.** *Le principe de Huygens est une des pièces à la base des conditions de viscosités utilisées pour résoudre l'équation eikonale (5.13) [Sethian 99].*

## 6.2.2 Adaptation discrète

Par analogie avec la formulation continue, nous représentons un front  $\Gamma$  comme le niveau zéro implicite d'une fonction d'ensembles de niveaux  $\phi : V \rightarrow \mathbb{R} \times \mathbb{R}$  vérifiant les propriétés

$$\phi(u,t) = \begin{cases} < 0 & \text{si } u \notin \Omega_t \\ \geq 0 & \text{si } u \in \Omega_t \end{cases} \quad (6.8)$$

Considérant l'interprétation morphologique de l'équation des ensembles de niveaux, nous pouvons transcrire directement l'équation continue en utilisant les équations aux différences partielles définies sur des graphes grâce aux processus morphologiques associés [Ta et al. 11].

### Équation du mouvement

Ainsi, en généralisant à toute vitesse et à toute norme la relation entre l'équation d'ensembles de niveaux et les opérateurs morphologiques, le pendant



discret de l'équation du mouvement d'un front par ensembles de niveaux peut s'écrire comme la combinaison de dilatation et érosion suivante

$$\frac{\partial \phi(u, t)}{\partial t} = \begin{cases} \mathcal{F}(u, t) \left\| \left( \nabla_w^+ \phi(u, t) \right) \right\|_p, & \text{si } \mathcal{F}(u, t) > 0 \\ \mathcal{F}(u, t) \left\| \left( \nabla_w^- \phi(u, t) \right) \right\|_p, & \text{si } \mathcal{F}(u, t) < 0 \\ 0, & \text{sinon.} \end{cases} \quad (6.9)$$

Cette formulation coïncide avec l'interprétation intuitive du mouvement d'un front discret, où l'évolution selon une vitesse positive consistant à ajouter les sommets adjacents au front correspond à une dilatation de l'ensemble  $\Omega_t$ . De même, l'évolution selon une vitesse négative consistant à retirer des sommets du front coïncide naturellement avec un processus d'érosion.

### Schéma numérique

À partir de la formulation discrète de l'équation des ensembles de niveaux (6.9), et en utilisant une discrétisation temporelle, nous obtenons le schéma de résolution itératif suivant

$$\begin{cases} \phi(u, t + \Delta t) &= \phi(u, t) \\ &+ \Delta t \left[ + \left( \mathcal{F}(u, t) \right)^+ \left\| \left( \nabla_w^+ \phi \right) (u, t) \right\|_p \right. \\ &\quad \left. - \left( \mathcal{F}(u, t) \right)^- \left\| \left( \nabla_w^- \phi \right) (u, t) \right\|_p \right] \\ \phi(u, 0) &= \phi^0(u) \end{cases} \quad (6.10)$$

où  $\phi^0 : V \rightarrow \mathbb{R}$  décrit la position du front à l'instant initial, et  $\Delta t$  correspond au pas de discrétisation temporelle.

Contrairement aux méthodes basées sur les équations aux dérivées partielles, aucune étape de discrétisation spatiale n'est nécessaire aux expressions discrètes des opérateurs différentiels.

### 6.2.3 Relation avec les schémas numériques classiques

Suivant la même démarche que [Ta et al. 08a], nous montrons dans ce paragraphe que sur une structure de graphe particulière, notre formulation discrète de l'équation du mouvement coïncide avec le schéma numérique de premier ordre proposé par [Sethian 99].

Soit  $\mathcal{G} = (V, E, w)$  un graphe pondéré représentant une grille régulière dans  $\mathbb{R}^m$ . Puisque nous sommes dans l'espace  $\mathbb{R}^m$ , nous associons à chaque sommet  $u$  de  $V$  un vecteur de coordonnées spatiales défini tel que  $u = (i_1 h_1, \dots, i_m h_m)^T$

### 6.3. Approximation de la solution

---

où  $i_j \in \mathbb{N}$  avec  $j = 1, \dots, m$  et où  $h_j$  correspond à l'espacement de la grille sur l'axe  $j$ . Ainsi, le voisinage d'un sommet  $u$  est défini comme  $N(u) = \{v \mid v = u \pm h_j e_j, \text{ avec } j = 1, \dots, m\}$  et où  $e_j = (q_k)_{1, \dots, m}^T$  est un vecteur de  $\mathbb{R}^m$  tel que  $q_k = 1$  si  $j = k$  et  $q_k = 0$  sinon.

La fonction de poids est définie de manière à être cohérente avec le pas de discrétisation  $h_j$ , telle que  $w_{uv} = \frac{1}{h_j} \forall v = u + h_j e_j$ .

Considérant une norme  $\mathcal{L}_2$ , et en utilisant une discrétisation temporelle, l'équation s'écrit

$$\phi^{n+1}(u) = \phi^n(u) + \Delta t \mathcal{F}(u, t) \sqrt{\sum_{j=1}^m \max\left(\frac{\phi^n(u \pm h_j e_j) - \phi^n(u)}{h_j}, 0\right)^2} \quad (6.11)$$

Puis, en utilisant les conventions de notations des différences finies avant et arrière,

$$D_u^{+j} f = \frac{f(u + h_j e_j) - f(u)}{h_j} \quad (6.12)$$

$$D_u^{-j} f = \frac{f(u) - f(u - h_j e_j)}{h_j}, \quad (6.13)$$

ainsi que la relation  $\min(0, x) = -\max(0, -x)$ , nous obtenons le schéma de discrétisation de premier ordre proposé par [Osher et Sethian 88] (5.22).

$$\phi^{n+1}(u) = \phi^n(u) + \Delta t \mathcal{F}(u, t) \sqrt{\sum_{j=1}^m \max\left(D_u^{+j} \phi^n, 0\right)^2 + \min\left(D_u^{-j} \phi^n(u), 0\right)^2} \quad (6.14)$$

## 6.3 Approximation de la solution

En l'état, résoudre l'équation (6.10) à tout instant  $t$  nécessite de connaître la position de  $\phi$  à l'instant  $t - 1$ . Ce qui, par extension, implique de connaître la solution de l'équation à l'instant initial et donc d'avoir un fort a priori sur l'instant d'arrivée du front en chaque point.

**Remarque.** Si et seulement si la fonction  $\phi_0$  est une fonction de distance au front à l'instant initial (avec une distance négative à l'extérieur du front), alors la fonction  $\phi$  obtenue par résolution de l'équation (6.10) coïncide avec la position du front à tout instant  $t$ .

Afin de pallier ce problème, nous proposons de simuler l'évolution du front via une approximation de la position du front à l'instant initial et l'introduction d'un temps artificiel.

### Approximation de la fonction $\phi$ à l'instant initial

Puisque la fonction  $\phi$  est inconnue, nous proposons de l'approximer par une fonction vérifiant la propriété première d'une fonction d'ensembles de niveaux, à savoir que cette fonction doit être négative à l'extérieur du front, positive à l'intérieur et nulle sur le front. À l'instant initial, une fonction vérifiant ces propriétés existe et correspond à la fonction caractéristique du front à l'instant initial (notée  $\psi_0$ ) (6.2). Ainsi, considérant cette fonction, le schéma numérique (6.10) devient

$$\begin{cases} \phi(u, t+1) = \phi(u, t) \\ \quad + \Delta t \left[ + (\mathcal{F}(u, t))^+ \left\| (\nabla_w^+ \phi)(u, t) \right\|_p \right. \\ \quad \quad \quad \left. - (\mathcal{F}(u, t))^- \left\| (\nabla_w^- \phi)(u, t) \right\|_p \right] \\ \phi(u, 0) = \psi_0(u) \end{cases} \quad (6.15)$$

Dans la suite de cette section, nous étudions ce schéma et montrons que sous certaines conditions sur le paramètre  $\Delta t$ , le domaine de définition de la fonction  $\phi$  peut être borné par l'intervalle  $[-1, 1]$ . Puis, en définissant également certaines conditions sur la fonction vitesse  $\mathcal{F}$ , nous montrons que le schéma numérique (6.15) converge vers une fonction  $\psi_\infty$  définissant la position exacte du front à l'instant  $t = \infty$ .

#### 6.3.1 Cas d'une vitesse constante dans le temps

Intéressons nous tout d'abord au cas où la fonction vitesse est constante par rapport au temps. Nous supposons que le front est susceptible d'atteindre (depuis l'instant initial) tous les sommets pour lesquels cette vitesse est positive et, de même, de se rétracter de tous les sommets pour lesquels cette vitesse est négative. Dans ce cas, le schéma (6.15) devient

$$\begin{cases} \phi(u, t+1) = \phi(u, t) \\ \quad + \Delta t \left[ + (\mathcal{F}(u))^+ \left\| (\nabla_w^+ \phi)(u, t) \right\|_p \right. \\ \quad \quad \quad \left. - (\mathcal{F}(u))^- \left\| (\nabla_w^- \phi)(u, t) \right\|_p \right] \\ \phi(u, 0) = \psi_0(u) \end{cases} \quad (6.16)$$

Lorsque le temps depuis l'instant initial est très grand ( $t \rightarrow \infty$ ), nous montrons qu'il existe un réel  $\Delta t > 0$  tel que le schéma (6.16) converge et que

### 6.3. Approximation de la solution

---

la fonction  $\phi$  possède une limite strictement égale à la fonction  $\psi_\infty$ .

**Remarque.** Nous rappelons qu'à l'instant  $t = \infty$ , le front a atteint tous les sommets où sa vitesse est positive et s'est entièrement retiré des sommets où sa vitesse est négative. Ainsi la fonction  $\psi_\infty : V \rightarrow \mathbb{R}$  est définie telle

$$\psi_\infty(u) = \begin{cases} -1 & \text{si } \mathcal{F}(u) < 0 \\ +1 & \text{si } \mathcal{F}(u) > 0. \end{cases} \quad (6.17)$$

Afin de démontrer la convergence du schéma (6.16), nous montrons tout d'abord que considérant la définition de la vitesse et la configuration initiale, il existe un réel  $\Delta t$  tel que la fonction  $\phi$  impliquée dans le schéma (6.16) est bornée. Puis, à partir de ce premier résultat, nous montrons que la fonction  $\phi$  est strictement croissante où décroissante par morceaux et possède donc une limite lorsque le temps  $t$  tend vers l'infini.

**Lemme 2.** Soit  $\phi : V \rightarrow [-1, 1]$  une fonction définie sur un graphe  $\mathcal{G} = (V, E, w)$ . Il existe au moins un réel  $\Delta t > 0$  tel que pour tout  $u \in V$  la fonction  $\phi$  vérifie la propriété suivante :

$$-1 \leq \phi(u) + \Delta t \begin{cases} \mathcal{F}(u) \left\| (\nabla_w^+ \phi)(u) \right\|_p & \leq 1, & \text{si } \mathcal{F}(u) > 0 \\ \mathcal{F}(u) \left\| (\nabla_w^- \phi)(u) \right\|_p & \leq 1, & \text{si } \mathcal{F}(u) < 0 \end{cases} \quad (6.18)$$

*Démonstration.* Considérons un sommet  $u$  tel que  $\mathcal{F}(u) > 0$ . Si  $u$  est un maximum local, alors  $\left\| (\nabla_w^+ \phi)(u) \right\|_p = 0$  et  $\phi(u) + 0 \geq 1$  quelque soit  $\Delta t$ . Si  $u$  n'est pas un maximum local, alors  $\phi(u) < 1$  et il existe  $\Delta t_u^+$  tel que

$$0 < \Delta t_u^+ \leq \frac{1 - \phi(u)}{\mathcal{F}(u) \left\| (\nabla_w^+ \phi)(u) \right\|_p}, \quad (6.19)$$

et vérifiant

$$-1 \leq \phi(u) + \Delta t_u^+ \mathcal{F}(u) \left\| (\nabla_w^+ \phi)(u) \right\|_p \leq 1. \quad (6.20)$$

De même, considérons un sommet  $u$  tel que  $\mathcal{F}(u) < 0$ . Si  $u$  est un minimum local, alors  $\left\| (\nabla_w^- \phi)(u) \right\|_p = 0$  et  $\phi(u) + 0 \leq -1$  quelque soit  $\Delta t$ . Si  $u$  n'est pas un minimum local, alors  $\phi(u) > -1$  et il existe  $\Delta t_u^-$  tel que

$$0 < \Delta t_u^- \leq \frac{-1 - \phi(u)}{\mathcal{F}(u) \left\| (\nabla_w^- \phi)(u) \right\|_p}, \quad (6.21)$$

et vérifiant

$$-1 \leq \phi(u) + \Delta t_u^- \mathcal{F}(u) \left\| \left( \nabla_w^+ \phi \right)(u) \right\|_p \leq 1. \quad (6.22)$$

Finalement il existe  $\Delta t = \min_{u \in V} (\Delta t_u^+, \Delta t_u^-)$ , qui vérifie la relation (6.18).

□

**Lemme 3.** Soit  $\phi : V \times \mathbb{R} \rightarrow \mathbb{R}$  une fonction vérifiant l'équation (6.16). Il existe au moins un réel  $\Delta t > 0$  tel que si  $\phi(u, 0) \in [-1, 1] \forall u \in V$ , alors  $\phi(u, t) \in [-1, 1] \forall u \in V, \forall t$ .

*Démonstration.* Puisque  $\phi$  vérifie l'équation (6.15), alors

$$\phi(u, 0) \in [-1, 1]. \quad (6.23)$$

D'après le lemme 2 nous pouvons écrire

$$\phi(u, 1) \in [-1, 1], \quad (6.24)$$

et par récurrence

$$\phi(u, t) \in [-1, 1]. \quad (6.25)$$

□

Ainsi, nous avons montré qu'il existe au moins un réel  $\Delta t > 0$  tel que la fonction  $\phi$  est bornée par l'intervalle  $[-1, 1]$ . Nous montrons maintenant que lorsque  $t$  tend vers l'infini et considérant ce  $\Delta t$ , cette fonction possède une limite localement définie comme la borne supérieure, respectivement inférieure de cet intervalle en fonction du signe de la vitesse  $\mathcal{F}$ , et coïncide avec la fonction indicatrice  $\psi_\infty$

### Preuve de convergence du schéma numérique

Si  $\Delta t$  est défini tel que la fonction  $\phi$  soit bornée sur l'intervalle  $[-1, 1]$ , alors le schéma (6.16) converge vers la fonction caractéristique exacte du front lorsque  $t$  tend vers l'infini.

**Proposition 6.** Lorsque  $t \rightarrow \infty$ , et pour un  $\Delta t$  vérifiant les relations du lemme 3, la fonction  $\phi : V \times \mathbb{R} \rightarrow \mathbb{R}$  vérifiant l'équation (6.16) possède une limite égale à la fonction  $\psi_\infty : V \rightarrow \mathbb{R}$ , tel que

$$\phi(u, \infty) = \psi_\infty(u) = \begin{cases} -1 & \text{si } \mathcal{F}(u) < 0 \\ +1 & \text{si } \mathcal{F}(u) > 0 \end{cases} \quad (6.26)$$

*Démonstration.* Intéressons nous tout d'abord au cas d'un sommet  $u$  tel que la vitesse  $\mathcal{F}(u)$  soit positive. Nous avons alors deux possibilités :

- Si  $\phi(u, 0) = 1$ , alors  $u$  est un maximum local et  $\phi(u, t)$  est constante et égale à 1 pour tout  $t$ .
- Si  $\phi(u, 0) = -1$ , nous pouvons montrer par construction que  $\phi(u, \infty) = 1$ . Considérons tout d'abord un sommet  $u \in \partial^+ \Omega_0$  appartenant à la bordure extérieure de l'ensemble  $\Omega_0$ . Alors il existe nécessairement un sommet  $v \in N(u) \cap \Omega_0$  tel que  $\phi(v, t) = 1 \forall t$ . Par définition la norme du gradient externe de  $\phi(u, t)$  est donc strictement positive tant que  $\phi(u, t) < 1$ . Ainsi, d'après le système (6.16) et le lemme 3, la fonction  $\phi$  en  $u$  est strictement croissante et converge vers 1 lorsque  $t \rightarrow \infty$ .

En suivant la même démarche, nous pouvons ensuite montrer que si  $u$  appartient à la bordure extérieure d'un ensemble  $\Omega_\tau$ ,  $\tau \in [0, \infty[$  alors  $\phi(u, t)$  tend vers 1 lorsque  $t \rightarrow \infty$ . Puis, par extension, il est trivial de montrer que la fonction  $\phi$  tend vers 1 lorsque  $t$  tend vers l'infini,  $\forall u$  tel que  $\mathcal{F}(u) > 0$ .

Une démonstration similaire peut-être utilisée pour montrer que la fonction  $\phi$  possède une limite  $\phi(u, \infty) = -1$  dans le cas où la vitesse  $\mathcal{F}(u)$  est négative. □

Ainsi, nous avons montré qu'en approximant la fonction  $\phi$  à l'instant initial par la fonction caractéristique  $\psi_0$ , le schéma numérique itératif associé à notre formulation converge vers la solution exacte  $\psi_\infty$ , lorsque  $t$  tend vers l'infini.

**Remarque.** *La position initiale du front étant une approximation, l'instant d'arrivée du front calculé en un sommet  $u$  n'est pas l'instant d'arrivée exact mais correspond à un temps artificiel induit par l'approximation de la position.*

### 6.3.2 Extension pour des vitesses dépendant du temps

Dans le cas où la fonction vitesse  $\mathcal{F} : V \times \mathbb{R} \rightarrow \mathbb{R}$  n'est plus constante et varie au cours du temps artificiel, la propriété de convergence du schéma (6.15) ne peut plus être garantie. En effet, si la fonction vitesse  $\mathcal{F}$  oscille infiniment entre des valeurs positives et négatives, la fonction  $\phi$  est infiniment et alternativement croissante puis décroissante et sa limite ne peut plus être définie (de même la convergence du schéma numérique associé ne peut-être garantie).

Cependant, en se basant sur la même approche que dans le cas d'une vitesse constante par rapport au temps, nous pouvons garantir qu'il existe au moins un réel  $\Delta t > 0$ , tel que la fonction  $\phi$  soit bornée sur l'intervalle  $[-1, 1]$ .

**Lemme 4.** *Soit  $\phi : V \times \mathbb{R} \rightarrow \mathbb{R}$  une fonction vérifiant l'équation (6.15). Il existe au moins un réel  $\Delta t > 0$  tel que si  $\phi(u, 0) \in [-1, 1] \forall u \in V$ , alors  $\phi(u, t) \in [-1, 1] \forall u \in V, \forall t$ .*

*Démonstration.* La preuve est similaire à celle proposée pour le lemme 3.

Puisque  $\phi$  vérifie l'équation (6.15), alors

$$\phi(u, 0) \in [-1, 1]. \quad (6.27)$$

D'après le lemme 2 nous pouvons écrire

$$\phi(u, 1) \in [-1, 1], \quad (6.28)$$

et par récurrence

$$\phi(u, t) \in [-1, 1]. \quad (6.29)$$

□

Ainsi, nous avons montré qu'il existe au moins un réel  $\Delta t > 0$ , tel que la fonction  $\phi$  est bornée par l'intervalle  $[-1, 1]$ .

**Remarque.** *Dans le cas où la vitesse varie dans le temps tout en conservant un signe constant ( $\mathcal{F}$  est localement strictement positive, ou strictement négative), la fonction  $\phi$  est localement strictement croissante, respectivement strictement décroissante, et le schéma converge.*

**Remarque.** *S'il existe un réel  $\tau$ , tel que le signe de la fonction  $\mathcal{F}$  soit constant pour tout  $t > \tau$ , alors le schéma converge également.*

## 6.4 Algorithme

Dans la section précédente, nous avons proposé une méthode permettant de simuler l'évolution d'un front  $\Gamma$  évoluant sur un graphe par la résolution d'un schéma numérique itératif et reposant sur une approximation de la position initiale du front. Nous avons montré qu'avec un choix judicieux du paramètre  $\Delta t$  la fonction  $\phi$  est bornée, ce qui garantit dans certains cas particulier la convergence du schéma numérique qui lui est associé.

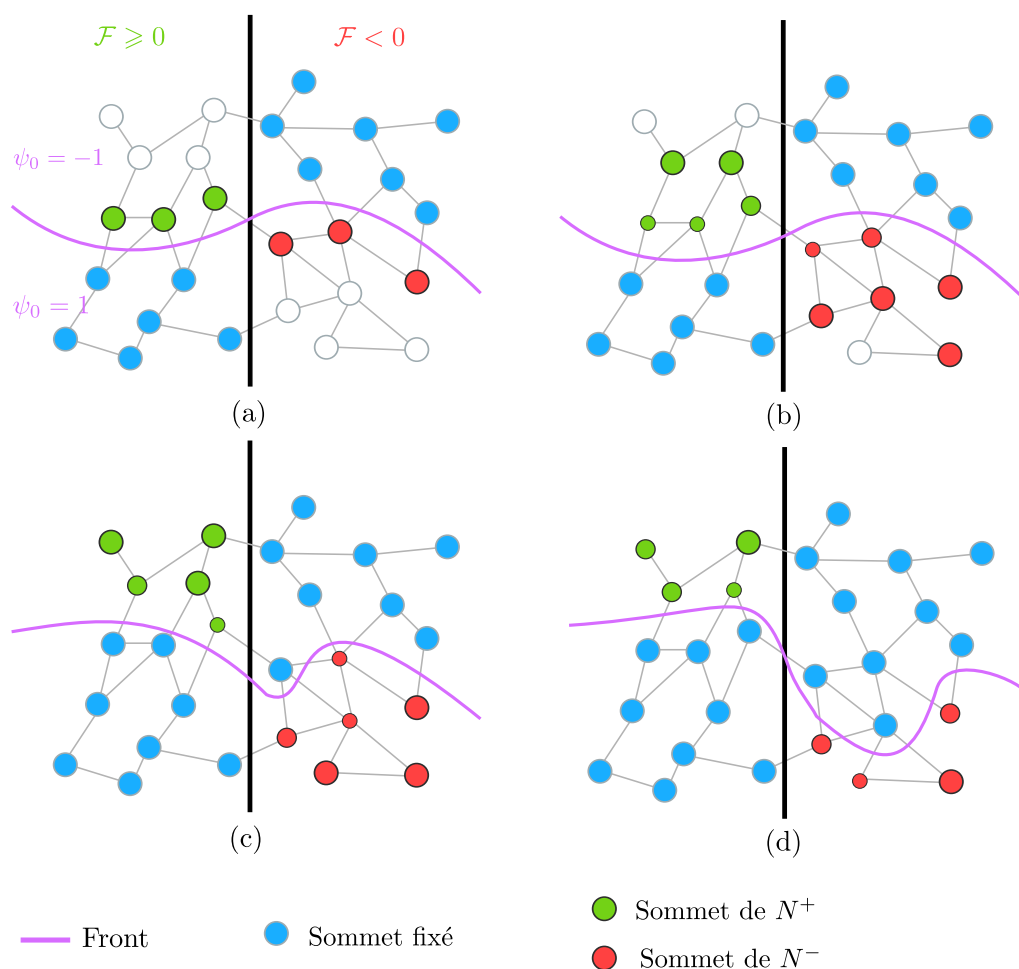


FIGURE 6.2 – Illustration de l’algorithme optimisé. Les sommets en bleu sont les sommets fixés (i.e., les sommets pour lesquels la fonction  $\phi$  a atteint sa limite locale) et ne sont donc pas pris en compte par l’algorithme. Les sommets en vert, respectivement rouge, sont les sommets candidats soumis à une vitesse positive, respectivement négative. Plus le cercle est petit, plus la fonction  $\phi$  est proche de sa limite locale (1, respectivement  $-1$ ). Enfin, les sommets en gris sont les sommets pour lesquels la norme du gradient est nulle, ils ne sont donc pas pris en compte par l’algorithme. À chaque itération, la fonction  $\phi$  est mise à jour pour les sommets candidats, si la fonction atteint sa limite locale (en fonction du signe de la vitesse), les sommets sont retirés des candidats et fixés. Les voisins des sommets candidats sont eux ajoutés à la liste des candidats.

Dans cette section, nous présentons tout d’abord l’algorithme itératif qui découle directement de ce schéma numérique. Puis, en se basant sur les



propriétés des gradients directionnels discrets, nous présentons une version optimisée de cet algorithme où le nombre de sommets à mettre à jour à chaque itération est restreint à un ensemble réduit de sommets situés à proximité du front.

L'algorithme 1 découle directement du schéma numérique (6.15) et ne nécessite pas de structures de données particulières. Le nombre d'itérations et la condition d'arrêt dépendent directement de l'application.

---

**Algorithme 1** : Résolution de l'équation de propagation non-stationnaire

---

**Entrées :**

- Un graphe  $\mathcal{G} = (V, E, w)$  ;
- Une fonction caractéristique  $\psi_0$  ;
- Un pas d'itération  $\Delta t$  ;

**Données :**

- Une fonction  $\phi_t$  qui représente la fonction  $\phi$  à l'instant courant ;
- Une fonction  $\phi_{t+1}$  qui représente la fonction  $\phi$  à l'instant suivant ;

**initialisation**

$$\phi_t \leftarrow \psi_0$$

**répéter**

**pour chaque**  $u \in V$  **faire**

$$\phi_{t+1}(u) \leftarrow \phi_t(u, t) + \Delta t \left[ + (\mathcal{F}(u, t))^+ \left\| (\nabla_w^+ \phi_1)(u, t) \right\|_p \right. \\ \left. - (\mathcal{F}(u, t))^- \left\| (\nabla_w^- \phi_1)(u, t) \right\|_p \right]$$

$$\phi_t \leftarrow \phi_{t+1}$$

**jusqu'à** *Condition d'arrêt (dépend de l'application) ;*

---

### 6.4.1 Optimisation

De par sa construction, cet algorithme est nécessairement coûteux et implique de nombreuses itérations pour atteindre la convergence. Néanmoins, nous pouvons alléger le calcul à chaque itération en réduisant l'ensemble des sommets à mettre à jour. En effet, pour de nombreux sommets (éloignés du front) la fonction  $\phi$  est localement constante et les normes de leurs gradients directionnels sont par conséquent nulles. De plus, nous avons montré qu'avec un  $\Delta t$  judicieusement choisi, la fonction  $\phi$  est bornée et que le schéma numérique converge localement vers 1, respectivement  $-1$ , pour les sommets dont la vitesse est définie strictement positive, respectivement strictement négative, sur un intervalle de temps suffisamment grand.

Ainsi, nous pouvons réduire l'ensemble des sommets pour lesquels il est nécessaire de mettre la fonction  $\phi$  à jour, en ne tenant compte que des sommets appartenant à deux bandes minces, notées  $N^+$  et  $N^-$ , situées au voisinage du front et définies telles que

- $N^+$  est l'ensemble des sommets pour lesquels la vitesse est strictement positive, dont la norme du gradient est non nulle et pour lesquels la fonction  $\phi$  n'a pas atteint sa limite locale (1),
- $N^-$  est l'ensemble des sommets pour lesquels la vitesse est strictement négative, dont la norme gradient est non nulle et pour lesquels la fonction  $\phi$  n'a pas atteint sa limite locale ( $-1$ ).

À l'instant initial, ces deux bandes minces sont définies comme des sous-ensembles de  $V$ , telles que

$$N^+ \leftarrow \{u \in \partial^+ \Omega_0 \mid \mathcal{F}(u) > 0\} \quad (6.30)$$

$$N^- \leftarrow \{u \in \partial^- \Omega_0 \mid \mathcal{F}(u) < 0\}. \quad (6.31)$$

À chaque itération, les bandes sont étendues aux sommets adjacents de ces bandes pour lesquels la fonction  $\phi$  n'a pas encore atteint sa limite locale, puisque la norme de leur gradient n'est plus nulle. De même, les sommets des deux bandes pour lesquels la fonction  $\phi$  a atteint sa limite locale sont retiré des bandes, garantissant une dimension restreinte à ces dernières.

Le processus entier est décrit par l'algorithme 2 et est illustré pour quelques itérations à la figure 6.2.

## 6.5 Expérimentations

Dans cette section, nous présentons l'intérêt de notre adaptation discrète de l'équation des ensembles de niveaux (6.9) et de l'algorithme optimisé associé algorithme 2. Tout d'abord, nous illustrons le comportement de l'algorithme avec un exemple jouet sur un simple graphe grille. Puis nous présentons les résultats de l'application de notre approche à l'évolution d'un front selon la courbure moyenne puis selon le modèle des contours actifs de [Chan et Vese 01].

### 6.5.1 Exemple jouet

L'exemple jouet est réalisé sur un graphe grille ( $V \in \mathbb{R}^2$ ), où l'évolution du front est contrôlée par une vitesse localement constante : soit positive ( $\mathcal{F} = 1$ ) soit négative ( $\mathcal{F} = -1$ ). Dans cet exemple, la fonction de poids est constante

---

**Algorithme 2** : Résolution de l'équation de propagation non-stationnaire  
(version optimisée)

---

**Entrées :**

- Un graphe  $\mathcal{G} = (V, E, w)$  ;
- Une fonction caractéristique  $\psi_0$  ;
- Un pas d'itération  $\Delta t$  ;

**Données :**

- Une fonction  $\phi_t$  qui représente la fonction  $\phi$  à l'instant courant ;
- Une fonction  $\phi_{t+1}$  qui représente la fonction  $\phi$  à l'instant suivant ;
- Un ensemble  $N^+$  des sommets candidats pour lesquels  $\mathcal{F} > 0$
- Un ensemble  $N^-$  des sommets candidats pour lesquels  $\mathcal{F} < 0$

**initialisation**

$$\phi_t \leftarrow \psi_0$$

$$N^+ \leftarrow \{u \in \partial^+ \Omega_0 \mid \mathcal{F}(u) > 0\}$$

$$N^- \leftarrow \{u \in \partial^- \Omega_0 \mid \mathcal{F}(u) < 0\}$$

**répéter**

**pour chaque**  $u \in N^+$  **faire**

$$\left[ \phi_{t+1}(u) \leftarrow \phi_t(u, t) + \Delta t \mathcal{F}(u, t) \left\| (\nabla_w^+ \phi_t)(u, t) \right\|_p \right]$$

**pour chaque**  $u \in N^-$  **faire**

$$\left[ \phi_{t+1}(u) \leftarrow \phi_t(u, t) + \Delta t \mathcal{F}(u, t) \left\| (\nabla_w^- \phi_t)(u, t) \right\|_p \right]$$

**pour chaque**  $u \in N^+$  **faire**

**si**  $\phi_{t+1}(u) = 1$  **alors**

$$\left[ N^+ = N^+ - \{u\} \right]$$

**pour chaque**  $v \in \mathcal{N}(u) - N^+$  **faire**

**si**  $\phi(v) \neq 1$  **et**  $\mathcal{F}(v) > 0$  **alors**

$$\left[ \left[ N^+ = N^+ \cup \{u\} \right] \right]$$

**pour chaque**  $u \in N^-$  **faire**

**si**  $\phi_{t+1}(u) = -1$  **alors**

$$\left[ N^- = N^- - \{u\} \right]$$

**pour chaque**  $v \in \mathcal{N}(u) - N^-$  **faire**

**si**  $\phi(v) \neq -1$  **et**  $\mathcal{F}(v) < 0$  **alors**

$$\left[ \left[ N^- = N^- \cup \{u\} \right] \right]$$

$$\phi_t \leftarrow \phi_{t+1}$$

**jusqu'à** *Condition d'arrêt (dépend de l'application) ;*

---

( $w = 1$ ). La figure figure 6.3 illustre l'évolution de ce front réalisée par notre algorithme optimisé au travers de quelques itérations. Le front  $y$  est représenté en bleu (contour et intérieur), la zone de la grille où la vitesse est positive est représentée en blanc et la zone de la grille où la vitesse est négative est représentée en gris. À chaque itération, la nouvelle position du front est indiquée, ainsi que les deux bandes minces respectivement représentées en vert (pour la bande  $N^+$ ) et rouge (pour la bande  $N^-$ ). Il apparaît clairement que le nombre de points pour lesquels la fonction  $\phi$  doit être mise à jour à chaque itération est grandement réduit.

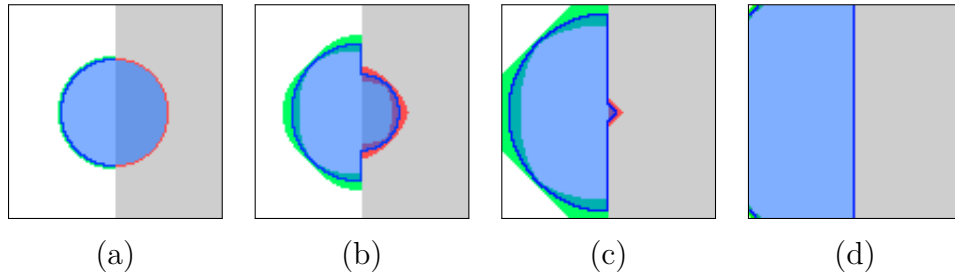


FIGURE 6.3 – Exemple jouet illustrant l'algorithme optimisé. Le front, représenté par la région bleue, évolue sur un graphe grille  $\in \mathbb{R}^2$ . La zone blanche à gauche indique la partie de la grille où la vitesse est positive, et la zone grise à droite indique la partie de la grille où la vitesse est négative. En vert, les sommets appartenant à la bande mince positive et en rouge, les sommets appartenant à la bande mince négative. (a) À l'instant initial. (b) Après 10 itérations. (c) Après 30 itérations. (d) Après 60 itérations.

### 6.5.2 Évolution selon la courbure moyenne

Un cas particulier bien connu de la méthode des ensembles de niveaux est l'évolution selon la courbure moyenne (Mean Curvature Motion) qui consiste à faire évoluer un front de manière à en minimiser la longueur et qui est exprimée par l'équation

$$\frac{\partial \phi}{\partial t} = \operatorname{div} \left( \frac{\nabla \phi}{\|\nabla \phi\|} \right) \|\nabla \phi\| \quad (6.32)$$

où le terme  $\operatorname{div} \left( \frac{\nabla \phi}{\|\nabla \phi\|} \right)$  correspond à la courbure du front.

Avec notre formalisme discret, l'évolution selon la courbure moyenne peut aisément être exprimée par l'équation

$$\frac{\partial \phi(u, t)}{\partial t} = \begin{cases} |\mathcal{K}_w^\phi(u)| \left\| \left( \nabla_w^+ \phi(u, t) \right) \right\|_p, & \text{si } \mathcal{K}_w^\phi(u) < 0 \\ -|\mathcal{K}_w^\phi(u)| \left\| \left( \nabla_w^- \phi(u, t) \right) \right\|_p, & \text{si } \mathcal{K}_w^\phi(u) > 0 \\ 0, & \text{sinon,} \end{cases} \quad (6.33)$$

où la vitesse  $\mathcal{F}$  est égale à l'opposée de la courbure  $\mathcal{K}_w^\phi$  (initialement définie à la section section 2.1.5).

La figure figure 6.4 illustre l'évolution d'un front dont la propagation est contrôlée par cette équation (6.33).

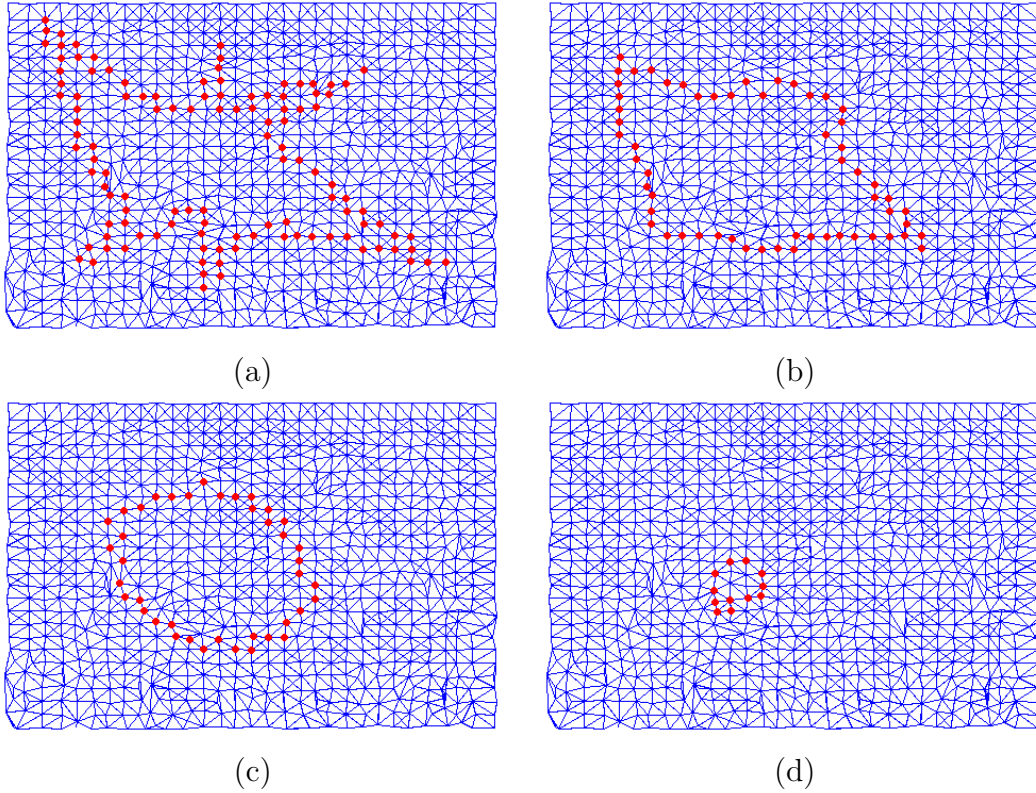


FIGURE 6.4 – Évolution d'un front sur graphe selon la courbure moyenne. La vitesse du front en chaque sommet est définie en fonction de la courbure moyenne en ce sommet. (a) La position initiale du front. (b) Après 20 itérations. (c) Après 60 itérations. (d) Après 100 itérations.

## 6.5. Expérimentations

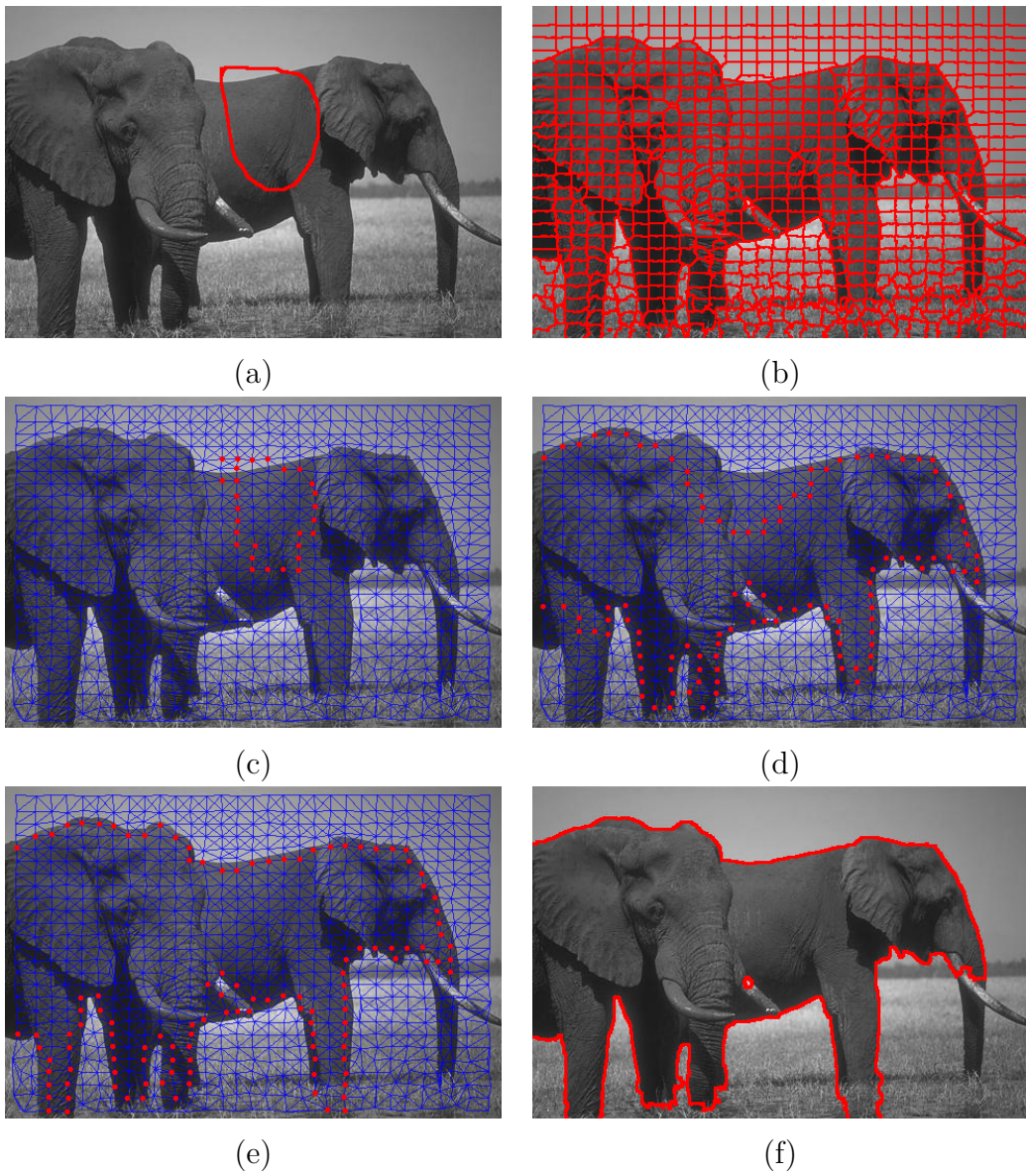


FIGURE 6.5 – Illustration des contours actifs sur graphe par le modèle de Chan et Vese. (a) Le contour initial. (b) La partition de l'image sur laquelle le graphe est construit. (c) La position du front sur le graphe à l'instant initial. (d) La position du front après 60 itérations. (e) La position du front après 300 itérations. (f) Le contour final retransposé sur l'image.



### 6.5.3 Contours actifs

Avec ce formalisme, il est également simple d'adapter certains modèles de contours actifs, comme le modèle de [Chan et Vese 01]. Notre choix d'illustrer les contours actifs sur graphe par ce modèle ne repose pas sur ses performances (de nombreux modèles plus récents [Boykov et al. 06, El Zehiry et al. 07, Grady et Alvinio 09] permettent d'obtenir de meilleurs résultats), mais sur sa simplicité. En effet, notre objectif ici n'est pas de proposer une nouvelle méthode de contour actifs mais de montrer l'adaptabilité de notre approche et des équations associées pour décrire différents modèles de propagation de front sur graphe.

Le modèle de Chan et Vese est basé sur la minimisation d'une énergie dépendant de la longueur du contour et de la variation d'une fonction d'intensité à l'intérieur et à l'extérieur du contour. Exprimé selon la méthode des ensembles de niveaux, il devient un problème d'évolution selon la courbure moyenne et s'écrit

$$\frac{\partial \phi}{\partial t} = \left[ \alpha \operatorname{div} \left( \frac{\nabla \phi}{\|\nabla \phi\|} \right) + \beta \left( (u_0 - c_2) - (u_0 - c_1) \right) \right] \|\nabla \phi\| \quad (6.34)$$

où  $u_0$  est une fonction d'intensité,  $c_1$  la moyenne de  $u_0$  à l'intérieur du contour et  $c_2$  la moyenne de  $u_0$  à l'extérieur du contour.

Transposé sur graphe, ce modèle repose sur l'équation du mouvement (6.9) où la vitesse est définie en fonction de la moyenne à l'intérieur et à l'extérieur du front telle que

$$\mathcal{F}(u) = \alpha \mathcal{K}_w \phi(u) + \beta \left( d_2(F(u) - \mu_{out}(t)) - d_2(F(u) - \mu_{in}(t)) \right), \quad (6.35)$$

où  $\mathcal{K}_w \phi(u)$  est la courbure moyenne (2.25),  $F(u)$  est un vecteur caractéristique associé au sommet  $u$ ,  $\mu_{out}$  est le vecteur caractéristique moyen pour les sommets à l'extérieur du front et  $\mu_{in}$  est le vecteur caractéristique moyen pour les sommets à l'intérieur du front, et  $d_2$  est la distance Euclidienne.

## 6.6 Conclusion

Dans ce chapitre, nous avons proposé une adaptation discrète de l'équation du mouvement définie par l'approche des ensembles de niveaux au domaine des graphes.

Cette adaptation repose sur une interprétation morphologique de la version non-stationnaire de l'équation continue et se base sur le formalisme des EdP.

## 6.6. Conclusion

---

Nous avons montré qu'en fonction du signe de la vitesse du front, son évolution peut être réalisée par une combinaison d'érosion et de dilatations discrètes.

Nous avons ainsi proposé une adaptation de l'équation des ensembles de niveaux non-stationnaire, ainsi qu'un schéma numérique de résolution de cette équation. Nous avons montré que ce schéma converge sous certaines conditions sur la vitesse du front. Nous avons proposé un algorithme itératif efficace implémentant ce schéma numérique et proposé une version optimisée permettant de réduire fortement la complexité de l'algorithme initial.

Finalement, nous avons illustré l'application de cette équation discrète au traitement des images et des données. Les résultats présentés le sont à titre qualitatif et non quantitatif, notre objectif étant de montrer l'adaptabilité et le potentiel de notre approche pour étendre des méthodes initialement définies sur des domaines continus pour le traitement d'images, aux graphes pour le traitement des données.





# Chapitre 7

## Équation stationnaire restreinte au cas d'une vitesse positive

### Sommaire

---

<b>7.1</b>	<b>Équation eikonale discrète</b>	<b>122</b>
<b>7.2</b>	<b>Existence et unicité de la solution</b>	<b>123</b>
<b>7.3</b>	<b>Schéma de résolution local</b>	<b>126</b>
<b>7.4</b>	<b>Schéma de résolution global</b>	<b>130</b>
<b>7.5</b>	<b>Algorithme</b>	<b>135</b>
<b>7.6</b>	<b>Relation avec les schémas numériques classiques</b>	<b>139</b>
<b>7.7</b>	<b>Expérimentations</b>	<b>139</b>
7.7.1	Calcul de distances géodésiques	140
<b>7.8</b>	<b>Conclusion</b>	<b>144</b>

---

À partir des équations d'ensembles de niveaux décrivant l'évolution d'un front sur un domaine continu, nous avons proposé dans le chapitre précédent une adaptation discrète sur graphe de la version non-stationnaire de ces équations.

Dans ce chapitre, nous présentons une version stationnaire et définie sur graphe de l'équation discrète proposée précédemment en se restreignant au cas où la vitesse est définie positive sur l'ensemble du graphe. Cette équation présente une forte analogie avec l'équation eikonale stationnaire continue et nous montrons que sur un graphe particulier (correspondant à une grille régulière) notre formulation coïncide avec certains schémas de discrétisation de cette équation. Nous montrons également que pour tout type de graphe et dans le cas d'une norme  $\mathcal{L}_\infty$ , notre formulation coïncide avec la fonction de distance utilisée dans les algorithmes de type Dijkstra. Nous présentons enfin

une preuve de l'existence et l'unicité des solutions de l'équation discrète, ainsi qu'une formulation explicite de ces solutions pour les normes  $\mathcal{L}_1, \mathcal{L}_2$  et  $\mathcal{L}_\infty$ . Enfin, nous présentons des schémas numériques de résolutions de l'équation pour les normes  $\mathcal{L}_1, \mathcal{L}_2$  et  $\mathcal{L}_\infty$ , ainsi qu'un algorithme rapide de résolution de cette équation, inspiré de l'algorithme du Fast Marching.

L'adaptation de la version stationnaire de l'équation du mouvement d'un front sur graphe, permet de regrouper les méthodes utilisant l'équation eikonale ou l'algorithme de Dijkstra sous une unique formulation générale. De plus, cette adaptation permet d'étendre les applications de l'équation eikonale aux données appartenant à des domaines non-euclidiens comme les graphes et les réseaux ou à des domaines non organisés qui peuvent être représentés sous forme de graphes pondérés.

## 7.1 Équation eikonale discrète

Suivant la même approche que dans le cas continu, l'équation non-stationnaire (6.9) peut-être réécrite sous la forme d'une équation stationnaire. Pour cela nous introduisons une fonction  $\mathcal{T} : V \rightarrow \mathbb{R}$ , dite fonction d'instant d'arrivée, associant à chaque sommet du graphe l'instant d'arrivée du front  $\Gamma$ . Par définition, et par analogie avec la relation continue (5.8), la relation entre la fonction d'instant d'arrivée et la fonction d'ensembles de niveaux  $\phi$  est donnée par la relation

$$\phi(u, t) = \text{sign}(\mathcal{F}(u))(t - \mathcal{T}(u)). \quad (7.1)$$

**Remarque.** L'opérateur  $\text{sign}$  permet de conserver un instant d'arrivée toujours positif quel que soit le signe de la fonction de vitesse.

Dans ce chapitre, nous considérons uniquement le cas où la fonction vitesse  $\mathcal{F}$  est définie positive sur l'ensemble du graphe et constante par rapport au temps. Aussi, considérant cette vitesse, la relation entre les fonctions  $\phi$  et  $\mathcal{T}$  devient

$$\phi(u, t) = t - \mathcal{T}(u). \quad (7.2)$$

Considérant cette relation, la version stationnaire de l'équation du mouvement d'un front peut être exprimée comme une simple réécriture de l'équation (6.9). Ainsi, en remplaçant  $\phi(u, t)$  par  $t - \mathcal{T}(u)$  et en ne tenant compte que du cas où la vitesse est positive, l'équation non-stationnaire peut être réécrite telle que

$$\frac{\partial(t - \mathcal{T}(u))}{\partial t} = \mathcal{F}(u) \left\| \left( \nabla_w^+ (t - \mathcal{T}(u)) \right) \right\|_p. \quad (7.3)$$

La fonction  $\mathcal{T}$  ne dépendant pas du temps, le membre de gauche de cette équation est constant et égal à 1, tel que

$$1 = \mathcal{F}(u) \left\| \left( \nabla_w^+ (t - \mathcal{T}(u)) \right) \right\|_p. \quad (7.4)$$

De même, les gradients pondérés directionnels étant insensibles à l'ajout de constantes et d'après la relation entre les normes des gradients directionnels internes et externes, l'équation devient finalement

$$\mathcal{F}(u) \left\| \left( \nabla_w^- \mathcal{T} \right) (u) \right\|_p = 1. \quad (7.5)$$

Cette équation présente par son écriture une forte analogie avec l'équation eikonale (5.7), et nous verrons par la suite que pour des configurations particulières (i.e., des graphes grilles), cette équation discrète est strictement équivalente à la discrétisation de l'équation eikonale continue.

Afin de simplifier les notations, nous introduisons une fonction de potentiel  $\mathcal{P} : V \rightarrow \mathbb{R}$  et définie telle que  $\mathcal{P}(u) = 1/\mathcal{F}(u)$  et l'équation devient

$$\left\| \left( \nabla_w^- \mathcal{T} \right) (u) \right\|_p - \mathcal{P}(u) = 0. \quad (7.6)$$

Finalement, l'équation du mouvement d'un front  $\Gamma$  à partir de l'instant initial s'écrit sous sa forme stationnaire comme

$$\begin{cases} \left\| \left( \nabla_w^- \mathcal{T} \right) (u) \right\|_p - \mathcal{P}(u) = 0 & u \in \Omega \\ \mathcal{T}(u) = 0 & u \in \Omega_0. \end{cases} \quad (7.7)$$

Une adaptation de l'équation eikonale non-stationnaire sur graphe, et reposant sur les mêmes opérateurs discrets, a également été proposée par [Ta et al. 08b]. Cette approche présente néanmoins l'inconvénient de nécessiter de nombreuses itérations pour converger.

## 7.2 Existence et unicité de la solution

Nous montrons maintenant que cette équation (7.7) possède une solution et que cette solution est unique. Pour cela, nous reprenons la démarche suivie par [Monneau 10] dans sa preuve de l'existence et de l'unicité de la solution du schéma numérique associé à l'algorithme du Fast Marching (algorithme 5.4), et l'adaptions à notre équation discrète.

Pour simplifier l'écriture des démonstrations, nous introduisons une fonction  $\mathcal{S} : V \times \mathcal{H}(V) \rightarrow \mathbb{R}$ , définie telle que

$$\mathcal{S}(u, f) = \left\| (\nabla_w^- f)(u) \right\|_p - P(u). \quad (7.8)$$

Et l'équation que l'on cherche à résoudre peut-être réécrite

$$\begin{cases} \mathcal{S}(u, f) = 0 & u \in \Omega \\ f(u) = 0 & u \in \Omega_0. \end{cases} \quad (7.9)$$

Nous allons tout d'abord introduire quelques propriétés préliminaires de cette fonction, nécessaires aux démonstrations d'existence et d'unicité de la solution.

**Propriété 1.** *La borne inférieure de la fonction  $\mathcal{S}$  est égale à  $-\mathcal{P}(u)$ .*

*Démonstration.* Par définition, la fonction max est définie de  $\mathcal{H}(V) \rightarrow \mathbb{R}^+$ . On a donc

$$\left( \inf_{f \in \mathcal{H}(v)} \nabla_w^- \right) (f(u)) = 0, \quad (7.10)$$

et donc

$$\left( \inf_{f \in \mathcal{H}(v)} \mathcal{S} \right) (u, f) = 0. \quad (7.11)$$

□

**Propriété 2.** *La limite supérieure de la fonction  $\mathcal{S}$  lorsque  $f(u)$  tend vers l'infini est égale à l'infini.*

*Démonstration.* Etant donné que  $\lim_{f(u) \rightarrow +\infty} \max(f(u) - f(v), 0) = +\infty$ , il est trivial de montrer que

$$\lim_{f(u) \rightarrow +\infty} \mathcal{S}(u, f) = +\infty \quad (7.12)$$

□

**Propriété 3.** *La fonction  $\mathcal{S}$  est monotone, et on a*

$$\frac{\partial \mathcal{S}(u, f)}{\partial f(v)} \leq 0, \quad \forall v \neq u. \quad (7.13)$$

*Démonstration.* Selon la définition du gradient directionnel, on a la relation

$$\frac{\partial \left\| (\nabla_w^- f)(u) \right\|_p}{\partial f(v)} = 0, \quad \forall v \neq u, \quad (7.14)$$

or par définition de la vitesse, le potentiel  $\mathcal{P}$  est toujours positif. □

**Propriété 4.** La fonction  $\mathcal{S}$  est invariante à l'ajout d'une constante, telle que

$$\mathcal{S}(u, f) = \mathcal{S}(u, \lambda + f), \quad \lambda \in \mathbb{R}. \quad (7.15)$$

*Démonstration.* Cette propriété découle directement de l'invariance du gradient directionnel.  $\square$

**Propriété 5.** Effet de la multiplication par une constante. On a

$$\mathcal{S}(u, \lambda f) = \lambda \mathcal{S}(u, f) + (\lambda - 1)\mathcal{P}(u), \quad \lambda \in \mathbb{R}. \quad (7.16)$$

*Démonstration.* Il s'agit d'une simple réécriture :

$$\begin{aligned} \mathcal{S}(u, \lambda f) &= \lambda \left\| (\nabla_w^- f)(u) \right\|_p - \mathcal{P}(u) \\ &= \lambda \mathcal{S}(u, f) + (\lambda - 1)\mathcal{P}(u). \end{aligned} \quad (7.17)$$

$\square$

À partir de ces propriétés, nous pouvons montrer l'existence et l'unicité de la solution de l'équation (7.7)

**Proposition 7.** L'équation (7.9) admet une solution.

*Démonstration.* Sachant que la fonction (7.8) est continue, et sachant les propriétés 1 et 2, on sait qu'il existe nécessairement une fonction  $f$  telle que  $\mathcal{S}(u, f) = 0$ .  $\square$

**Proposition 8.** L'équation (7.9) admet une solution unique.

*Démonstration.* D'après la proposition 7, nous savons que l'équation (7.9) possède au moins une solution. Nous allons prouver que cette solution est unique en procédant par l'absurde.

Soient  $f$  et  $g$ , deux solutions distinctes de l'équation (7.9) telles que

$$\max_A (f(u) - g(u)) > 0. \quad (7.18)$$

Ainsi, considérant un réel  $\lambda > 1$  vérifiant

$$\max_A (f(u) - \lambda g(u)) > 0, \quad (7.19)$$

on note  $M_\lambda = \max_A (f(u) - \lambda g(u))$  et  $u_\lambda$  le sommet pour lequel  $M_\lambda = f(u_\lambda) - \lambda g(u_\lambda)$ .

Si l'on introduit une fonction  $h$ , définie telle que  $h(u) = M_\lambda + \lambda g(u)$ , il est possible de déduire que

$$\begin{cases} f(u) \leq h(u) \forall u \in A, \\ f(u_\lambda) = h(u_\lambda). \end{cases} \quad (7.20)$$

Ceci implique les inégalités suivantes

$$\begin{aligned} \mathcal{S}(u_\lambda, f) &\geq \mathcal{S}(u_\lambda, h) && \text{(en raison de la propriété 3)} \\ &\geq \mathcal{S}(u_\lambda, \lambda g) && \text{(en raison de la propriété 4)} \\ &\geq \lambda \mathcal{S}(u_\lambda, g) + (\lambda - 1)P(u) && \text{(en raison de la propriété 5)} \\ &\geq (\lambda - 1)P(u_\lambda). \end{aligned} \quad (7.21)$$

Or, par définition,  $(\lambda - 1)P(u_\lambda)$  est strictement positif, et on a

$$\mathcal{S}(u_\lambda, f) > 0, \quad (7.22)$$

ce qui entre en contradiction avec la définition de l'équation (7.9)  $\square$

Nous avons montré que notre adaptation de l'équation eikonale (7.7) sur graphe possède une solution unique. Dans les Sections suivantes, nous allons proposer des schémas de résolution de cette équation, ainsi qu'un algorithme rapide.

### 7.3 Schéma de résolution local

Pour simplifier la compréhension du raisonnement, nous présentons tout d'abord les schémas de résolution locaux de notre adaptation de l'équation eikonale sur graphe. Par schémas locaux, nous entendons des schémas permettant de connaître la solution en un sommet  $u$ , en supposant que la solution est connue est fixée pour les autres sommets du graphe (où du moins, et comme nous le verrons par la suite, certains sommets).

#### Résolution pour une norme $\mathcal{L}_p$

Nous allons tout d'abord nous intéresser au cas non trivial de la norme  $\mathcal{L}_p$ , avec  $0 < p < \infty$ . Dans ce cas, en développant l'écriture du gradient directionnel, notre équation devient :

$$\sum_{v \in V} \left( \gamma_{uv} (\mathcal{T}(v) - \mathcal{T}(u))^- \right)^p - \mathcal{P}^p(u) = 0, \quad (7.23)$$

puis

$$\sum_{v \in \mathcal{N}(u)} \left( \gamma_{uv} (\mathcal{T}(u) - \mathcal{T}(v))^+ \right)^p - \mathcal{P}^p(u) = 0 \quad (7.24)$$

grâce à la relation  $(-x)^+ = -(x)^-$ .

Afin de simplifier les écritures, nous introduisons les notations suivantes. Étant donné un sommet  $u$ , on note  $\mathcal{S}_u : \mathbb{R}^+ \times W \subseteq V \rightarrow \mathbb{R}$  la fonction définie par

$$\mathcal{S}_u(\mathcal{T}, W) = \sum_{i=1}^n \left( h_i(x - a_i)^+ \right)^p - \mathcal{C}^p \quad (7.25)$$

où  $x = \mathcal{T}(u)$ ,  $n = \text{card}(W)$ ,  $A = \{a_i = \mathcal{T}(v_i) \mid v_i \in W \text{ avec } i = 1, \dots, n\}$ ,  $H = \{h_i = \gamma_{uv_i} \mid v_i \in W \text{ avec } i = 1, \dots, n\}$  et  $\mathcal{C} = \mathcal{P}(u)$ .

Cette fonction  $\mathcal{S}_u$  est une simple réécriture (avec un paramètre supplémentaire  $W$  permettant de contrôler le nombre de sommets impliqués dans le calcul du gradient), de la fonction  $\mathcal{S}$  (7.8). Dans le cas où  $W = V$ , on a

$$\mathcal{S}_u(\mathcal{T}, V) = \mathcal{S}(u, \mathcal{T}), \quad (7.26)$$

et il est trivial de montrer que  $\mathcal{S}_u(\mathcal{T}, W)$  vérifie également les propriétés de  $\mathcal{S}$  présentées dans la section précédente.

Nous allons montrer que cette équation peut être simplifiée et que la dépendance de la solution  $x$  aux instants d'arrivées des autres sommets du graphe peut-être réduite à un minimum de sommets vérifiant certaines conditions, que nous allons détailler ci dessous.

**Proposition 9.** *L'ensemble  $V_u = \{v \in \mathcal{N}(u) \mid \mathcal{T}(v) < \mathcal{T}(u)\}$  des sommets voisins de  $u$  et possédant un temps d'arrivée inférieur vérifie l'égalité suivante  $\mathcal{S}_u(\mathcal{T}, V) = \mathcal{S}_u(\mathcal{T}, V_u)$ .*

*Démonstration.* Les deux propriétés sont triviales à démontrer. Si  $v$  n'appartient pas au voisinage de  $u$ , le poids  $w_{uv}$  est nul et par conséquent la contribution de  $v$  est nulle. De même, si l'instant d'arrivée  $a_i$  associé au sommet  $v$  est nul, la quantité  $(x - a_i)^+$  est égale à zéro et la contribution de  $v$  est nulle.  $\square$

Ainsi, considérant la proposition 9, l'équation (7.24), peut être réécrite

$$\mathcal{S}_u(\mathcal{T}, V_u) = 0, \quad \text{où } V_u = \{v \in \mathcal{N}(u) \mid \mathcal{T}(v) < \mathcal{T}(u)\} \quad (7.27)$$

où seuls les sommets apportant une contribution à la résolution entrent en compte dans le calcul de la solution  $\bar{x}$ . Cette simplification peut facilement



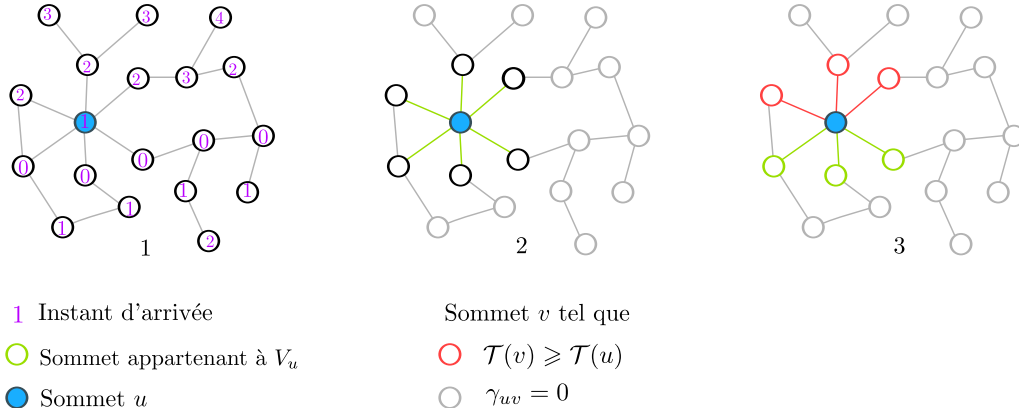


FIGURE 7.1 – Interactions entre les sommets pour la résolution locale. 1- Le sommet  $u$  en bleu et l'instant d'arrivée connu en chaque sommet du graphe. 2- Les sommets non adjacents (en gris) ne possèdent pas d'arêtes communes au sommet  $u$  et n'apportent aucune contribution. Ils sont donc ignorés. 3- Les sommets dont l'instant d'arrivée est supérieur à celui en  $u$  (en rouge) ne sont pas non plus pris en compte puisque leur contribution est nulle.

être interprétée géométriquement. L'instant d'arrivée du front en un sommet  $u$  ne dépend que de son instant d'arrivée aux sommets d'où il est susceptible d'atteindre  $u$ . La sélection des sommets contribuant effectivement au calcul de la solution locale en un sommet  $u$ , en fonction des interactions entre ces sommets, est illustrée à la figure 7.1

**Proposition 10.** *Selon la définition de l'ensemble  $V_u$ , la fonction  $\mathcal{S}_u$  peut être réécrite telle que*

$$\mathcal{S}_u(\mathcal{T}, V_u) = \sum_{i=1}^n \left( h_i(x - a_i) \right)^p - \mathcal{C}^p \quad (7.28)$$

*Démonstration.* Par définition de l'ensemble  $V_u$ , on sait que

$$\begin{aligned} x &> a_i \\ x - a_i &> 0 \\ (x - a_i)^+ &= (x - a_i) \end{aligned} \quad (7.29)$$

pour tout sommet  $v_i$  appartenant à  $V_u$ . □

Avec cette écriture libérée du maximum, notre adaptation de l'équation eikonale sur graphe (7.27) devient

$$\mathcal{S}_u(\mathcal{T}, V_u) = \sum_{i=1}^n \left( h_i(x - a_i) \right)^p - \mathcal{C}^p = 0. \quad (7.30)$$

Cette équation possède une solution unique (d'après la proposition 8), qui peut être calculée de façon triviale. En développant cette équation pour isoler l'inconnue  $x$ , et dans le cas où  $p = 1$ , la solution  $\bar{x}$  est donnée par

$$\bar{x} = \frac{\sum_{i=1}^n h_i a_i + \mathcal{C}}{\sum_{i=1}^n h_i}. \quad (7.31)$$

De même, dans le cas où  $p = 2$ , la solution de l'équation est donnée par

$$\bar{x} = \frac{\sum_{i=1}^n h_i^2 a_i + \sqrt{\sum_{i=1}^n \left( h_i^2 \mathcal{C}^2 - \sum_{j>i} h_i^2 h_j^2 (a_i - a_j)^2 \right)}}{\sum_{i=1}^n h_i^2}. \quad (7.32)$$

Ainsi, considérant un sommet  $u$ , nous avons montré que, dans le cas de la norme  $\mathcal{L}_p$  et en supposant l'instant d'arrivée connu aux sommets atteints avant  $u$ , la solution locale de notre adaptation de l'équation eikonale peut être obtenue en résolvant l'équation  $\mathcal{S}_u(V_u) = 0$  dont nous avons fourni une formulation explicite de la solution pour  $p = 1$  et  $p = 2$ .

### Résolution pour une norme $\mathcal{L}_\infty$

Intéressons nous maintenant au cas de la norme  $\mathcal{L}_\infty$ . La démarche est similaire au cas de la norme  $\mathcal{L}_p$  et nous ne fournirons donc pas les étapes intermédiaires. En développant la norme du gradient directionnel, l'équation (7.7) devient

$$\max_{v \in V} \left( \gamma_{uv} \left( \mathcal{T}(v) - \mathcal{T}(u) \right)^- \right) - \mathcal{P}(u) = 0, \quad (7.33)$$

puis, en utilisant les mêmes notations que pour le cas  $\mathcal{L}_p$ ,

$$\mathcal{S}_u(\mathcal{T}, V_u) = \max_{i=1}^n \left( h_i(x - a_i)^+ \right) - \mathcal{C}. \quad (7.34)$$

où  $V_u$  est le même sous-ensemble des voisins de  $u$  que défini dans le paragraphe précédent. En effet, il est trivial de montrer que la proposition 9 reste vrai avec cette nouvelle écriture de la fonction  $\mathcal{S}_u$ .

**Proposition 11.** *Selon la définition de l'ensemble  $V_u$ , la fonction  $\mathcal{S}_u$  peut être réécrite telle que*

$$\mathcal{S}_u(\mathcal{T}, V_u) = \max_{i=1}^n (h_i(x - a_i)) - \mathcal{C} \quad (7.35)$$

*Démonstration.* La démonstration est identique à celle proposée pour la proposition 10.  $\square$

Finalement, avec cette écriture, l'équation eikonale sur graphe pour une norme  $\mathcal{L}_\infty$  devient

$$\mathcal{S}_u(\mathcal{T}, V_u) = \max_{i=1}^n (h_i(x - a_i)) - \mathcal{C} = 0. \quad (7.36)$$

Enfin, toujours en suivant la même approche que pour la cas de la norme  $\mathcal{L}_p$ , la solution explicite de l'équation (7.7), pour une norme  $\mathcal{L}_\infty$  est donnée par

$$\bar{x} = \min_{i=1}^n \left( a_i + \frac{\mathcal{C}}{h_i} \right). \quad (7.37)$$

Finalement, nous avons montré que, similairement au cas d'une norme  $\mathcal{L}_p$ , la solution de l'équation eikonale sur graphe pour un sommet  $u$  et avec un à priori sur l'instant d'arrivée de ses voisins coïncide avec la solution de l'équation  $\mathcal{S}_u(V_u) = 0$  dont nous avons fourni une formulation explicite de la solution.

## 7.4 Schéma de résolution global

Dans les sections précédentes, nous avons montré que la solution de notre équation en un sommet  $u$  ne dépend que de la solution aux voisins de  $u$  dont les instants d'arrivée sont inférieurs à celui en  $u$ . Et ceci quelle que soit la norme. Nous allons maintenant montrer que la solution globale (i.e., en tout sommet du graphe) peut aisément être calculée selon un schéma de résolution récursif et reposant sur la propriété de causalité (propriété 9) des solutions locales.

Considérons l'ensemble initial  $\Omega_0$ , ainsi que sa bordure extérieure  $\partial^+\Omega_0$ . Soit  $\mathcal{T}^0 : V \rightarrow \mathbb{R}$  une fonction vérifiant l'équation

$$\begin{cases} \mathcal{S}_u(\mathcal{T}_0, V_u^0) = 0, & \forall u \in \partial^+\Omega_0 \\ \mathcal{T}_0(u) = \mathcal{T}(u) = 0, & \forall u \in \Omega_0 \end{cases} \quad (7.38)$$

où  $V_u^0 = \{v \in \mathcal{N}(u) \cap \Omega_0\}$ . La fonction  $\mathcal{T}_0$  peut-être interprétée comme une fonction d'instant d'arrivée supposée, associant à chaque sommet  $u \in \partial^+\Omega_0$ , l'instant d'arrivée du front en ne considérant que la position initiale du front (et non sa position au cours du temps).

Nous pouvons montrer que pour certains sommets, cette fonction coïncide avec la fonction d'instant d'arrivée  $\mathcal{T}$  du front.

**Proposition 12.** *Si  $\mathcal{T}_0(u) = \min_{v \in \partial^+ \Omega_0} (\mathcal{T}_0(v))$  alors  $\mathcal{T}(u) = \mathcal{T}_0(u)$ .*

*Démonstration.* Si  $\mathcal{T}_0(u) = \min(\mathcal{T}_0(v))$ , alors il n'existe pas de sommet  $v \in \mathcal{N}(u)/\Omega_0$ , tel que  $v$  soit atteint par  $\Gamma$  avant  $u$ . Et donc, selon la propriété de causalité, on a  $V_u = V_u^0$ , ce qui implique que  $\mathcal{T}(u) = \mathcal{T}_0(u)$ .  $\square$

Ainsi, d'après la proposition 12, l'instant d'arrivée du premier sommet atteint par le front peut-être calculé comme étant le minimum des valeurs d'une fonction  $\mathcal{T}_0$  définie à l'instant initial. Si l'on note  $t_1$  cette valeur minimale, on peut écrire la relation suivante

$$\Omega_{t_1} = \Omega_0 \cup \{u \in \partial^+ \Omega_0 \mid \mathcal{T}_0(u) = \min_{v \in \partial^+ \Omega_0} (\mathcal{T}_0(v))\}, \quad (7.39)$$

puisque par définition l'ensemble  $\Omega_{t_1}$  est l'ensemble des sommets déjà atteints par le front à l'instant  $t_1$ . Les différentes notions et étapes nécessaires pour le calcul de  $\mathcal{T}_0$  afin d'établir le sous-ensemble  $\Omega_{t_1}$  sont illustrés par la figure 7.2.

Considérons maintenant la fonction  $\mathcal{T}_{t_1} : V \rightarrow \mathbb{R}$ , définie telle que

$$\begin{cases} \mathcal{S}_u(\mathcal{T}_{t_1}, V_u^{t_1}) = 0, & \forall u \in \partial^+ \Omega_{t_1} \\ \mathcal{T}_{t_1}(u) = \mathcal{T}(u), & \forall u \in \Omega_{t_1} \end{cases} \quad (7.40)$$

où  $V_u^{t_1} = \{v \in \mathcal{N}(u) \cap \Omega_{t_1}\}$ , et supposant que  $\mathcal{T}(u)$  est connue pour tout  $u \in \Omega_{t_1}$ .

**Proposition 13.** *Si  $\mathcal{T}_{t_1}(u) = \min_{v \in \partial^+ \Omega_{t_1}} (\mathcal{T}_{t_1}(v))$  alors  $\mathcal{T}(u) = \mathcal{T}_{t_1}(u)$ .*

*Démonstration.* La démonstration est la même que pour la proposition 12, en remplaçant l'instant 0 par l'instant  $t_1$ .  $\square$

Et ainsi, si l'on note  $t_2$ , le minimum de la fonction  $\mathcal{T}_{t_1}$  sur la bordure extérieure, nous avons la relation

$$\Omega_{t_2} = \Omega_{t_1} \cup \{u \in \partial^+ \Omega_{t_1} \mid \mathcal{T}_{t_1}(u) = \min_{v \in \partial^+ \Omega_{t_1}} (\mathcal{T}_{t_1}(v))\}, \quad (7.41)$$

et tel que l'instant d'arrivée du front  $\Gamma$  en tout sommet de  $\Omega_{t_2}$  est donné par la fonction  $\mathcal{T}_{t_2}$ . De même que pour l'instant 0, le processus est illustré par la

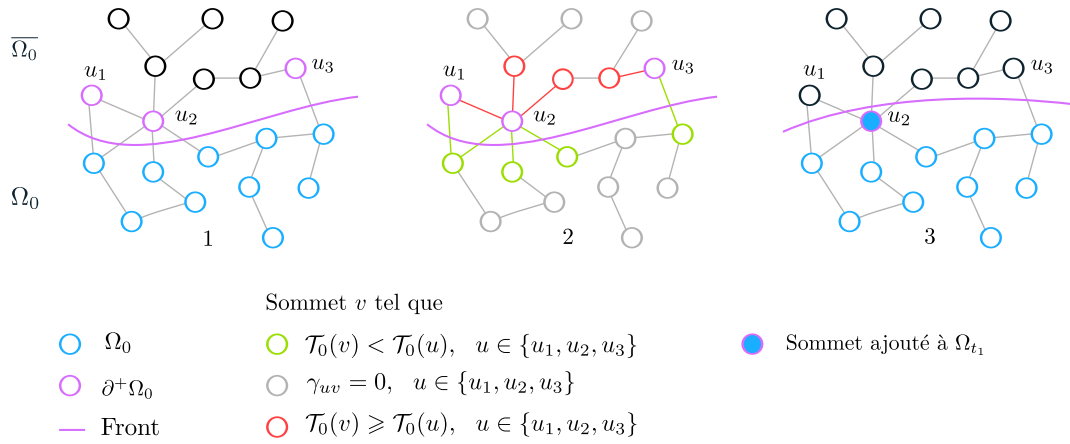


FIGURE 7.2 – 1- Un front représenté à l’instant initial par le sous-ensemble  $\Omega_0$ , et les sommets appartenant à sa bordure extérieure ( $u_1, u_2, u_3$ ). 2- La valeur de la fonction  $\mathcal{T}_0$  en ces sommets ne dépend que des sommets adjacents pour lesquels la fonction  $\mathcal{T}_0$  est inférieure. En gris, les sommets non-adjacents. En rouge, les sommets adjacents mais dont on sait que la valeur de la fonction  $\mathcal{T}_0$  sera supérieure. 3- Finalement le sommet  $u_2$  pour lequel la valeur de la fonction  $\mathcal{T}_0$  est minimale est atteint par le front à l’instant  $\mathcal{T}_0(u_2)$  et ajouté au sous-ensemble  $\Omega_{t_1}$ , avec  $t_1 = \mathcal{T}_0(u_2)$ .

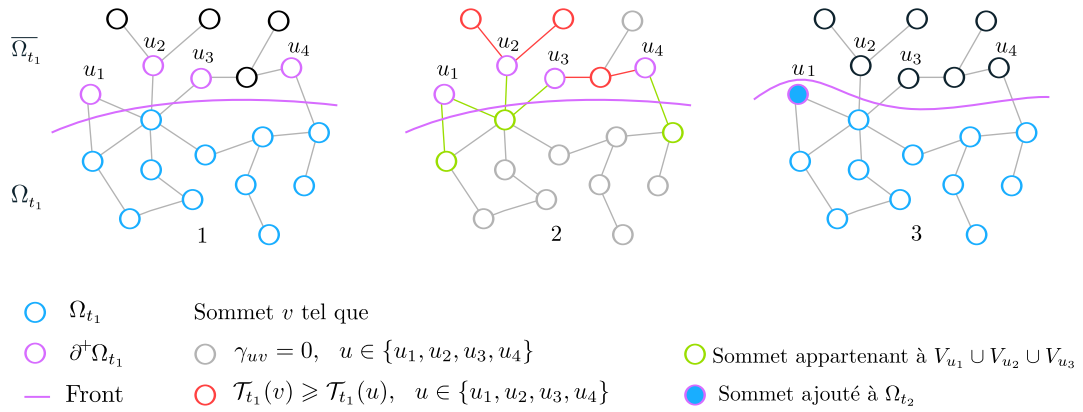


FIGURE 7.3 – La légende et le principe sont similaires à ce qui est présenté à la figure 7.2. À travers ces deux figures, et au delà de l’illustration de la résolution des différentes fonctions de la suite  $(\mathcal{T}_n)$ , nous voyons apparaître le processus itératif de résolution tel qu’il sera repris dans l’algorithme 3.

figure 7.3

Nous pouvons dégager de ces deux propositions (proposition 12 et proposition 13) les deux premières étapes d'un schéma récursif de résolution de l'équation eikonale sur graphe. Ce schéma repose sur une suite de fonctions à valeurs réelles, notée  $(\mathcal{T}_n)_{n \in \mathbb{N}}$ , où la fonction  $\mathcal{T}_n : V \rightarrow \mathbb{R}$  est définie par  $\mathcal{T}_n = \mathcal{T}_{t_n}$  et telle que la relation entre  $\mathcal{T}_n$  et  $\mathcal{T}_{n-1}$  soit donnée par

$$\begin{cases} \mathcal{T}_0(u) = 0, & \forall u \in \Omega_n \\ \mathcal{T}_n(u) = \mathcal{T}_{n-1}(u), & \forall u \in \Omega_n \\ \mathcal{S}_u(\mathcal{T}_n, V_u^n) = 0, & \forall u \in \partial^+ \Omega_n \\ \Omega_n = \Omega_{n-1} \cup \{u \in \partial^+ \Omega_{n-1} \mid \mathcal{T}_{n-1}(u) = t_n = \min_{v \in \partial^+ \Omega_{n-1}} (\mathcal{T}_{n-1}(v))\}, \end{cases} \quad (7.42)$$

où  $V_u^n = \{u \in N(u) \cap \Omega_n\}$ . Par souci de clarté dans les écritures, nous notons  $\Omega_n$  le sous ensemble des sommets à l'intérieur du front à l'instant  $t_n$ .

Du système (7.42), il apparaît que les valeurs de la fonction  $\mathcal{T}_{n-1}$  pour certains sommets du graphe (les sommets déjà atteints par le front à l'instant  $t_n$ ) sont conservées dans la fonction  $\mathcal{T}_n$ . Cette conservation peut-être étendue à un plus grande nombre de sommets.

**Proposition 14.** *Soit  $u \in \partial^+ \Omega_{n-1} \cap \partial^+ \Omega_n$ . Si  $u$  n'est voisin d'aucun sommet atteint par le front entre les instants  $t_{n-1}$  et  $t_n$ , alors  $\mathcal{T}_n(u) = \mathcal{T}_{n-1}(u)$ .*

*Démonstration.* Étant donné les conditions sur le voisinage de  $u$ , on a  $V_u^n = V_u^{n-1} \subseteq \Omega_n$ . Or, puisque  $\mathcal{T}_n(v) = \mathcal{T}_{n-1}(v)$ ,  $\forall v \in \Omega_n$ , on a

$$\mathcal{S}_u(\mathcal{T}_n, V_u^n) = \mathcal{S}_u(\mathcal{T}_{n-1}, V_u^{n-1}) \quad (7.43)$$

et finalement

$$\mathcal{T}_n(u) = \mathcal{T}_{n-1}(u). \quad (7.44)$$

□

Ainsi, la variation entre  $\mathcal{T}_n$  et  $\mathcal{T}_{n-1}$  ne concerne qu'un petit nombre de sommets : Les sommets dont un voisin au moins a été atteint par le front entre les instants  $t_{n-1}$  et  $t_n$ . La relation entre  $\mathcal{T}_n$  et  $\mathcal{T}_{n-1}$  devient donc

$$\begin{cases} \mathcal{T}_0(u) = 0, & \forall u \in \Omega_n \\ \mathcal{T}_n(u) = \mathcal{T}_{n-1}(u), & \forall u \in \Omega_n \cup (\partial^+ \Omega_n \cap \overline{U_n}) \\ \mathcal{S}_u(\mathcal{T}_n, V_u^n) = 0, & \forall u \in U_n \\ \Omega_n = \Omega_{n-1} \cup \{u \in \partial^+ \Omega_{n-1} \mid \mathcal{T}_{n-1}(u) = t_n = \min_{v \in \partial^+ \Omega_{n-1}} (\mathcal{T}_{n-1}(v))\}, \end{cases} \quad (7.45)$$

où  $U_n = \{u \in \partial^+ \Omega_n \mid \exists v \in \Omega_{n-1} \cap \Omega_n \text{ et } u \in N(v)\}$  est l'ensemble des sommets pour lesquels il peut exister une variation entre  $\mathcal{T}_{n-1}$  et  $\mathcal{T}_n$ .

Enfin, nous montrons que cette suite  $(\mathcal{T}_n)$  est monotone et qu'elle converge vers la solution  $\mathcal{T}$  de notre équation stationnaire discrète. Afin de garantir la propriété de monotonie (que nous démontrons ci-dessous), la fonction  $\mathcal{T}_n$  est définie comme nulle pour tous les sommets au delà de la bordure extérieure de  $\Omega_n$  :

$$\mathcal{T}_n(u) = 0, \forall u \in V - (\Omega_n \cup \partial^+ \Omega_n). \quad (7.46)$$

**Propriété 6.** *La suite  $(\mathcal{T}_n)_{n \in \mathbb{N}}$  est monotone*

*Démonstration.* Nous avons plusieurs cas de figures. Les deux premiers sont triviaux et concernent les sommets pour lesquels  $\mathcal{T}_n$  est définie comme égale à  $\mathcal{T}_{n-1}$ , ainsi que les sommets candidats qui étaient hors de la bordure extérieure du front à l'instant  $t_{n-1}$ , et pour lesquels la fonction  $\mathcal{T}_{n-1}$  est définie comme nulle. Si un sommet  $u$  appartient à l'un de ces deux cas, nous avons la relation  $\mathcal{T}_n(u) \geq \mathcal{T}_{n-1}(u)$ .

Le dernier cas de figure concerne les sommets candidats pour lesquels la valeur de  $\mathcal{T}_{n-1}$  est non nulle. Par définition, nous savons que

- $\mathcal{T}_n(u)$  est solution de  $\mathcal{S}_u(\mathcal{T}_n, V_u^n)$
- $\mathcal{T}_{n-1}(u)$  est solution de  $\mathcal{S}_u(\mathcal{T}_{n-1}, V_u^{n-1})$

Toujours par définition, on sait qu'il existe au moins un sommet  $v \in V_u^n$ , tel que  $v \notin V_u^{n-1}$  et tel que

$$\mathcal{T}_n(v) \geq \mathcal{T}_{n-1}(w), \forall w \in V_u^{n-1}. \quad (7.47)$$

Ce sommet est en fait le dernier sommet atteint par le front (à l'instant  $t_n$ ). Par construction, il a un instant d'arrivée supérieur ou égal à celui de tous les sommets précédemment atteints.

Ainsi, d'après l'inégalité (7.47) et la propriété de monotonie du schéma  $\mathcal{S}_u$ , on a  $\mathcal{T}_n(u) \geq \mathcal{T}_{n-1}(u)$ , ce qui complète la preuve.  $\square$

**Propriété 7.** *La suite  $(\mathcal{T}_n)_{n \in \mathbb{N}}$  converge et sa limite est  $\mathcal{T}$ .*

*Démonstration.* Par construction, on sait qu'il existe un entier  $k > 0$  tel que  $\Omega_k = V$  et tel que pour tout entier  $n > k$  on ait  $\Omega_n = \Omega_k$ . Or, par définition, on sait que

$$\mathcal{T}_n(u) = \mathcal{T}(u), \forall u \in \Omega_n \quad (7.48)$$

et donc, considérant l'entier  $k$ , on a la relation

$$\begin{aligned}\mathcal{T}_k(u) &= \mathcal{T}(u), \quad \forall u \in \Omega_k = V \\ \mathcal{T}_n(u) &= \mathcal{T}(u), \quad \forall u \in \Omega_n, \quad \forall n > k.\end{aligned}\tag{7.49}$$

□

Nous avons montré que la solution de l'équation eikonale sur graphe peut-être obtenue à la convergence d'une suite de fonctions  $(\mathcal{T}_n)$ , et ainsi calculée récursivement. Dans la section suivante, nous proposons un algorithme rapide de résolution de l'équation eikonale sur graphe basé sur ce schéma récursif.

## 7.5 Algorithme

Dans la section précédente, nous avons présenté une méthode de résolution de notre équation stationnaire basée sur une suite récurrente convergeant vers la solution exacte. Cette méthode de résolution présente intrinsèquement tous les ingrédients d'un algorithme de résolution efficace puisqu'elle décrit l'ordre de parcours des noeuds et permet de minimiser le nombre de calculs en chaque sommet. Intuitivement, l'algorithme consiste à calculer une solution locale sur une bande mince initialisée à partir des sommets de  $\partial^+\Omega_0$  et en fonction des instants d'arrivée aux sommets déjà à l'intérieur du front (soit, à l'instant initial,  $\Omega_0$ ). Ensuite, à chaque itération, le sommet de la bande mince pour lequel l'instant d'arrivée est minimum est ajouté à l'ensemble des sommets intérieurs. Ses voisins qui étaient jusqu'alors ignorés sont ajoutés à la bande mince et une solution locale est calculée pour chacun des voisins (qu'ils soient nouvellement ajoutés ou déjà présents dans la bande mince). En effet, d'après les propriétés de la suite récursive, la solution locale reste constante entre deux itérations, excepté pour les voisins du sommet qui vient d'être atteint. Enfin, l'algorithme se termine lorsque tous les sommets ont été ajoutés à l'intérieur du front (en fonction de la topologie du graphe).

Cet algorithme est très similaire sur le principe à l'algorithme du Fast Marching qui résout la même équation (dans le cas où la norme est  $\mathcal{L}_2$ ) sur une grille discrète. En utilisant les mêmes conventions de notations que dans l'algorithme de Sethian, notre algorithme consiste en une liste de sommets actifs (à l'intérieur du front), une bande mince et un ensemble de sommets lointains. Ainsi, à tout instant  $t_n$ , la liste des sommets actifs, la bande mince et l'ensemble des sommets lointains coïncident respectivement avec  $\Omega_n$ ,  $\partial^+\Omega_n$  et  $V - (\Omega_n \cup \partial^+\Omega_n)$ . L'algorithme intégral est présenté à l'algorithme 3 et illustré sur quelques itérations par la figure 7.4.



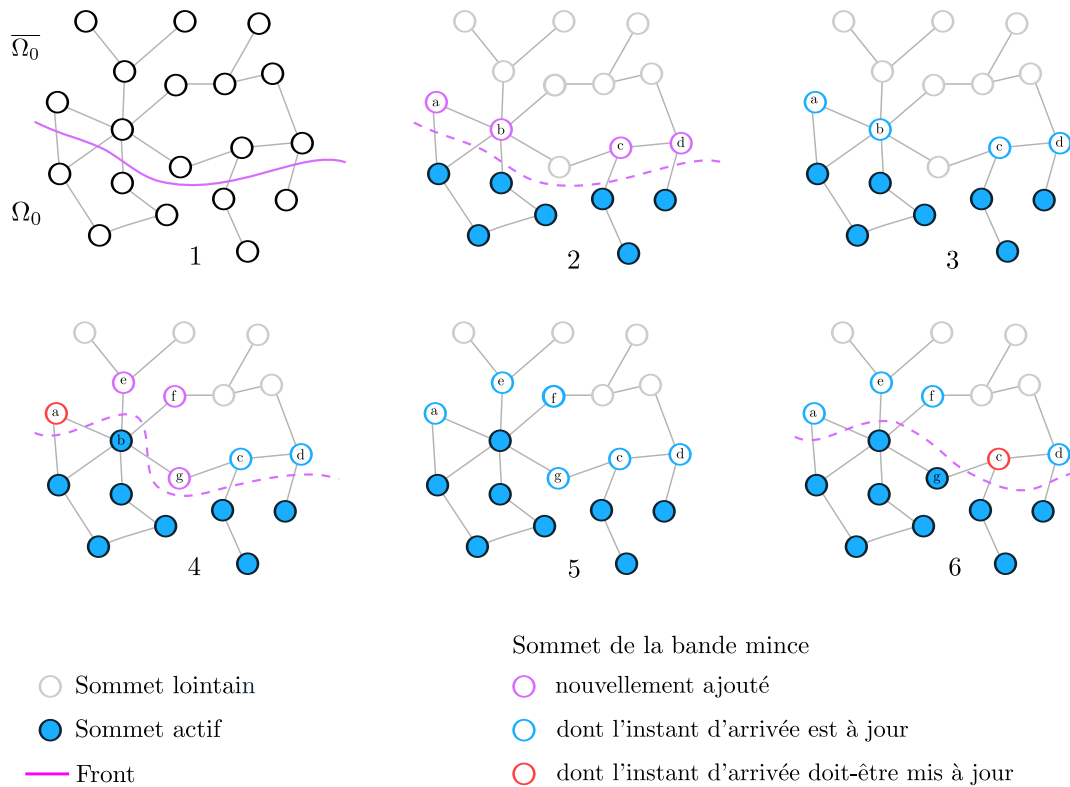


FIGURE 7.4 – Illustration de l’algorithme sur quelques itérations. Au travers des six figures présentées, nous illustrons l’initialisation ainsi que deux itérations de notre algorithme. 1- La figure représente la situation initiale d’un front représenté par deux sous-ensembles  $\Omega_0$  et  $\overline{\Omega}_0$  et évoluant sur un graphe. 2- Initialisation de l’algorithme. Les sommets appartenant à l’ensemble  $\Omega_0$  sont ajoutés à la liste active et les sommets adjacents mais hors d’ $\Omega_0$  (a,b,c,d) sont ajoutés à la bande mince. Les sommets restants sont dit lointains. 3- L’instant d’arrivée potentielle aux sommets de la bande mince est mis à jour et la bande mince est réordonnée. 4- Le sommet b, ayant l’instant d’arrivée potentielle du front minimum, est retiré de la bande mince et ajouté à la liste active. Les sommets adjacents à b (e,f,g) sont ajoutés à la bande mince. L’instant d’arrivée au voisinage du sommet a étant modifiée, l’instant d’arrivée potentielle au sommet a n’est plus à jour et doit être recalculé. Ce n’est pas le cas des sommets c et d pour lesquels aucun calcul ne sera nécessaire à l’itération suivante. 5 - L’instant d’arrivée potentielle aux sommets de la bande mince est mis à jour et la bande mince est réordonnée. Le sommet g, ayant l’instant d’arrivée potentielle minimum, est ensuite retiré de la bande mince et ajouté à la liste active. Aucun sommet n’est en situation d’être ajouté à la bande mince. Seul l’instant d’arrivée potentielle au sommet c doit être mis à jour lors de l’itération suivante.

---

**Algorithme 3** : Résolution de l'équation de propagation stationnaire.

---

**Entrées :**

- Un graphe  $\mathcal{G} = (V, E, w)$  ;
- Un sous ensemble  $\Omega_0 \subset V$  ;
- Une fonction potentiel  $\mathcal{P}$  ;

**Sorties :**

- Une fonction d'instant d'arrivée  $\mathcal{T}$  ;

**Données :**

- L'ensemble  $A$  des sommets actifs ;
- L'ensemble  $NB$  des sommet de la bande mince, ordonné selon  $\mathcal{T}$  ;
- L'ensemble  $FA$  des sommets lointains ;

**initialisation**

$A \leftarrow \Omega_0$

$NB \leftarrow \partial^+ \Omega_0$

$FA \leftarrow V - (\Omega_0 \cup \partial^+ \Omega_0)$

$\mathcal{T}(u) \leftarrow 0, \forall u \in A$

$V_u = A \cap N(u), \mathcal{T}(u) \leftarrow \text{solution locale}(u, \mathcal{T}, V_u), \forall u \in NB$

**répéter**

$u \leftarrow$  premier élément de  $NB$ .

$NB \leftarrow NB - \{u\}; A \leftarrow A \cup \{u\}$

**pour chaque**  $v \in N(u) \cap \bar{A}$  **faire**

$V_v = A \cap N(v)$

$\mathcal{T}(v) \leftarrow \text{solution locale}(v, \mathcal{T}, V_v)$

**si**  $v \in FA$  **alors**

$FA \leftarrow FA - \{v\}; NB \leftarrow NB \cup \{v\}$

**sinon**

$\lfloor$  mettre à jour la position de  $v$  dans  $NB$

**jusqu'à**  $NB = \emptyset$  ;

---

### Propriétés de l'algorithme

L'algorithme implémentant exactement l'évolution de la suite  $(\mathcal{T}_n)$  définie à l'équation (7.45), il en reprend les propriétés de monotonie (propriété 6) et de convergence (propriété 7).

### Performances et complexité de l'algorithme

Le coût  $C_p^i$  pour calculer l'instant d'arrivée potentielle en un sommet  $u$  candidat et en fonction de  $p$  est donné par

$$\begin{aligned} C_1^i(u) &= D_i(u) && \text{dans le cas où } p = 1 \\ C_2^i(u) &= D_i^2(u) && \text{dans le cas où } p = 2 \\ C_\infty^i(u) &= D_i(u) && \text{dans le cas où } p = \infty \end{aligned} \quad (7.50)$$

où la quantité  $D_i(u)$  correspond au nombre de voisins de  $u$  pour lesquels l'instant d'arrivée est déjà connu. Afin de généraliser ce coût, nous considérons la quantité  $D_i(u)$  comme constante et égale au degré maximum d'un sommet de  $V$ . Nous introduisons ainsi la constante  $D$ , qui majore  $D_i$ , définie telle que

$$D = \max_{u \in V} (\text{card}(\mathcal{N}(u))). \quad (7.51)$$

Ainsi, nous obtenons une version majorée et constante du coût de calcul du pseudo temps d'arrivée en un sommet candidat.

$$\begin{aligned} C_1^i &= D \geq C_1^i(u) \quad \forall u \in V \\ C_2^i &= D^2 \geq C_2^i(u) \quad \forall u \in V \\ C_\infty^i &= D \geq C_\infty^i(u) \quad \forall u \in V. \end{aligned} \quad (7.52)$$

Le coût d'activation d'un sommet  $u$ , c'est à dire le coût pour faire passer un sommet de la bande mince à la liste des sommets actifs, dépend lui aussi du degré de  $u$ . En effet, à chaque fois qu'un voisin de  $u$  est activé, il est nécessaire de calculer un nouveau pseudo instant d'arrivée en  $u$ . Le coût d'activation majoré, noté  $C_p^a$  est donc défini par

$$C_p^a = DC_p^i, \quad p \in \{1, 2, \infty\} \quad (7.53)$$

Finalement, la complexité de notre algorithme est donnée par

$$\mathcal{O}(C_p^a N \log(N)), \quad (7.54)$$

où  $N$  désigne le nombre total de sommets du graphe et où le facteur  $\log(N)$  correspond au coût de maintien du tas ordonné des pseudo instants d'arrivée de la bande mince.

En pratique, les graphes sur lesquels est appliqué cet algorithme ne sont que très faiblement connectés et le degré maximum est très inférieur au nombre total de sommets ( $C_p^a \ll N$ ).

## 7.6 Relation avec les schémas numériques classiques

### Discrétisation de l'équation eikonale

De la même manière que dans la section 6.2, et en reprenant les mêmes notations, nous montrons dans ce paragraphe que sur une structure de graphe particulière, notre formulation discrète de l'équation du mouvement coïncide avec le schéma numérique de premier ordre proposé par [Sethian 99].

Soit  $\mathcal{G} = (V, E, w)$  un graphe pondéré représentant une grille régulière dans  $\mathbb{R}^m$ . En reprenant les notations déjà utilisées, et considérant une norme  $\mathcal{L}_2$  l'équation stationnaire s'écrit

$$\sqrt{\sum_{j=1}^m \max\left(\frac{\mathcal{T}(u \pm h_j e_j) - \mathcal{T}(u)}{h_j}, 0\right)^2} - \mathcal{P}(u) = 0 \quad (7.55)$$

Puis, en utilisant les conventions de notations des différences finies avant et arrière, ainsi que la relation  $\min(0, x) = -\max(0, -x)$ , nous obtenons le schéma de discrétisation de premier ordre proposé par [Osher et Sethian 88] pour la version stationnaire de l'équation du mouvement (5.23).

$$\sqrt{\sum_{j=1}^m \max\left(D_u^- j, 0\right)^2 + \min\left(D_u^+ j, 0\right)^2} - \mathcal{P}(u) = 0 \quad (7.56)$$

### Algorithme de type Dijkstra

Dans le cas où  $p = \infty$ , la solution explicite de notre équation stationnaire coïncide avec la fonction de distance utilisée dans les algorithmes de type Dijkstra. De plus, lorsqu'il est muni de cette solution locale, notre algorithme se comporte comme un algorithme de type Dijkstra.

Ainsi, notre formulation stationnaire de l'équation du mouvement sur graphe peut-être considérée comme une généralisation des équations du mouvement des domaines Euclidiens aux graphes. L'algorithme associé peut quand à lui être considéré comme une généralisation et une unification des algorithmes de Fast marching et de type Dijkstra.

## 7.7 Expérimentations

Dans cette section, nous illustrons le comportement de notre adaptation discrète de la version stationnaire de l'équation des ensembles de niveaux

appliquée sur différents types de graphes et présentons nous intéresser pour le calcul de géodésiques.

### 7.7.1 Calcul de distances géodésiques

La fonction d'instant d'arrivée  $\mathcal{T}$  d'un front en tout point du domaine, solution de l'équation (7.7), est équivalente à une carte de distance depuis l'origine du front. Ainsi, il est aisé de calculer des cartes de distances géodésiques sur graphe (suivant les arêtes) et plus courts chemins sur graphe avec notre algorithme rapide (algorithme 3).

#### Sur une image

La figure 7.5 illustre l'application de notre formulation de l'équation d'un front sur graphe, pour le calcul de géodésiques sur une image (l'image du tangramme). Les résultats présentés sont obtenus à partir d'un graphe grille en 4-connectivité, selon différentes fonctions de poids et pour des valeurs de  $p \in [1, 2, \infty]$ .

Dans chaque cas, la carte de distance depuis un point initial et avec une fonction de poids constante  $\mathcal{P} = 1$ , est présentée sous la forme d'un dégradé de couleurs allant du rouge (faible distance) au violet (grande distance).

La première ligne présente des résultats pour une fonction de poids constante  $w_1 = 1$  et un point initial situé dans le coin en haut à gauche de l'image. Avec une telle fonction de poids et dans le cas où  $p = 2$ , nous retrouvons le résultat obtenu avec l'algorithme du Fast Marching [Sethian 99]. De même, dans le cas où  $p = \infty$ , notre algorithme devient un algorithme de Dijkstra, et nous retrouvons le résultat obtenu avec ce type d'algorithmes.

La seconde ligne présente des résultats à partir d'un point initial central, pour une fonction de poids  $w_2$  favorisant la direction horizontale, et définie telle que

$$w_2 = \begin{cases} 1 & \text{si } v_x = u_x \\ 0.25 & \text{si } v_y = u_y \end{cases} \quad (7.57)$$

Dans ce cas, la propagation est plus rapide dans la direction horizontale et les iso-contours se trouvent étirés dans cette direction.

La troisième ligne présente les résultats obtenus à partir d'un point initial situé dans le coin en haut à gauche de l'image, pour une fonction de poids  $w_3$  dépendant de l'image du tangramme. Cette fonction est construite à partir de la mesure de similarité  $g_4$  (1.16) basée sur l'intensité des pixels de l'image.

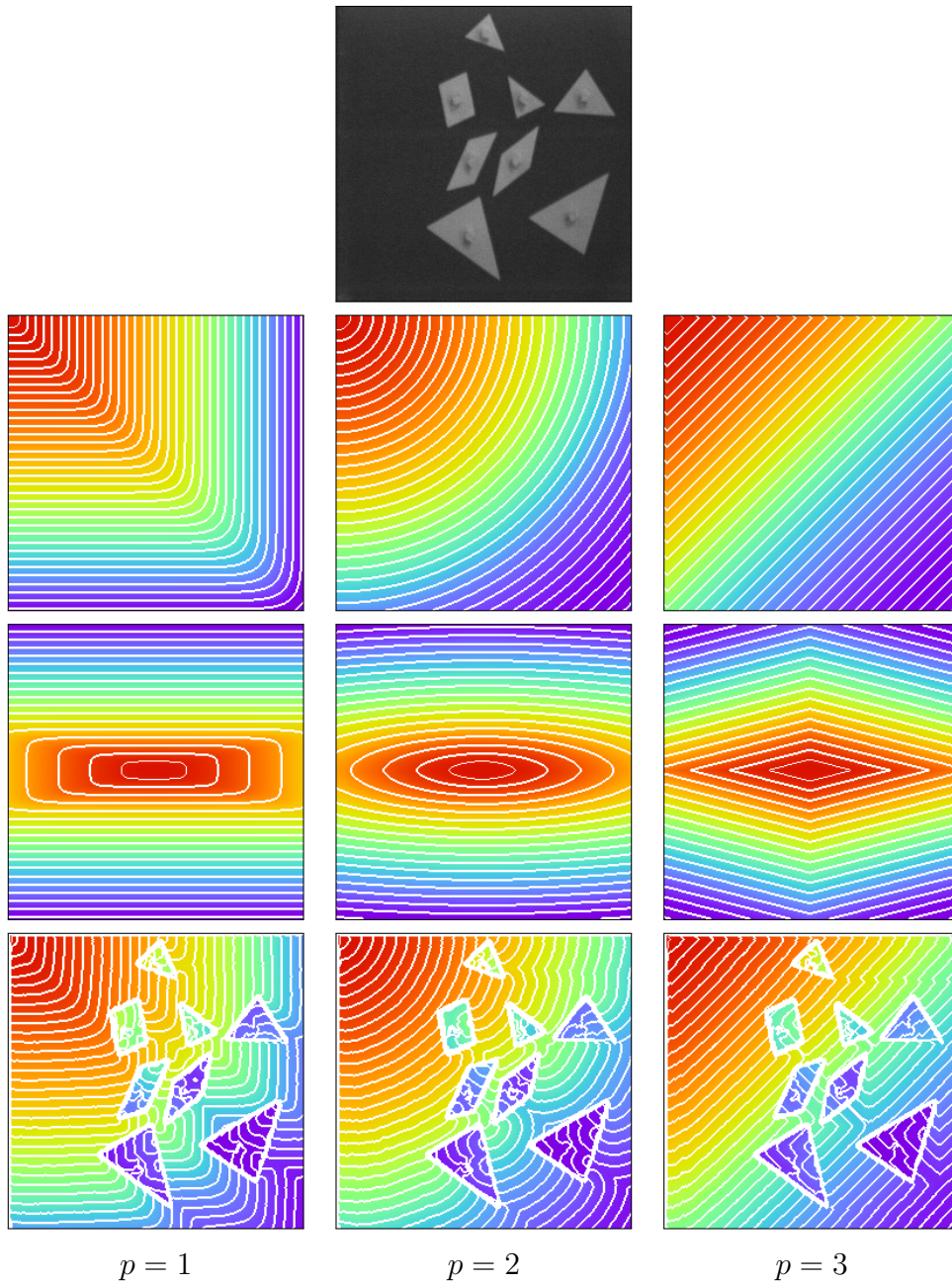


FIGURE 7.5 – Calcul de géodésiques sur une image. Les résultats sont présentés avec un graphe de 4-voisinage, pour différentes fonctions de poids et des valeurs de  $p \in 1, 2, \infty$ . Voir le texte pour plus de détails

Selon cette construction, la fonction de poids tend vers 0 entre deux pixels dissimilaires (un contour interne de l'image), et tend vers 1 entre deux pixels

similaires (une zone homogène de l'image). Ainsi, la propagation du front est rapide dans les zones homogènes de l'image et lente sur ses contours internes.

### Sur un RAG

La figure 7.6 illustre l'application de notre formulation pour le calcul de géodésique sur une image représentée par un RAG (voir la définition d'un RAG section [1.2.3]). La première image présente l'image initiale et les contours des partitions en superposition (en blanc). Ces contours coïncident avec les contours internes de l'image originale. La seconde image présente le graphe d'adjacence des régions construit à partir de cette partition (en rouge). Et finalement, la dernière image présente la carte de distance obtenue à partir de la région en haut à gauche. Dans ce cas, la fonction de poids  $w_4$  est construite à partir de la mesure de similarité  $g_4$  (1.16) basée sur l'intensité moyenne des pixels de chaque région. Ici encore, la propagation du front est rapide sur les zones homogènes de l'image et lente au voisinage des discontinuités.

Ici, un simple seuillage de la carte de distance permet d'extraire l'ensemble des pièces du jeu de tangramme.

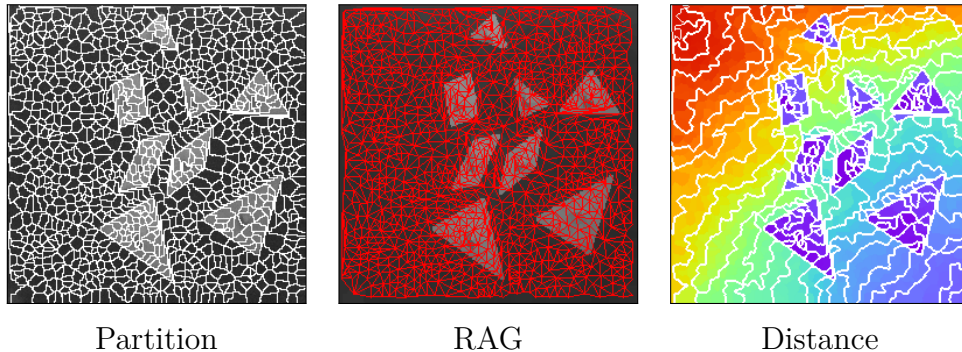


FIGURE 7.6 – Calcul de géodésique sur un RAG.

### Sur un maillage 3D texturé

La figure 7.7 présente plusieurs cartes de distance calculées sur un maillage 3D texturé. La première carte est obtenue en utilisant une fonction de poids constante ( $w = 1$ ) ce qui entraîne une diffusion homogène du front et qui ne dépend que de la topologie du maillage. La seconde carte de distance est quand à elle obtenue en utilisant une fonction de poids dépendant de la texture définie en chaque sommet. Tout comme dans le cas d'une image, le front évolue



rapidement sur les zones de couleur homogène et se trouve ralenti au niveau des discontinuités de la texture. Dans cet exemple, et pour les deux fonctions de poids, nous ne tenons pas compte des coordonnées spatiales des sommets.

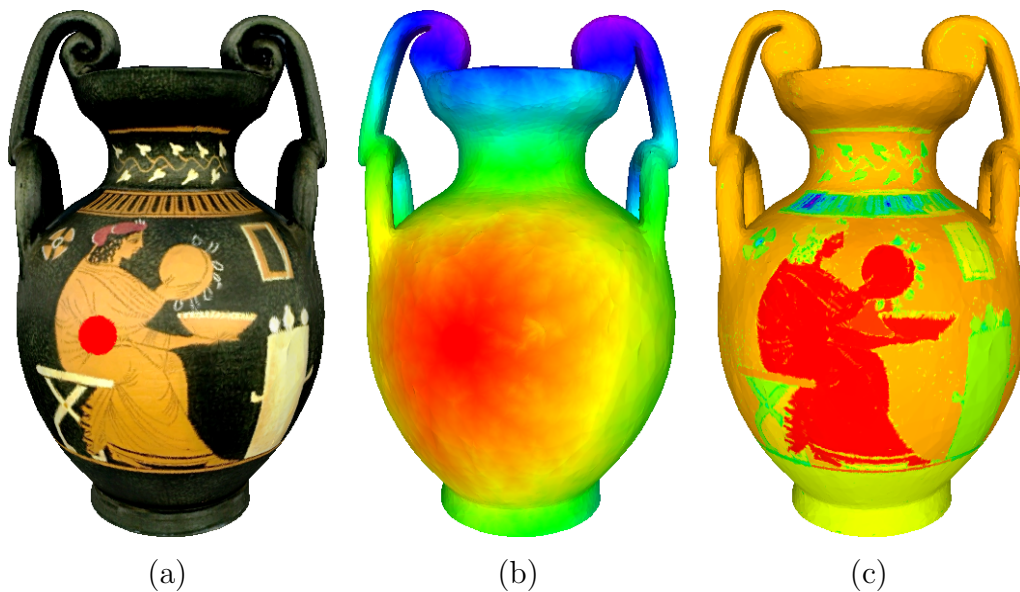


FIGURE 7.7 – Calcul de géodésique sur un maillage texturé. (a) Le maillage texturé sur lequel est superposé le point initial (en rouge). (b) La carte de distance obtenue pour une fonction de poids constante ( $w = 1$ ). (c) La carte de distance obtenue pour une fonction de poids dépendant de la texture en chaque sommet ( $w = g_4$ ).

### Sur un graphe de données

La figure 7.8 présente une carte de distance calculée sur un graphe de données. Le graphe présenté est un graphe des  $k$ -pppv (voir la définition d'un  $k$ -ppv section [1.2.2]) construit à partir de la base d'images USPS. Pour le calcul de la fonction de poids, chaque image est représentée comme un unique vecteur de longueur  $28^2$  contenant de haut en bas et de gauche à droite l'intensité de tous les pixels de l'image. Ainsi, la similarité entre deux images est calculée en comparant leurs deux vecteurs respectifs. La carte de distance est calculée en tout point du graphe depuis une unique image située en haut à droite du graphe.



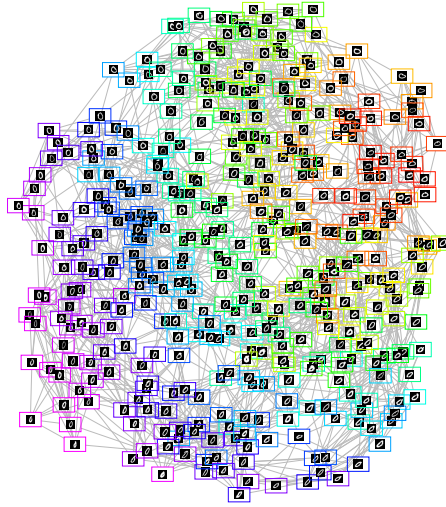


FIGURE 7.8 – Calcul de géodésique sur un graphe de données.

### Sur un graphe de topologie irrégulière

Enfin, la (7.9) illustre le comportement de notre algorithme sur un graphe de topologie irrégulière. Ce graphe est constitué d'un ensemble de points appartenant à l'espace  $\mathbb{R}^2$ . La fonction de poids  $w$  est construite à partir de la mesure de similarité  $g_4$  (1.16) basée sur le vecteur position de chaque point. La distance est calculée à partir du point rouge sur la figure de gauche. La figure de droite présente la carte de distance.

## 7.8 Conclusion

Dans ce chapitre, nous avons présenté une version stationnaire de l'équation (6.9) introduite au chapitre précédent en nous restreignant au cas où la vitesse est définie positive sur l'ensemble du graphe.

Nous avons montré que pour un graphe euclidien et une norme  $\mathcal{L}_2$  cette équation coïncide avec la discrétisation d'Osher-Sethian de l'équation eikonale définie en continu. Nous avons également montré que pour une norme  $\mathcal{L}_\infty$  notre formulation coïncide avec la fonction de distance utilisée dans les algorithmes de type Dijkstra.

Nous avons présenté une preuve de l'existence et de l'unicité de la solution de cette équation pour les normes  $\mathcal{L}_1, \mathcal{L}_2$  et  $\mathcal{L}_\infty$ , ainsi qu'une formulation explicite de ces solutions.

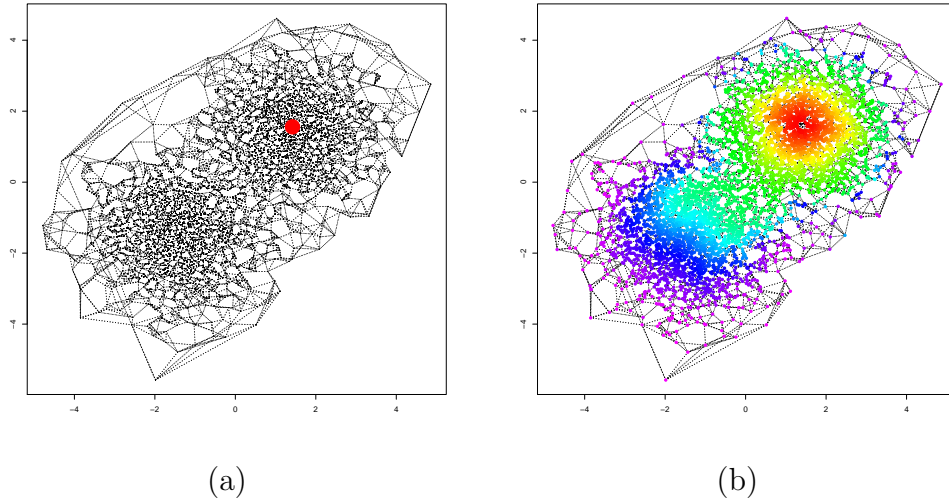


FIGURE 7.9 – Calcul de géodésique sur un graphe de topologie irrégulière. (a) Le graphe et le point de départ de la diffusion (en rouge). (b) La carte de distance calculée avec notre algorithme.

Nous avons également proposé un schéma de résolution de cette équation ainsi qu'un algorithme rapide implémentant ce schéma. Cet algorithme peut être considéré comme une généralisation de l'algorithme du Fast Marching aux graphes et se comporte comme un algorithme de type Dijkstra lorsque  $p = \infty$ .

Finalement, nous avons illustré l'application de cette équation discrète pour le calcul de géodésiques sur différents types de graphes.

Dans le chapitre suivant, nous proposons d'étendre les équations et le cadre présenté ici au cas où plusieurs fronts évoluent simultanément sur un même graphe. Cette extension ouvre la voie à des applications plus abouties pour le traitement des images et des données.



# Chapitre 8

## Généralisation à la propagation simultanée de plusieurs fronts

### Sommaire

---

<b>8.1 Définitions et notations</b> . . . . .	<b>148</b>
<b>8.2 Équation de propagation du front global</b> . . . . .	<b>149</b>
<b>8.3 Schémas de résolution et algorithme</b> . . . . .	<b>151</b>
8.3.1 Schéma de résolution local . . . . .	151
8.3.2 Schéma de résolution global . . . . .	153
8.3.3 Algorithme . . . . .	154
<b>8.4 Expérimentations</b> . . . . .	<b>154</b>
8.4.1 Exemple jouet . . . . .	154
8.4.2 Partitionnement et simplification de graphes . . . . .	156
8.4.3 Segmentation semi-supervisée d'images . . . . .	159
8.4.4 Classification semi-supervisée de données . . . . .	165
<b>8.5 Conclusion</b> . . . . .	<b>166</b>

---

Dans le chapitre précédent, nous avons présenté une nouvelle formulation de l'équation stationnaire du mouvement sur graphes pondérés. Cette équation, ainsi que l'algorithme de résolution associé, permettent de décrire l'évolution au cours du temps d'un front défini sur un graphe, en se limitant au cas d'un front unique se propageant dans la direction de la normale.

Dans ce chapitre, nous proposons d'étendre la formulation précédemment introduite et de nous intéresser au cas où plusieurs fronts évoluent simultanément sur un même graphe. De même que pour le chapitre précédent, nous restreignons le problème aux fronts évoluant selon une vitesse constante et définie positive sur l'ensemble du graphe. De plus, nous considérons que ces

fronts ne peuvent se chevaucher (i.e., à un instant  $t$ , un sommet n'appartient qu'à un seul front) et qu'ils n'ont aucune influence mutuelle, excepté qu'ils sont mutuellement bloquants (sans pour autant se repousser). Nous montrons dans ce chapitre que l'ensemble de ces fronts peut être représenté par un unique front les rassemblant et que l'évolution simultanée de ces fronts peut-être calculée en résolvant une version modifiée de l'équation du mouvement pour le front unique.

## 8.1 Définitions et notations

Soient  $\Gamma_1, \dots, \Gamma_N$ ,  $N$  fronts évoluant simultanément sur un graphe  $\mathcal{G}$ . Ces fronts sont respectivement définis à l'instant initial par les sous-ensembles  $\Omega_0^1, \dots, \Omega_0^N$ , avec  $\Omega_0^i \cap \Omega_0^j = \emptyset$ ,  $\forall i \neq j$  (nous rappelons que les fronts ne peuvent se chevaucher).

Considérant un front  $\Gamma_i$ . On note  $\mathcal{F}_i : V \rightarrow \mathbb{R}^+$  la fonction de vitesse contrôlant l'évolution du front  $\Gamma_i$  en tout sommet du graphe.

On note  $\Omega^i$  le sous-ensemble des sommets de  $V$  qui sont atteints par le front  $\Gamma_i$ . Par définition, on a

$$\begin{aligned} \Omega^i \cap \Omega^j &= \emptyset, \quad \forall i \neq j \\ \bigcup_{i \in [1, N]} \Omega^i &= V \end{aligned} \tag{8.1}$$

On note  $\mathcal{T}^i : \Omega^i \rightarrow \mathbb{R}^+$  la fonction associant à chaque sommet atteint par  $\Gamma_i$  l'instant d'arrivée du front, telle que  $\mathcal{T}^i$  vérifie l'équation

$$\begin{cases} \mathcal{S}_u(\mathcal{T}^i, V_u, \mathcal{F}_i) = 0, \quad \forall u \in \Omega^i \\ \mathcal{T}^i(u) = 0, \quad \forall u \in \Omega_0^i \end{cases} \tag{8.2}$$

où  $V_u = \{v \in \Omega_i \cap \mathcal{N}(u) \mid \mathcal{T}(v) < \mathcal{T}(u)\}$  et où  $\mathcal{S}_u$  désigne indifféremment l'équation du mouvement d'un front quelque soit la norme.

**Remarque.** *Puisque  $\Omega_i$  est l'ensemble des sommets atteints par le front  $\Gamma_i$ , seul le front  $\Gamma_i$  est susceptible d'atteindre ces sommets. Ainsi, la propagation concurrente du front  $\Gamma_i$  sur l'ensemble  $V$  peut-être vue comme une propagation sans contrainte sur l'ensemble  $\Omega_i$ . C'est pourquoi la fonction  $\mathcal{T}^i$  est solution de l'équation (8.2).*

Soit  $L : V \rightarrow [1, N]$  une fonction de marqueur, associant à chaque sommet du graphe l'indice de l'unique front l'atteignant, et définie telle que

$$L(u) = i, \forall u \in \Omega^i \text{ et } \forall i \in [1, N] \quad (8.3)$$

### Front global

Nous cherchons à représenter l'ensemble de ces fronts par un front global les regroupant, et défini tel que son évolution corresponde à l'évolution des fronts regroupés. Soit  $\Gamma$  un front représenté à l'instant initial par l'ensemble  $\Omega_0 = \bigcup_{i \in [1, N]} \Omega_0^i$ . On note  $\mathcal{T} : V \rightarrow \mathbb{R}^+$  la fonction associant à chaque sommet du graphe, l'instant d'arrivée de l'unique front l'atteignant, de telle sorte que

$$\mathcal{T}(u) = \mathcal{T}^i(u) \forall u \in \Omega^i \text{ et } \forall i \in [1, N] \quad (8.4)$$

On note  $\mathcal{F} : V \rightarrow \mathbb{R}^+$  la fonction associant à chaque sommet du graphe la vitesse de l'unique front l'atteignant, de telle sorte que

$$\mathcal{F}(u) = \mathcal{F}_i(u) \forall u \in \Omega^i \text{ et } \forall i \in [1, N] \quad (8.5)$$

## 8.2 Équation de propagation du front global

Dans cette section, nous montrons que la fonction d'instant d'arrivée du front global est solution d'une version modifiée de l'équation eikonale sur graphe. En effet, l'équation stationnaire (7.7) présentée dans le chapitre précédent n'est pas suffisante pour décrire l'évolution du front global. En effet, elle ne prend pas en compte les contraintes de l'évolution simultanée des différents fronts, tout particulièrement la contrainte d'absence d'interactions entre les fronts, et l'interdiction des chevauchement. Cette version adaptée pour tenir compte de ces contraintes est définie par l'équation

$$\begin{cases} \mathcal{S}_u(\mathcal{T}, V_u, \mathcal{F}) = 0, \forall u \in V \\ \mathcal{T}(u) = 0, \forall u \in \Omega_0, \end{cases} \quad (8.6)$$

où le sous ensemble  $V_u$  est restreint aux sommets qui sont atteints par le même front que  $u$ .  $V_u$  est donc défini tel que

$$V_u = \{v \in \Omega_i \cap N(u) \mid \mathcal{T}(v) \leq \mathcal{T}(u)\} \quad (8.7)$$

Nous montrons que la solution  $\mathcal{T}$  de l'équation (8.6) coïncide avec les fonctions d'instant d'arrivée de chaque front.

**Proposition 15.** *Si  $\mathcal{T}(u) = \mathcal{T}^i(u)$  et  $\mathcal{F}(u) = \mathcal{F}_i(u)$ ,  $\forall u \in \Omega^i, i \in [1, N]$ , alors  $\mathcal{T}$  vérifie l'équation (8.6). Et réciproquement.*

*Démonstration.* En remplaçant  $\mathcal{T}^i$  par  $\mathcal{T}$  dans (8.2), l'équation de propagation du front  $\Gamma^i$  devient

$$\begin{cases} \mathcal{S}_u(\mathcal{T}, V_u, \mathcal{F}), \forall u \in \Omega^i, i \in [1, N] \\ \mathcal{T}(u) = 0, \forall u \in \Omega_0^i \end{cases} \quad (8.8)$$

où  $V_u = \{v \in \Omega_i \cap N(u) \mid \mathcal{T}(v) \leq \mathcal{T}(u)\}$ . Puis, étant donné les relations

$$\begin{aligned} \bigcup_{i \in [1, N]} \Omega_0^i &= \Omega_0 \\ \bigcup_{i \in [1, N]} \Omega^i &= V \end{aligned} \quad (8.9)$$

on a

$$\begin{cases} \mathcal{S}_u(\mathcal{T}, V_u, \mathcal{F}) = 0, \forall u \in V \\ \mathcal{T}(u) = 0, \forall u \in \Omega_0, \end{cases} \quad (8.10)$$

avec  $V_u = \{v \in \Omega_i \cap N(u) \mid \mathcal{T}(v) \leq \mathcal{T}(u)\}$ .

Pour la réciproque, supposons que  $\mathcal{T}$  soit l'unique solution de l'équation (8.6). Alors on a

$$\begin{aligned} \mathcal{S}_u(\mathcal{T}, V_u, \mathcal{F}) &= 0, \forall u \in V \\ \mathcal{S}_u(\mathcal{T}, V_u, \mathcal{F}) &= 0, \forall u \in \Omega^i, i \in [1, N] \end{aligned} \quad (8.11)$$

où  $V_u = \{v \in \Omega_i \cap N(u) \mid \mathcal{T}(v) \leq \mathcal{T}(u)\}$ . Étant donné que  $v \in \Omega_i$ , nous avons nécessairement  $L(u) = L(v)$ , ce qui implique la relation suivante

$$\begin{aligned} V_u &= \{v \mid \mathcal{T}(v) \leq \mathcal{T}(u) \text{ et } L(v) = L(u)\} \\ &= \{v \in \Omega^i \mid \mathcal{T}(v) \leq \mathcal{T}(u)\} \\ &= V_u^i \end{aligned} \quad (8.12)$$

Et donc

$$\begin{cases} \mathcal{S}_u(\mathcal{T}, V_u^i, \mathcal{F}_i) = 0, \forall u \in \Omega^i, i \in [1, N] \\ \mathcal{T}(u) = 0, \forall u \in \Omega_0^i, \end{cases} \quad (8.13)$$

Étant donné que cette équation possède une solution unique, on a nécessairement  $\mathcal{T}(u) = \mathcal{T}^i(u)$  et  $\mathcal{F}(u) = \mathcal{F}_i(u)$ .  $\square$

Ainsi, l'instant d'arrivée de chaque front peut-être calculée en résolvant l'équation de propagation discrète modifiée du front global.

### Existence et unicité de la solution

Étant donné que l'équation (8.6) est un cas particulier de l'équation proposée au chapitre 7, elle en partage les propriétés. L'existence et l'unicité de la solution de cette équation peut donc être démontrée de manière similaire.

## 8.3 Schémas de résolution et algorithme

Dans cette section, nous considérons toujours le schéma  $\mathcal{S}_u$  de manière générale, indépendamment du choix de la norme du gradient.

### 8.3.1 Schéma de résolution local

Considérant un sommet  $u \in V$  et supposant que l'instant d'arrivée et la fonction de marqueurs sont connus en tout sommet  $v$  voisin de  $u$  et tel que  $\mathcal{T}(v) \geq \mathcal{T}(u)$ , l'équation à résoudre s'écrit

$$\mathcal{S}_u(\mathcal{T}, V_u, \mathcal{F}) = 0 \quad (8.14)$$

où  $V_u = \{v \in \mathcal{N}(u) \mid \mathcal{T}(v) \leq \mathcal{T}(u) \text{ et } L(v) = L(u)\}$  n'est pas connu.

Soit  $\mathcal{T}^i : N(u) \cap \Omega^i \rightarrow \mathbb{R}$  une fonction d'instant d'arrivée potentielle, qui associe au sommet  $u$  l'instant d'arrivée du front  $\Gamma_i$  (en négligeant les interactions avec les autres fronts). La fonction  $\mathcal{T}^i$  vérifie l'équation

$$\begin{cases} \mathcal{S}_u(\mathcal{T}^i, V_u^i, \mathcal{F}_i) = 0 \\ \mathcal{T}^i(v) = \mathcal{T}(v), \forall v \in N(u) \cap \Omega^i \end{cases} \quad (8.15)$$

où  $V_u^i = \{v \in \mathcal{N}(u) \cap \Omega^i \mid \mathcal{T}(v) \leq \mathcal{T}(u)\}$ .

Nous montrons tout d'abord que l'instant d'arrivée du front global en un sommet  $u$  coïncide avec l'instant d'arrivée potentielle du front pour lequel cet instant est minimum.

**Proposition 16.**  $\mathcal{T}(u) = \min_{i \in [1, N]} (\mathcal{T}^i(u))$  et  $L(u) = i \mid \mathcal{T}^i(u) = \mathcal{T}(u)$ .

*Démonstration.* La preuve est triviale, le front qui atteint réellement le sommet  $u$  est nécessairement celui qui potentiellement l'atteint en premier et donc celui pour lequel l'instant d'arrivée  $\mathcal{T}^i(u)$  est minimum.  $\square$



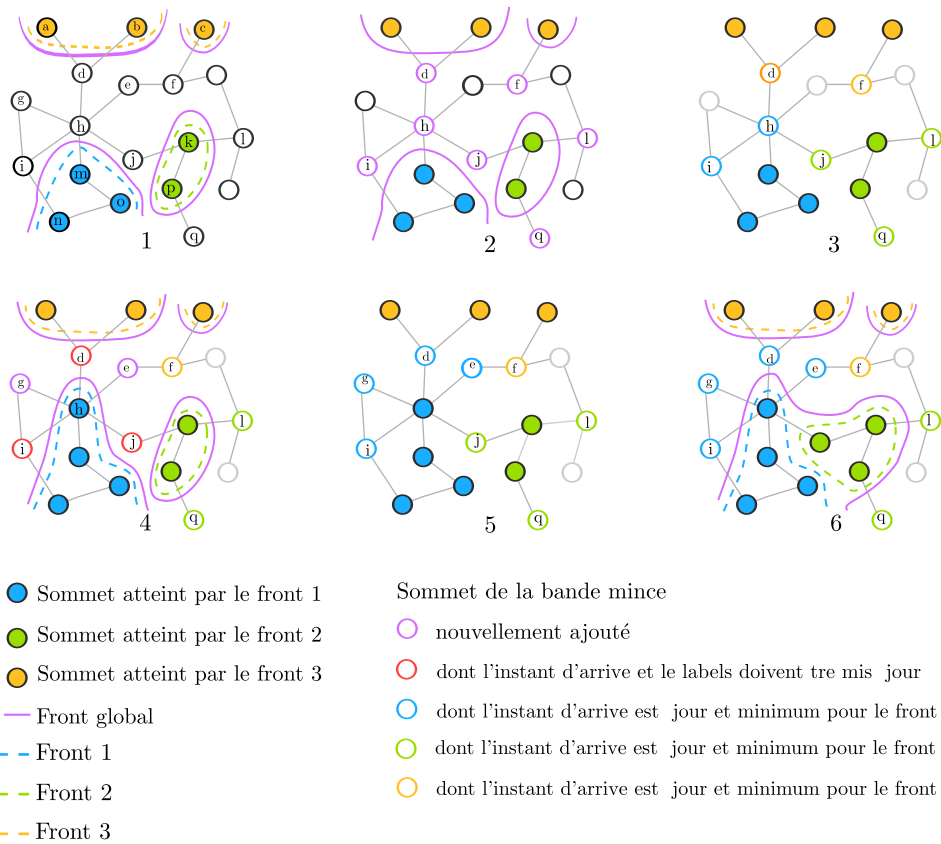


FIGURE 8.1 – Illustration de l’algorithme dans le cas de multiples fronts. 1- Trois fronts évoluant sur un graphe sont représentés par un front global. 2- Les sommets adjacents à ce front (d,f,h,i,j,l,q) sont placés dans la *bande mince*. 3- Les instants d’arrivée et le marqueur associés sont mis à jour pour les sommets de la *bande mince*. 4- Le sommet (h), possédant le plus petit instant d’arrivée, est retiré de la *bande mince* et ajouté aux sommets *actifs* du front 1 (correspondant à son marqueur). Les instants d’arrivée et marqueurs des sommets (d,i,j) doivent être mis à jours, puisque la configuration de leur voisinage a changé. Les sommets (g,e) sont ajoutés à la *bande mince*. Les sommets (f,l,q) sont fixés puisque leur voisinage reste inchangé. 5- Les sommets de la *bande mince* (d,i,j) sont mis à jour. Contrairement à l’étape 3 le sommet d est marqué par le front 1, celui ci lui associant un instant d’arrivée minimum. 6 - Le sommet (j), possédant le plus petit instant d’arrivée est retiré de la *bande mince* et ajouté aux sommets *actifs* du front 2. Les instants d’arrivée au voisinage des sommets de la *bande mince* n’ayant pas été modifiés, l’étape suivante peut donc être réalisée sans aucune mise à jour.

Ainsi, la solution locale de l'équation en un sommet  $u$  peut-être obtenue en résolvant localement l'équation du mouvement pour chaque front susceptible d'atteindre  $u$  et en sélectionnant l'instant d'arrivée et le marqueur de celui qui atteint le sommet en premier.

### 8.3.2 Schéma de résolution global

Le schéma de résolution global est similaire au schéma proposé pour l'équation d'un front unique. En se basant sur les démonstrations du chapitre précédent, il est trivial de montrer que l'évolution du front peut-être décrite par une suite  $(\mathcal{T}_n)_{n \in \mathbb{N}}$  définie par le système

$$\begin{cases} \mathcal{T}_0(u) = 0, \forall u \in \Omega_0 \\ \mathcal{T}_n(u) = \mathcal{T}_{n-1}(u), \forall u \in \Omega_n \cup (\partial^+ \Omega_n \cap \overline{U}_n) \\ \mathcal{S}_u(\mathcal{T}, V_u, \mathcal{F}) = 0, \forall u \in U_n \\ \Omega_n = \Omega_{n-1} \cup \{u \in \partial^+ \Omega_{n-1} \mid \mathcal{T}_{n-1}(u) = t_n = \min_{v \in \partial^+ \Omega_{n-1}} (\mathcal{T}_{n-1}(v))\}, \end{cases} \quad (8.16)$$

où  $V_u = \{v \in \mathcal{N}(u) \cap \Omega^i \mid \mathcal{T}(v) \leq \mathcal{T}(u)\}$  et, où  $U_n = \{u \in \partial^+ \Omega_n \mid \exists v \in \Omega_{n-1} \cap \Omega_n \text{ et } u \in \mathcal{N}(v)\}$  est l'ensemble des sommets pour lesquels il peut exister une variation entre  $\mathcal{T}_{n-1}$  et  $\mathcal{T}_n$ .

Cette suite est un cas particulier de la suite  $(\mathcal{T}_n)$  présentée dans le chapitre précédent reposant sur une simple restriction du voisinage pour la résolution. Elle vérifie donc également les propriétés de monotonie et de convergence vers la fonction  $\mathcal{T}$ .

Finalement, nous avons montré que la propagation simultanée de plusieurs fronts peut-être exprimée comme la propagation d'un front unique regroupant ces fronts et décrite par une version légèrement modifiée de l'équation stationnaire du mouvement d'un front. Dans la section suivante, nous proposons un algorithme de résolution de l'équation du mouvement de ces fronts, lui-même inspiré de l'algorithme dédié à un front unique.

**Remarque.** Ces schémas numériques peuvent également être appliqués dans le cas où la vitesse varie au cours du temps tout en restant positive. Il suffit de mettre à jour l'instant d'arrivée potentielle de tous les sommets de la bande mince à chaque instant où la vitesse se trouve modifiée.

### 8.3.3 Algorithme

De même que pour les schémas numériques, l'algorithme de résolution de l'équation stationnaire du mouvement de fronts multiples est une version légèrement modifiée de l'algorithme dédié à la propagation d'un front unique. La principale différence tient à l'introduction de la fonction de marqueurs qui permet de connaître à tout instant la position de chaque front. Une version détaillée de l'algorithme est décrite par l'algorithme 4.

## 8.4 Expérimentations

Dans cette section, nous montrons l'intérêt et le comportement de notre algorithme pour le traitement de données représentées par des graphes. Pour cela, nous illustrons le comportement de l'algorithme au travers d'un exemple jouet sur une grille régulière, puis nous présentons diverses expérimentations appliquées au partitionnement de données et d'images par la propagation simultanée d'un grand nombre de fronts et montrant l'intérêt de notre approche pour ce type d'applications. Enfin nous présentons plusieurs exemples semi-supervisés de segmentation d'images et de classification de données, toujours basés sur la propagation simultanée de plusieurs fronts évoluant sur un graphe.

### 8.4.1 Exemple jouet

Tout d'abord, nous illustrons le comportement de l'algorithme au travers d'un exemple jouet présentant la propagation simultanée de cinq fronts évoluant sur une grille régulière de  $\mathbb{R}^2$ . Le graphe est un graphe de 4-connexité. La figure 8.2 montre l'évolution des cinq fronts selon une vitesse constante et pour deux configurations différentes de la fonction de poids.

La première configuration (première ligne, figure 8.2) utilise une fonction de poids constante ( $w = 1$ ) telle que la propagation des fronts est uniforme et les fronts ne sont stoppés que lorsqu'ils rencontrent un autre front.

La seconde configuration (seconde ligne, figure 8.2) utilise une fonction de poids construite pour tenir compte des discontinuités de la couleur de fond de la grille. Ainsi, le poids est élevé (il tend vers 1) dans les zones homogènes de la grille et les fronts évoluent de manière uniforme. Au contraire, le poids est faible (il tend vers 0) au voisinage des discontinuités de la couleur de fond de la grille et la propagation des fronts s'y trouve grandement ralentie. Ainsi, la rencontre des fronts se situe donc logiquement au niveau de ces discontinuités. Ceci illustre l'importance de la fonction de poids dans la propagation des fronts.

---

**Algorithme 4** : Propagation simultanée de plusieurs fronts.

---

**Entrées :**

- Un graphe  $\mathcal{G} = (V, E, w)$  ;
- Une liste de sous-ensemble  $\Omega_0^i \subset V$  ;
- Une liste de fonction potentiel  $\mathcal{P}_i$  ;

**Sorties :**

- Une fonction d’instant d’arrivée  $\mathcal{T}$  ;
- Une fonction de marqueurs  $L$  ;
- Une fonction de marqueurs temporaires  $L^T$  associant à chaque sommet de  $NB$  le front à partir duquel l’instant d’arrivée a été calculé ;

**Données :**

- L’ensemble  $A$  des sommets actifs ;
- L’ensemble  $NB$  des sommet de la bande mince, ordonné selon  $\mathcal{T}$  ;
- L’ensemble  $FA$  des sommets lointains ;

**initialisation**

$$A \leftarrow \bigcup_{i \in [1, N]} \Omega_0^i$$

$$NB \leftarrow \partial^+ A$$

$$FA \leftarrow V - (A \cup NB)$$

$$\mathcal{T}(u) \leftarrow 0, \quad \forall u \in A$$

$$L(u) = i, \quad \forall u \in \Omega_0^i$$

$$V_u = \{v \in \mathcal{N}(u) \mid \mathcal{T}(v) \leq \mathcal{T}(u) \text{ et } L(v) = L(u)\}, \forall u \in NB ;$$

$$(\mathcal{T}(u), L^T(u)) \leftarrow \text{solution locale}(u, \mathcal{T}, V_u), \forall u \in NB ;$$

**répéter**

$u \leftarrow$  premier élément de  $NB$ .

$NB \leftarrow NB - \{u\}$ ;  $A \leftarrow A \cup \{u\}$ ;  $L(u) \leftarrow L^T(u)$ ;

**pour chaque**  $v \in N(u) \cap \bar{A}$  **faire**

$$V_v = V_v^i \mid \max_{w \in V_v^i} \left( \frac{\mathcal{T}(w)}{\gamma_{vw}} \right) = \min_{j \in [1, N]} \left( \max_{w \in V_v^j} \left( \frac{\mathcal{T}(w)}{\gamma_{vw}} \right) \right) ;$$

$$(\mathcal{T}(v), L^T(v)) \leftarrow \text{solution locale}(v, \mathcal{T}, V_v)$$

**si**  $v \in FA$  **alors**

└  $FA \leftarrow FA - \{v\}$ ;  $NB \leftarrow NB \cup \{v\}$

**sinon**

└ mettre à jour la position de  $v$  dans  $NB$

**jusqu’à**  $NB = \emptyset$  ;

---

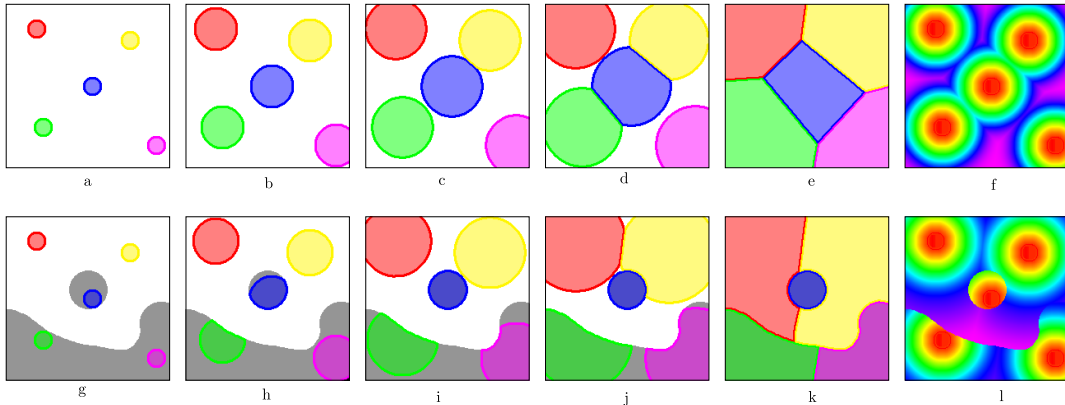


FIGURE 8.2 – Exemple jouet de la propagation simultanée de plusieurs fronts sur un graphe grille. Chaque ligne présente l'évolution de 5 fronts au travers de 5 étapes ainsi que la carte de distance finale du front global. Dans les deux cas, les fronts évoluent selon une vitesse constante  $\mathcal{F} = 1$ . Dans le premier cas, la fonction de poids est constante ( $w = 1$ ) et les fronts évoluent sans contraintes jusqu'à ce qu'ils se bloquent mutuellement. Dans le second cas, la fonction de poids est définie pour porter la similarité entre la couleur des points du domaine (blanc ou gris). La propagation des fronts est donc fortement ralentie au niveau discontinuités du fond, et les fronts ralentis se bloquent mutuellement sur ces discontinuités.

### 8.4.2 Partitionnement et simplification de graphes

Dans cette section, nous montrons l'intérêt et l'efficacité de notre approche pour le partitionnement (ou la simplification) de graphes basés sur une métrique, au travers de quelques exemples d'applications. De nombreuses méthodes de partitionnement reposant sur une métrique ont été développées et nous recommandons au lecteur intéressé par une revue complète et détaillée de se référer à [Arbeláez et Cohen 06, Bai et Sapiro 09, Falcão et al. 04] et aux références connexes.

#### Partitionnement de graphes

Tout d'abord, nous rappelons la définition d'une partition de graphe basée sur une métrique. Soit  $\mathcal{G} = (V, E, w)$  un graphe pondéré et  $C_{\mathcal{G}}(u, v)$  l'ensemble

des chemins connectant deux sommets  $u, v \in V$ . Soit  $\rho : E \rightarrow \mathbb{R}$  une métrique définie telle que

$$\rho(u, v) = \min_{\rho \in \mathcal{C}_{\mathcal{G}}(u, v)} \sum_{i=1}^{j-1} (f(u_{i+1}) - f(u_i)). \quad (8.17)$$

Étant donné un ensemble  $M = \{l_i\}_{i=1, \dots, K}$  de  $K$  marqueurs et un ensemble  $S = \{s_i\}_{i=1, \dots, K} \subset V$  de  $K$  graines (où  $s_i$  est la graine du marqueur  $l_i$ ), l'énergie  $\rho_S : V \rightarrow \mathbb{R}^m$  induit par la métrique  $\rho$  pour toute graine de  $S$  est définie telle que

$$\rho_S(u) = \min_{s_i \in S} (\rho(s_i, u)), \quad \forall u \in V. \quad (8.18)$$

La région  $R_i$  associée à un marqueur  $l_i$  est l'ensemble des sommets qui sont plus proches de  $s_i$  que de toutes les autres graines selon la métrique  $\rho$ . Ainsi, la région  $R_i$  est définie telle que

$$R_i = \{u \in V : \rho(s_i, u) \leq \rho(s_j, u)_{j=1, \dots, K}\}. \quad (8.19)$$

Étant donné un ensemble de marqueurs  $L$  (de graines  $S$ ) et une métrique  $\rho$ , la partition de  $\mathcal{G}$  est finalement donnée par l'ensemble

$$R(M, \rho) = \{R_i, \forall l_i \in L\}. \quad (8.20)$$

Ainsi, considérant un graphe  $\mathcal{G} = (V, E, w)$ , construire une partition de  $\mathcal{G}$  revient à trouver le plus court chemin entre chaque sommet et l'ensemble des graines, ce qui peut facilement être réalisé via un algorithme de type Dijkstra.

Cette formulation générale du partitionnement de graphe basé sur une métrique peut facilement être reliée à notre formulation pour la propagation de fronts multiples sur graphe. En effet, si l'on considère chaque graine  $s_i$  comme le point de départ d'un front  $\Gamma_i$ , la relation entre la fonction d'instant d'arrivée  $\mathcal{T}$  du front global regroupant les fronts  $\Gamma_i$  et l'énergie  $\rho_S$  est donnée par

$$\begin{aligned} \mathcal{T}(u) &= \min(\mathcal{T}_i(u)) \\ &= \min_{s_i \in S} (\rho(u, s_i)) \\ &= \rho_S(u). \end{aligned} \quad (8.21)$$

Ainsi, une région est définie par

$$R_i = \{u \in V : L(u) = i\}. \quad (8.22)$$

Finalement, nous remarquerons que la notion de partitionnement de graphe est intrinsèquement liée à notre formulation, qui produit naturellement une partition du graphe.

## Super Vertices

Dans ce paragraphe, et en se basant sur la capacité de notre algorithme à créer des partitions, nous proposons une généralisation de la méthode des Super Pixels aux graphes, que nous appelons Super Vertices.

Initialement développée par [Ren et Malik 03], la méthode des Super Pixels permet de réduire la complexité d'une image en regroupant les pixels la composant en régions qui préservent les contours intérieurs de l'image. Avec les Turbo Pixels, [Levinshtein et al. 09] ont proposé une implémentation des Super Pixels dans laquelle la partition de l'image en régions est obtenue par la dilatation simultanée et contrôlée d'une grille régulière de marqueurs jusqu'à recouvrir l'intégralité de l'image. La dilatation est basée sur une approche par ensembles de niveaux et dépend de la structure de l'image. Ainsi, les contours des régions coïncident avec les contours de l'image.

Dans ce manuscrit, nous proposons d'étendre cette approche de la simplification d'images aux graphes en se basant sur notre algorithme rapide de propagation de fronts multiples (algorithme 4). Étant donné que cette approche regroupe des sommets et non des pixels, nous l'appelons Super Vertices. Comme dans le cas des Turbo Pixels, notre approche utilise un ensemble de marqueurs placés régulièrement sur le graphe, et propagés jusqu'à ce que chaque sommet ait été marqué. La dilatation des marqueurs se fait non plus via un algorithme itératif d'évolution, mais via la propagation simultanée de plusieurs fronts (un front étant associé à chaque marqueur). Cette propagation est effectuée en une passe par l'algorithme de propagation de fronts multiple (algorithme 4) et selon une vitesse constante  $\mathcal{F} = 1$ , commune à tous les fronts.

## Illustrations

Dans le cas d'un maillage 3D, la notion de grille régulière n'étant pas présente, les marqueurs initiaux sont placés de manière semi-aléatoire sur les nœuds du maillage de manière à ce que les marqueurs ne soient pas trop proches et présents de façon homogène sur la surface du maillage. La figure 8.3 présente le résultat d'une partition en Super Vertices appliquée à des maillages 3D. Pour chaque maillage, la position des marqueurs initiaux, la carte de distance correspondant à chaque front et la partition sont illustrés. Les lecteurs intéressés par une revue des différentes méthodes de la littérature pour la segmentation et le partitionnement de maillages 3D peuvent se référer à [Shamir 08].

Dans le cas d'images 2D, l'initialisation du processus de partitionnement est très similaire à celle utilisé par [Levinshtein et al. 09] pour les Turbo Pixels : les marqueurs sont placés de manière homogène à partir d'une grille régulière

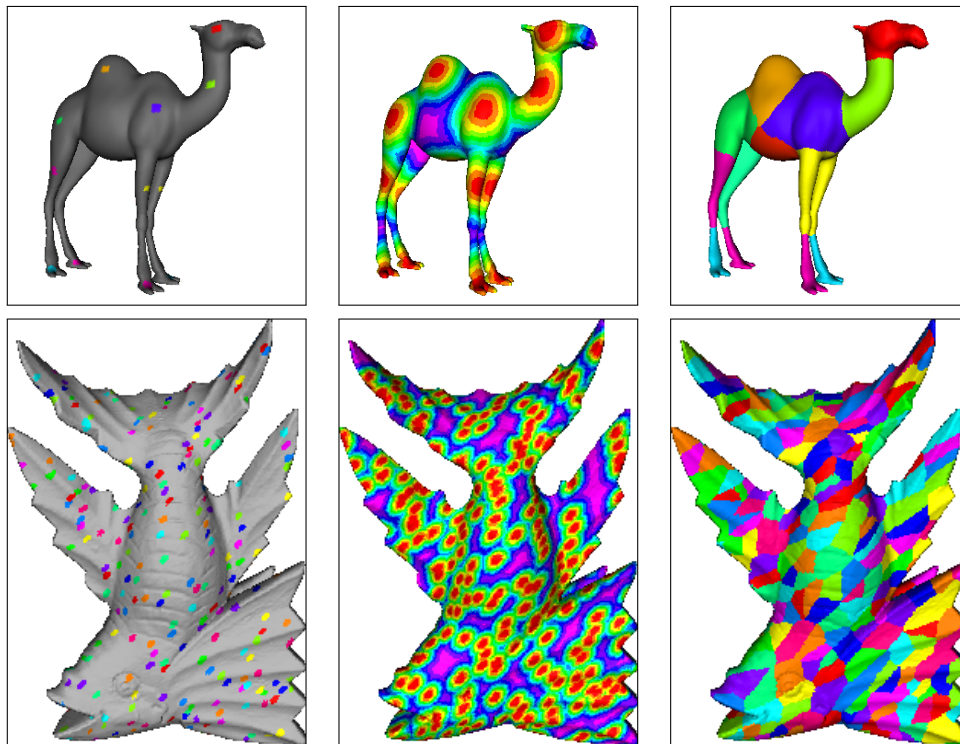


FIGURE 8.3 – Super vertices sur maillage 3D. La première colonne présente le mesh original avec l'ensemble des graines pour la décomposition en Super Vertices. La seconde colonne présente la carte de distance issue de la propagation simultanée de tous les fronts. La dernière colonne présente la carte de régions obtenue.

légèrement déformée afin de prévenir la présence de marqueurs à proximité immédiate des contours de l'image. Cette déformation est effectuée en déplaçant la position des marqueurs dans la direction du gradient décroissant. La figure 8.4 présente le résultat d'une telle partition appliquée à une image 2D, ainsi que la carte de distance associée à chaque front.

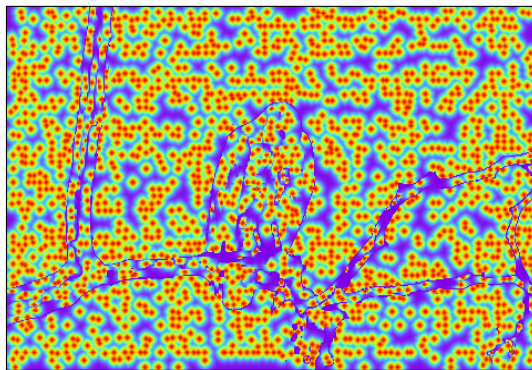
### 8.4.3 Segmentation semi-supervisée d'images

Dans cette section, nous montrons le comportement et l'efficacité de notre méthode de propagation de fronts pour la segmentation semi-supervisée d'images. Nous présentons tout d'abord une comparaison qualitative et quantitative avec la récente méthode du Power Watershed, puis nous montrons l'intérêt et l'apport de notre approche pour la segmentation d'images, via deux

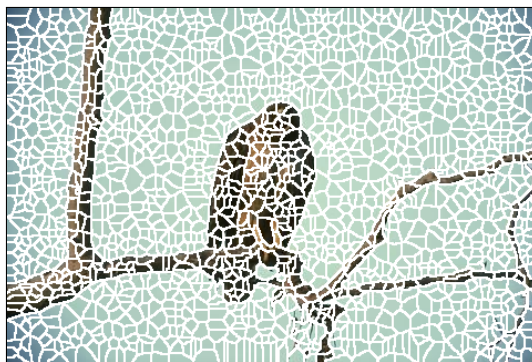




Original + graines



Carte de distance



Partition

FIGURE 8.4 – Super vertices sur image 2D. La première image présente l'image originale avec les graines pour la décomposition en Super vertices. La seconde image présente la carte de distance issue de la propagation simultanée de tous les fronts. La dernière image présente la carte de région obtenue, où la couleur de chaque région est la couleur moyenne des pixels de la région.

représentations non classiques d'une image via l'utilisation de graphes non locaux ou basés sur une partition de l'image (RAG).

### État de l'art des approches basées graphe

Dans le domaine de la segmentation d'image, les approches basées sur une représentation d'une image par un graphe ont connu un intérêt grandissant ces dernières années. En effet, de nombreux algorithmes reposant sur des graphes ont été proposés, tel que les Graph cuts [Boykov et Jolly 01], les Random Walks [Grady 06], l'Image Forest Transform [Falcão et al. 04], les méthodes des chemins minimaux [Dijkstra 59, Bai et Sapiro 09], la ligne de partage des eaux [Najman et Schmitt 94, Bertrand 05, Cousty et al. 10] ou encore plus récemment le Power Watershed [Couprie et al. 11b] qui unifie certaines des méthodes précédemment citées.

En effet, [Sinop et Grady 07, Couprie et al. 11b] ont montré que toutes ces méthodes peuvent être vues comme des cas particuliers d'une méthode générale dépendant de deux paramètres  $p$  et  $q$  et reposant sur la minimisation de l'énergie

$$E_{p,q}(f) = \sum_{u \sim v \in E} w_{uv}^p |f(u) - f(v)|^q, \quad (8.23)$$

où  $f$  représente la configuration souhaitée.

De manière générale, et dans le cas où l'on cherche à séparer le graphe en deux classes distinctes  $A$  et  $B$ , l'algorithme associé à cette méthode consiste à trouver une fonction  $f$  satisfaisant l'équation (8.23) en résolvant le système suivant

$$\begin{cases} f(u) = 1 & \forall u \in S_A \\ f(u) = 0 & \forall u \in S_B \\ f = \arg \min_g E_{p,q}(g), \end{cases} \quad (8.24)$$

puis seuiller la fonction  $f$  pour obtenir une partition en deux classes. Ainsi, la fonction de marqueur  $L : V \rightarrow \{A, B\}$  (qui associe à chaque sommet une classe) est définie telle que

$$L(u) = \begin{cases} B & \text{if } f(u) < \frac{1}{2} \\ A & \text{if } f(u) \geq \frac{1}{2}. \end{cases} \quad (8.25)$$

**Remarque.** L'équation (8.23) peut facilement être interprétée en terme d'une EdP telle que

$$E_{p,q}(f) = \sum_{u \sim v \in E} \|(\nabla_{w'} f)(u)\|_q, \quad (8.26)$$

où le poids  $w'$  dépend de  $w$ ,  $p$  et  $q$  tel que  $w' = w^{2p/q}$

Dans les cas impliquant plus de deux classes ( $N$  classes), il est nécessaire de résoudre  $N$  systèmes similaires à (8.24) où le marqueur A correspond à la classe et le marqueur B à toutes les autres classes (méthode du un contre tous). Finalement, la fonction de marqueur  $L : V \rightarrow \{C_i\}_{i=1,\dots,N}$  (qui associe à chaque sommet une classe) est définie telle que

$$L(u) = C_i \mid f_i(u) = \max_{j=1,\dots,N} f_j(u) \quad (8.27)$$

L'inconvénient majeur d'une telle approche est qu'il est nécessaire de résoudre autant de fois le système (8.24) qu'il y'a de classes à séparer, cette résolution étant elle même effectuée par un algorithme itératif nécessitant de parcourir l'ensemble des sommets un grand nombre de fois.

### Comparaison avec le Power Watershed

Dans ce paragraphe, nous proposons une comparaison quantitative et qualitative entre notre approche et la méthode récente du Power Watershed qui est une des méthodes les plus efficaces parmi celles proposées dans le paragraphe précédent. Les résultats présentés pour le Power Watershed ont été obtenus à partir du code source fourni sur le site web de [Couprie et al. 11b].

La comparaison est effectuée sur les images de la base Microsoft Grabcut. Le problème considère une segmentation en deux classes (objet d'intérêt et fond) à partir d'une version érodée des marqueurs initiaux proposés par la base (les méthodes basées sur une croissance de régions étant toujours favorisées lorsque les marqueurs initiaux sont trop proches des bords). La figure 8.5 présente les résultats obtenus avec le Power Watershed, comparés aux résultats obtenus avec notre algorithme de propagation de fronts, pour des valeurs de  $p \in \{1, 2, \infty\}$ .

### Segmentation d'images texturées

Par construction, un graphe peut représenter les interactions non locales. Nous montrons dans ce paragraphe l'intérêt de notre méthode pour la segmentation d'images texturées via l'utilisation de graphes non locaux construits à partir de ces images.

La figure 8.6 présente de manière comparative des résultats de segmentation obtenus à partir d'une construction locale du graphe d'une part et d'une

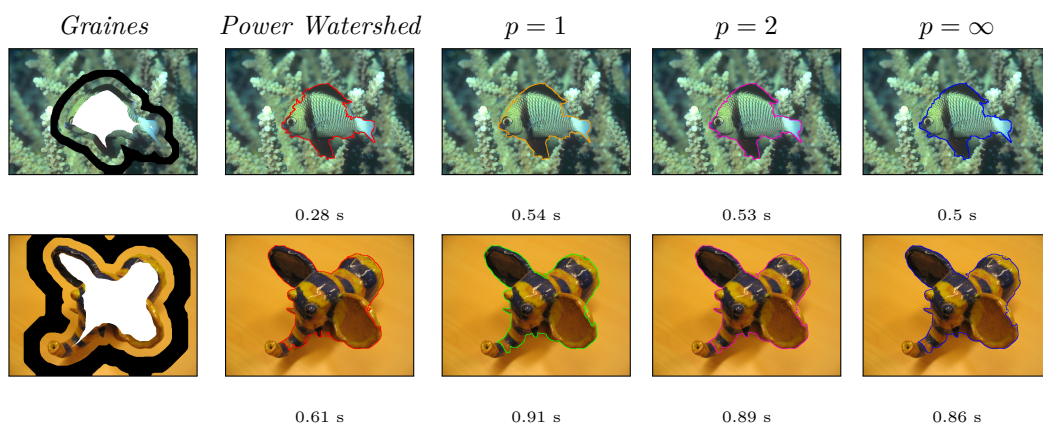


FIGURE 8.5 – Comparaison avec le Power Watershed. La première colonne présente l’image originale avec les marqueurs initiaux (graines). Les quatre suivantes présentent le résultat de la segmentation et le temps de calcul pour les quatre méthodes comparées.

construction non locale d’autre part. Dans chaque cas, la fonction vitesse est constante ( $\mathcal{F} = 1$ ) et la fonction de poids est définie comme une mesure de similarité ( $g_4$ ) entre les vecteurs couleurs des pixels, ou des patches dans le cas non local. Dans le cas local, le graphe est construit comme un graphe de 4-connexité et chaque pixel est représenté simplement par son vecteur couleur. Les interactions sont donc limitées au voisinage immédiat. Dans le cas non local, le graphe est construit en considérant un voisinage étendu à tous les pixels présents dans une fenêtre de dimension  $11 \times 11$  centrée sur le pixel et chaque pixel est représenté par un patch de vecteurs couleurs de dimension  $3 \times 3$  centré sur lui même. Les interactions sont ainsi élargies à un plus grand nombre de pixels, ce qui permet de bien mieux prendre en compte la texture. Nous remarquons une bien meilleure qualité de la segmentation dans le cas d’un graphe non local, notamment au niveau de la queue du tigre pour la deuxième image.

### Segmentation utilisant un RAG étendu

Un avantage de notre approche basée sur des graphes est que nous pouvons utiliser notre algorithme sur n’importe quelle représentation d’une image pour peu que cette représentation soit assimilable à un graphe pondéré. L’objectif de ce paragraphe est donc d’illustrer à la fois cette adaptivité de notre algorithme à tous types de graphes et l’avantage que nous pouvons en tirer pour la segmentation d’objets similaires mais non spatialement connectés d’une image.

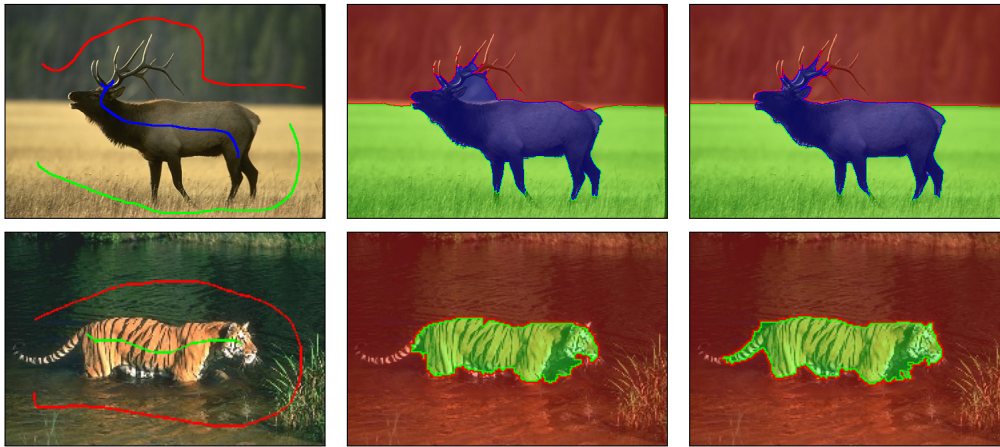


FIGURE 8.6 – Segmentation d’images texturées par graphe non local. La première colonne présente les images originales avec les marqueurs initiaux en surimpression. La seconde présente le résultat de la segmentation via un graphe local. La dernière présente le résultat de la segmentation via un graphe non local.

La figure 8.7 présente une image contenant plusieurs objets d’intérêt que nous souhaitons extraire (les éléphants) et les deux marques placées par un utilisateur pour la segmentation (les deux points rouges font référence à un même marqueur). Avec cette configuration, il n’est pas possible d’extraire l’ensemble des éléphants avec la seule marque bleue et sur un graphe de 4-connexité car il n’existe pas d’arêtes entre les différents éléphants. Ajouter des arêtes supplémentaires entre tous les pixels, afin de rendre le graphe non local, augmenterait de manière très significative la complexité de l’algorithme tout en entraînant un fort biais.

Aussi, nous proposons de remplacer la représentation de l’image correspondant à un graphe de 4-connexité, par un graphe plus petit et possédant des arêtes entre les parties similaires et non connectées de l’image. Pour cela, nous utilisons une version simplifiée de l’image (une partition obtenue par Super Vertices), sur laquelle nous construisons un graphe d’adjacence de régions. Afin de connecter les régions distantes, le graphe est étendu de manière à ce qu’une arête supplémentaire soit ajoutée entre chaque région et ses deux régions les plus similaires (au sens de la couleur moyenne) sur l’ensemble de l’image. Ceci est illustré sur la figure 8.7(c).

Enfin, la propagation des deux fronts associés aux marqueurs de l’utilisateur est effectuée sur ce RAG étendu où le nombre réduit de sommets ne pénalise pas la rapidité de l’algorithme. Le résultat de la propagation est illustré sur la figure 8.7(d) où la frontière finale entre les deux fronts est présentée en rouge.



## 8.4. Expérimentations

Finalement, nous avons montré que cette approche présente les avantages suivants :

- Rapidité de calcul. En réduisant le nombre de sommet (environ 98% de sommets en moins dans l'exemple de la figure 8.7), la construction du graphe des  $k$ -ppv et la propagation des fronts impliquant des interactions non-locales sont fortement accélérées.
- Prise en compte des interactions non locales.
- Nombre réduit de marqueurs initiaux. Avec un seul marqueur initial il devient possible d'extraire des objets non connectés.

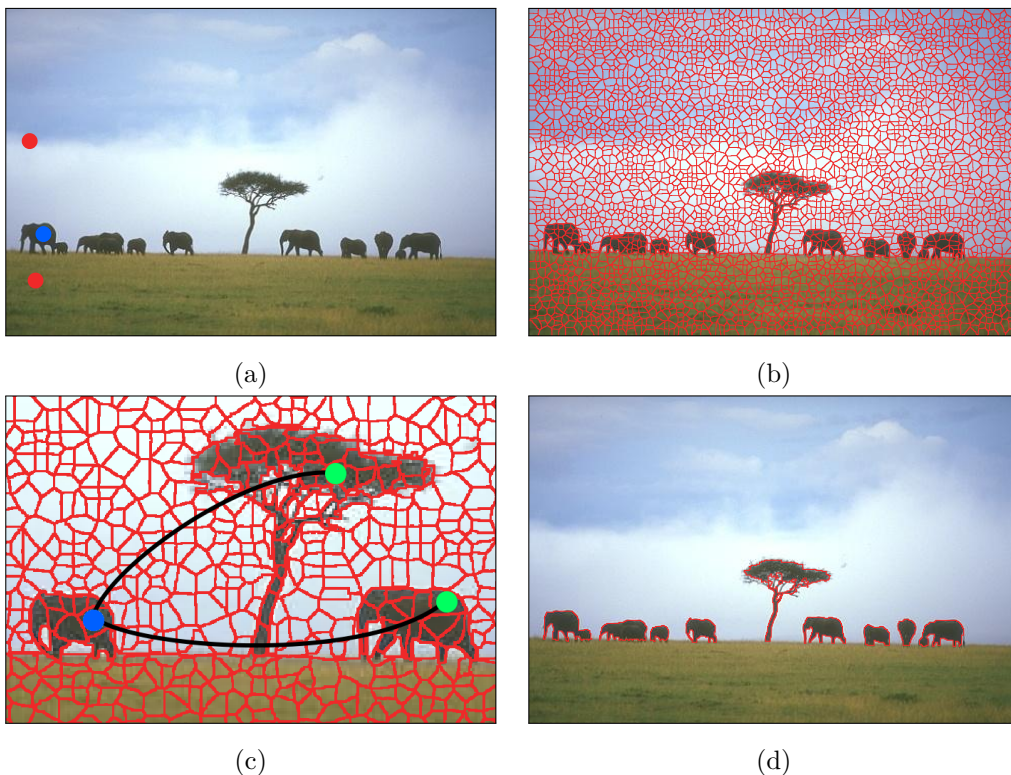


FIGURE 8.7 – Segmentation d'images RAG. (a) L'image originale et les marqueurs de l'utilisateur. (b) Les contours des régions de la partition de l'image. (c) Illustration des arêtes supplémentaires du RAG pour un sommet. (d) Les contours de la segmentation.

### 8.4.4 Classification semi-supervisée de données

Dans ce paragraphe, nous illustrons l'intérêt de notre approche appliquée à la classification semi-supervisée de données. La classification semi-supervisée

consiste à associer une classe (ou un marqueur) à chaque élément d'un ensemble de données dont une partie a été préalablement marquée.

La littérature scientifique est riche de méthodes de classification semi-supervisées. Nous pouvons citer par exemple les méthodes hiérarchiques, les réseaux de neurones, ou les méthodes basées sur la théorie des graphes. À notre connaissance, les méthodes basées sur la propagation de fronts n'ont jamais été appliquées à la classification semi-supervisées de données. La seule exception étant l'approche utilisant la version non-stationnaire de l'équation eikonale proposée par [Ta et al. 09a].

La figure 8.8 présente deux exemples de classification semi-supervisée de données utilisant notre approche. Dans un souci de clarté, les deux bases de données utilisées sont des bases de données d'images. La première correspond à un échantillon de la base USPS de chiffres manuscrits et où chaque image de la base est représentée par le vecteur d'intensités de l'ensemble des pixels de l'image. La seconde correspond à un ensemble de noyaux cellulaires segmentés depuis une lame cytologique et où chaque noyau est représenté par un vecteur de descripteurs (forme, couleur, etc.). Dans les deux cas, le graphe construit est un graphe des  $k$ -ppv et la fonction de poids est construite selon la fonction de similarité  $g_4$  appliquée aux vecteurs caractéristiques de chaque image. La première ligne de la figure 8.8 présente le graphe construit pour la base USPS avec les marqueurs initiaux définis par nos soins puis les marqueurs propagés sur l'ensemble du graphe par deux fronts. La seconde ligne de la figure 8.8 présente le graphe construit pour la base de noyaux cellulaires avec les marqueurs provenant d'une base de référence et le résultat de la classification.

Nous pouvons remarquer que nous obtenons des résultats de classification satisfaisants, avec des taux de classification approchant les 100 % dans chacun des cas.

Toutes ces expérimentations montrent les potentialités de notre approche basée sur des graphes pour la segmentation d'images et la classification de données de toutes sortes.

## 8.5 Conclusion

Dans ce chapitre, nous avons poursuivi notre objectif de proposer une formulation et un algorithme général pour la propagation simultanée de plusieurs fronts sur un même graphe.

Pour cela, nous avons étendu la formulation introduite au chapitre précédent pour nous intéresser au cas où plusieurs fronts évoluent simultanément

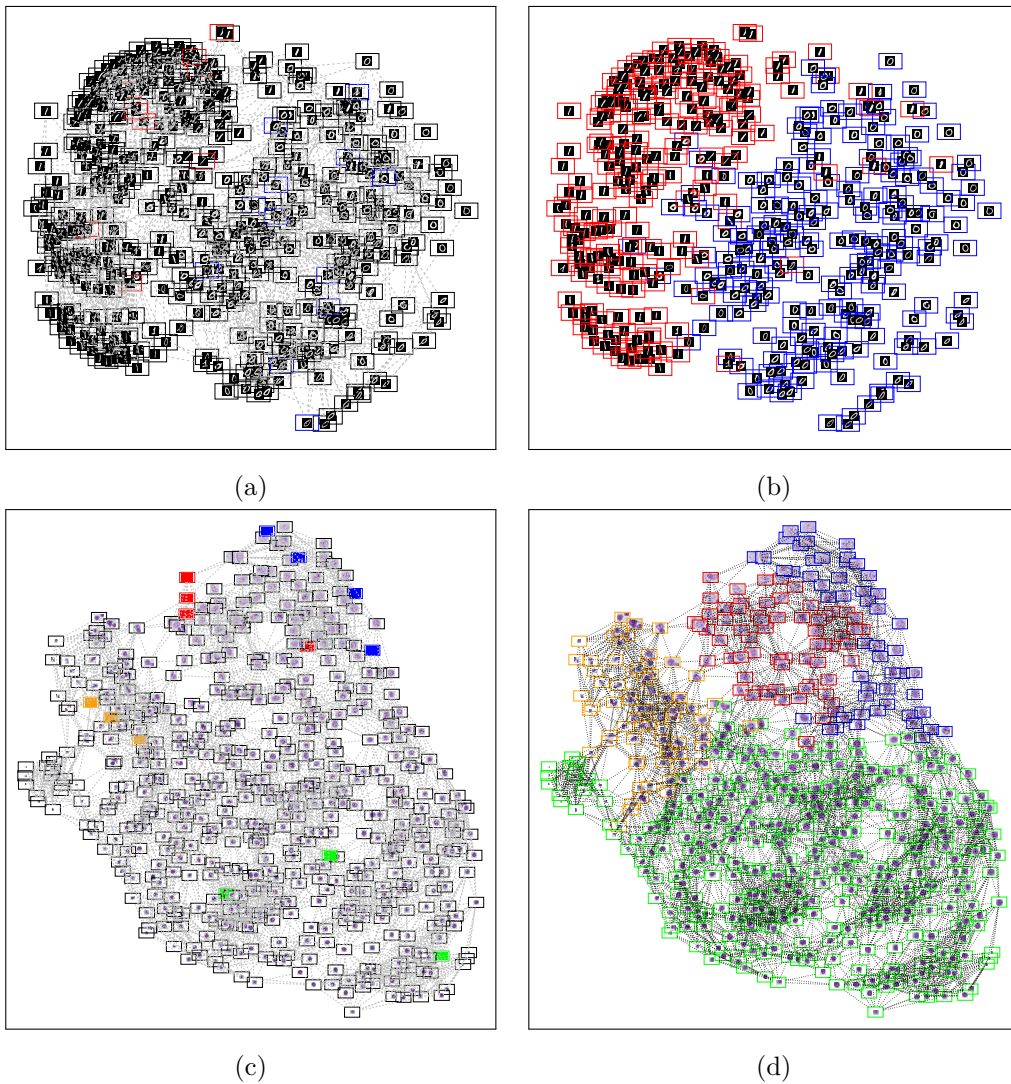


FIGURE 8.8 – Classification semi-supervisée de données.

sur un même graphe et sous certaines contraintes sur les interactions entre ces fronts.

Nous avons montré que l'évolution simultanée de ces fronts peut être réduite à la propagation d'un front global regroupant ces fronts, dont l'évolution est contrôlée par une version modifiée de l'équation stationnaire présentée dans le cas d'un front unique. Nous avons également proposé un schéma de résolution de l'équation modifiée ainsi qu'un algorithme rapide de résolution dont la complexité ne dépend pas du nombre de fronts.



Finalement, nous avons illustré le comportement et l'intérêt de notre approche appliquée au traitement des images, des maillages et des données. Nous avons notamment montré l'intérêt de la propagation simultanée de multiples fronts pour la simplification de graphes via une généralisation des super pixels (que nous appelons super vertices). Nous avons également comparé notre approche avec une des méthodes les plus efficaces de segmentation d'image basée sur des graphes (le power watershed) et présenté de nombreux exemples de segmentation d'image, illustrant l'intérêt d'une méthode adaptative qui permet de manipuler différentes représentations d'une image comme les RAG ou les graphes non-locaux. Enfin, nous avons illustré l'intérêt de notre approche pour la classification semi-supervisée de données.

Dans le chapitre suivant, nous étendons une seconde fois notre formulation pour considérer le cas le plus général où plusieurs fronts évoluent simultanément sur un même graphe sans restriction sur le signe de leurs vitesses.

# Chapitre 9

## Généralisation sans restrictions sur le signe de la vitesse

### Sommaire

---

<b>9.1</b>	<b>Cas d'un front unique</b>	<b>170</b>
9.1.1	Relation entre propagation interne et externe	170
9.1.2	Représentation d'un front comme deux fronts	172
<b>9.2</b>	<b>Extension au cas de fronts multiples</b>	<b>173</b>
9.2.1	Définition du front global et des fronts secondaires	175
9.2.2	Schémas de résolution et algorithme	177
<b>9.3</b>	<b>Expérimentations</b>	<b>181</b>
9.3.1	Exemple jouet	181
9.3.2	Contours actifs	181
<b>9.4</b>	<b>Conclusion</b>	<b>186</b>

---

Dans ce chapitre, nous nous intéressons finalement au cas le plus général, où plusieurs fronts peuvent évoluer simultanément sur un graphe sans restriction de signe sur leur fonctions de vitesse.

Nous étudions tout d'abord le cas d'un front unique évoluant selon une vitesse positive sur une partie du graphe, et selon une vitesse négative sur l'autre partie. Nous montrons ainsi qu'un front peut être décomposé en deux fronts : l'un représentant l'ensemble des sommets pour lesquels la vitesse est positive, et l'autre représentant le complémentaire de l'ensemble des sommets pour lesquels la vitesse est négative et que l'évolution simultanée de ces deux fronts coïncide avec l'évolution du front entier. Les lecteurs intéressés peuvent trouver une approche similaire sur le principe, qui a été proposée par [Forcadel et al. 08, Carlini et al. 08] pour la propagation d'un front sur une grille, sans restriction sur le signe de la vitesse.

Puis, nous étendons cette approche dans le cas où plusieurs fronts évoluent simultanément sur le graphe, chacun de ces fronts n'ayant aucune restriction sur le signe de leur fonction de vitesse.

Enfin, nous montrons que notre approche peut-être très simplement généralisée au cas où la vitesse du front dépend également du temps.

## 9.1 Cas d'un front unique

Nous montrons tout d'abord que l'évolution d'un front  $\Gamma$  entièrement soumis à une vitesse négative peut-être exprimée comme l'évolution d'un front  $\Gamma^c$  opposé à  $\Gamma$  et entièrement soumis à une vitesse positive.

Puis, dans le cas où le front  $\Gamma$  est soumis à une vitesse positive ou négative, nous montrons que l'évolution peut être exprimée comme l'évolution combinée de deux pseudo-fronts signés, l'un noté  $\Gamma^+$  et représentant l'évolution des parties du front soumises à une vitesse positive et l'autre noté  $\Gamma^-$  et représentant l'évolution des parties du front soumises à une vitesse négative.

### 9.1.1 Relation entre propagation interne et externe

Dans cette section, nous montrons que la propagation interne (ou rétraction) d'un front soumis à une vitesse négative  $\mathcal{F}$ , est équivalente à une propagation externe (ou dilatation) du front opposé soumis à une vitesse opposée à  $\mathcal{F}$ .

Considérons un front  $\Gamma$ , défini à tout instant  $t$  par le sous-ensemble  $\Omega_t$ , tel que  $u \in \Omega_t$  si l'instant d'arrivée  $\mathcal{T}$  de  $\Gamma$  en  $u$  est inférieur ou égal à  $t$ . L'évolution de ce front est contrôlée par une vitesse, notée  $\mathcal{F}$ , définie négative sur l'ensemble du graphe.

La position de ce front à tout instant  $t$  est donné par le niveau zéro d'une fonction d'ensembles de niveaux notée  $\phi : V \times \mathbb{R} \rightarrow \mathbb{R}$  vérifiant l'équation non-stationnaire du mouvement d'un front se rétractant (6.9)

$$\frac{\partial \phi(u, t)}{\partial t} = \mathcal{F}(u) \left\| \left( \nabla_w^- \phi \right) (u) \right\|_p, \quad (9.1)$$

et par définition (7.1), nous avons la relation  $\mathcal{T}(u) = t - \phi(u, t)$ .

### 9.1. Cas d'un front unique

---

Soit  $\Gamma^c$  le front opposé à  $\Gamma$ , défini à tout instant  $t$  par le sous-ensemble  $\Omega_t^c = \overline{\Omega}_t \cup \{v \in \Omega_t \mid \phi(v, t) = 0\}$  et tel que  $u \in \Omega_t^c$  si l'instant d'arrivée  $\mathcal{T}$  de  $\Gamma$  en  $u$  est inférieur ou égal à  $t$ . Ainsi, les instant d'arrivée des deux fronts coïncident et si l'on note  $\mathcal{T}^c$  l'instant d'arrivée du front  $\Gamma^c$ , nous avons l'égalité suivante  $\mathcal{T}^c = \mathcal{T}$ . L'évolution de ce front est contrôlée par une vitesse, notée  $\mathcal{F}^c$ , et la position de ce front à tout instant  $t$  est donnée par le niveau zéro d'une fonction d'ensembles de niveaux  $\phi^c : V \times \mathbb{R} \rightarrow \mathbb{R}$ . Nous montrons que si  $\mathcal{F}$  est négative, alors  $\mathcal{F}^c$  est nécessairement positive.

**Proposition 17.** *Si  $\Omega_t^c = \overline{\Omega}_t \cup \{v \in \Omega_t \mid \phi(v, t) = 0\}$  et  $\mathcal{T} = \mathcal{T}^c$ , alors  $\mathcal{F}^c \geq 0$ .*

*Démonstration.* Nous procédons par l'absurde. Supposons que  $\mathcal{F}^c < 0$ . Alors nous pouvons écrire l'égalité

$$\begin{aligned} \phi^c(u, t) &= \mathcal{T}(u) - t \\ &= \phi(u, t). \end{aligned} \tag{9.2}$$

Cette égalité implique donc qu'il existe  $u \in \overline{\Omega}_t \subset \Omega_t^c$  tel que  $\phi^c(u) < 0$ . Ce qui est contradictoire avec la définition de  $\phi^c$ .  $\square$

Ainsi, d'après la proposition 17, la vitesse  $\mathcal{F}^c$  du front complémentaire à  $\Gamma$  est positive sur l'ensemble des sommets, et  $\phi^c$  vérifie l'équation non-stationnaire du mouvement d'un front se dilatant (6.9)

$$\frac{\partial \phi^c(u, t)}{\partial t} = \mathcal{F}^c(u) \left\| (\nabla_w^+ \phi^c)(u) \right\|_p. \tag{9.3}$$

De plus, étant donné le signe de  $\mathcal{F}^c$ , on a

$$\phi^c(u, t) = t - \mathcal{T}(u) = -\phi(u, t). \tag{9.4}$$

**Proposition 18.** *Si  $\mathcal{T} = \mathcal{T}^c$ , alors  $\mathcal{F}^c = -\mathcal{F}$  et  $\mathcal{T}$  vérifie*

$$\mathcal{F}^c(u) \left\| (\nabla_w^- \mathcal{T})(u) \right\|_p = 1 \tag{9.5}$$

*Démonstration.*  $\mathcal{T} = \mathcal{T}^c$  implique que  $\phi^c = -\phi$ . Donc

$$\begin{aligned} \frac{\partial \phi^c(u, t)}{\partial t} &= -\mathcal{F}(u) \left\| (\nabla_w^- (-\phi^c))(u) \right\|_p \\ \frac{\partial \phi^c(u, t)}{\partial t} &= -\mathcal{F}(u) \left\| (\nabla_w^+ \phi^c)(u) \right\|_p \\ \mathcal{F}^c(u) \left\| (\nabla_w^+ \phi^c)(u) \right\|_p &= -\mathcal{F}(u) \left\| (\nabla_w^+ \phi^c)(u) \right\|_p \\ \mathcal{F}^c(u) &= -\mathcal{F}(u) \end{aligned} \tag{9.6}$$

$\square$

Ainsi, d'après la proposition 18, la fonction d'instant d'arrivée du front  $\Gamma$  en tout sommet du graphe vérifie l'équation eikonale sur graphe avec une vitesse  $\mathcal{F}^c = -\mathcal{F}$ . Finalement, grâce à la dualité entre front et sous-ensemble de  $V$ , la rétractation d'un front  $\Gamma$  peut-être exprimée simplement par l'équation d'évolution

$$\begin{cases} \left\| \left( \nabla_w^- \mathcal{T} \right) (u) \right\|_p - \mathcal{P}(u) = 0 \\ \mathcal{T}(u) = 0, \forall u \in \Omega_0^c, \end{cases} \quad (9.7)$$

où  $\mathcal{P}(u) = 1/|\mathcal{F}(u)|$ . Cette équation correspond exactement à l'équation stationnaire du mouvement du front  $\Gamma^c$  opposé à  $\Gamma$  et peut être réécrite sous la forme du schéma

$$\begin{cases} \mathcal{S}_u(\mathcal{T}, V_u, -\mathcal{F}) = 0 \\ \mathcal{T}(u) = 0, \forall u \in \Omega_0^c, \end{cases} \quad (9.8)$$

et résolue via l'algorithme 3 proposé dans le chapitre 7.

### 9.1.2 Représentation d'un front comme deux fronts

Nous avons jusqu'à présent considéré que la vitesse  $\mathcal{F}$  était négative sur l'ensemble des sommets. Dans le cas où le signe de la vitesse n'est pas constant, nous allons montrer que l'évolution d'un front  $\Gamma$  peut-être exprimée comme l'évolution de deux pseudo-fronts signés  $\Gamma^+$  et  $\Gamma^-$  représentant respectivement les parties du front se dilatant ou se rétractant.

Soit  $\Gamma$  un front évoluant sur un graphe  $\mathcal{G} = (V, E, w)$  et défini à l'instant  $t$  par le sous ensemble  $\Omega_t$ . Ce front évolue sous l'influence d'une vitesse  $\mathcal{F} : V \rightarrow \mathbb{R}$  et son instant d'arrivée en chaque sommet de  $G$  est donné par une fonction  $\mathcal{T} : V \rightarrow \mathbb{R}$ .

Soient  $V^+$  le sous-ensemble de  $V$  où la vitesse  $\mathcal{F}$  est positive, et  $V^-$  le sous-ensemble de  $V$  où  $\mathcal{F}$  est strictement négative.

Soit  $\Gamma^+$ , la partie du front  $\Gamma$  évoluant sur  $V^+$ , telle que le front  $\Gamma^+$  soit défini à l'instant  $t$  par le sous ensemble  $\Omega_t^+ = V^+ \cap \Omega_t$ . Soit  $\mathcal{T}^+ : V \rightarrow \mathbb{R}$  la fonction d'instant d'arrivée du front  $\Gamma^+$  en tout sommet  $u \in V^+$  et vérifiant l'équation

$$\begin{cases} \mathcal{S}_u(\mathcal{T}^+, V_u, \mathcal{F}) = 0, \quad \forall u \in V^+ \\ \mathcal{T}^+(u) = 0, \forall u \in \Omega_0^+. \end{cases} \quad (9.9)$$

Soit  $\Gamma^-$ , la partie du front  $\Gamma$  évoluant sur  $V^-$ , telle que le front  $\Gamma^-$  soit défini à l'instant  $t$  par le sous ensemble  $\Omega_t^- = V^- - \Omega_t^c$ . Soit  $\mathcal{T}^- : V \rightarrow \mathbb{R}$  la fonction d'instant d'arrivée du front  $\Gamma^-$  en tout sommet  $u \in V^-$  et vérifiant.

$$\begin{cases} \mathcal{S}_u(\mathcal{T}^-, V_u, -\mathcal{F}) = 0, & \forall u \in V^- \\ \mathcal{T}^-(u) = 0, & \forall u \in \Omega_0^-. \end{cases} \quad (9.10)$$

**Proposition 19.** *L'évolution du front  $\Gamma$  coïncide avec l'évolution simultanée de  $\Gamma^+$  et  $\Gamma^-$ , et nous avons la relation*

$$\mathcal{T}(u) = \begin{cases} \mathcal{T}^+(u), & \forall u \in V^+ \\ \mathcal{T}^-(u), & \forall u \in V^- \end{cases} \quad (9.11)$$

*Démonstration.* Par définition,  $\mathcal{T}$  et  $\mathcal{T}^+$  sont solution de  $\mathcal{S}_u = 0$ ,  $\forall u \in V^+$ . Cette solution étant unique, on a  $\mathcal{T}(u) = \mathcal{T}^+(u)$ ,  $\forall u \in V^+$ . De même, d'après la proposition 18,  $\mathcal{T}$  et  $\mathcal{T}^-$  sont solution de  $\mathcal{S}_u = 0$ ,  $\forall u \in V^-$ , cette solution étant unique, on a  $\mathcal{T}(u) = \mathcal{T}^-(u)$ ,  $\forall u \in V^-$ .  $\square$

Ainsi, l'évolution du front  $\Gamma$  coïncide avec l'évolution des deux pseudo-fronts  $\Gamma^+$  et  $\Gamma^-$  évoluant sur deux ensembles disjoints de  $V$ . En effet, par définition, on a  $V^+ \cap V^- = \emptyset$  et  $V^+ \cup V^- = V$ .

Finalement, les deux pseudo-fronts  $\Gamma^+$  et  $\Gamma^-$  vérifient les mêmes propriétés que les fronts concurrents présentés dans le chapitre précédent et leur évolution peut donc facilement être calculée en résolvant l'équation de propagation d'un front global réunissant ces deux fronts et implémentée par l'algorithme 4. Ceci est illustré pour quelques itérations à la figure 9.1.

**Remarque.** *L'évolution du front  $\Gamma^-$  correspondant à un retrait du front  $\Gamma$ , le marqueur qui lui est associé par la fonction  $L$  est zéro (ce qui signifie aucun front).*

## 9.2 Extension au cas de fronts multiples

Dans cette section, nous généralisons encore la formulation au cas où plusieurs fronts évoluent sur le graphe, chacun de ces fronts étant sous l'influence d'une vitesse positive sur une partie du graphe et négative sur une autre partie.



De même, soit  $\Gamma_i^-$  la partie du front  $\Gamma_i$  évoluant sur  $V_i^-$ , telle que le front  $\Gamma_i^-$  soit défini à l'instant  $t$  par le sous-ensemble  $\Omega_t^{i-} = V_i^- - \Omega_t^i$ . Soit  $\mathcal{T}_i^- : V \rightarrow \mathbb{R}$  la fonction d'instant d'arrivée potentielle du front  $\Gamma_i^-$  en tout sommet  $u \in V_i^-$  (en négligeant l'influence des autres fronts) et vérifiant

$$\begin{cases} \mathcal{S}_u(\mathcal{T}_i, V_u, |\mathcal{F}_i|) = 0, & \forall u \in V_i^- \\ \mathcal{T}_i^-(u) = 0, & \forall u \in \Omega_0^{i-}. \end{cases} \quad (9.13)$$

D'après la proposition 19 nous avons la relation

$$\mathcal{T}_i(u) = \begin{cases} \mathcal{T}_i^+(u), & \forall u \in V_i^+ \\ \mathcal{T}_i^-(u), & \forall u \in V_i^-, \end{cases} \quad (9.14)$$

où  $\mathcal{T}_i$  est la fonction d'instant d'arrivée potentielle du front  $\Gamma_i$ .

### 9.2.1 Définition du front global et des fronts secondaires

Suivant la même approche que celle définie pour des fronts évoluant uniquement dans la direction de la normale (voir la section 8.1), nous cherchons à représenter l'ensemble de ces fronts par un front global les regroupant tel que l'évolution de ce front corresponde à l'évolution des fronts regroupés.

Nous avons vu dans la section précédente qu'un front dont le signe de la vitesse n'est pas constant peut être représenté par deux fronts disjoints définis sur deux sous-ensembles de l'ensemble des sommets, disjoints eux aussi. Nous avons également présenté au chapitre 8, une formulation permettant de décrire l'évolution simultanée de plusieurs fronts comme un front global.

Malheureusement, cette formulation implique que les fronts soient strictement disjoints à tout instant de leur évolution et cette condition ne peut être garantie en ce qui concerne les fronts représentés par des pseudo-fronts signés. En effet, les propriétés de non chevauchement et de blocage mutuel des fronts réels ne permettent pas de garantir ces mêmes propriétés pour les pseudo-fronts signés. Il est d'ailleurs trivial de montrer que dans de nombreux cas particuliers ces fronts se chevauchent (voir la figure 9.2).

Afin de pallier ce problème, nous proposons d'introduire, en plus du front global représentant l'évolution de l'ensemble des fronts positifs, des fronts secondaires (un par front négatif) évoluant simultanément avec le front global, mais possédant des instants d'arrivées distincts. L'évolution de  $k$  fronts réels est donc réalisée comme l'évolution simultanée d'un front global et de  $k$  fronts secondaires correspondant chacun à la partie négative d'un front réel.



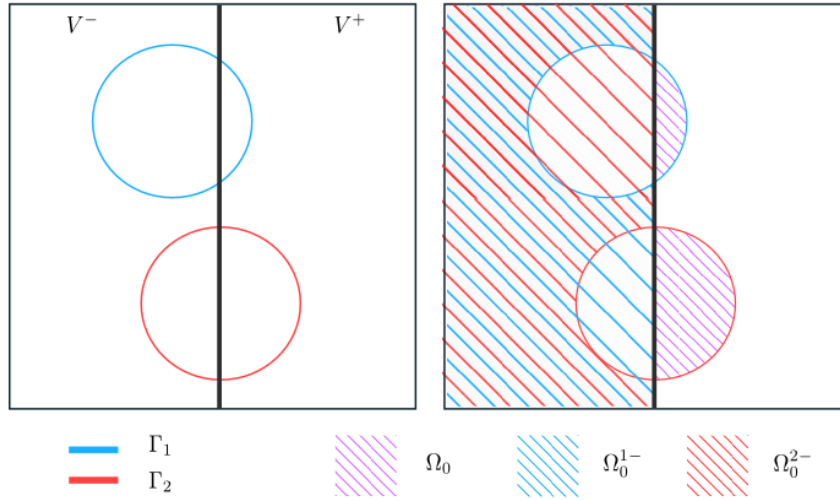


FIGURE 9.2 – Illustration du chevauchement des fronts. Soient  $\Gamma_1$  et  $\Gamma_2$ , deux fronts soumis à une vitesse positive à droite et négative à gauche. Les sous-ensembles  $\Omega_0^{1-}$  et  $\Omega_0^{2-}$  correspondant aux pseudo-fronts négatifs de ces deux fronts se chevauchent et leur union est égale à  $V^-$ .

**Remarque.** *Il est impossible de représenter l'ensemble des fronts négatifs comme un front unique, l'intersection de leur sous-ensembles respectifs n'étant pas vide.*

Avec cette configuration, les fronts secondaires correspondent exactement aux fronts négatifs  $\Gamma_i^-$ . Considérant un front secondaire  $\Gamma_i^-$ , l'instant d'arrivée de ce front en chaque sommet de  $\Omega_0^i$  est défini par la fonction  $\mathcal{T}_i^-$  vérifiant (9.13). En effet,  $\Gamma_i^-$  correspondant à la rétractation du front  $\Gamma_i$ , il ne peut se propager qu'à l'intérieur de l'ensemble  $\Omega_0^i \cup V_i^-$ . De plus, puisque les fronts réels ne peuvent se chevaucher, nous avons la garantie que le front  $\Gamma_i^-$  sera nécessairement le premier à atteindre les sommets de l'ensemble  $\Omega_0^i \cup V_i^-$ .

Similairement au cas où les vitesses sont toujours positives, le front global  $\Gamma$  est représenté à l'instant initial par le sous-ensemble  $\Omega_0$  défini tel que

$$\Omega_0 = \bigcup_{i \in [1, N]} \Omega_0^{i+}. \quad (9.15)$$

Pour définir la fonction d'arrivée  $\mathcal{T} : V \rightarrow \mathbb{R}$  associée à ce front, nous devons considérer deux cas. Soit  $u$  un sommet du graphe. Si le sommet  $u$  n'appartient à aucun front, alors le front global peut s'y propager librement et l'instant d'arrivée  $\mathcal{T}$  peut-être calculé comme le minimum des instants d'arrivées de tous les fronts positifs (voir le chapitre précédent). Au contraire, si le sommet  $u$

appartient à un front qui se rétracte, alors le front global ne peut s'y propager qu'une fois que le front secondaire s'est rétracté du sommet  $u$ . Dans ce cas, l'instant d'arrivée en ce sommet correspond au maximum entre l'instant où le front secondaire se rétracte et l'instant où le front global est en situation de se propager. Ainsi, la fonction d'instant d'arrivée du front global est définie de telle sorte que

$$\mathcal{T}(u) = \begin{cases} \max(\mathcal{T}_j^-(u), \mathcal{T}_i^+(u)) & \forall u \in \Omega_0^j \cup \Omega^i \text{ et } \forall i \in [1, N] \\ \mathcal{T}_i^+(u) & \forall u \in \Omega^i \text{ et } \forall i \in [1, N] \end{cases} \quad (9.16)$$

Finalement, le problème de l'évolution de  $k$  fronts réels soumis à des vitesses positives et négative revient à trouver  $k + 1$  fonctions correspondant aux fonctions d'instant d'arrivée des  $k$  fronts secondaires et du front global. Nous montrons dans la section suivante que ce problème peut être résolu simplement en utilisant les différents schéma numériques présentés dans les deux chapitres précédents et implémenté comme une version modifiée de notre algorithme pour la propagation simultanée de fronts multiples.

### 9.2.2 Schémas de résolution et algorithme

Dans cette section, nous présentons les schémas de résolution pour un front secondaire et pour le front global.

#### Schéma pour un front secondaire

Dans le cas d'un front négatif, le schéma de résolution est strictement identique au schéma de résolution pour un front unique. En effet, considérant un front  $\Gamma_i^-$ , nous savons que ce front se propage uniquement sur  $V_i^- - \Omega_0^i$  et qu'il sera nécessairement le premier à atteindre les sommets de cet ensemble.

Par souci de clarté, nous notons ici  $A_0 = V_i^- - \Omega_0^i$ . En reprenant les démonstrations et schémas de résolution (local et global) présentés au chapitre 7, il est trivial de montrer que l'évolution du front peut-être décrite par une suite  $(f_n)_{n \in \mathbb{N}}$  définie par le système

$$\begin{cases} f_0(u) = 0, \forall u \in A_0 \\ f_n(u) = f_{n-1}(u), \forall u \in A_n \cup (\partial^+ A_n \cap \overline{U_n}) \\ \mathcal{S}_u(f_n, V_u, \mathcal{F}) = 0, \forall u \in U_n \\ A_n = A_{n-1} \cup \{u \in \partial^+ A_{n-1} \mid f_{n-1}(u) = t_n = \min_{v \in \partial^+ \Omega_{n-1}} (f_{n-1}(v))\}, \end{cases} \quad (9.17)$$

où  $V_u = \mathcal{N}(u) \cap \mathcal{A}_n$  et, où  $U_n = \{u \in \partial^+ A_n \mid \exists v \in A_{n-1} \cap A_n \text{ et } u \in \mathcal{N}(v)\}$  est l'ensemble des sommets pour lesquels il peut exister une variation entre  $f_{n-1}$  et  $f_n$ .

De même, il est trivial de montrer que cette suite est monotone et converge vers la solution  $\mathcal{T}_i^-$ .

### Schéma pour le front global

Dans le cas du front global, le schéma de résolution est une version modifiée du schéma de résolution présenté au chapitre précédent, pour la propagation simultanée de plusieurs fronts. La différence tient au maximum introduit dans l'équation (9.16) qui se retrouve dans le calcul de la solution locale.

### Cas où la vitesse dépend du temps

Cette approche générale pour la propagation simultanée de plusieurs fronts, peut aisément être appliquée dans le cas où la vitesse  $\mathcal{F}$  dépend également du temps. Dans ce cas, la propagation du front peut être vue comme une succession de petits pas d'évolution, chaque pas correspondant à un intervalle de temps sur lequel la fonction  $\mathcal{F}$  est constante. Ainsi, nous pouvons résoudre l'équation de propagation sur chaque intervalle  $I_n$ , où l'instant d'arrivée et la position du front sur l'intervalle  $I_n$  sont initialisés avec l'instant d'arrivée et la position du front à la fin de l'intervalle  $I_{n-1}$ .

### Algorithme

L'algorithme 5 associé est une version modifiée de l'algorithme présenté au chapitre précédent. À chaque itération, le sommet  $u$  ayant l'instant d'arrivée supposé minimum (sans distinction entre le front global et les fronts secondaires) est retiré de la liste des sommets candidats. Si le sommet est sur un front secondaire, alors la fonction d'instant d'arrivée associée à ce front est mise à jour pour les sommets au voisinage de  $u$  et susceptibles d'être atteints par le front secondaire. Afin que le sommet  $u$  puisse être atteint par le front global (il est maintenant libre de tout front), son marqueur est mis à zéro. Au contraire, si le sommet n'appartient à aucun front, la fonction d'instant d'arrivée associée au front global est mise à jour pour les sommets au voisinage de  $u$  et susceptibles d'être atteints par le front global. Le marqueur du front réel qui a permis de calculer l'instant d'arrivée en  $u$  est associé à ce sommet. La cohérence entre les différentes fonctions d'instant d'arrivée est implicitement

maintenue, puisque si un sommet candidat est au voisinage à la fois du front global et d'un front secondaire, le front global sera négligé à cette itération, mais le sommet sera maintenu dans la bande mince. Voir la figure 9.3 pour une illustration de l'algorithme pour quelques itérations.

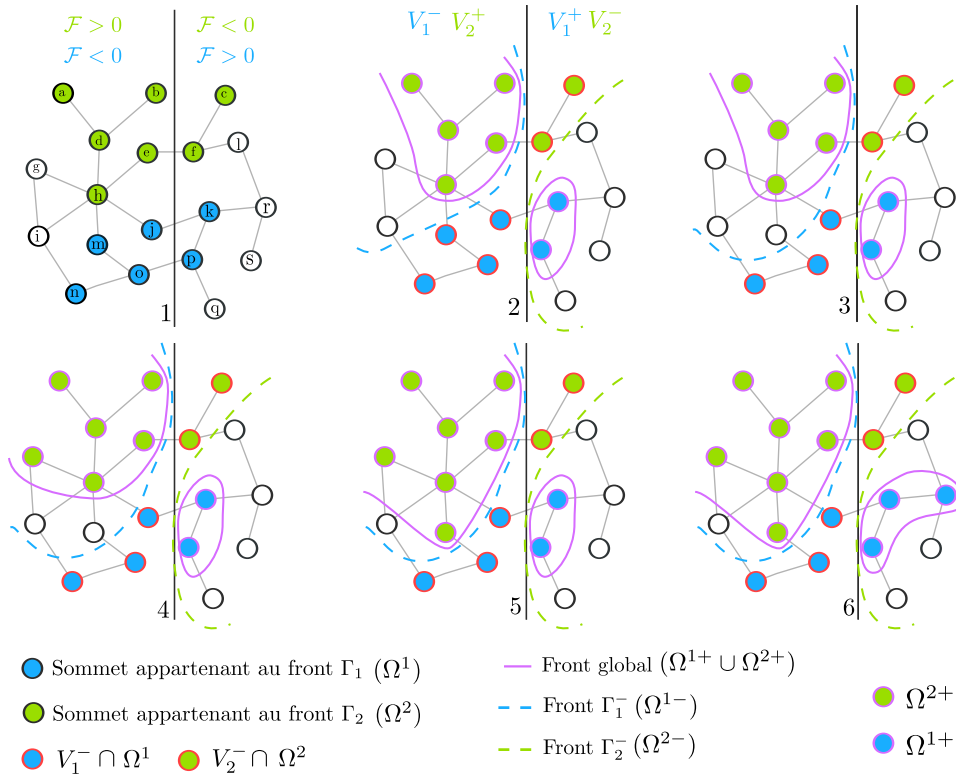


FIGURE 9.3 – Illustration de l'algorithme pour quelques itérations. 1. Deux fronts  $\Gamma_1$  et  $\Gamma_2$  évoluent sur un graphe, soumis à des vitesses positives et négatives (pour plus de clarté, la vitesse des deux fronts est opposée, chaque front se propageant sur une partie du graphe, et se rétractant sur l'autre). 2. Les sommets des deux fronts soumis à une vitesse positive sont rassemblés sous un front global (en rose). Deux fronts secondaires (en bleu et vert) sont ajoutés pour représenter la rétractation de chacun des fronts réels. Ces fronts sont définis comme les complémentaires des fronts réels, dans le sous ensemble du graphe où leurs vitesses sont négatives. Chaque front secondaire est indépendant. 3. Le front  $\Gamma_1^-$  se propage sur le sommet  $h$  (plus petit instant d'arrivée) et le sommet se trouve libéré du front  $\Gamma_1$ . 4. Le front global se propage sur le sommet  $g$  sous l'influence des sommets de  $\Gamma_2$ . 5. Le sommet  $h$  étant libre, le front global peut s'y propager (en accord avec la fonction d'instant d'arrivée du front global). 6. Le front global se propage sur le sommet  $r$  sous l'influence de  $\Gamma_1$ .

**Algorithme 5** : Propagation simultanée de plusieurs fronts sans restrictions sur le signe de leurs vitesses.

---

**Entrées :**

- Un graphe  $\mathcal{G} = (V, E, w)$  ;
- Une liste de sous-ensembles  $\Omega_0^i \subset V$  ;
- Une liste de fonctions de potentiels  $\mathcal{P}_i$  ;

**Sorties :**

- Une fonction d’instant d’arrivée  $\mathcal{T}$  pour le front global ;
- Une liste de fonctions d’instant d’arrivée  $\mathcal{T}_i^-$  pour les fronts secondaires ;
- Une fonction de marqueurs  $L$  ;
- Une fonction de marqueurs temporaires  $L^T$  associant à chaque sommet le front positif à partir duquel l’instant d’arrivée à été calculé ;

**Données :**

- L’ensemble  $A$  des sommets actifs du front global ;
- Une liste de sous-ensemble  $A_i$  des sommets actifs des fronts secondaires ;
- L’ensemble  $NB$  des sommets de la bande mince, ordonné selon  $\mathcal{T}$  et  $\mathcal{T}_i^-$  ;

**initialisation**

$$A \leftarrow \bigcup_{i \in [1, N]} \Omega_0^{i+} ; A_i \leftarrow (V_i^- - \Omega_0^i) ; NB \leftarrow \partial^+ A \bigcup_{i \in [1, N]} \partial^+ A_i$$

$$\mathcal{T}(u) \leftarrow 0, \quad \forall u \in A ; \mathcal{T}_i^-(u) \leftarrow 0, \quad \forall u \in A_i$$

$$L(u) = i, \quad \forall u \in \Omega_0^i ; L(u) = 0, \quad \forall u \in (V - \bigcup_{i \in [1, N]} \Omega_0^i)$$

**pour chaque  $u \in NB$  faire**

- si**  $L(u) = i \neq 0$  **alors**
  - └  $\mathcal{T}_i^-(u) \leftarrow$  solution locale( $u, \mathcal{T}_i^-, V_u$ ),  $\forall u \in NB$
- sinon**
  - └  $(\mathcal{T}(u), L^T(u)) \leftarrow$  solution locale( $u, \mathcal{T}, V_u$ ),  $\forall u \in NB$

**répéter**

- └  $u \leftarrow$  premier élément de  $NB$  ;  $NB \leftarrow NB - \{u\}$  ;
- si**  $L(u) = i \neq 0$  **alors**
  - └  $A_i \leftarrow A_i \cup \{u\}$  ;  $L(u) = 0$
  - pour chaque**  $v \in N(u) \cap \bar{A}_i \cap V_i^-$  **faire**
    - └  $\mathcal{T}_i^-(v) \leftarrow$  solution locale( $v, \mathcal{T}_i^-, V_v$ )
    - └ mettre à jour la position de  $v$  dans  $NB$
- sinon**
  - └  $A \leftarrow A \cup \{u\}$  ;  $i = L^T(u)$  ;  $L(u) = L^T(u)$
  - pour chaque**  $v \in N(u) \cap \bar{A} \cap V_i^+$  **tel que**  $L(v) = 0$  **faire**
    - └  $(\mathcal{T}(v), L^T(v)) \leftarrow$  solution locale( $v, \mathcal{T}, V_v$ )
    - └ mettre à jour la position de  $v$  dans  $NB$

**jusqu’à**  $NB = \emptyset$  ;

---

## 9.3 Expérimentations

Dans cette section, nous montrons l'intérêt et le comportement de notre algorithme pour le traitement de données représentées par des graphes. Pour cela, nous illustrons le comportement de l'algorithme au travers d'un exemple jouet sur une grille régulière, puis nous présentons diverses expérimentations appliquées au partitionnement de données et d'images par la propagation simultanée de plusieurs fronts évoluant sans restriction sur le signe de leur vitesse.

### 9.3.1 Exemple jouet

Tout d'abord, nous illustrons le comportement de l'algorithme au travers d'un exemple jouet présentant la propagation simultanée de 3 fronts évoluant sur une grille régulière de  $\mathbb{R}^2$ . Le graphe est un graphe grille en 4-connexité et la fonction de poids est constante ( $w = 1$ ). La figure 9.4 montre l'évolution des 3 fronts évoluant selon des vitesses positive ou négative. Le front vert évolue selon une vitesse  $\mathcal{F}_1$  définie positive sur le fond gris et négative sur le fond blanc. Au contraire, les fronts bleu et rouge évoluent selon une vitesse  $\mathcal{F}_2$  définie négative sur le fond gris et positive sur le fond blanc.

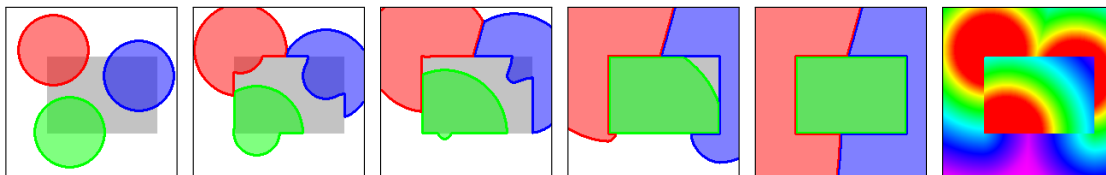


FIGURE 9.4 – Exemple jouet de la propagation simultanée de plusieurs fronts soumis à des vitesses positives et négatives. La propagation est réalisée sur une grille régulière pour un graphe grille en 4-connexité et une fonction de poids constante ( $w = 1$ ). Voir le texte pour plus de détails.

### 9.3.2 Contours actifs

Dans ce paragraphe, nous illustrons le comportement de notre algorithme (l'algorithme 5) appliqué à la propagation de contours actifs. L'évolution de ces contours est définie selon une version modifiée du modèle de [Chan et Vese 01] dans laquelle nous ne tenons pas compte de la courbure. La propagation est donc uniquement contrôlée par la différence entre la moyenne à l'intérieur et à l'extérieur du contour d'une fonction caractéristique  $F$  définie sur les sommets (le vecteur couleur, dans le cas d'une image).

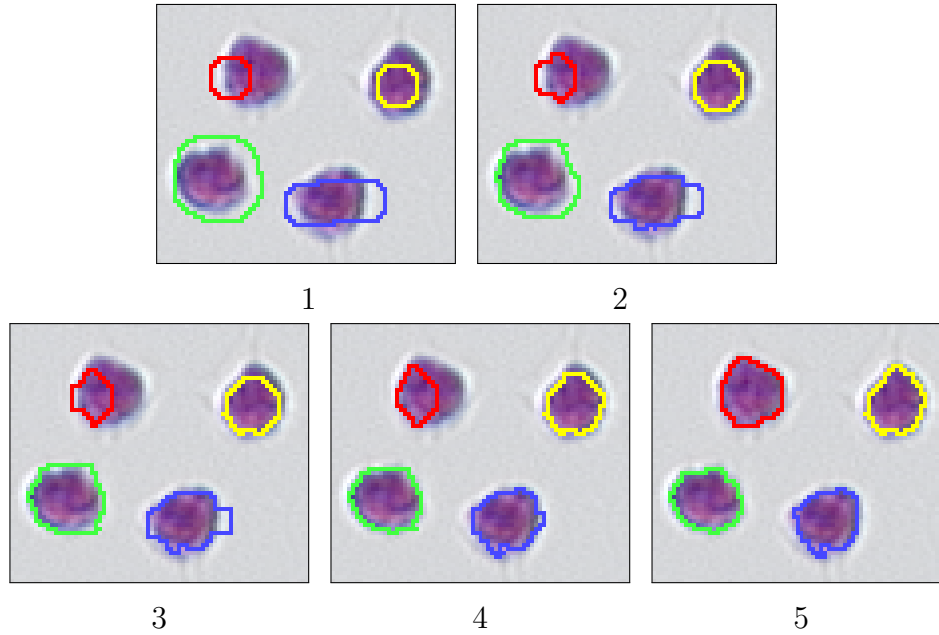


FIGURE 9.5 – Contours actifs multiples sur une image. Les cinq images présentent l'évolution des différents contours à différents instants.

Chaque contour est représenté par un front  $\Gamma_i$  distinct dont la propagation est contrôlée par une vitesse propre  $\mathcal{F}_i$ , définie selon une version modifiée de l'adaptation discrète du modèle de Chan et Vese (voir la section 6.5.3 pour plus de détails sur cette adaptation). Ainsi, puisque les différentes fonctions de vitesse ne sont pas constantes dans le temps, la propagation des différents contours actifs est réalisée comme une succession de propagations sur de courts intervalles de temps, et telle que la fonction de vitesse  $\mathcal{F}_i$  est définie pour un front  $\Gamma_i$  sur l'intervalle  $[t, t + \tau]$  telle que

$$\mathcal{F}_i(u) = d_2\left(F(u) - \mu_o^i u(t)\right)^2 - d_2\left(F(u) - \mu_i^i n(t)\right)^2 \quad (9.18)$$

où  $\mu_o^i u^i$  correspond à la moyenne d'une fonction caractéristique  $F$  à l'extérieur du front  $\Gamma_i$ , et  $\mu_i^i n^i$  correspond à la moyenne à l'intérieur du front  $\Gamma_i$ .

Afin de simuler l'évolution de la vitesse dans le temps, nous définissons un pas de temps  $\tau$  contrôlant la fréquence de mise à jour des fonctions de vitesse. Sur chaque intervalle  $[t, t + \tau]$ , les fonctions de vitesse sont considérées comme constantes, et la propagation des fronts est réalisée par l'algorithme 5. À l'instant  $t + \tau$ , les fonctions de vitesse sont mises à jour et l'algorithme est réinitialisé tout en conservant la nouvelle position du front. Ceci est répété jusqu'à une condition d'arrêt dépendant de l'application.

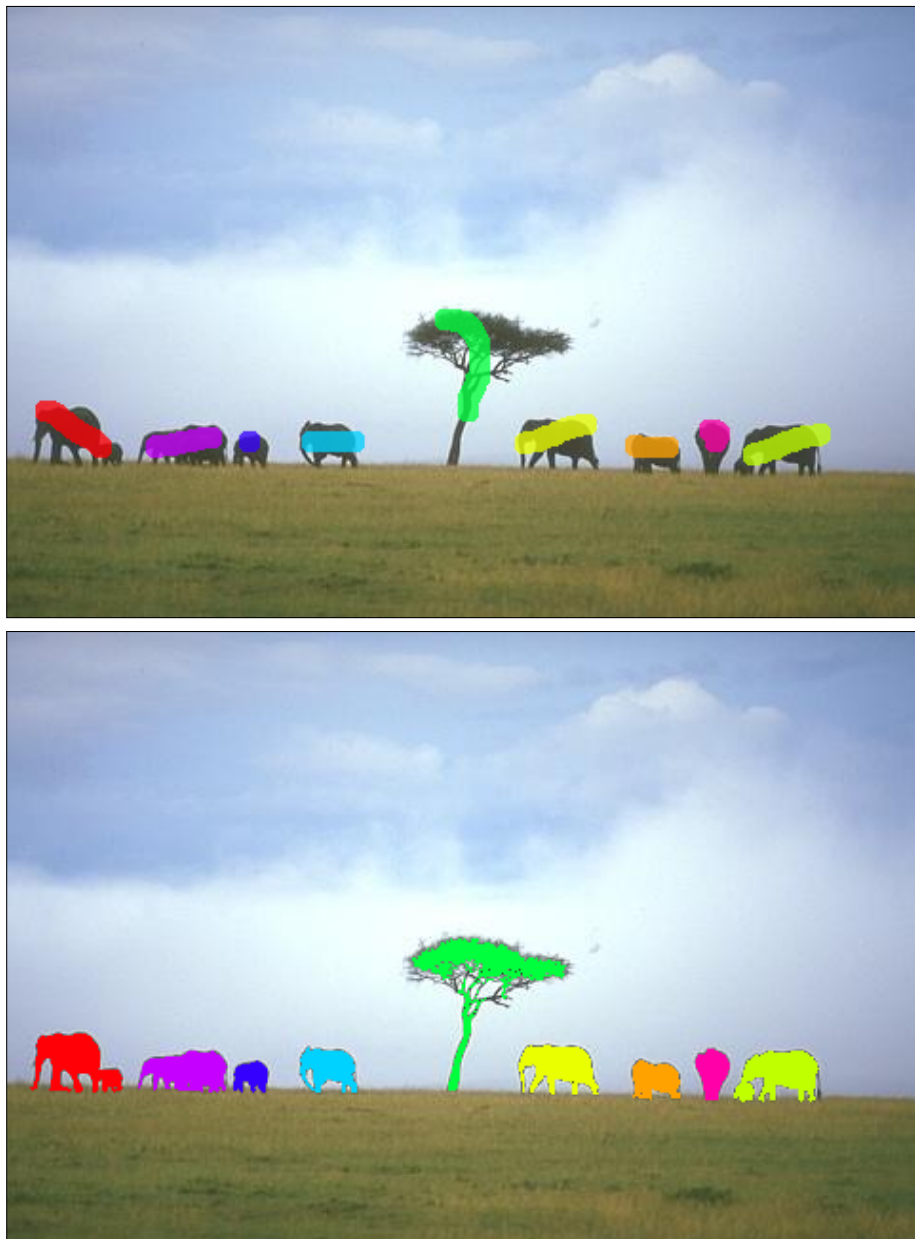


FIGURE 9.6 – Contours actifs multiples sur une image.

### Sur une image

La figure 9.5 présente un exemple de contours actifs multiples pour la segmentation semi-supervisée d'images. L'image contient 4 objets d'intérêt que nous souhaitons extraire à l'aide de quatre fronts dont les marqueurs initiaux ont été volontairement placés de façon grossière de manière à ce qu'ils intègrent



à la fois des éléments du fond et de l'objet à extraire. La propagation simultanée de ces fronts est réalisée par l'algorithme 5 et la fonction de vitesse  $\mathcal{F}_i$  de chaque front est calculée en fonction de la couleur des pixels à l'intérieur et à l'extérieur des fronts (selon notre version modifiée du modèle de Chan et Vese). La figure 9.5 présente différentes étapes de l'évolution simultanée des quatre contours. La propagation s'arrête d'elle-même lorsque la bande mince est vide.

La figure 9.6 présente un second exemple de contours actifs multiples sur une image mettant en œuvre la propagation simultanée de neuf fronts. Ici encore, les contours ont été initialisés de façon grossière par l'utilisateur. Notre approche montre ici tout son intérêt pour la segmentation semi-supervisée d'objet multiples, autorisant une certaine marge d'erreur à l'utilisateur lors de la définition des marqueurs initiaux.

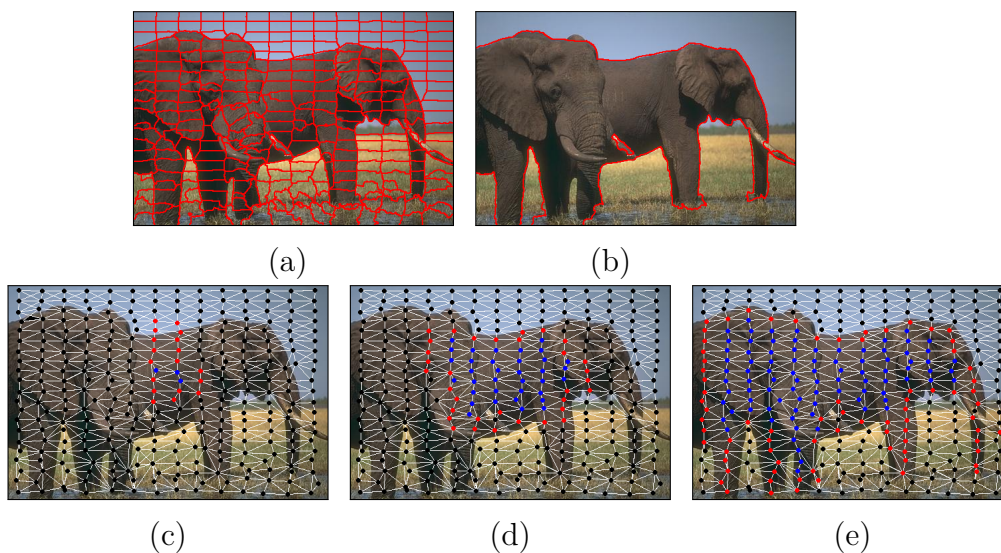


FIGURE 9.7 – Contours actifs sur un RAG. (a) La partition de l'image à partir de laquelle est construit le RAG. (b) Le résultat de l'évolution du contour sur le RAG, reporté sur l'image initiale. (c), (d), (e) La position du contour au cours du temps.

### Sur un RAG

La figure 9.7 illustre l'adaptabilité de notre approche et présente l'évolution d'un contour actif sur un RAG construit à partir d'une partition d'image. L'approche utilisée pour l'évolution de ce contour actif est en tous points similaire à celle utilisée dans le cas d'une image. En revanche, grâce au nombre réduit de sommets, la propagation du contour ne coûte que quelques dizaines

### 9.3. Expérimentations

d'itérations de l'algorithme contrairement à plusieurs milliers si nous avons utilisé l'image entière.

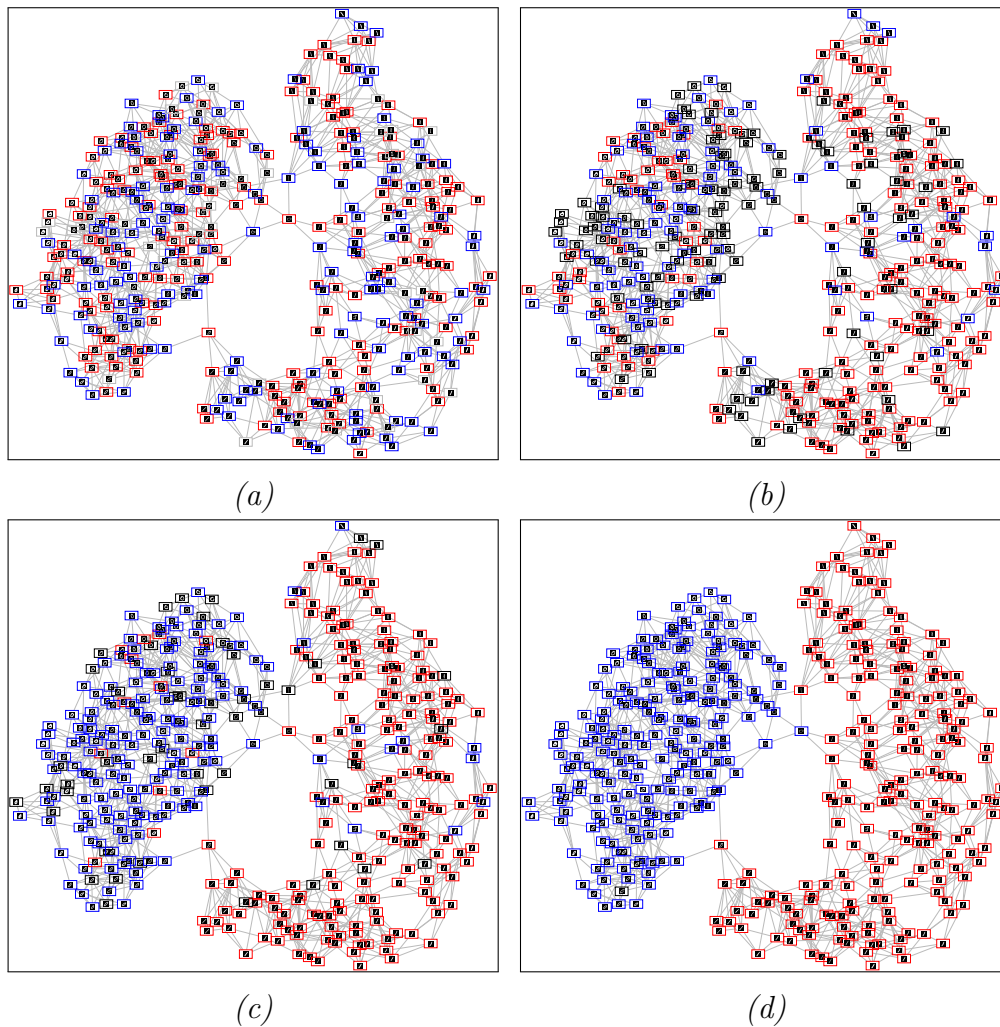


FIGURE 9.8 – Contours actifs multiples sur un graphe de données. (a) Le front à l’instant initial. (b) (c) Différentes étapes de l’évolution des deux fronts. (d) Le résultat de la classification.

#### Sur un graphe de données

Finalement, la figure 9.8 illustre à nouveau l’adaptabilité de notre approche et présente un exemple d’application le traitement des données. Le graphe est un graphe des  $k$ -ppv (avec  $k = 5$ ) construit à partir d’une base de 400 imagettes de zéros et de uns tirées la base USPS. Nous supposons pour cet exemple qu’il

existe une classification de ces imagerie, contenant de nombreuses erreurs et ignorant certaines imagerie. La figure 9.8.(a) présente le graphe avec les marqueurs de cette classification initiale. Afin de pallier ce problème, nous proposons de réattribuer le bon marqueur à chaque imagerie en associant un front à chaque classe et en les faisant évoluer selon notre approche des contours actifs sur graphe. La figure 9.8 présente le graphe avec la position initiale des deux fronts, établie à partir de la classification erronée puis différentes étapes de la propagation de ces fronts. Lorsque les fronts se stabilisent, nous pouvons remarquer la bonne classification de l'ensemble des imagerie.

## 9.4 Conclusion

Dans ce chapitre, nous avons présenté une formulation générale pour la propagation simultanée de plusieurs fronts sur un même graphe sans restriction sur le signe de leur vitesse.

Cette formulation est une extension des deux formulations présentées dans les deux chapitres précédent et repose sur le principe de décomposer chaque front en deux fronts, la partie de chaque front évoluant selon une vitesse positive étant représentée par son complémentaire de manière à ne considérer que des fronts évoluant selon une vitesse positive.

Finalement, nous avons illustré l'application de notre formulation générale au traitement des images et des données.

## Quatrième partie

### Application à l'aide au diagnostic en cytopathologie



# Contexte

Durant les deux dernières décennies, les pathologistes et les biologistes ont été témoins d'avancées majeures dans les technologies informatiques, tant au niveau matériel que logiciel. Le premier apport majeur de ces technologies a concerné la gestion des informations du patient et a conduit à de nombreux systèmes de connaissances médicales. De nos jours, ces systèmes de gestion des connaissances traditionnels se concentrent principalement sur le savoir médical mais abordent également l'aide au diagnostic ou au pronostic, l'analyse d'images, ainsi que l'enseignement à distance. Parmi tous ces champs, les systèmes de traitement d'images sont devenus très prometteurs. La raison principale est que les avancées des technologies de traitement d'images ont conduit à une importante augmentation de l'usage et de la manipulation d'images digitales (et ce dans tous les domaines).

Plus précisément, les systèmes de traitement et d'analyse d'image se sont développés en pathologie quantitative. Traditionnellement, les pathologistes établissent leur diagnostic à partir de l'observation d'un spécimen au microscope. Au cours de cet examen, ils mesurent des caractéristiques visuelles importantes pour leur décision telles que la taille, la forme, la teinte ou encore la texture. Ce procédé est complexe et chronophage or, les images digitales peuvent être analysées pour en extraire de l'information, traitées pour être améliorée, archivée pour participer à l'établissement du savoir médical, ou encore transmises sur des réseaux numériques pour faciliter la communication et l'échange entre les pathologistes. Ces composants peuvent être chacun intégrés au cœur d'un système d'information pour établir la base d'un outil numérique d'aide au diagnostic. Les techniques du traitement d'image sont d'un intérêt réel pour les pathologistes car elles peuvent se révéler plus fiables que les méthodes d'évaluation classiques (basées sur des critères visuels définis grossièrement et difficilement reproductibles). De telles techniques sont sur le point d'améliorer significativement la productivité et la capacité de diagnostic des pathologistes. En particulier, la détection de certains symptômes cancéreux rares est un problème majeur non encore résolu et pour lequel le traitement d'images peut être très utile.

---

Cependant, l'analyse quantitative des images médicales nécessite l'extraction, en amont, d'une grande quantité de données. Cette extraction implique de pouvoir effectuer une segmentation automatique ou interactive des objets d'intérêts depuis les images médicales. Si l'on considère le travail quotidien d'un pathologiste, impliquant une grande masse de données à traiter et à analyser, une automatisation de l'étape de segmentation se révèle alors absolument nécessaire, offrant de plus une reproductibilité des résultats et libérant les experts d'une lourde charge de travail.

En microscopie cellulaire, l'objectif de la segmentation est l'extraction d'objets cellulaires à partir de prélèvements étalés sur de minces lames microscopiques qui sont numérisées. Ce problème est un problème difficile en raison de la grande variation de ces objets et de leurs caractéristiques, due en particulier aux méthodes de coloration et d'étalement des prélèvements sur les lames. La littérature présente, à ce sujet, plusieurs stratégies de segmentations. Nous pouvons citer par exemple les méthodes basées sur l'analyse d'histogrammes, la classification de pixels, les opérateurs morphologiques ou encore les équations aux dérivées partielles. Leurs performances dépendent fortement d'à priori sur les objets cellulaires et leur coloration.

L'analyse des objets extraits et leur classification est certainement l'étape la plus importante puisque ce sont les résultats de la classification qui vont influencer le pathologiste lors de l'établissement du diagnostic. La classification de données est un domaine de recherche à part entière et la littérature scientifique est riche de méthodes de classifications. Nous pouvons citer, par exemple, les méthodes hiérarchiques, les k-moyennes, les réseaux de neurones, ou encore les méthodes de coupe basées sur la théorie des graphes.

Dans cette partie, nous proposons une application concrète des EdP présentées dans les deux précédentes parties de ce manuscrit pour l'aide au diagnostic en cytopathologie. Nous présentons tout d'abord un schéma général du processus d'aide au diagnostic appliqué à la microscopie quantitative. Ce schéma propose une approche unifiée de l'extraction et de la classification des noyaux cellulaires reposant sur l'utilisation des graphes et du formalisme des EdP. Enfin, nous présentons le logiciel Antarctic développé au cours de cette thèse et destiné à l'aide au diagnostic en cytopathologie. Ce logiciel doté d'une architecture client-serveur se présente sous la forme d'une application web et regroupe deux outils dont les utilisations sont la plupart du temps interdépendantes. D'une part, un outil automatique dédié à l'aide au diagnostic et offrant à un pathologiste la possibilité d'obtenir à distance une classification des cellules d'une lame cytologique téléchargée sur le serveur. De l'autre, un outil collaboratif de gestion et de consultation d'une base de connaissances

qui s'étoffe au fur et à mesure de l'ajout de nouvelles lames ainsi que des contributions (sous la forme d'expertises de cellules) des pathologistes.





# Chapitre 10

## Schéma général

### Sommaire

---

<b>10.1 Extraction des noyaux</b> . . . . .	<b>195</b>
10.1.1 Prétraitement . . . . .	195
10.1.2 Segmentation des noyaux . . . . .	197
10.1.3 Post-traitement . . . . .	198
<b>10.2 Extraction des caractéristiques</b> . . . . .	<b>200</b>
10.2.1 Caractéristiques de forme . . . . .	200
10.2.2 Caractéristiques de texture . . . . .	201
<b>10.3 Classification des noyaux</b> . . . . .	<b>202</b>

---

Dans ce chapitre, nous proposons un schéma général d'aide au diagnostic appliqué à la microscopie quantitative pour la cytologie. Ce schéma complet repose sur une approche unifiée de l'extraction et de la classification des noyaux cellulaires, impliquant le formalisme des EdP sur graphes. Nous montrons en effet que de par l'adaptabilité de ce formalisme à tous les types de graphes nous pouvons appliquer nos modèles à la fois aux images (pour la segmentation des noyaux) et aux données (pour la classification des noyaux).

L'ensemble des illustrations présentées dans ce chapitre proviennent de prélèvements cytologiques fournis dans le cadre d'une collaboration avec le service de cytopathologie du Centre Hospitalier Public du Cotentin, l'association Cœur et Cancer et le Conseil Régional de Basse-Normandie.

Ces prélèvements sont colorées par la coloration de Feulgen et sont présentés sous la forme d'images de grandes dimensions (environ  $40000^2$  pixels) qui sont découpées en images plus petites ( $1024^2$  pixels). Cette coloration est principalement utilisée en histologie mais présente néanmoins un grand intérêt en cytologie car elle permet une détection facilitée de l'ADN contenu dans

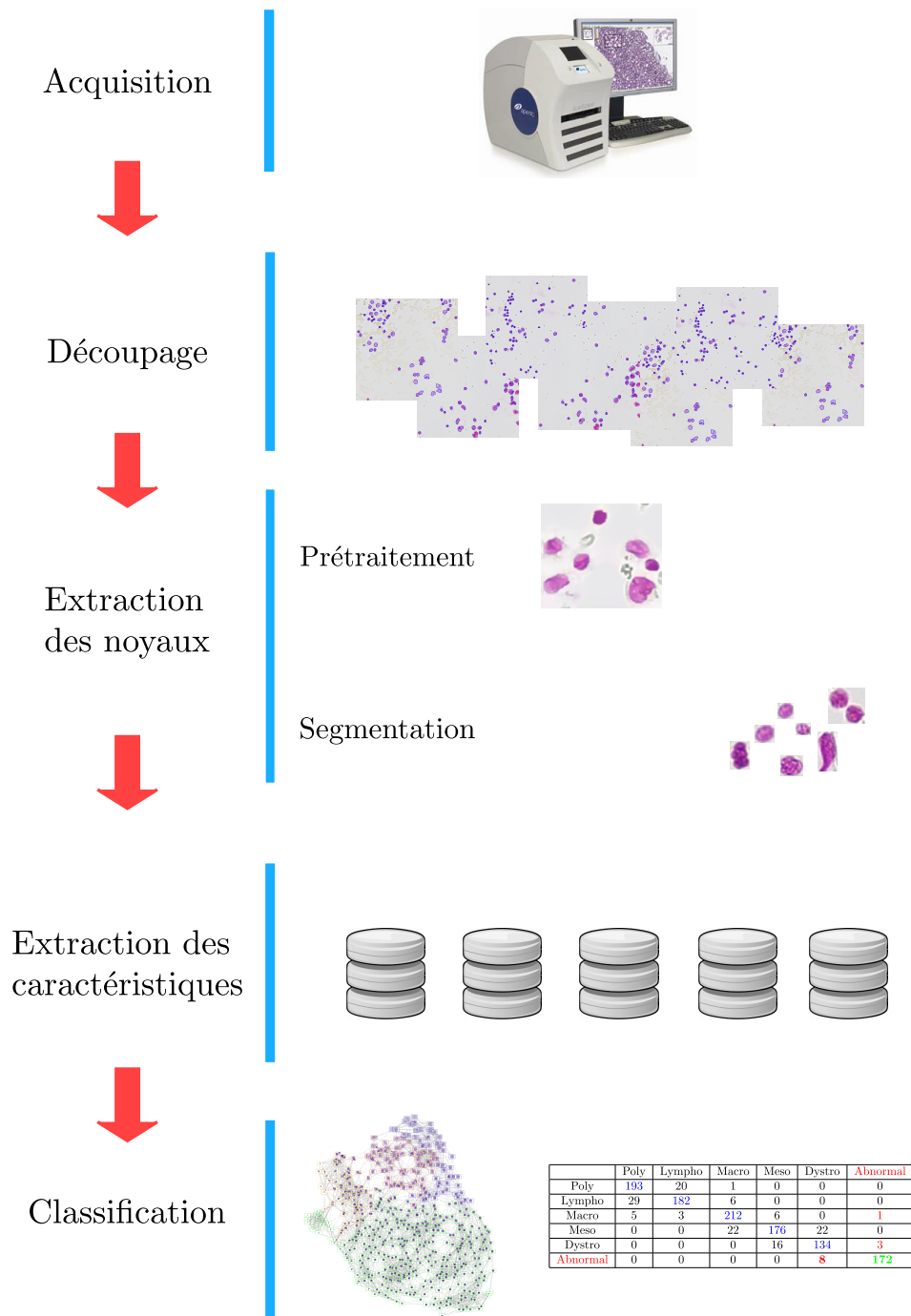


FIGURE 10.1 – Schéma général de l'aide au diagnostic en microscopie quantitative.

les cellules. Avec cette coloration, seuls les brins d'ADN (et donc les noyaux) sont colorés (en rose) avec des zones plus foncées représentant les mottes de chromatine. L'image peut donc être vue comme facilement séparable en deux classes bien distinctes. Cependant, sur l'image numérisée les contours des objets cellulaires sont très souvent irréguliers et flous en raison de phénomènes de réfringence et de la présence d'artefacts biologiques (tels que du mucus) sur les lames.

Ces illustrations ont toutes été réalisées en utilisant les différents opérateurs et algorithmes présentés dans les précédentes parties de ce manuscrit.

## 10.1 Extraction des noyaux

Le processus d'extraction des noyaux peut être décomposé en trois étapes significatives que nous détaillons dans cette section. La première étape, que nous désignons sous le terme étape de prétraitement, regroupe l'ensemble des traitements appliqués en amont à l'image microscopique afin de faciliter la segmentation des noyaux. La seconde étape, qui est à proprement parler l'étape d'extraction, consiste à séparer les parties de l'image correspondant aux noyaux du reste de l'image (que nous désignerons par la suite par le terme fond de l'image). Enfin, la troisième et dernière étape, que nous désignons sous le terme d'étape de post-traitement, regroupe l'ensemble des traitements appliqués aux objets segmentés afin d'améliorer leur qualité.

### 10.1.1 Prétraitement

De manière générale, l'objectif de l'étape de prétraitement est de renforcer le contraste entre les éléments du fond et les objets d'intérêt (les noyaux). Les performances de la plupart des méthodes de segmentation, qu'elles soit basées sur une classification des pixels ou des méthodes de diffusion de marqueurs, sont en effet fortement liées au degré de séparabilité des différents objets de l'image.

Ce prétraitement peut être effectué via une simple transformation couleur. En effet, la représentation d'une image par ses trois composantes rouge, verte et bleue n'est pas nécessairement la plus adaptée pour mettre en valeur les différences entre le fond et l'objet. Il est souvent possible de tirer avantage d'une autre représentation de la couleur (TSL, YUV, CMJN, etc.) ou même d'une combinaison entre les différents canaux de l'image RVB.

Ce prétraitement peut également être effectué par des méthodes de lissage adaptatif, c'est à dire qui conservent les zones de fort gradient mais lissent fortement les zones où le gradient est plus faible. Parmi ces méthodes, nous pouvons citer celles basées sur des EDP, telles que la régularisation par  $p$ -laplacien ou les approches morphologiques. Considérant le cadre des EdP sur graphe, les opérateurs laplacien normalisé, NLE ou NLD peuvent aisément être utilisés pour effectuer ce lissage.

### Illustration

La figure 10.2 présente un exemple de prétraitement basé sur une simple transformation couleur. L'image transformée (au milieu) est une image composée d'un unique canal et obtenue par une simple combinaison linéaire des trois canaux rouge, vert et bleu. Cette combinaison (le rapport entre des différences entre le bleu et le vert d'une part et le rouge et le vert d'autre part) permet de renforcer l'opposition entre le rose vif des noyaux cellulaires et le rose-gris au contour de ces noyaux. On remarquera le fort contraste entre les objets cellulaires et le fond.

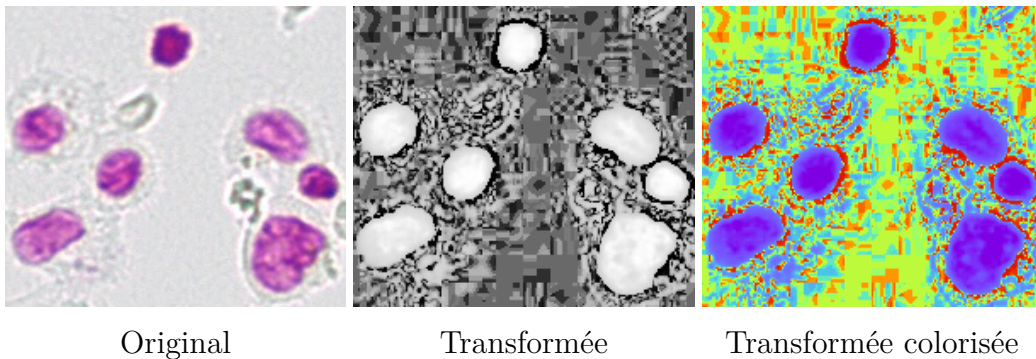


FIGURE 10.2 – Transformée couleur. Les histogrammes des images résultats ont été égalisés pour une meilleure visualisation.

La figure 10.3 présente un exemple de lissage préservant les contours obtenu grâce à quelques itération de l'équation de diffusion du  $p$ -Laplacien normalisé (4.27). L'image originale utilisée dans cet exemple est l'image d'intensité obtenue par transformation couleur et présentée à la figure 10.3. On remarquera la préservation des forts gradients malgré le fort lissage des régions plus homogènes.

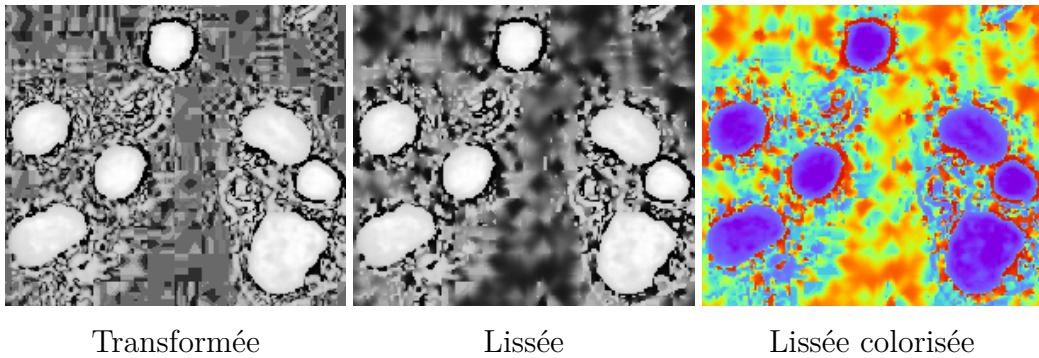


FIGURE 10.3 – Lissage par diffusion du  $p$ -Laplacien normalisé. Les histogrammes des images ont été égalisés pour une meilleure visualisation.

### 10.1.2 Segmentation des noyaux

En microscopie cellulaire, l’objectif de la segmentation est l’extraction d’objets cellulaires depuis les lames cytologiques numérisées. Ce problème est un problème difficile en raison de la grande variation de ces objets et de leurs caractéristiques, due en particulier aux méthodes de coloration et d’étalement des prélèvements sur les lames. La littérature présente, tous domaines d’application confondus, de nombreuses stratégies de segmentations [Meijering 12]. Nous pouvons citer par exemple les méthodes basées sur l’analyse d’histogrammes, la classification de pixels, les opérateurs morphologiques ou encore les équations aux dérivées partielles. Leurs performances dépendent fortement de l’a priori sur les caractéristiques des objets à extraire.

Dans le contexte de la microscopie, une approche classique pour la segmentation est d’utiliser une méthode de croissance de régions depuis un ensemble de graines (telle que la méthode de la ligne de partage des eaux par exemple). Le choix de ces graines pouvant être automatique ou interactif. Récemment, [Ta et al. 09b] ont proposé une approche de segmentation par partition d’une énergie reposant sur le formalisme des EdP sur graphe. De la même manière, notre approche pour la propagation de fronts sur graphes peut-être utilisée pour ce problème de segmentation.

#### Illustration

La figure 10.4 illustre une méthode de segmentation automatique des noyaux, reposant sur notre algorithme de propagation de fronts et via une approche basée sur des contours actifs (voir la section 9.3.2). La position initiale du front est obtenue par un simple seuillage de l’image d’intensité (par a priori). Le front est ensuite propagé selon une vitesse définie en fonction de

l'intensité moyenne à l'intérieur et à l'extérieur du front (la vitesse est positive sur les sommets de faible intensité, nulle sur les sommets d'intensité moyenne et négative sur les sommets d'intensité forte). Cette approche, qui peut être assimilée au modèle de Chan et Vese privé de la courbure, permet à la fois une grande précision et une faible sensibilité aux erreurs d'initialisation.

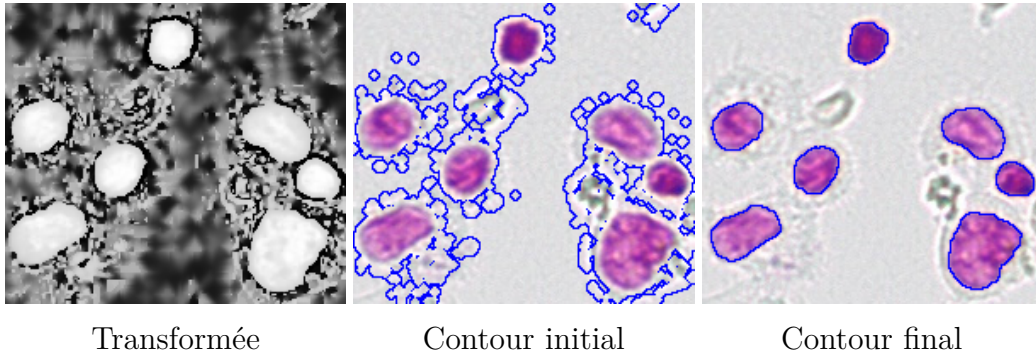


FIGURE 10.4 – Extraction automatique des noyaux.

### 10.1.3 Post-traitement

L'objectif de l'étape de post-traitement est de raffiner le résultat de la segmentation.

#### Illustration

La figure 10.5 illustre une méthode de raffinement des contours des objets segmentés par la courbure. Chaque contour est assimilé à un front distinct, qui est propagé sur quelques itérations en fonction de la courbure. La propagation est réalisée suivant l'approche présentée au chapitre 9 et la vitesse de propagation du front est définie telle que  $\mathcal{F} = (\mathcal{K}_w \psi_0)$ , où  $\psi_0$  est la fonction indicatrice de la position du front. Une telle définition de la vitesse permet de lisser le front (par de petites dilatations et rétractions en fonction du signe de la vitesse), tout en garantissant que le front ne s'écarte pas du contour des noyaux (grâce à la fonction  $w$ ).

Finalement, la figure 10.6 présente quelques résultats de segmentation obtenus avec notre approche basée sur le formalisme des EdPS sur graphe.



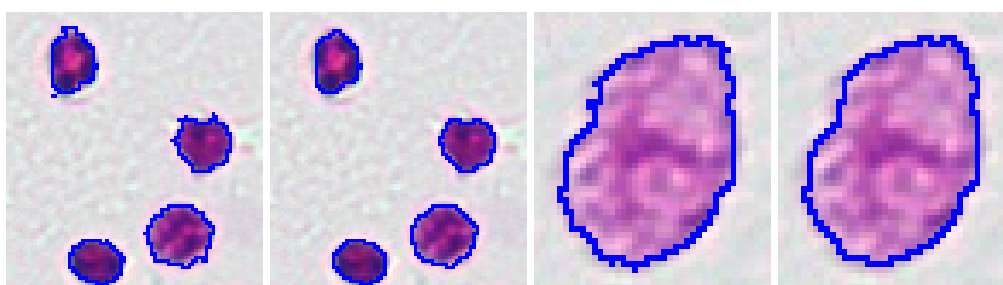


FIGURE 10.5 – Lissage des contours par la courbure.

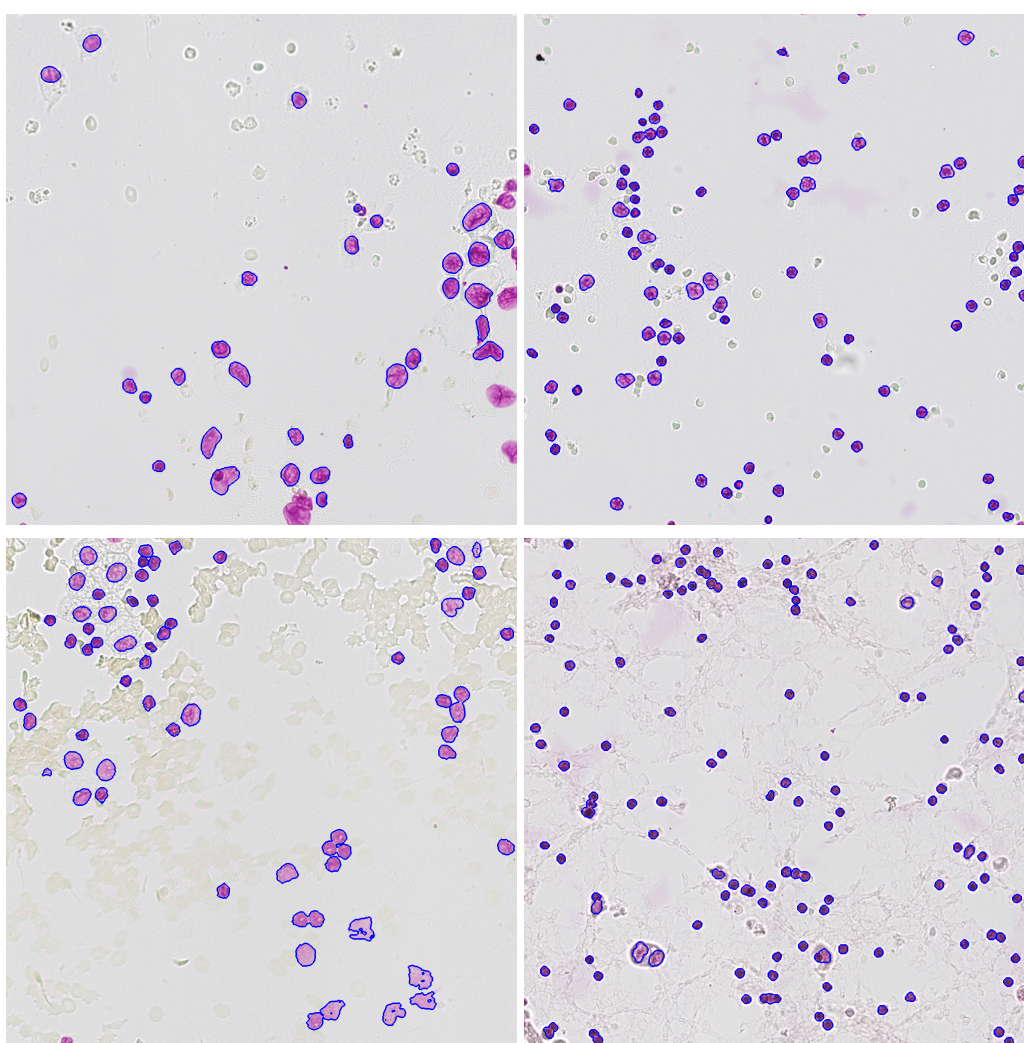


FIGURE 10.6 – Résultats de segmentation.



## 10.2 Extraction des caractéristiques

Dans cette section, nous présentons l'étape d'extraction des caractéristiques. L'objectif de cette étape est d'extraire un ensemble d'attributs caractéristiques depuis les noyaux extraits et permettant de discriminer ces noyaux en vue de l'étape de classification.

Ces caractéristiques correspondent toutes à des critères visuels simples, les mêmes qu'utilisés par les pathologistes lors de l'expertise au microscope d'une lame cytologique. L'utilisation d'outils de traitement d'image permet de réaliser des mesures précises et reproductibles de ces caractéristiques (ce qui ne peut être le cas lors d'une expertise visuelle au microscope).

Avec une telle approche, il devient également aisé pour la communauté médicale d'alimenter massivement des bases de connaissances. Ces bases pouvant être exploitées pour la recherche, notamment par le biais d'études statistiques afin de mettre en évidence de nouveaux symptômes et particularités à l'échelle non plus de chaque cellule, mais du prélèvement entier.

Ces caractéristiques peuvent être réparties en trois catégories : les caractéristiques de forme, de couleur et de texture. Nous présentons dans les paragraphes suivants quelques exemples de caractéristiques de forme et de texture.

### 10.2.1 Caractéristiques de forme

La taille et la forme des noyaux correspondent au premier critère visuel utilisé par les pathologistes pour repérer les cellules tumorales. En effet, les cellules tumorales sont, en raison de leur affection, généralement plus grosses que les cellules saines. Parmi ces caractéristiques, nous pouvons citer

- La surface du noyau. La taille du noyau (quantifiée par la surface) est un des trois critères principaux et indispensable permettant au pathologiste de déterminer si une cellule est ou non tumorale. En effet, la très grande majorité des noyaux cellulaires présents sur une lame cytologique ont une taille très inférieure à celle des noyaux des cellules tumorales. Plus généralement, la surface permet de discriminer les différents types de noyaux, au delà du simple *distinguo* entre les cellules saines et tumorales. Chaque type de noyaux (polynucléaire, macrophage, mésothéliale, etc.) étant présent dans un intervalle restreint de surfaces.
- La convexité, la circularité et l'élongation. Ces trois critères, permettent de caractériser la forme des noyaux, et plus particulièrement la présence d'excroissances. Ce critère permet de distinguer les cellules normales,

généralement compactes et sans excroissances, des cellules tumorales présentant des irrégularités de contour.

Ces trois critères permettent également de repérer les cellules macrophages, dont la forme typique d'un haricot est facilement détectable par une mesure de convexité ou encore les lymphocytes qui sont presque toujours assimilables à des disques.

De nombreuses approches simples ont été développées pour extraire ces caractéristiques. Certaines, comme la convexité ou l'élongation peuvent aisément être calculées via une carte de distance depuis le centre de masse des noyaux.

## 10.2.2 Caractéristiques de texture

### Caractéristiques liées à la chromatine

La chromatine est la forme sous laquelle se présente l'ADN dans le noyau. Sa quantité et sa répartition comptent parmi les principaux indicateurs (avec la surface) utilisés par les pathologistes pour repérer les cellules tumorales. Une grande quantité de chromatine, principalement localisée sur les bords de la cellule (on parle alors de margination de la chromatine) alerte toujours le pathologiste sur la nature cancéreuse de la cellule.

- La quantité totale de chromatine. Cette caractéristique peut être mesurée par un simple seuillage sur certaines composantes couleur du noyau, ou encore par le calcul d'une version adaptée de la densité optique intégrée (DOI). La DOI est une mesure de la quantité de lumière absorbée par un objet. Elle est définie comme l'intégrale sur l'objet du rapport logarithmique entre l'intensité lumineuse d'un rayon avant et après avoir traversé ce milieu. Sous certaines colorations (notamment la coloration Feulgen), la quantité de colorant retenue par une zone du noyau est proportionnelle à la quantité de chromatine présente dans cette zone. La DOI est donc un bon estimateur de la quantité de chromatine présente dans le noyau.
- Le nombre de mottes. Lorsque la cellule est tumorale, la chromatine est visible est répartie en mottes. Cette caractéristique peut être mesurée en créant une partition fine du noyau (par la méthode de la ligne de partage des eaux, ou notre approche des super vertices) . Le nombre de régions contenant une grande quantité de chromatine peut ensuite aisément être quantifié, par exemple par une analyse de la couleur de chaque région. Cette caractéristique est illustrée à la figure 10.8.
- La margination de la chromatine. Une autre particularité visuelle des cellules tumorales tient en ce que la chromatine se concentre principalement

à la périphérie de leurs noyaux. Cette caractéristique peut être mesurée en découpant le noyau en de fines bandes concentriques (à partir d'une carte de distance), et estimant la quantité de chromatine à l'intérieur des bandes les plus marginales. Cette caractéristique est illustrée pour un noyau à la figure 10.7

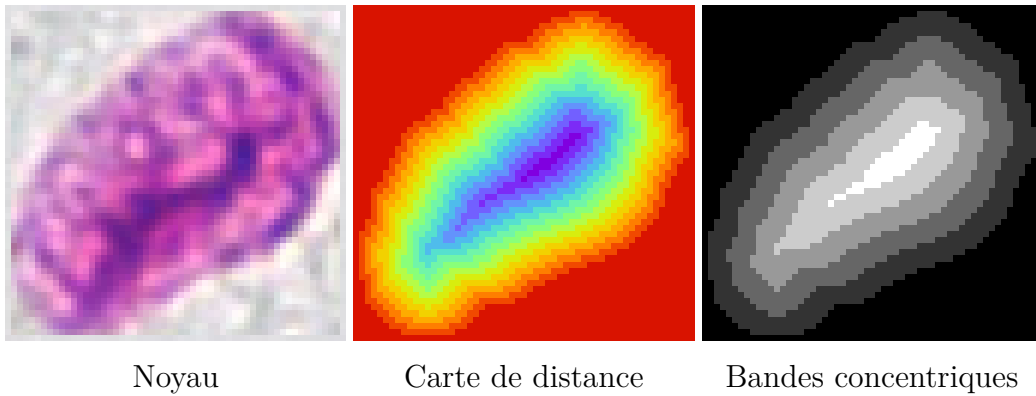


FIGURE 10.7 – Calcul de la margination de la chromatine. Cette caractéristique est calculée en mesurant la quantité de chromatine dans la bande la plus marginale (située à la plus grande distance du centre du noyau)

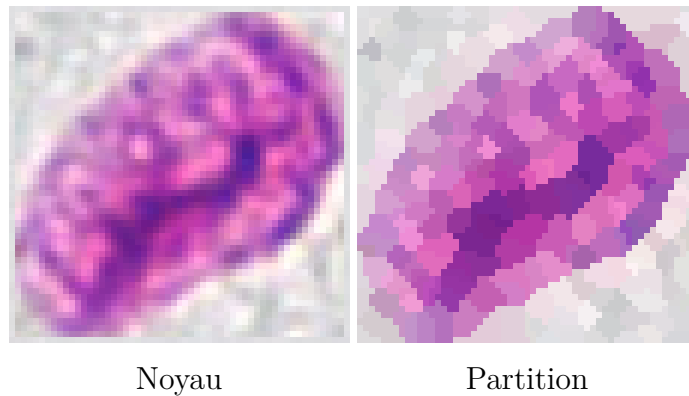


FIGURE 10.8 – Calcul du nombre de mottes de chromatine à partir d'une partition en régions de l'image du noyau.

### 10.3 Classification des noyaux

Enfin, dans cette section, nous présentons la dernière étape de ce schéma général à savoir l'étape de classification des noyaux. Cette étape est certainement la plus importante puisque ce sont les résultats de la classification qui

vont influencer le pathologiste lors de l'établissement du diagnostic. Cette étape est fortement liée aux deux précédentes puisque le résultat de la classification dépend principalement de la qualité de la segmentation des noyaux ainsi que de l'extraction de leurs caractéristiques.

La littérature scientifique est riche de méthodes de classifications. Nous pouvons citer, par exemple, les méthodes hiérarchiques, les  $k$ -moyennes, les réseaux de neurones, ou encore les méthodes de coupe basées sur la théorie des graphes.

#### **Illustration**

La figure 10.9 présente le résultat d'une classification effectuée sur un échantillon de noyaux (un cas réel contenant entre  $6 \times 10^4$  et  $3 \times 10^5$  noyaux). La classification a été réalisée par notre méthode de propagation de front multiples sur graphe. Chaque front correspond à une classe et est initialisé à partir d'une base de référence créée et validée par des experts. Le graphe est un graphe des  $k$ -ppv (avec  $k = 5$ ) construit à partir de l'ensemble fusionné des noyaux à classer et des noyaux de la base de référence. La fonction caractéristique utilisée pour construire la mesure de similarité associée à chaque noyau le vecteur de ses caractéristiques. Cette approche présente l'avantage d'être très rapide (quelques secondes pour classer  $6 \times 10^4$  cellules) et produit des résultats très prometteurs.

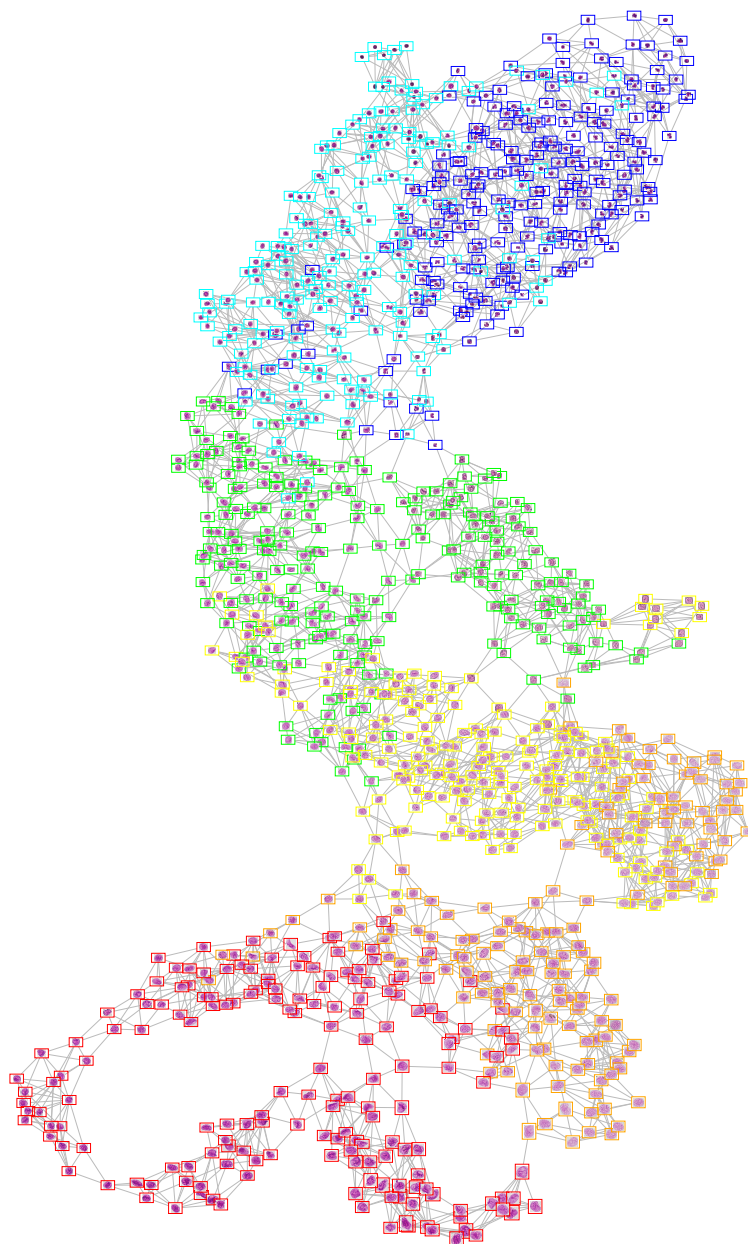


FIGURE 10.9 – Classification de noyaux en 6 classes par notre approche de propagation de fronts sur graphe.

# Chapitre 11

## Antarctic

### Sommaire

---

<b>11.1 Un outil automatique pour l'aide au diagnostic . .</b>	<b>206</b>
<b>11.2 Une base de connaissances interactive et collaborative . . . . .</b>	<b>209</b>
<b>11.3 Conclusion . . . . .</b>	<b>212</b>

---

Dans ce chapitre, nous présentons le logiciel Antarctic qui a été développé au cours de cette thèse. À l'origine conçu comme un support au travail de recherche et ne permettant que la visualisation de différents résultats de segmentation et de classification, il a été étendu au fur à mesure de l'avancement pour faciliter les interactions (comme l'expertise ou la validation) avec les pathologistes du Centre Hospitalier Public du Cotentin situés sur un site distant. Puis, sous l'impulsion des pathologistes, le logiciel a été entièrement repensé pour être à la fois pleinement fonctionnel pour les pathologistes (avec le traitement automatique des lames cytologiques, et la gestion quantitative des connaissances), tout en conservant son rôle de support à la recherche.

Antarctic est conçu selon une architecture client-serveur. Côté serveur, l'application concrétise le schéma général présenté dans le chapitre précédent et intègre les outils nécessaires à chacune des étapes de la chaîne de traitement des images et des données. L'application intègre également une base de données regroupant des millions de cellules, leurs caractéristiques, ainsi que des bases de références créées, modifiables et consultables par les pathologistes. Côté client, l'application se présente sous la forme d'une interface web utilisable depuis n'importe quel navigateur. Depuis cette interface, les pathologistes peuvent ajouter de nouveaux cas, consulter les résultats de la classification, examiner les caractéristiques extraites, et consulter ou contribuer à la base de connaissances.

Le logiciel regroupe en fait deux outils dont les utilisations sont la plupart du temps interdépendantes. D'une part, un outil automatique dédié à l'aide au diagnostic, et offrant à un pathologiste la possibilité de soumettre un nouveau cas et d'obtenir en retour le résultat de la classification des cellules. De l'autre, un outil collaboratif de gestion et de consultation d'une base de connaissances, qui s'étoffe au fur et à mesure des nouveaux cas ajoutés ainsi que des contributions (sous la forme d'expertises de cellules) des pathologistes.

## 11.1 Un outil automatique pour l'aide au diagnostic

Dans cette section, nous présentons l'aspect aide au diagnostic du logiciel. Dans ce mode d'utilisation, le pathologiste commence par télécharger sur le serveur l'image d'une nouvelle lame cytologique qu'il souhaite analyser. Cette lame est alors découpée en tuiles, chacune de ces tuiles étant ensuite segmentée pour en extraire les différents noyaux qui sont à leur tour traités pour en extraire des caractéristiques. L'ensemble de ces informations : les noyaux et leurs caractéristiques, sont ensuite stockées dans la base de connaissances et pourront être consultés par d'autres utilisateurs du logiciel (les cas étant tous anonymes).

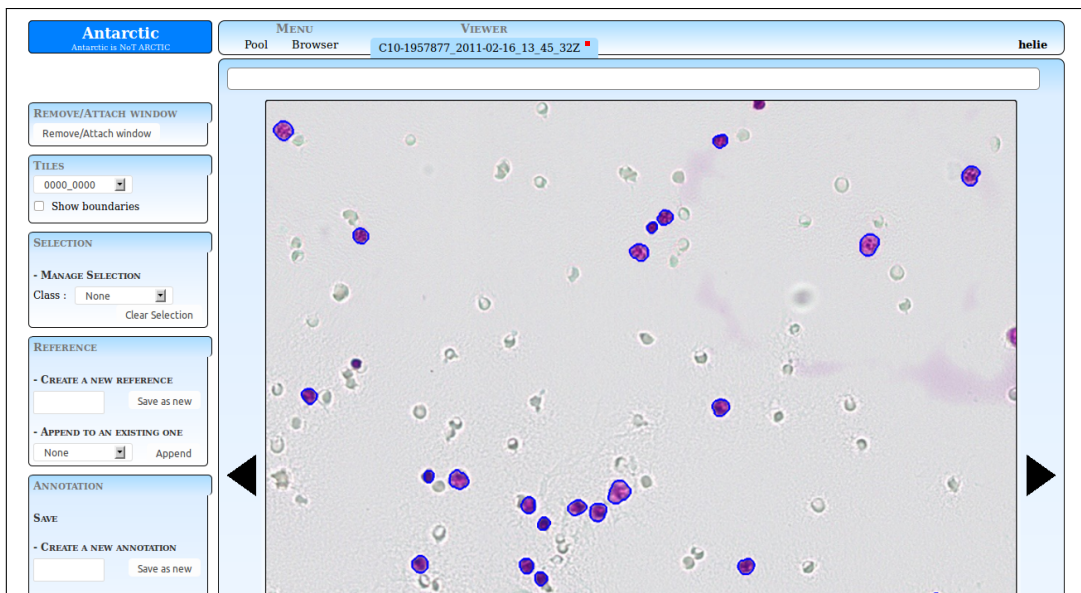


FIGURE 11.1 – Visualisation du résultat de la segmentation.

### 11.1. Un outil automatique pour l'aide au diagnostic

Le pathologiste peut alors contrôler la qualité de la segmentation via le visualiseur de lames (voir la figure 11.1). Si la segmentation est correcte, le pathologiste peut ensuite effectuer une classification automatique des cellules en choisissant simplement une base de références, c'est à dire une liste de cellules ne faisant pas partie du cas et dont la classe a été définie et validée par des pathologistes (voir la figure 11.2). Une fois la classification effectuée, le résultat est stocké dans la base de connaissances et consultable immédiatement par le pathologiste.

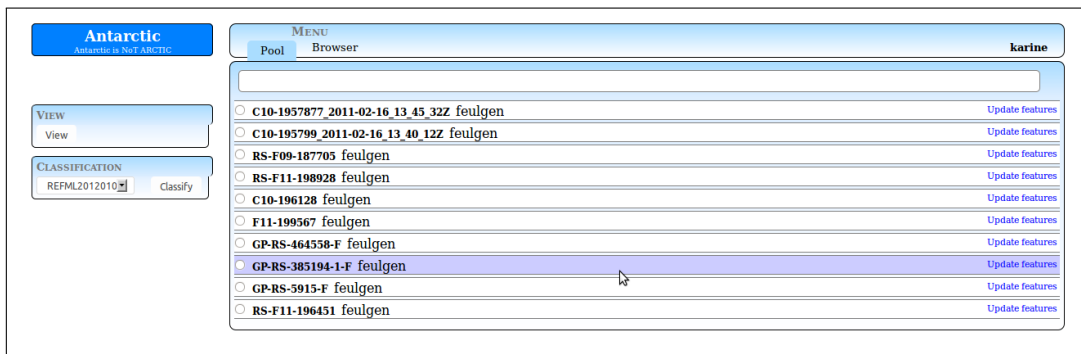


FIGURE 11.2 – Interface pour l'aide au diagnostic.

Cette consultation (voir la figure 11.3) est interactive, et le pathologiste peut consulter chaque cellule, ses caractéristiques, mais aussi le degré de confiance dans la classification ou encore le contexte clinique de la cellule (c'est à dire la cellule replacée sur la lame cytologique, voir la figure 11.4). Sur cette page, le pathologiste peut également visualiser le graphe construit pour la classification, et ainsi situer la position de la cellule parmi l'ensemble des cellules.

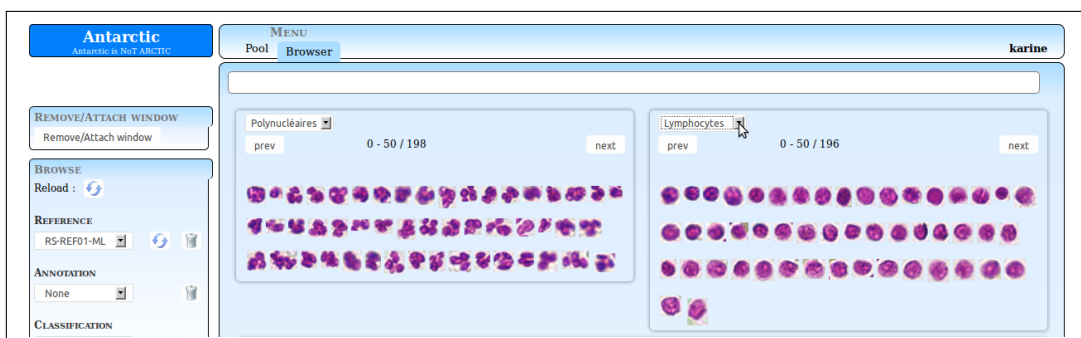


FIGURE 11.3 – Visualisation du résultat de la classification.

Il est possible d'effectuer le processus de classification plusieurs fois, avec différentes bases de références, et de comparer les résultats obtenus.



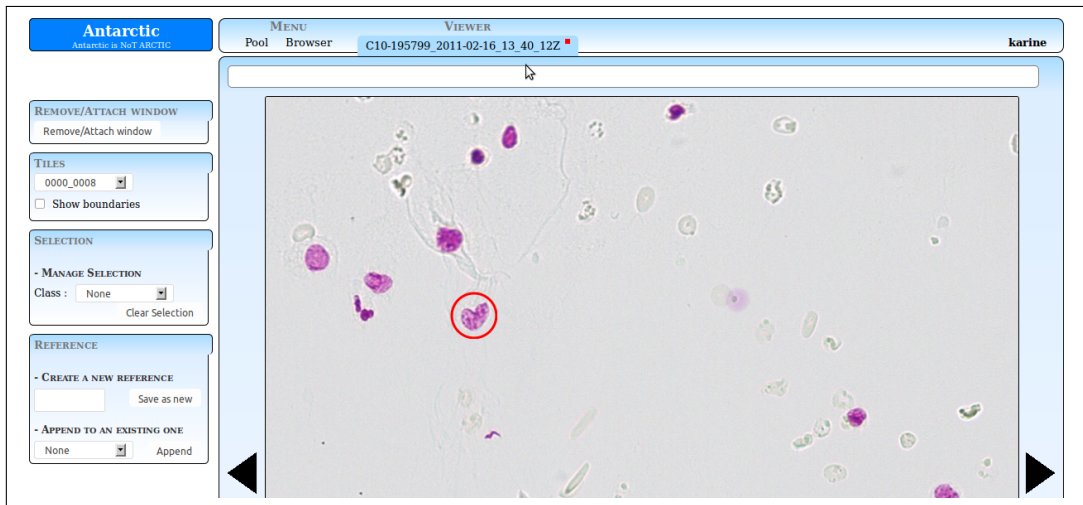


FIGURE 11.4 – Visualisation d'un noyau dans son contexte.

Le pathologiste peut également annoter les résultats de la classification et ainsi participer à la base de connaissances que nous présentons dans la section suivante.

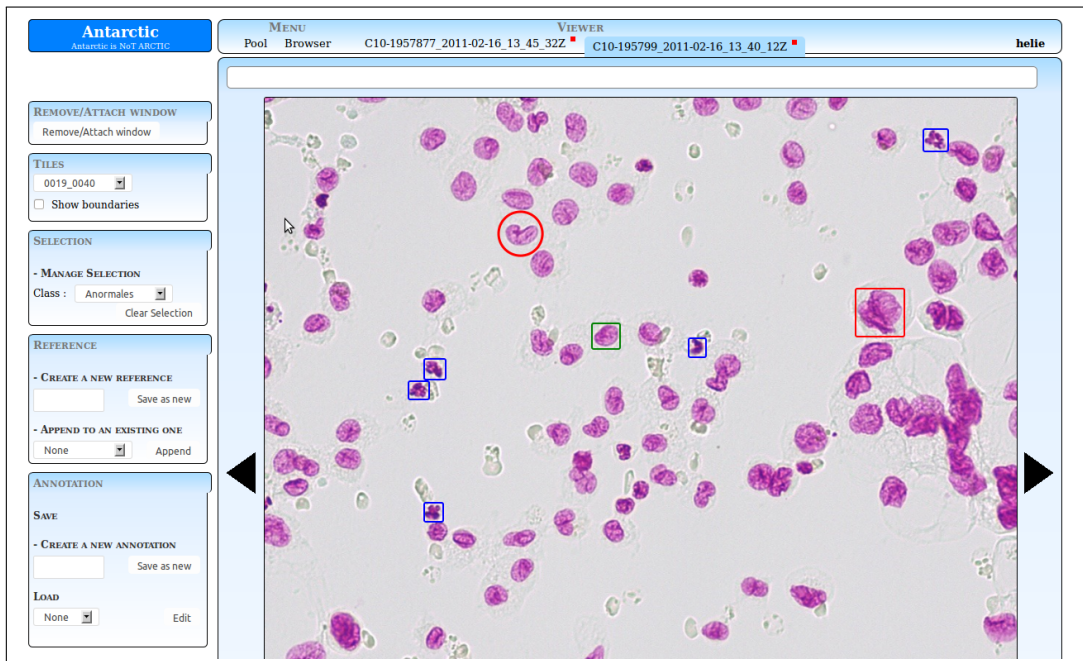


FIGURE 11.5 – Annotation collaborative de noyaux.

## 11.2 Une base de connaissances interactive et collaborative

Dans cette section, nous présentons la base de connaissances associée à l'outil d'aide au diagnostic. Cette base est interactive, c'est à dire qu'elle est maintenue, consultée et enrichie par les pathologistes eux même. En ajoutant de nouveaux cas (que ce soit pour l'aide au diagnostic ou non), les pathologistes alimentent la base de données en nouvelles cellules qu'ils peuvent à loisir consulter, annoter ou expertiser. L'interface permet aux pathologistes de naviguer entre différentes représentations des cellules, que ce soit du point de vue des caractéristiques extraites ou de leur contexte clinique.

Cette base est également une base collaborative, puisque chaque pathologiste peut contribuer à l'expertise de cellules qu'il a lui même ajouté mais aussi des cellules ajoutées par d'autres experts. De même, chaque pathologiste peut créer ses propres bases de références pour la classification, qui sont ensuite mises à la disposition des autres pathologistes : soit pour l'aide au diagnostic de leurs propres cas, soit pour être intégrées à d'autres bases de références. Cette collaboration est possible et grandement facilitée par l'architecture client-serveur d'Antarctic qui permet à des pathologistes de contribuer à distance et en quelques clics et sans avoir à installer de logiciel dédié ni à stocker de base de donnée.



FIGURE 11.6 – Navigateur.

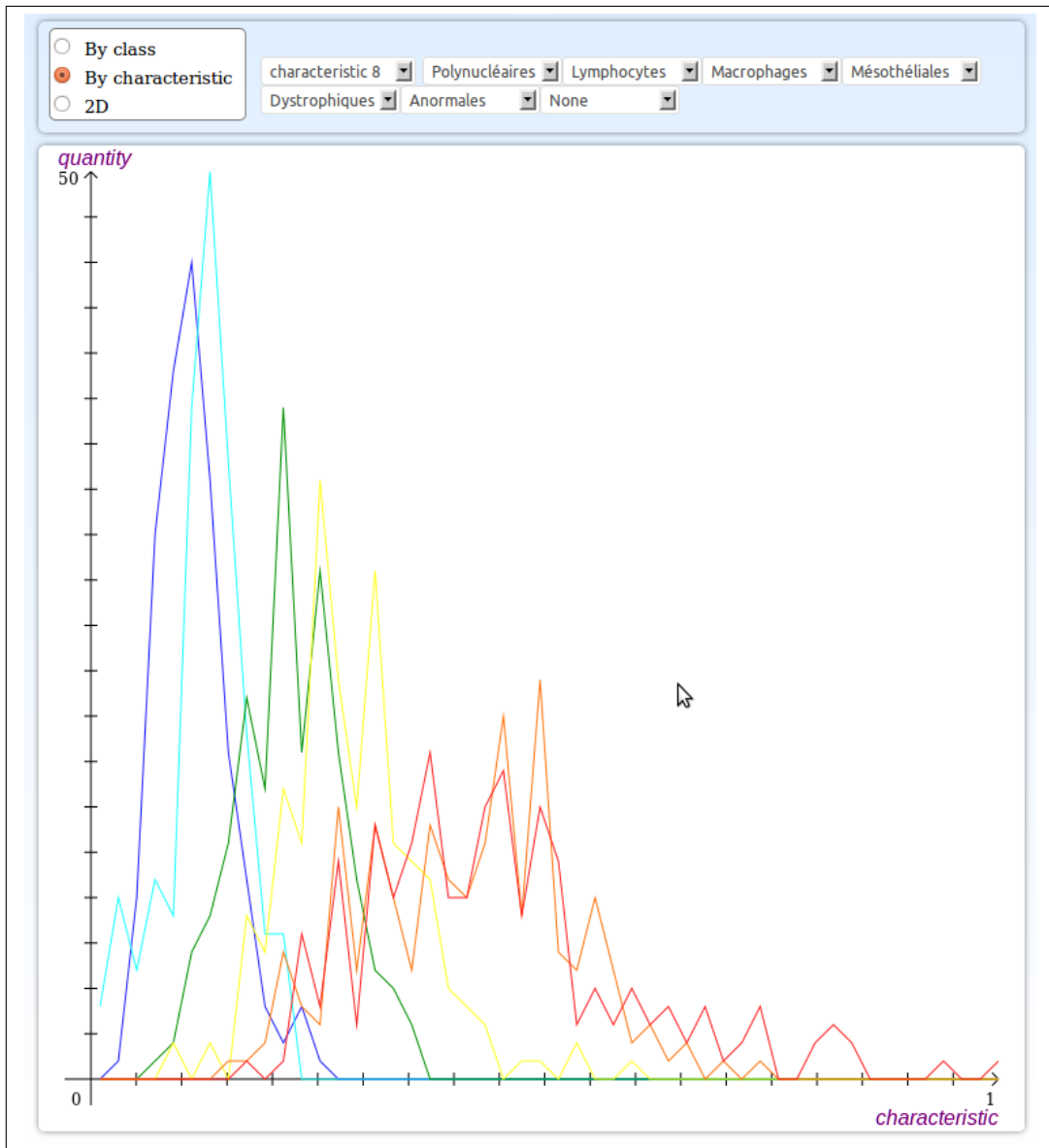


FIGURE 11.7 – Visualisation des caractéristiques.

La figure 11.8 illustre le visualiseur de lame cytologique permettant au pathologiste d'annoter les noyaux. Ces annotations peuvent être utilisées pour créer des bases de références (utilisée lors de l'étape de classification) ou consultées par l'ensemble des utilisateurs du logiciel.

Le logiciel intègre également un navigateur permettant de parcourir les différentes bases de données associées à la base de connaissance. Ce navigateur est présenté à la figure 11.6. Le pathologiste peut ainsi consulter et modifier les

### 11.3. Une base de connaissances interactive et collaborative

différentes bases de référence présentes dans l'application ou encore consulter des résultats de la classification automatique. Le navigateur intègre également différents outils liés à la visualisation des caractéristiques et de leur distribution. La figure 11.7 illustre la visualisation de la distribution du nombre de noyaux classe pour une caractéristique. La figure 11.8 illustre la visualisation de la distribution des classes de noyaux en fonction de deux caractéristiques.

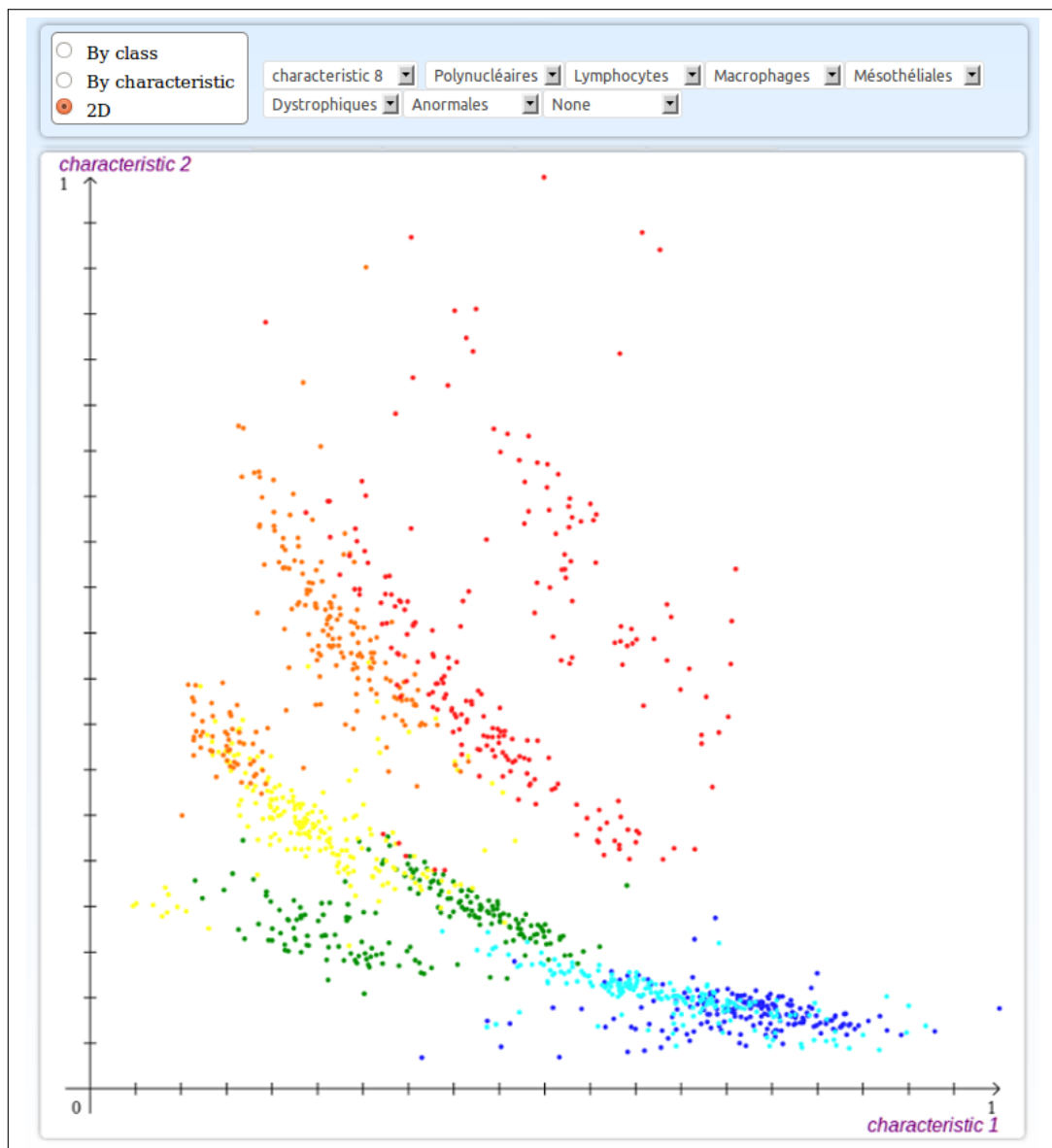


FIGURE 11.8 – Visualisation de la distribution des caractéristiques.

## **11.3 Conclusion**

Dans ce chapitre, nous avons présenté le logiciel Antarctic développé au cours de cette thèse pour l'aide au diagnostic en cytopathologie et la gestion d'une base de connaissances médicales.

Ce logiciel est à l'heure actuelle en phase de validation et rassemble plusieurs millions de noyaux cellulaires ainsi que leurs caractéristiques. Son développement est toujours actif pour l'étendre notamment à différents types de colorations et permettre son utilisation à grande échelle.

# Conclusion générale



---

Dans cette conclusion générale, nous présentons une synthèse des principales contributions de ce manuscrit ainsi que les perspectives de nos travaux.

Dans ce manuscrit, nous nous sommes attachés à transcrire certaines équations aux dérivées partielles définies dans le domaine continu vers des adaptations discrètes exploitant le formalisme des équations aux différences partielles définies sur des graphes. Nous nous sommes intéressés au  $p$ -laplacien normalisé ainsi qu'à l'équation des ensembles de niveaux pour la propagation d'un front et étendu leurs formulations et leurs champs d'applications au domaine des graphes. Le potentiel et l'intérêt de ces approches ont été illustrés dans des applications telles que le calcul de géodésiques, la segmentation d'images et de texture, la simplification de graphes, la classification de données, la restauration d'images et de textures ou encore l'inpainting.

### **Nouvelle famille d'opérateurs $p$ -laplaciens.**

Dans cette partie, nous avons présenté une nouvelle famille d'opérateurs  $p$ -laplaciens, s'appuyant sur le cadre des EdP sur graphe.

#### *L'opérateur laplacien infini non-local*

Dans un premier temps, motivés par la volonté d'étendre les applications du laplacien infini du domaine des images à celui des données non organisées, nous avons proposé une adaptation de cet opérateur au domaine des graphes de topologie arbitraire. Cette adaptation se base sur une transcription du laplacien infini continu par EdP permettant une nouvelle définition de l'opérateur laplacien infini non-local sur graphe. Nous avons montré que pour des graphes euclidiens particuliers cet opérateur coïncide avec la définition du laplacien infini de Hölder et coïncide également avec les fonctions de certains jeux stochastiques qui sont liés au laplacien infini continu. Nous avons également montré que l'opérateur laplacien infini non-local peut être interprété comme un opérateur morphologique combinant les opérateurs d'érosion et de dilatation non-locaux définis dans la première partie de ce manuscrit. Enfin, nous avons proposé une étude du problème de Dirichlet associé à l'équation du laplacien infini et montré que ce problème possède une solution unique. Finalement, nous avons illustré l'application de cet opérateur à des problèmes d'interpolation pour le traitement des images et des données représentées par des graphes.

#### *L'opérateur $p$ -laplacien normalisé*

Dans un second temps, nous avons proposé une nouvelle formulation, non-locale, du  $p$ -laplacien normalisé sur graphes. Cette formulation peut être



considérée comme l'analogue discret du  $p$ -laplacien des jeux défini en continu et s'exprime sous la forme d'une combinaison linéaire du laplacien et du laplacien infini non-local. Nous avons montré que pour certaines constructions de graphes, cet opérateur coïncide avec certaines EDP non-locales continues et coïncide également avec les fonctions de jeu d'une version bruitée du jeu du Tug-of-War. Nous avons également montré que l'opérateur  $p$ -laplacien normalisé peut être interprété comme un opérateur morphologique combinant des processus d'érosion et de dilatation non-locaux et un filtre moyenneur non-local. Cette interprétation est basée sur la définition des opérateurs morphologiques non-locaux introduits dans la première partie de ce manuscrit. Enfin, nous avons proposé une étude du problème de Dirichlet associé à notre opérateur et montré que ce problème possède une solution unique. De même, nous avons proposé une étude du problème de diffusion impliquant l'opérateur  $p$ -laplacien normalisé non-local et montré que le processus de diffusion associé peut être exprimé comme un processus de filtrage moyenneur non-local. Finalement, nous avons illustré l'application de cet opérateur à des problèmes d'interpolation et de diffusion pour le traitement des images, des maillages et des données représentées par des graphes.

### *Perspectives*

Parmi les perspectives, nous envisageons d'étudier le cas où  $1 \leq p \leq 2$  et de proposer une extension de notre adaptation du  $p$ -laplacien normalisé pour considérer ce cas. Nous projetons également d'étudier une nouvelle famille de  $p$ -laplaciens dérivée de la famille présentée dans ce manuscrit et incluant un terme de gradient. Nous pensons que cette famille correspond également à une large classe d'opérateurs  $p$ -laplaciens, et se conçoit comme l'extension naturelle de nos travaux sur le  $p$ -laplacien normalisé.

### **Équations de propagation de fronts sur graphe**

Dans cette partie, suivant notre objectif de transcription d'EDP définies sur des domaines continus au domaine des graphes pour le traitement des données, nous avons proposé une adaptation des équations des ensembles de niveaux pour la propagation d'un front au domaine des graphes pondérés. Au delà de la transcription des équations reposant sur le formalisme des EdP, nous avons proposé plusieurs extensions à la formulation initiale ne considérant qu'un front unique pour l'étendre au cas où plusieurs fronts évoluent simultanément sur un même graphe et sans restriction sur le signe de leurs vitesses.

### *Adaptation discrète de l'équation des ensembles de niveaux non-stationnaire*

---

Dans un premier temps, nous avons proposé une adaptation discrète de l'équation du mouvement définie par l'approche des ensembles de niveaux au domaine des graphes. Cette adaptation repose sur une interprétation morphologique de la version non-stationnaire de l'équation continue et se base sur le formalisme des EdP. Nous avons ainsi montré qu'en fonction du signe de la vitesse du front, son évolution peut être interprétée et réalisée comme une combinaison d'érosions et de dilatations discrètes. Nous avons également proposé un schéma numérique de résolution de cette équation et montré que ce schéma converge sous certaines conditions sur la vitesse du front. Nous avons proposé un algorithme itératif efficace implémentant ce schéma numérique puis proposé une version optimisée permettant de réduire fortement la complexité de l'algorithme initial. Finalement, nous avons illustré l'application de cette équation discrète au traitement des images et des données et montré le potentiel de notre approche pour étendre des méthodes définies sur des domaines continus pour le traitement des images, aux graphes pour le traitement des données.

#### *Équation stationnaire discrète et algorithme rapide*

Dans un second temps et en nous restreignant au cas où la vitesse est définie positive sur l'ensemble du graphe, nous avons présenté une version stationnaire de notre adaptation discrète de l'équation des ensembles de niveaux. Nous avons montré que pour un graphe euclidien et une norme  $\mathcal{L}_2$ , cette équation coïncide avec la discrétisation d'Osher-Sethian de l'équation eikonale définie en continu. Nous avons également montré que pour une norme  $\mathcal{L}_\infty$  notre formulation coïncide avec la fonction de distance utilisée dans les algorithmes de type Dijkstra. Nous avons présenté une preuve de l'existence et de l'unicité de la solution de cette équation pour les normes  $\mathcal{L}_1, \mathcal{L}_2$  et  $\mathcal{L}_\infty$ , ainsi qu'une formulation explicite de ces solutions. Nous avons également proposé un schéma de résolution de cette équation ainsi qu'un algorithme rapide implémentant ce schéma. Cet algorithme peut être considéré comme une généralisation de l'algorithme du Fast Marching aux graphes et se comporte comme un algorithme de type Dijkstra lorsque  $p = \infty$ . Finalement, nous avons illustré l'application de cette équation discrète pour le calcul de géodésiques sur différents types de graphes.

#### *Extension à la propagation simultanée de plusieurs fronts*

Poursuivant notre objectif de proposer une formulation et un algorithme général pour la propagation simultanée de plusieurs fronts sur un même graphe, nous avons proposé une première extension à notre formulation pour nous intéresser au cas où plusieurs fronts évoluent simultanément sur un

même graphe selon une vitesse positive et sous certaines contraintes sur les interactions entre ces fronts. Nous avons montré que l'évolution simultanée de ces fronts peut être réduite à la propagation d'un front global regroupant ces fronts, dont l'évolution est contrôlée par une version modifiée de l'équation stationnaire présentée dans le cas d'un front unique. Nous avons également proposé un schéma de résolution de l'équation modifiée ainsi qu'un algorithme rapide de résolution dont la complexité ne dépend pas du nombre de fronts. Finalement, nous avons illustré le comportement et l'intérêt de notre approche appliquée au traitement des images, des maillages et des données. Nous avons notamment montré l'intérêt de la propagation simultanée de multiples fronts pour la simplification de graphes via une généralisation des super pixels (que nous appelons super vertices). Nous avons également comparé notre approche avec une des méthodes les plus efficaces de segmentation d'images basée sur des graphes (le Power Watershed) et présenté de nombreux exemples de segmentation d'images, illustrant l'intérêt d'une méthode adaptative qui permet de manipuler différentes représentations d'une image comme les RAG ou les graphes non-locaux. Enfin, nous avons illustré l'intérêt de notre approche pour la classification semi-supervisée de données.

*Extension au cas le plus général sans restrictions sur le signe de la vitesse*

Finalement, nous avons présenté une formulation générale pour la propagation simultanée de plusieurs fronts sur un même graphe sans restriction sur le signe de leur vitesse. Cette formulation est une extension des deux formulations présentées dans les deux chapitres précédent et repose sur le principe de décomposer chaque front en deux fronts, la partie de chaque front évoluant selon une vitesse positive étant représentée par son complémentaire de manière à ne considérer que des fronts évoluant selon des vitesses positives. Enfin, nous avons illustré le comportement et l'intérêt de notre approche appliquée au traitement des images et des données, via une adaptation du modèle de contours actifs de Chan et Vese.

*Perspectives*

Ces travaux sur la propagation de fronts sur graphe ont montré le potentiel de notre approche et ouvrent la voie à de nombreuses applications effectives et réelles en traitement des images des maillages et des données. En ce sens, nous avons réalisé des études préliminaires pour la reconstruction de surfaces 3D par la méthode des ensembles de niveaux sur graphe. Cette approche repose sur un graphe de densité adaptative et permet de réduire les temps de calcul. Nous pouvons également envisager d'étudier l'intérêt de notre approche pour l'organisation hiérarchique de graphes de données. Il serait en effet intéressant

---

d'utiliser l'approche des super vertices pour des opérations de groupement et de partitionnement pour l'organisation de grandes bases de données.

### **Application à l'aide au diagnostic en cytopathologie**

Dans cette partie, nous avons proposé une application concrète des EdP présentées dans les deux précédentes parties de ce manuscrit pour l'aide au diagnostic en cytopathologie. Nous avons présenté un schéma général du processus d'aide au diagnostic appliqué à la microscopie quantitative. Ce schéma propose une approche unifiée de l'extraction et de la classification des noyaux cellulaires reposant sur l'utilisation des graphes et du formalisme des EdP. Nous avons également présenté le logiciel Antarctic développé au cours de cette thèse et destiné à l'aide au diagnostic en cytopathologie. Ce logiciel doté d'une architecture client-serveur se présente sous la forme d'une application web et regroupe deux outils dont les utilisations sont la plupart du temps interdépendantes. D'une part, un outil automatique dédié à l'aide au diagnostic et offrant à un pathologiste la possibilité d'obtenir à distance une classification des cellules d'une lame cytologique téléchargée sur le serveur. De l'autre, un outil collaboratif de gestion et de consultation d'une base de connaissances qui s'étoffe au fur et à mesure de l'ajout de nouvelles lames ainsi que des contributions (sous la forme d'expertises de cellules) des pathologistes.

#### *Perspectives*

Parmi les perspectives liées à cette application, nous pensons tout d'abord à la validation de notre approche et des résultats obtenus, pour le traitement de lames cytologiques en coloration de Feulgen. Pour cela, nous envisageons d'élargir le cercle des pathologistes ayant accès à notre application via le logiciel Antarctic. Ceci implique de poursuivre le développement du logiciel pour permettre une utilisation à grande échelle. Enfin, nous envisageons également d'étendre le domaine d'application de notre schéma à d'autres pathologies et méthodes de coloration des lames cytologiques.



# Liste des publications

- [Chakik et al. 12] A. EL CHAKIK, X. DESQUESNES et A. ELMOATAZ : Fast 3d surface reconstruction from point clouds using graph-based fronts propagation. *In Proceedings of ACCV 12*, 2012.
- [Desquesnes et al. 10] X. DESQUESNES, A. ELMOATAZ, O. LÉZORAY et V.T. TA : Efficient algorithms for image and high dimensional data processing using eikonal equation on graphs. *In International Symposium on Visual Computing*, volume LNCS 6454, pages 647–658, 2010.
- [Desquesnes et al. 11a] X. DESQUESNES, A. ELMOATAZ et O. LÉZORAY : Une nouvelle méthode de diffusion géométrique sur graphe pour la segmentation d’images et la classification de données de grandes dimensions. *In Colloque du Groupe de Recherche et d’Étude du Traitement du Signal et des Images*, 2011.
- [Desquesnes et al. 11b] X. DESQUESNES, A. ELMOATAZ et O. LÉZORAY : Pdes level sets on weighted graphs. *In International Conference on Image Processing (IEEE)*, pages 3377 – 3380, 2011.
- [Desquesnes et al. 12a] X. DESQUESNES, A. ELMOATAZ et O. LÉZORAY : Eikonal equation adaptation on weighted graphs : fast geometric diffusion process for local and non-local image and data processing. *Journal of Mathematical Imaging and Vision*, 2012.
- [Desquesnes et al. 12b] X. DESQUESNES, A. ELMOATAZ et O. LÉZORAY : Generalized fronts propagation on weighted graphs. *In Proceedings of Algorithm 12*, 2012.
- [Desquesnes et al. 12c] X. DESQUESNES, A. ELMOATAZ et O. LÉZORAY : Pdes-based morphology on graphs for cytological

- slides segmentation and clustering. *In International Symposium on Biomedical Imaging (IEEE)*, 2012.
- [Elmoataz et al. 11] A. ELMOATAZ, X. DESQUESNES, Z. LAKHDARI et O. LÉZORAY : On the infinity laplacian equation on graph with applications to image and manifolds processing. *In International Conference on Approximation Methods and Numerical Modelling in Environment and Natural Resources*, 2011.
- [Elmoataz et al. 12] A. ELMOATAZ, X. DESQUESNES et O. LÉZORAY : Non-local morphological pdes and p-laplacian equation on graphs with applications in image processing and machine learning. *IEEE Journal of Selected Topics in Signal Processing*, 2012.

# Bibliographie

- [Andreu-Vaillo et al. 10] F. ANDREU-VAILLO, J. M. MAZÓN, J. D. ROSSI et J. J. TOLEDO-MELERO : *NonLocal Diffusion Problems*. Amer. Math.Soc., 2010.
- [Arbeláez et Cohen 06] P. A. ARBELÁEZ et L. D. COHEN : A metric approach to vector-valued image segmentation. *International Journal of Computer Vision*, 69(1): 119–126, 2006.
- [Arehart et al. 93] A.B. AREHART, L. VINCENT et B.B. KIMIA : Mathematical morphology : The hamilton jacobi connection. pages 215–219, 1993.
- [Arias et al. 11] P. ARIAS, G. FACCILOLO, V. CASELLES et G. SAPIRO : A variational framework for exemplar-based image inpainting. *Int. J. of Comp. Vis.*, 93(3):319–347, 2011.
- [Aronsson 67] G. ARONSSON : Extension of functions satisfying lipschitz conditions. *Arkiv för Matematik*, 6(6): 551–561, 1967.
- [Aronsson et al. 04] G. ARONSSON, M. G. CRANDALL et P. JUUTINEN : A tour of the theory of absolutely minimizing functions. *Bull. Amer. Math. Soc.*, 41:439–505, 2004.
- [Bai et Sapiro 09] X. BAI et G. SAPIRO : Geodesic matting : A framework for fast interactive image and video segmentation and matting. *Int. J. Comput. Vis.*, 82:113–132, 2009.
- [Belkin et Niyogi 03] M. BELKIN et P. NIYOGI : Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.*, 15(6):1373–1396, 2003.



- [Bensoussan et Menaldi 03] A. BENSOUSSAN et J.-L. MENALDI : Difference equations on weighted graphs. *Convex Anal.*, 12(1):13–44, 2003.
- [Bertrand 05] G. BERTRAND : On topological watersheds. *Journal of Mathematical Imaging and Vision*, 22(2-3):217–230, 2005.
- [Bougleux et al. 07] S. BOUGLEUX, A. ELMOATAZ et M. MELKEMI : Discrete regularization on weighted graphs for image and mesh filtering. *In Proc. SSVM*, pages 128–139, 2007.
- [Bougleux et al. 09] S. BOUGLEUX, A. ELMOATAZ et M. MELKEMI : Local and nonlocal discrete regularization on weighted graphs for image and mesh processing. *International Journal of Computer Vision*, 84: 220–236, 2009.
- [Boykov et al. 06] Y. BOYKOV, V. KOLMOGOROV, D. CREMERS et A. DELONG : An integral solution to surface evolution pdes via geo-cuts. *In Proc. ECCV*, pages 409–422, 2006.
- [Boykov et Jolly 01] Y. BOYKOV et M.-P. JOLLY : Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. *In Proc. ICCV*, volume 1, pages 105–112, 2001.
- [Bruss 82] A. BRUSS : The eikonal equation : Some results applicable to computer vision. *J. Math. Phys.*, 23: 890–896, 1982.
- [Buades et al. 08] A. BUADES, B. COLL et J.-M. MOREL : Nonlocal image and movie denoising. *Int. J. Comput. Vis.*, 76:123–139, 2008.
- [Canny 86] J. CANNY : A computational approach to edge detection. *IEEE Trans. on Pattern Analysis Machine Intelligence*, 8:679–698, 1986.
- [Carlini et al. 06] E. CARLINI, E. CRISTIANI et N. FORCADEL : A non-monotone fast marching scheme for a hamilton-jacobi equation modelling dislocation dynamics. *In Numerical Mathematics and Advanced Applications*, pages 723–731. 2006.

- 
- [Carlini et al. 08] E. CARLINI, M. FALCONE, N. FORCADEL et R. MONNEAU : Convergence of a generalized fast-marching method for an eikonal equation with a velocity-changing sign. *SIAM J. Numer. Anal.*, 46:2920–2952, 2008.
- [Caselles et al. 97] V. CASELLES, R. KIMMEL et G. SAPIRO : Geodesic active contours. *Intl. Journal of Comput. Vision*, 22(1):61–79, 1997.
- [Caselles et al. 98] V. CASELLES, J.M. MOREL et C. SBERT : An axiomatic approach to image interpolation. *IEEE Trans. on Image Processing*, 7(3):376–386, 1998.
- [Chambolle et al. 12] A. CHAMBOLLE, E. LINDGREN et R. MONNEAU : A h ölder infinity laplacian. *ESAIM : Control Optimisation and Calculus of Variations*, 2012.
- [Chan et al. 01] T. F. CHAN, S. OSHER et J. SHEN : The digital tv filter and nonlinear denoising. *IEEE Trans. Image Process.*, 10:231–241, 2001.
- [Chan et Vese 01] T. F. CHAN et L. A. VESE : Active contours without edges. *IEEE Trans. Image Process.*, 10(2):266–277, 2001.
- [Chung 97] F. R. K. CHUNG : *Spectral Graph Theory*. American Mathematical Society, 1997.
- [Coifman et Maggioni 6b] R. R. COIFMAN et M. MAGGIONI : Diffusion wavelets. *Appl. Comput. Harmon. Anal.*, 21(1):53–94, 2006b.
- [Cong et al. 04] G. CONG, M. ESSER, B. PARVIN et G. BEBIS : Shape metamorphism using p-laplacian equation. *In ICPR*, volume 4, pages 15–18, 2004.
- [Couprie et al. 11a] C. COUPRIE, L. GRADY, L. NAJMAN et H. TALBOT : Power watersheds : A unifying graph-based optimization framework. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(7):1384–1399, 2011.
- [Couprie et al. 11b] C. COUPRIE, L. GRADYN, L. NAJMAN et H. TALBOT : Power watershed : A unifying graph-based optimization framework. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(7):1384–1399, 2011.
- [Courant et al. 28] R. COURANT, K. FRIEDRICHS et H. LEWY : On the partial difference equations of mathematical physics. *Math. Ann.*, 100:32–74, 1928.

- [Cousty et al. 10] J. COUSTY, G. BERTRAND, L. NAJMAN et M. COUPRIE : Watershed cuts : Thinnings, shortest path forests, and topological watersheds. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(5): 925–939, 2010.
- [Crandall 08] M.G. CRANDALL : A visit with the infinity laplace equations, 2008.
- [Crandall et al. 07] M.G. CRANDALL, G. GUNNARSSON et P. WANG : Uniqueness of  $\infty$ -harmonic functions and the eikonal equation. *Comm. Partial Differential Equations*, 32(10–12):1587–1615, 2007.
- [Danielsson 80] P. DANIELSSON : Euclidean distance mapping. *Computer Graphics and Image Processing*, 14: 227–248, 1980.
- [Diestel 05] R. DIESTEL : *Graph Theory (Graduate Texts in Mathematics)*. Springer, 2005.
- [Dijkstra 59] E. W. DIJKSTRA : A Note on Two Problems in Connection with Graphs. *Numerical Mathematics*, 1:269–271, 1959.
- [Drábek 07] P. DRÁBEK : The  $p$ -laplacian – mascot of non-linear analysis. *Acta Mathematica Universitatis Comenianae. New Series*, 76(1):85–98, 2007.
- [Efros et Leung 99] A. A. EFROS et T. K. LEUNG : Texture synthesis by non-parametric sampling. *In Proceedings of International Conference on Computer Vision (ICCV)*, volume 2, pages 1033–1038, 1999.
- [El Zehiry et al. 07] N. EL ZEHIRY, S. XU, P. SAHOO et A. ELMAGHRABY : Graph cut optimization for the mumford-shah model. *In Proc. VIIP*, pages 182–187, 2007.
- [Elion et Vese 07] C. ELION et L. A. VESE : An image decomposition model using the total variations and the infinity laplacian. *In SPIE*, volume 6498W, pages 1–10, 2007.
- [Elmoataz et al. 08a] A. ELMOATAZ, O. LÉZORAY et S. BOUGLEUX : Nonlocal discrete regularization on weighted graphs : A framework for image and manifold processing. *IEEE Trans. Image Process.*, 17(7): 1047–1060, 2008.

- 
- [Elmoataz et al. 08b] A. ELMOATAZ, O. LÉZORAY, S. BOUGLEUX et V.-T. TA : Unifying local and nonlocal processing with partial difference operators on weighted graphs. *In International Workshop on Local and Non-Local Approximation in Image Processing (LNLA)*, pages 11–26, 2008.
- [Falcão et al. 04] A. X. FALCÃO, J. STOLFI et R. de ALEN-CAR LOTUFO : The image foresting transform : Theory, algorithms, and applications. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26:19–29, 2004.
- [Faugeras 97] R. FAUGERAS, O. Keriven : Level set method and the stereo problem. *Scale-Space Theory in Computer Vision*, pages 272–283, 1997.
- [Forcadel et al. 08] N. FORCADEL, C. LE GUYADER et C. GOUT : Generalized fast marching method : applications to image segmentation. *Numerical Algorithms*, 48: 189–211, 2008.
- [Ghoniem et al. 11] M. GHONIEM, A. ELMOATAZ et O. LÉZORAY : Discrete infinity harmonic functions : towards a unified interpolation framework on graphs. *In Int. Conf. on Image Process. (IEEE)*, 2011.
- [Gilboa et Osher 07] G. GILBOA et S. OSHER : Nonlocal linear image regularization and supervised segmentation. *Multiscale Model. Simul.*, 2(6):595–630, 2007.
- [Gilboa et Osher 08] G. GILBOA et S. OSHER : Nonlocal operators with applications to image processing. *Multiscale Model. Simul.*, 7:1005–1028, 2008.
- [Grady 06] L. GRADY : Random walks for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28:1768–1783, 2006.
- [Grady et Alvino 09] L. GRADY et C. V. ALVINO : The piecewise smooth mumford-shah functional on an arbitrary graph. *IEEE Trans. Image Proc.*, 18:2547–2561, 2009.
- [Grady et Polimeni 10] L. GRADY et J. R. POLIMENI : *Discrete Calculus : Applied Analysis on Graphs for Computational Science*. Springer, 2010.

- [Guillot et Le Guyader 09] L. GUILLOT et C. LE GUYADER : Extrapolation of vector fields using the infinity laplacian and with applications to image segmentation. *In ICSSVMCV*, volume LNCS 5567, pages 87–99, 2009.
- [Hammond et al. 11] D. K. HAMMOND, P. VANDERGHEYNST et R. GRIBONVAL : Wavelets on graphs via spectral graph theory. *Appl. Comput. Harmon. Anal.*, 30(2):129–150, 2011.
- [Heijmans et al. 92] H. J. A. M. HEIJMANS, P. NACKEN, A. TOET et L. VINCENT : Graph morphology. *Journal of Visual Communication and Image Representation*, pages 24–38, 1992.
- [Heijmans et Vincent 92] H. J. A. M. HEIJMANS et L. VINCENT : Graph morphology in image analysis. *Mathematical Morphology in Image Processing*, pages 171–203, 1992.
- [Hein et al. 08] M. HEIN, U. VON LUXBURG et M. MAIER : Influence of graph construction on graph-based clustering measures. *In Advances in Neural Information Processing Systems*, pages 1025–1032, 2008.
- [Hein et Bühler 10] M. HEIN et T. BÜHLER : An inverse power method for nonlinear eigenproblems with applications in 1-spectral clustering and sparse pca. *In NIPS*, pages 847–855, 2010.
- [Hein et Maier 06] M. HEIN et M. MAIER : Manifold denoising. *In Proc. NIPS*, pages 561–568, 2006.
- [Hirani 03] A. N. HIRANI : Discrete exterior calculus. *Ph.D. thesis*, 2003.
- [Jansen et Oonincx 05] M. H. JANSEN et P. J. OONINCX : *Second Generation Wavelets and Applications*. Springer, 2005.
- [Jaromsky et Toussaint 92] J. W. JAROMSKY et G. T. TOUSSAINT : Relative neighborhood graphs and their relatives. *In Proceedings of the IEEE*, volume 80, 1992.
- [Jeong et Whitaker 08] W.-K. JEONG et R. T. WHITAKER : A fast iterative method for eikonal equations. *J. Sci. Comput.*, 30(5):2512–2534, 2008.

- 
- [Juutinen et al. 99] P. JUUTINEN, P. LINDQVIST et J.J. MANFREDI : The  $\infty$ -eigenvalue problem. *Arch. Rational Mech. Anal.*, 148:89–105, 1999.
- [Juutinen et Kawohl 06] P. JUUTINEN et B. KAWOHL : On the evolution governed by the infinity laplacian. *Matematische Annalen*, 335(4):819–851, 2006.
- [Kawohl 11] D. KAWOHL : Variations on the  $p$ -laplacian. *Cont. Math.*, (540):35–46, 2011.
- [Kichenassamy et al. 95] S. KICHENASSAMY, A. KUMAR, P. OLVER et A. TANNENBAUM : Gradient flows and geometric active contour models. *In ICCV*, pages 810–815, 1995.
- [Kimmel et Sethian 98] R. KIMMEL et J. A. SETHIAN : Computing geodesic paths on manifolds. *In Proc. Natl. Acad. Sci.*, pages 8431–8435, 1998.
- [Kurucz et al. 09] M. KURUCZ, A. BENCZÚR, K. CSALOGÁNY et L. LUKÁCS : Spectral clustering in social networks. *In Advances in Web Mining and Web Usage Analysis*, volume 5439, pages 1–20. 2009.
- [Le Gruyer 05] E. LE GRUYER : On absolutely minimizing lip-schitz extensions and pde  $\nabla_{\infty}(u) = 0$ . *Nonlinear Differential Equations and Applications*, 14:09–55, 2005.
- [Leventon et al. 00] M. LEVENTON, W. GRIMSON et O. FAUGERAS : Statistical shape influence in geodesic active contours. *In IEEE CCVPR*, pages 316–323, 2000.
- [Leveque 02] R. J. LEVEQUE : *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002.
- [Levinshtein et al. 09] A. LEVINSHTEIN, A. STERE, K.N. KUTULAKOS, D.J. FLEET, S.J. DICKINSON et K. SIDDIQI : Turbopixels : Fast superpixels using geometric flows. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(12):2290–2297, 2009.
- [Lindquist 06] P. LINDQUIST : *Notes on the  $p$ -Laplace equation*. University of Jyväskylä, Dept. of Mathematics and Statistics, 2006.
- [Luxburg 07] U. von LUXBURG : A tutorial on spectral clustering. *J. Stat. Comput.*, 17:395–416, 2007.

- [Lézoray et al. 07] O. LÉZORAY, A. ELMOATAZ et S. BOUGLEUX : Graph regularization for color image processing. *Computer Vision and Image Understanding (CVIU)*, 107(1-2):38–55, 2007.
- [Malladi et al. 95] R. MALLADI, J. A. SETHIAN et B. C. VEMURI : Shape modeling with front propagation : A level set approach. *IEEE Trans. on Pattern Anal. and Machine Intel.*, 17, 1995.
- [Manfredi et al. a] J. J. MANFREDI, M. PARVIAINEN et J. D. ROSSI : Dynamic programming principle for tug-of-war games with noise. *To appear in ESAIM Control Optim Calc. Var.*,.
- [Manfredi et al. b] J. J. MANFREDI, M. PARVIAINEN et J. D. ROSSI : On the definition and properties of p-harmonious functions. *To appear in Ann. Sc. Norm. Super. Pisa Cl. Sci.*
- [Meijering 12] E. MEIJERING : Cell segmentation : 50 years down the road. *IEEE Signal Processing Magazine*, 29(5): 140–145, 2012.
- [Monneau 10] R. MONNEAU : Introduction to the fast marching method. *HAL*, 2010.
- [Najman et Schmitt 94] L. NAJMAN et M. SCHMITT : Watershed of a continuous function. *Signal Processing*, 38(1):99 – 112, 1994.
- [Neuberger 06] J. M. NEUBERGER : Nonlinear elliptic partial difference equations on graphs. *Exp. Math.*, 15(1): 91–107, 2006.
- [Noh et Woodward 76] W. F. NOH et P. WOODWARD : Slic (simple line interface calculation). *In Proc. of Int. Conf. on Fluid Dynamics*, volume 59, pages 330–340, 1976.
- [Osher et Sethian 88] S. OSHER et J. A. SETHIAN : Fronts propagating with curvature dependent speed : Algorithms based on hamilton-jacobi formulations. *J. Comput. Phys.*, 79:12–49, 1988.
- [Osher et Shen 00] S. OSHER et J. SHEN : Digitized pde method for data restoration. *Anal. Comput. Methods. Appl. Math.*, pages 751–771, 2000.

- 
- [Paragios et Deriche 02] N. PARAGIOS et R. DERICHE : Geodesic active contours for supervised texture segmentation. *In IEEE CCVPR*, volume 2, 2002.
- [Park et Chung 11] J.-H. PARK et S.-Y. CHUNG : Positive solutions for discrete boundary value problems involving the p-laplacian with potential terms. *Comput. Math. Appl.*, 61:17–29, 2011.
- [Peres et al. 09] Y. PERES, O. SCHRAMM, S. SHEFFIELD et D. B. WILSON : Tug-of-war and the infinity laplacian. *J. Amer. Math. Soc.*, 22:167–210, 2009.
- [Qian et al. 07] J. QIAN, Y.-T. ZHANG et H.-K. ZHAO : Fast sweeping methods for eikonal equations on triangular meshes. *SIAM J. Numer. Anal.*, 45:83–107, January 2007.
- [Ren et Malik 03] X. REN et J. MALIK : Learning a classification model for segmentation. *In Proc. ICCV*, pages 10–, 2003.
- [Rouy et Tourin 92] E. ROUY et A. TOURIN : A viscosity solutions approach to shape-from-shading. *SIAM J. Numer. Anal.*, 3:867–884, 1992.
- [Schoenlieb et Bertozzi 11] C.-B. SCHOENLIEB et A. BERTOZZI : Unconditionally stable schemes for higher order inpainting. *Comm. Math. Sci.*, 2(9):413–457, 2011.
- [Sethian 85] J. A. SETHIAN : Curvature and the evolution of fronts. *Comm. in Math. Physics*, 101:487–499, 1985.
- [Sethian 90] J. A. SETHIAN : Numerical algorithms for propagating interfaces : Hamilton-jacobi equations and conservation laws. *J. of Differential Geometry*, 31:131–161, 1990.
- [Sethian 99] J. A. SETHIAN : *Level Set Methods and Fast Marching Methods : Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Science*. Cambridge University Press, 1999.
- [Shamir 08] A. SHAMIR : *Computer Graphics Forum*, 27(6): 1539–1556, 2008.



- [Shi et Malik 00] J. SHI et J. MALIK : Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, 2000.
- [Sinop et Grady 07] A. K. SINOP et L. GRADY : A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm. *In Proc. ICCV*, pages 1–8, 2007.
- [Ta et al. 08a] V.-T. TA, A. ELMOATAZ et O. LÉZORAY : Partial difference equations on graphs for mathematical morphology operators over images and manifolds. *In Proc. ICIP*, pages 801–804, 2008.
- [Ta et al. 08b] V.-T. TA, A. ELMOATAZ et O. LÉZORAY : Partial difference equations over graphs : Morphological processing of arbitrary discrete data. *In Proc. ECCV*, LNCS 5304, pages 668–680, 2008.
- [Ta et al. 09a] V.-T. TA, A. ELMOATAZ et O. LÉZORAY : Adaptation of eikonal equation over weighted graphs. *In Proc. SSVM*, page to appear, 2009.
- [Ta et al. 09b] V.-T. TA, O. LÉZORAY, A. ELMOATAZ et S. SCHÜPP : Graph-based tools for microscopic cellular image segmentation. *Pattern Recognition.*, 42:1113–1125, 2009.
- [Ta et al. 11] V.-T. TA, A. ELMOATAZ et O. LÉZORAY : Nonlocal pdes-based morphology on weighted graphs for image and data processing. *IEEE Trans. Image Process.*, 20(6):1504–1516, 2011.
- [Torre et Poggio 86] V. TORRE et T. POGGIO : On edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8:147–163, 1986.
- [Tsai et al. 04] Y.H.R. TSAI, L.T. CHENG, S. OSHER et H.K. ZHAO : Fast sweeping algorithms for a class of hamilton-jacobi equations. *SIAM journal on numerical analysis*, pages 673–694, 2004.
- [Tsitsiklis 95] J.N. TSITSIKLIS : Efficient algorithms for globally optimal trajectories. *IEEE Trans. Autom. Control*, 40(9):1528–1538, 1995.
- [Yuille et Poggio 86] A. L. YUILLE et T. A. POGGIO : Scaling theorems for zero crossings. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8:15–25, 1986.

- 
- [Zabusky et Overman 84] N. J. ZABUSKY et E. A. II OVERMAN : Regularization of contourdynamical algorithms. i. tangential regularization. *J. Comput. Physics.*, 52: 351–373, 1984.
- [Zhao 05] H. ZHAO : Fast sweeping method for eikonal equations. *Math. Comput.*, 74:603–627, 2005.
- [Zhou et Schölkopf 04] D. ZHOU et B. SCHÖLKOPF : A regularization framework for learning from graph data. *In Proceedings of Conference on Machine Learning 2004 Workshop on Statistical Relational Learning and Its Connections to Other Fields*, pages 132–137, 2004.
- [Zhou et Schölkopf 06] D. ZHOU et B. SCHÖLKOPF : *Discrete Regularization*, chapitre 3.13, pages 221–232. Adaptive computation and machine learning. 2006.



# Liste des figures

1	Exemples de graphes. . . . .	4
1.1	Construction de graphes sur un domaine non organisé . . . . .	19
1.2	Construction de graphes sur une image . . . . .	20
1.3	Exemples de patches. . . . .	21
1.4	Construction d'un graphe à partir d'une carte de régions . . . . .	22
1.5	Construction d'un graphe à partir d'un maillage 3D . . . . .	23
3.1	Illustration d'inpainting d'images . . . . .	56
3.2	Segmentation d'image par interpolation . . . . .	58
3.3	Classification de données par interpolation . . . . .	58
4.1	Exemple jouet . . . . .	69
4.2	Simplification d'images scalaires . . . . .	70
4.3	Simplification de textures sur un maillage 3D . . . . .	71
4.4	Segmentation semi-supervisée d'images naturelles. . . . .	73
4.5	Comparaison avec l'état de l'art des méthodes de segmentation semi-supervisées basées sur des graphes . . . . .	74
4.6	Segmentation d'une images naturelles basée sur des RAG . . . . .	75
4.7	Classification semi-supervisée de données réelles . . . . .	76
4.8	Segmentation de maillage 3D texturé . . . . .	77
4.9	Inpainting sur un maillage 3D. . . . .	78
4.10	Inpainting. . . . .	79
5.1	Illustration de l'algorithme Fast Marching . . . . .	95
5.2	Illustration de l'algorithme Fast Iterative Method . . . . .	96
5.3	Illustration de l'algorithme du Generalized Fast Marching . . . . .	97
6.1	Représentation discrète d'un front . . . . .	102

6.2	Illustration de l'algorithme optimisé . . . . .	111
6.3	Exemple jouet . . . . .	115
6.4	Évolution selon la courbure moyenne . . . . .	116
6.5	Contours actifs sur graphe . . . . .	117
7.1	Interactions entre les sommets pour la résolution locale . . . . .	128
7.2	Illustration de la résolution de la fonction $\mathcal{T}_0$ . . . . .	132
7.3	Illustration de la résolution de la fonction $\mathcal{T}_{t_1}$ . . . . .	132
7.4	Illustration de l'algorithme sur quelques itérations . . . . .	136
7.5	Calcul de géodésiques sur une image. . . . .	141
7.6	Calcul de géodésiques sur un RAG. . . . .	142
7.7	Calcul de géodésique sur un maillage texturé. . . . .	143
7.8	Calcul de géodésiques sur un graphe de données. . . . .	144
7.9	Calcul de géodésiques sur un graphe de topologie irrégulière. . .	145
8.1	Illustration de l'algorithme dans le cas de multiples fronts. . . .	152
8.2	Exemple jouet. . . . .	156
8.3	Super vertices sur maillage 3D . . . . .	159
8.4	Super vertices sur image 2D . . . . .	160
8.5	Comparaison avec le Power Watershed . . . . .	163
8.6	Segmentation d'images texturées par graphe non local . . . . .	164
8.7	Segmentation d'images RAG . . . . .	165
8.8	Classification semi-supervisée de données . . . . .	167
9.1	Évolution d'un front sans restrictions sur le signe de la vitesse .	174
9.2	Illustration du chevauchement des fronts . . . . .	176
9.3	Illustration de l'algorithme général. . . . .	179
9.4	Exemple jouet . . . . .	181
9.5	Contours actifs multiples sur une image . . . . .	182
9.6	Contours actifs multiples sur une image . . . . .	183
9.7	Contours actifs sur un RAG . . . . .	184
9.8	Contours actifs multiples sur un graphe de données . . . . .	185
10.1	Schéma général . . . . .	194
10.2	Transformée couleur . . . . .	196
10.3	Lissage par diffusion du $p$ -Laplacien normalisé . . . . .	197
10.4	Extraction automatique des noyaux. . . . .	198

---

10.5 Lissage des contours par la courbure . . . . .	199
10.6 Résultats de segmentation . . . . .	199
10.7 Calcul de la margination de la chromatine . . . . .	202
10.8 Calcul du nombre de mottes de chromatine . . . . .	202
10.9 Classification des noyaux . . . . .	204
11.1 Visualisation du résultat de la segmentation . . . . .	206
11.2 Interface pour l'aide au diagnostic . . . . .	207
11.3 Visualisation du résultat de la classification . . . . .	207
11.4 Visualisation d'un noyau dans son contexte . . . . .	208
11.5 Annotation collaborative de noyaux . . . . .	208
11.6 Navigateur . . . . .	209
11.7 Visualisation des caractéristiques . . . . .	210
11.8 Visualisation de la distribution des caractéristiques . . . . .	211



# Liste des algorithmes

1	Résolution de l'équation de propagation non-stationnaire . . . . .	112
2	Résolution de l'équation de propagation non-stationnaire (version optimisée) . . . . .	114
3	Résolution de l'équation de propagation stationnaire . . . . .	137
4	Propagation simultanée de plusieurs fronts . . . . .	155
5	Propagation simultanée de plusieurs fronts sans restrictions sur le signe de leurs vitesses . . . . .	180





# Index

## A

adjoint.....	27
aide au diagnostic .....	193
antarctic.....	205
arête.....	14

## B

base de connaissances.....	209
bordure	
externe .....	15
interne.....	15

## C

calcul discret.....	25
$n$ -connexité.....	20
contours actifs.....	118
courbure.....	30
cytologie.....	193

## D

dérivée.....	28
dérivée directionnelle.....	32
degré.....	15
différence .....	27
différence directionnelle .....	31, 32
différence pondérée .....	27
diffusion .....	67
dilatation .....	33
problème de Dirichlet.....	52
distance	
de chebyshev.....	16
de manhattan .....	16

euclidienne.....	16
divergence .....	28
domaine	
non organisé.....	17
organisé.....	19

## E

eikonale.....	90, 122
ensembles de niveaux .....	89
érosion.....	33
espace Hilbertien .....	26

## F

fast iterative method.....	95
fast marching .....	94
filtre moyennneur.....	34, 36
fonction de jeu.....	48
fonction de poids.....	13, 14
fonctions sur graphe .....	26
front.....	100
front global .....	149
front secondaire.....	175

## G

géodésique.....	140
gradient .....	28
gradient directionnel .....	32
graphe	
complet.....	18
de $\varepsilon$ -voisinage.....	18
des $k$ ppv.....	18
non-local.....	21

<p style="padding-left: 40px;">pondéré ..... 13</p> <p><b>I</b></p> <p>image ..... 19</p> <p>image texturée ..... 21</p> <p>inpainting ..... 55</p> <p>intégrale discrète ..... 26</p> <p>interpolation ..... 55</p> <p><b>L</b></p> <p>laplacien ..... 29</p> <p><math>p</math>-laplacien anisotrope ..... 31</p> <p><math>p</math>-laplacien isotrope ..... 29</p> <p><math>p</math>-laplacien normalisé ..... 62</p> <p>laplacien infini ..... 46</p> <p>laplacien infini non local ..... 49</p> <p><b>M</b></p> <p>métrique ..... 16</p> <p>maillage 3D ..... 23</p> <p>microscopie ..... 193</p> <p><b>N</b></p> <p>NLD ..... 36</p> <p>NLE ..... 37</p> <p>NLM ..... 36</p>	<p>norme ..... 29</p> <p><b>O</b></p> <p>opérateurs sur graphe ..... 26</p> <p><b>P</b></p> <p>partitionnement ..... 156</p> <p>patch ..... 21</p> <p>propagation externe ..... 170</p> <p>propagation interne ..... 170</p> <p><b>R</b></p> <p>RAG ..... 22</p> <p>restauration ..... 68</p> <p><b>S</b></p> <p>similarité ..... 16</p> <p>simplification ..... 69</p> <p>sommet ..... 13</p> <p>super vertices ..... 158</p> <p><b>T</b></p> <p>tug of war bruité ..... 63</p> <p>tug-of-war ..... 47</p> <p><b>V</b></p> <p>voisinage ..... 14</p>
---	---

**Résumé :** Cette thèse s'intéresse à la transcription d'équations aux dérivées partielles vers des domaines discrets en exploitant le formalisme des équations aux différences partielles définies sur des graphes pondérés. Dans une première partie, nous proposons une transcription de l'opérateur  $p$ -laplacien normalisé au domaine des graphes comme une combinaison linéaire entre le laplacien infini non-local et le laplacien normalisé (ces deux opérateurs étant discrets). Cette adaptation peut être considérée comme une nouvelle classe d'opérateurs  $p$ -laplaciens sur graphes, qui interpolent entre le laplacien infini non-local et le laplacien normalisé. Dans une seconde partie, nous nous intéressons aux équations de propagation de fronts sur des graphes de topologie arbitraire. Ces équations sont obtenues par la transcription de la méthode des ensembles de niveaux, définie en continu, vers une formulation discrète définie sur le domaine des graphes. Au delà de la transcription en elle-même, nous proposons une formulation générale et des algorithmes efficaces pour la propagation simultanées de plusieurs fronts évoluant sur un graphe. Les approches proposées dans ces deux premières parties donnent lieu à de nombreuses applications en segmentation d'images et classification de données que nous illustrons dans ce manuscrit. Enfin, dans une troisième partie, nous présentons une application à l'aide au diagnostic informatisé concrétisant l'emploi des différents outils proposés dans les deux premières parties. Nous présentons également le logiciel Antarctic développé au cours de cette thèse.

**Indexation RAMEAU :** Équations aux différences, Laplacien, Ensemble de niveaux, méthodes d', Traitement d'images – Techniques numériques

---

**Title :** Fronts propagation and normalized  $p$ -laplacien on graphs : Algorithms and applications to images and data processing

**Abstract :** This work deals with the transcription of continuous partial derivative equations to arbitrary discrete domains by exploiting the formalism of partial difference equations defined on weighted graphs. In the first part, we propose a transcription of the normalized  $p$ -Laplacian operator to the graph domains as a linear combination between the non-local infinity Laplacian and the normalized Laplacian (both in their discrete version). This adaptation can be considered as a new class of  $p$ -Laplacian operators on graphs that interpolate between non-local infinity Laplacian and normalized Laplacian. In the second part, we present an adaptation of fronts propagation equations on weighted graphs. These equations are obtained by the transcription of the continuous level sets method to a discrete formulation on the graphs domain. Beyond the transcription in itself, we propose a very general formulation and efficient algorithms for the simultaneous propagation of several fronts on a single graph. Both transcription of the  $p$ -Laplacian operator and level sets method enable many applications in image segmentation and data clustering that are illustrated in this manuscript. Finally, in the third part, we present a concrete application of the different tools proposed in the two previous parts for computer aided diagnosis. We also present the Antarctic software that was developed during this PhD.

**RAMEAU Index :** Difference equations, Laplacian operator, Level set methods, Image Processing – Digital Techniques

**Discipline :** Informatique et applications

Université de Caen Basse-Normandie, ENSICAEN, CNRS  
GREYC - équipe image  
6 Bvd Maréchal Juin, 14050 Caen cedex, France