

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

Université Mentouri Constantine

Faculté des Sciences de l'Ingénieur
Département d'Informatique

N° d'ordre: / /2009
Série: /INF/2009

Thèse de Doctorat en Sciences

Spécialité: Informatique

Intitulée:

Algorithmes évolutionnaires quantiques pour le recalage et la segmentation multiobjectif d'images

Présentée par:
Hichem Talbi

Directeur de thèse:
Pr. Mohamed Batouche

Soutenue le: 27 / 10 / 2009

Devant le jury constitué de:

Pr. Mohamed Benmohammed	Université Mentouri Constantine	Président
Dr. Allaoua Chaoui	Université Mentouri Constantine	Rapporteur
Pr. Hamid Seridi	Université 08 Mai 1945 Guelma	Examineur
Dr. Salim Chikhi	Université Mentouri Constantine	Examineur
Dr. Azzedine Bilami	Université Hadj Lakhdar Batna	Examineur

A mon fils, Mounib.

Remerciements

Je tiens à remercier d'abord ma mère, mon père, ma femme et tous les membres de ma grande famille pour leur soutien et leurs encouragements tout au long de ce travail.

J'exprime également ma profonde gratitude à mon directeur de thèse, Pr. Mohamed Batouche pour m'avoir encadré, orienté et guidé pour achever ce travail. Son soutien moral, ses qualités humaines et son esprit scientifique m'ont été de vrais propulseurs pour réaliser ce travail de recherche.

Merci à Dr. Allaoua Chaoui qui a accepté de me co-encadrer. Ses conseils et orientations m'ont permis de finaliser cette thèse dans les meilleures conditions.

Mes sincères remerciements à Pr. Mohamed Benmohammed pour avoir accepté d'être président du jury de soutenance. Je remercie également Pr. Hamid Seridi, Dr. Salim Chikhi et Dr. Azzedine Bilami qui ont accepté de juger ce travail et dont les remarques et suggestions auront un grand impact pour le raffinement de cette thèse.

Je remercie tous mes amis notamment Amer Draa, Anis Kadi, Younes Djaghloul, Salim Benayoune, Nabil Menadi, Amer Chabbi et Fouad Chebaiki pour leur soutien continu.

Je n'oublie pas tous les collègues enseignants chercheurs de l'Université Mentouri et de l'Université Emir Abdelkader, en particulier Salah Boudjemaa, Mustapha Quinten, Mourad Bouzenada et Nabil Belala.

Merci à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail.

Résumé

Une voie de recherche qui est très active de nos jours est celle qui essaye de développer de nouvelles approches s'inspirant des principes de la théorie quantique pour traiter plus efficacement les différents problèmes, notamment ceux portant sur l'optimisation. Dans le cadre de cette thèse, je m'intéresse à l'hybridation entre les algorithmes évolutionnaires et les principes de l'informatique quantique. Cette hybridation vise à améliorer le comportement des algorithmes évolutionnaires en leur donnant plus de chances pour avoir un bon équilibre entre renforcement et diversification au cours du processus d'optimisation. Cette hybridation offre de meilleures solutions via une convergence plus rapide utilisant des populations de tailles nettement inférieures grâce notamment au principe de superposition d'états. Les nouveaux opérateurs évolutionnaires quantiques qu'on a défini ont permis de résoudre des problèmes célèbres tels que le problème du sac à dos, le problème du voyageur de commerce et le problème des N reines de manière plus efficace. L'application de nos algorithmes évolutionnaires quantiques pour le recalage et la segmentation multiobjectif d'images a montré leur puissance et leur efficacité ce qui est très prometteur pour leur exploitation dans toutes les tâches liées à la vision par ordinateur.

Mots clés

Informatique quantique, recalage d'images, segmentation d'images, algorithmes évolutionnaires, optimisation multiobjectif.

Abstract

An active research direction nowadays is that trying to develop new approaches inspired from quantum theory principles for dealing more efficiently with the different problems, especially optimization ones. In this thesis, I am interested in the hybridization between evolutionary algorithms and the quantum computing principles. This hybridization aims to ameliorate the evolutionary algorithms behaviour by giving them more chances to have a good equilibrium between reinforcement and diversification along the optimization process. This hybridization gives better solutions through a faster convergence process using populations of definitely smaller size, in particular thanks to the states superposition principle. The new quantum evolutionary operators that we have defined have allowed the resolution of famous problems such as the knapsack problem, the travelling salesman problem and the N-Queens problem more efficiently. The application of our algorithms for image registration and multiobjective image segmentation has shown their power and efficiency which are promising for their employment in every tasks related to computer vision.

Keywords

Quantum Computing, Image Registration, Image Segmentation, Evolutionary Algorithms, Multiobjective Optimisation.

Principales publications de l'auteur

Reuves avec comité de lecture

1. Hichem Talbi, Mohamed Batouche, Amer Draa. **"A Quantum-Inspired Genetic Algorithm for Multi-Source Affine Image Registration"**. *International Conference on Image Analysis and Recognition (ICLAR'2004)*, Porto, September 2004, Lecture Notes in Computer Science: Image Analysis and Recognition Springer-Verlag Press, LNCS 3211 pp. 147-154.
2. Hichem Talbi, Amer Draa, Mohamed Batouche. **"A Novel Quantum-Inspired Evolutionary Algorithm for Multi-Sensor Image Registration"**. *International Arabic Journal on Information Technology*. Vol.3. N°1., January 2006. ISSN: 1683-3198, pp. 9-15.
3. Hichem Talbi, Mohamed Batouche, Amer Draa. **"A Quantum-Inspired Evolutionary Algorithm for Multiobjective Image segmentation"**. *International Journal of Mathematical, Physical and Engineering Sciences*, Volume 1, Number2, 2007, ISSN 2070-3880, pp. 109-114.

Actes de conférences internationales avec comité de lecture:

1. Hichem Talbi, Amer Draa, Mohamed Batouche. **"A Quantum Genetic Algorithm for Image Registration"**. *In the Proceedings of the International Conference on Information & Communication Technologies: from Theory to Applications (ICTTA'2004)*, Damascus, April 2004. IEEE Press, ISBN: 0-7803-8482-2/04.
2. Amer Draa, Hichem Talbi, Mohamed Batouche. **"A Quantum-Inspired Differential Evolution Algorithm for Rigid Image Registration"**. *In the proceedings of the International Conference on Computational Intelligence (ICCI 2004)*, Istanbul, December 2004, ISBN 975-98458 pp.408-411.
3. Amer Draa, Hichem Talbi, Mohamed Batouche. **"Une approche génétique quantique pour la fusion d'images multi-sources"**. *7^{ème} Colloque Africain sur la Recherche en Informatique (CARI'2004)*, Hammamet, November 2004, pp. 75-82.
4. Amer Draa, Hichem Talbi, Mohamed Batouche. **"Une approche génétique quantique pour le recalage d'images médicales multi-sources"**. *Journées d'études Algéro-Française en Imagerie Médicale (JETIM'2004)*, Blida, November 2004.

5. Hichem Talbi, Amer Draa, Mohamed Batouche. **"A New Quantum-Inspired Genetic Algorithm for Solving the Travelling Salesman Problem"**. *In the Proceedings of the IEEE International Conference on Industrial Technology. (ICIT'2004)*, Hammamet, December 2004, Volume 3, pp.1192 - 1197, IEEE Press, ISBN 0-7803-8662-0/04.
6. Amer Draa, Hichem Talbi, Mohamed Batouche. **"A Quantum-Inspired Genetic Algorithm for Solving the N-Queens Problem"**. *In the Proceedings of the 7th International Symposium on Programming and Systems, (ISPS'2005)*, Algiers, May 2005, pp.145-152.
7. Hichem Talbi, Mohamed Batouche, Amer Draa. **"A Quantum-Inspired Evolutionary Algorithm for Multiobjective Image segmentation"**. *In the Proceedings of World Academy of Science, Engineering and Technology*, Volume 25, November 2007, Venice, ISSN 1307-6884, pp. 205-210.

Actes de conférences nationales avec comité de lecture :

1. Amer Draa, Hichem Talbi, Mohamed Batouche. **" Un algorithme génétique quantique pour le recalage d'images multi-sources"**. *4^{ème} Séminaire National en Informatique de Biskra (SNIB'2004)*, Biskra, May 2004.
2. Amer Draa, Hichem Talbi, Mohamed Batouche. **"Une hybridation génétique quantique pour la résolution du problème du voyageur de commerce"**. *2^{èmes} Journées de l'Informatique pour l'Entreprise (JIE'2004)*, Blida, October 2004.

Table des matières

RESUME	III
ABSTRACT	IV
PRINCIPALES PUBLICATIONS DE L'AUTEUR	V
TABLE DES MATIERES	VII
LISTE DES FIGURES	X
LISTE DES TABLEAUX	XII
INTRODUCTION GENERALE	1
CHAPITRE 1 INFORMATIQUE QUANTIQUE	6
1.1. INTRODUCTION	6
1.2. FONDEMENTS PHYSIQUES DE L'INFORMATIQUE QUANTIQUE	7
<i>Expérience de polarisation des photons</i>	7
Explication	8
1.3. CONCEPTS DE BASE	11
1.3.1. Espace d'état et notation de Dirac	11
1.3.2. Bit quantique	12
1.3.3. Registre quantique	13
1.3.4. Les cinq principes de l'informatique quantique	13
- Superposition d'états	13
- Interférence	13
- Enchevêtrement ou intrication	14
- Non déterminisme et non duplication	14
- La réversibilité	14
1.3.5. Calcul quantique	15
1.3.5.1. Portes quantiques	15
1.3.5.2. Circuits quantiques	17
1.3.6. Algorithmes quantiques	19
1.4. ALGORITHMES QUANTIQUES CELEBRES	19
1.4.1. Algorithme de Shor	19
1.4.2. Algorithme de Grover	23
1.4.3. Algorithme de Deutsch - Jozsa	25
1.5. ORDINATEURS QUANTIQUES	28
1.5.1. Un peu d'histoire	28
1.5.2. Réalisation de l'ordinateur quantique	29
1.5.3. Principales difficultés	31

1.6. CONCLUSION -----	33
CHAPITRE 2 RECALAGE ET SEGMENTATION D'IMAGES-----	34
2.1. INTRODUCTION -----	34
2.2. RECALAGE D'IMAGE -----	35
2.2.1. Définition du recalage d'images -----	35
2.2.2. Classification des transformations géométriques -----	36
2.2.2.1. Transformations rigides -----	36
2.2.2.2. Similitudes -----	36
2.2.2.3. Transformations affines -----	36
2.2.2.4. Transformations projectives -----	37
2.2.2.5. Transformations locales paramétriques -----	37
2.2.3. Types et domaines d'application du recalage -----	38
2.2.3.1. Types de recalage -----	38
2.2.3.1. Domaines d'application du recalage -----	39
2.2.4. Méthodes de recalage-----	41
2.2.4.1. Méthodes basées primitives (géométriques)-----	41
2.2.4.2. Méthodes iconiques -----	45
- Méthodes approchées pour le recalage -----	53
2.3. SEGMENTATION D'IMAGES -----	53
2.3.1. Définition -----	53
2.3.2. Applications de la segmentation d'images-----	54
2.3.3. Méthodes de segmentation d'images -----	55
2.4. CONCLUSION-----	56
CHAPITRE 3 OPTIMISATION COMBINATOIRE ET OPTIMISATION MULTIOBJECTIF-----	58
3.1. INTRODUCTION -----	58
3.2. OPTIMISATION COMBINATOIRE -----	59
3.2.1. Définition -----	59
3.2.2. Les problèmes NP-difficile et les problèmes NP-complet-----	59
3.2.3. Méthodes approchées pour la résolution des problèmes d'optimisation combinatoire -----	64
3.3. OPTIMISATION MULTIOBJECTIF -----	64
3.3.1. Définition -----	64
3.3.2. Formulation du problème -----	65
3.3.3. Optimalité de Pareto et dominance -----	65
3.3.3.1. Notion d'optimum en optimisation multiobjectif-----	65
3.3.3.2. Définition de la dominance -----	66
3.3.3.3. Ensemble de Pareto -----	66
3.3.4. Un peu d'histoire -----	66
3.3.5. Optimisation multiobjectif et algorithmes évolutionnaires -----	67
3.3.5.1. Pourquoi les algorithmes évolutionnaires?-----	67
3.3.5.2. Vue d'horizon des algorithmes évolutionnaires pour l'optimisation multiobjectif-----	68
3.4. CONCLUSION -----	79

CHAPITRE 4 ALGORITHMES EVOLUTIONNAIRES QUANTIQUES (AEQ)	80
4.1. INTRODUCTION	80
4.2. ALGORITHMES EVOLUTIONNAIRES QUANTIQUES	82
<i>Présentation</i>	82
<i>Discussion</i>	84
4.3. NOUVEAUX ALGORITHMES EVOLUTIONNAIRES QUANTIQUES	86
4.4. APPLICATION A LA RESOLUTION DE PROBLEMES D'OPTIMISATION COMBINATOIRE	88
4.4.1. <i>Problème du sac à dos</i>	88
- Effet de la valeur de l'angle de rotation	89
- Effet de l'opération de mutation	94
- Effet de l'opération de permutation	96
- Effet de l'opération de croisement	98
- Effet de la combinaison de la permutation avec le croisement quantique	99
4.4.2. <i>Problème du voyageur de commerce</i>	100
4.4.3. <i>Problème des N-Reines</i>	107
4.5. CONCLUSION	111
CHAPITRE 5 ALGORITHMES EVOLUTIONNAIRES QUANTIQUES POUR LE RECALAGE ET LA SEGMENTATION MULTIOBJECTIF D'IMAGES	113
5.1. INTRODUCTION	113
5.2. ALGORITHMES EVOLUTIONNAIRES QUANTIQUES POUR LE RECALAGE D'IMAGES	114
5.2.1. <i>Description</i>	114
5.2.2. <i>Résultats expérimentaux</i>	117
5.2.2.1. Recalage rigide	118
5.2.2.2. Recalage rigide avec facteur d'échelle	123
5.2.2.2. Recalage affine	126
5.2.3. <i>Comparaison aux algorithmes génétiques classiques</i>	129
5.2.3.1. Temps de calcul	129
5.2.3.1. Qualité du recalage	130
5.3. ALGORITHMES EVOLUTIONNAIRES QUANTIQUES POUR LA SEGMENTATION MULTIOBJECTIF D'IMAGES	132
5.3.1. <i>Motivation</i>	132
5.3.2. <i>Description de l'algorithme</i>	133
5.3.3. <i>Résultats expérimentaux</i>	138
CONCLUSION GENERALE	142
BIBLIOGRAPHIE	146

Liste des figures

FIGURE 1.1. L'AJOUT DU FILTRE A-----	7
FIGURE 1.2. L'AJOUT DU FILTRE C-----	8
FIGURE 1.3. L'AJOUT DU FILTRE B-----	8
FIGURE 1.4. LA MESURE EST UNE PROJECTION SUR LA BASE CHOISIE. -----	9
FIGURE 1.5. L'EFFET DE CHAQUE FILTRE SUR LA PORTION DE LUMIERE OBSERVEE -----	11
FIGURE 1.6. CIRCUIT QUANTIQUE REPRESENTANT LE CONTROLLED-NOT-----	17
FIGURE 1.7. CONTROLLED-CONTROLLED-NOT -----	18
FIGURE 1.8. OPERATIONS A UNE SEULE ENTREE -----	18
FIGURE 1.9. CIRCUIT QUANTIQUE-----	18
FIGURE 2.1. RECALAGE D'IMAGES-----	35
FIGURE 2.2. TRANSFORMATIONS GEOMETRIQUES. -----	41
FIGURE 2.3. CLASSIFICATION GENERALE DES METHODES DE RECALAGE D'IMAGES SELON LE TYPE DE PRIMITIVES QU'ELLES UTILISENT -----	46
FIGURE 2.4. HISTOGRAMME CONJOINT DE DEUX IRM CEREBRALES (COUPES CORONALES REPRESENTEES) CALCULE POUR DEUX TRANSFORMATIONS DIFFERENTES [ROCHE 2001] -----	49
FIGURE 2.5. HISTOGRAMME CONJOINT D'UN COUPLE SCANNER/IRM (COUPES AXIALES REPRESENTEES) CALCULE POUR DEUX TRANSFORMATIONS DIFFERENTES [ROCHE 2001] -----	49
FIGURE 3.1. LE CIRCUIT DE HAMILTON REPRESENTE PAR LA LIGNE GRASSE -----	60
FIGURE 3.2. LE PROBLEME DU CIRCUIT DE HAMILTON AVEC LES REPONSES «OUI» ET «NON» -----	62
FIGURE 3.3. LA CLASSE DES PROBLEMES NP SUPPOSE QUE $P \neq NP$. -----	63
FIGURE 4.1. EFFET DE LA PORTE D SUR LE QUBIT -----	83
FIGURE 4.2. MUTATION QUANTIQUE -----	86
FIGURE 4.3. PERMUTATION QUANTIQUE -----	87
FIGURE 4.4. CROISEMENT QUANTIQUE-----	87
FIGURE 4.5. PERFORMANCE MOYENNE D'UN AGQ EN FONCTION DE L'ANGLE DE ROTATION -----	91
FIGURE 4.6. DISTRIBUTION DE X POUR DES VALEURS DE L'ANGLE DE ROTATION VARIANT 1° ET 30° ET P VARIANT ENTRE 1 ET 100-----	93
FIGURE 4.7. EVOLUTION DE LA VALEUR DU MEILLEUR RESULTAT DANS LA MEILLEURE EXECUTION -	94
FIGURE 4.8. EXEMPLE DU PROBLEME DE VOYAGEUR DE COMMERCE -----	100
FIGURE 4.9. LA MATRICE DES DISTANCES DE L'INSTANCE "GR24.TSP" -----	102
FIGURE 4.10. LA REPRESENTATION DE LA SOLUTION DE L'INSTANCE DE LA FIGURE 4.8-----	102
FIGURE 4.11. CODAGE GENETIQUE QUANTIQUE -----	103
FIGURE 4.12. SCHEMA GENERAL DE L'ALGORITHME -----	103

FIGURE 4.13. OPERATION DE CROISEMENT -----	104
FIGURE 4.14. UNE SOLUTION DU PROBLEME DES 8 REINES-----	108
FIGURE 4.15. EXEMPLE D'UNE CONFIGURATION A 4 REINES -----	109
FIGURE 5.1. ALGORITHME EVOLUTIONNAIRE QUANTIQUE POUR LE RECALAGE D'IMAGES -----	117
FIGURE 5.2. IMAGES IRM -----	118
FIGURE 5.3. PAIRE D'IMAGES: IRM-TOMOGRAPHIE CALCULEE A RAYON X (CTI) -----	121
FIGURE 5.4. IMAGES IRM -----	123
FIGURE 5.5. PAIRE D'IMAGES: IRM-TOMOGRAPHIE CALCULEE A RAYON X (CTI) -----	124
FIGURE 5.6. RECALAGE AFFINE D'IMAGES MONOMODALES -----	126
FIGURE 5.7. EVOLUTION DE LA MEILLEURE VALEUR DE L'INFORMATION MUTUELLE-----	127
FIGURE 5.8. RECALAGE AFFINE D'IMAGES MULTIMODALES (IRM-CTI) -----	127
FIGURE 5.9. EVOLUTION DE LA MEILLEURE VALEUR DE L'INFORMATION MUTUELLE-----	128
FIGURE 5.10. RECALAGE AFFINE D'IMAGES MULTIMODALES (IRM- SCINTIGRAPHIE)-----	128
FIGURE 5.11. EVOLUTION DE LA MEILLEURE VALEUR DE L'INFORMATION MUTUELLE -----	129
FIGURE 5.12. UN AUTRE EXEMPLE DE RECALAGE AFFINE D'IMAGES MULTIMODALES -----	129
FIGURE 5.13. COMPARAISON DES TEMPS DE CALCULS DE AEQ, AGC-15 ET AGC-50-----	130
FIGURE 5.14. COMPARAISON DE L'EVOLUTION DE LA MEILLEURE VALEUR DE L'INFORMATION MUTUELLE DE AEQ ET AGC-50-----	131
FIGURE 5.15. DESCRIPTION SOMMAIRE DE L'ALGORITHME DE SEGMENTATION-----	133
FIGURE 5.16. CODAGE D'UNE SOLUTION -----	134
FIGURE 5.17. L'ALGORITHME EVOLUTIONNAIRE QUANTIQUE POUR LA SEGMENTATION MULTIOBJECTIF D'IMAGES-----	135
FIGURE 5.18. L'IMAGE "CAMERAMAN" DE TAILLE 256x256 -----	138
FIGURE 5.19. L'EVOLUTION DE LA TAILLE DE L'ENSEMBLE DES SOLUTIONS NON DOMINEES -----	139
FIGURE 5.20. LES VALEURS DES FONCTIONS OBJECTIFS POUR LES SOLUTIONS NON DOMINEES -----	140
FIGURE 5.21. QUELQUES SOLUTIONS NON DOMINEES ISSUE DU MOQEA -----	141

Liste des tableaux

TABLEAU 2.1. TYPES DES PRIMITIVES -----	44
TABLEAU 2.2. MESURES DE SIMILARITE UTILES-----	52
TABLEAU 3.1. TECHNIQUES DE PREMIERE GENERATION NE CONSIDERANT PAS L'OPTIMALITE AU SENS DE PARETO -----	70
TABLEAU 3.2. TECHNIQUES DE PREMIERE GENERATION CONSIDERANT L'OPTIMALITE PARETO	72
TABLEAU 4.1. DIRECTION DE LA ROTATION -----	83
TABLEAU 4.2. EFFET DU CHANGEMENT DE LA VALEUR DE L'ANGLE DE ROTATION DANS KSP POUR NC=1 -----	89
TABLEAU 4.3. EFFET DU CHANGEMENT DE LA VALEUR DE L'ANGLE DE ROTATION DANS KSP POUR NC=2 -----	89
TABLEAU 4.4. EFFET DU CHANGEMENT DE LA VALEUR DE L'ANGLE DE ROTATION DANS KSP POUR NC=3 -----	90
TABLEAU 4.5. EFFET DU CHANGEMENT DE LA VALEUR DE L'ANGLE DE ROTATION DANS KSP POUR NC=4 -----	90
TABLEAU 4.6. EFFET DU CHANGEMENT DE LA VALEUR DE L'ANGLE DE ROTATION DANS KSP- RECAPITULATION -----	90
TABLEAU 4.7. EFFET DE L'OPERATION DE MUTATION DANS KSP POUR $\theta=\pi/10$ -----	95
TABLEAU 4.8. EFFET DE L'OPERATION DE MUTATION DANS KSP POUR $\theta=\pi/15$ -----	95
TABLEAU 4.9. EFFET DE L'OPERATION DE MUTATION DANS KSP POUR $\theta=\pi/20$ -----	95
TABLEAU 4.10. EFFET DE L'OPERATION DE PERMUTATION DANS KSP POUR $\theta=\pi/10$ -----	96
TABLEAU 4.11. EFFET DE L'OPERATION DE PERMUTATION DANS KSP POUR $\theta=\pi/15$ -----	97
TABLEAU 4.12. EFFET DE L'OPERATION DE PERMUTATION DANS KSP POUR $\theta=\pi/20$ -----	97
TABLEAU 4.13. EFFET DE L'OPERATION DE CROISEMENT DANS KSP POUR $\theta=\pi/10$ -----	98
TABLEAU 4.14. EFFET DE L'OPERATION DE CROISEMENT DANS KSP POUR $\theta=\pi/15$ -----	98
TABLEAU 4.15. EFFET DE L'OPERATION DE CROISEMENT DANS KSP POUR $\theta=\pi/20$ -----	98
TABLEAU 4.16. EFFET DE LA COMBINAISON DE LA PERMUTATION AVEC LE CROISEMENT DANS KSP POUR $\theta=\pi/10$ -----	99
TABLEAU 4.17. EFFET DE LA COMBINAISON DE LA PERMUTATION AVEC LE CROISEMENT DANS KSP POUR $\theta=\pi/15$ -----	99

TABLEAU 4.18. EFFET DE LA COMBINAISON DE LA PERMUTATION AVEC LE CROISEMENT DANS KSP POUR $\theta=\pi/20$ -----	100
TABLEAU 4.19. RESUME DES RESULTATS DE L'APPLICATION DES AEQ POUR LA RESOLUTION DU TSP-----	107
TABLEAU 4.20. RESUME DES RESULTATS DE L'APPLICATION DES AEQ POUR LA RESOLUTION DU PROBLEME DES N-REINES POUR N=4 A 7-----	109
TABLEAU 4.21. RESUME DES RESULTATS DE L'APPLICATION DES AEQ POUR LA RESOLUTION DU PROBLEME DES N-REINES POUR N=8 A 10 -----	110
TABLEAU 4.22. RESUME DES RESULTATS DE L'APPLICATION DES AEQ POUR LA RESOLUTION DU PROBLEME DES N-REINES POUR N=11 A13 -----	110
TABLEAU 4.23. RESUME DES RESULTATS DE L'APPLICATION DES AEQ POUR LA RESOLUTION DU PROBLEME DES N-REINES POUR N=14 A 16-----	111
TABLEAU 5.1. RECALAGE RIGIDE DES IMAGES DE LA FIGURE 5.2 -----	119
TABLEAU 5.2. EVOLUTION DE LA MEILLEURE VALEUR DE L'INFORMATION MUTUELLE AU COURS DU RECALAGE RIGIDE DES IMAGES DE LA FIGURE 5.2-----	120
TABLEAU 5.3. VALEUR MOYENNE DE L'INFORMATION MUTUELLE SUR LES 10 EXECUTIONS---	120
TABLEAU 5.4. ECART TYPE DE L'INFORMATION MUTUELLE SUR LES 10 EXECUTIONS -----	120
TABLEAU 5.5. RECALAGE RIGIDE DES IMAGES DE LA FIGURE 5.3 -----	121
TABLEAU 5.6. EVOLUTION DE LA MEILLEURE VALEUR DE L'INFORMATION MUTUELLE AU COURS DU RECALAGE RIGIDE DES IMAGES DE LA FIGURE 5.3-----	122
TABLEAU 5.7. VALEUR MOYENNE DE L'INFORMATION MUTUELLE SUR LES 10 EXECUTIONS---	122
TABLEAU 5.8. ECART TYPE DE L'INFORMATION MUTUELLE SUR LES 10 EXECUTIONS -----	122
TABLEAU 5.9. RECALAGE RIGIDE DES IMAGES DE LA FIGURE 5.4 -----	123
TABLEAU 5.10. EVOLUTION DE LA MEILLEURE VALEUR DE L'INFORMATION MUTUELLE AU COURS DU RECALAGE RIGIDE DES IMAGES DE LA FIGURE 5.4 -----	124
TABLEAU 5.11. VALEUR MOYENNE ET ECART TYPE DE L'INFORMATION MUTUELLE SUR LES 10 EXECUTIONS-----	124
TABLEAU 5.12. RECALAGE RIGIDE DES IMAGES DE LA FIGURE 5.5-----	125
TABLEAU 5.13. EVOLUTION DE LA MEILLEURE VALEUR DE L'INFORMATION MUTUELLE AU COURS DU RECALAGE RIGIDE DES IMAGES DE LA FIGURE 5.5 -----	125
TABLEAU 5.14. VALEUR MOYENNE ET ECART TYPE DE L'INFORMATION MUTUELLE SUR LES 10 EXECUTIONS-----	125
TABLEAU 5.15. RESUME DE LA DYNAMIQUE DE L'ENSEMBLE DES SOLUTIONS NON DOMINEES	139

Introduction générale

L'informatique a connu depuis le milieu du siècle précédent des progrès énormes. Elle est devenue un élément incontournable dans tous les secteurs de la vie. Avec des machines de plus en plus puissantes et des techniques de programmation de plus en plus innovatrices et performantes, on a de plus en plus l'audace de s'attaquer à la résolution automatique de problèmes très complexes qui étaient jusque là hors de portée de l'outil informatique. Cependant et malgré les milliards d'instructions que les machines actuelles peuvent exécuter en une seconde de temps, nous rencontrons toujours des problèmes à forte combinatoire qui ne sont pas solvables de manière exacte en un temps raisonnable.

Pour pousser encore plus loin les limites de l'informatique et lui permettre de traiter encore plus de problèmes, plusieurs voies et stratégies ont été adoptées:

- La première voie est la plus classique. Elle a bénéficié de plus d'investissements depuis les débuts de l'ère informatique pour tenter d'améliorer incessamment la vitesse de calcul des ordinateurs et leur capacité de stockage. Cette tendance s'appuie fondamentalement sur l'utilisation de matériaux possédant les meilleurs caractéristiques physiques, la miniaturisation des composants et l'adoption d'architectures de plus en plus sophistiquées. La technologie des processeurs multi-cœurs, ou processeurs à plusieurs unités de calcul, est l'une des plus favorisées en ce moment.
- Une deuxième voie est celle qui vise à exploiter l'expansion fabuleuse qu'ont connu ces dernières années les réseaux informatiques. Le développement et l'utilisation de grilles de calcul sont d'actualité. Un réseau comme Internet avec plusieurs centaines de millions de machines connectées offrirait une puissance de calcul extraordinaire qui servirait à traiter des problèmes en un temps des centaines de millions de fois plus court que sur une simple machine.

- Une troisième tendance vise le développement d'approches nouvelles pour traiter les différents problèmes. Les techniques de l'intelligence artificielle et celles inspirées de phénomènes naturels divers (algorithmes génétiques, réseaux de neurones, colonies de fourmis, essaims de particules et autres) se sont révélées très prometteuses pour résoudre des problèmes complexes de manière efficace.
- Une quatrième voie est celle qui cherche à construire des ordinateurs totalement différents des ordinateurs actuels. L'ordinateur neuronal, l'ordinateur optique, l'ordinateur à ADN et l'ordinateur quantique sont des candidats pour remplacer les ordinateurs actuels.

Le développement des ordinateurs actuels est basé en premier lieu sur l'amélioration des circuits et composants électroniques. Des records du niveau d'intégration des processeurs sont régulièrement battus et les transistors qui les composent actuellement sont de tailles très réduites. On devrait très prochainement atteindre "*The Wall*", la barrière des 20 nm de section pour un transistor qui représente seulement quelques dizaines d'atomes et qui est vue comme limite ultime de la technologie du silicium. Au-delà de cette limite, ce ne sont plus les lois classiques de la physique qui gouvernent mais celles de la physique quantique.

La théorie quantique est actuellement la plus plausible pour décrire et expliquer les différents phénomènes physiques aussi bien à l'échelle macroscopique qu'à l'échelle de l'infiniment petit. Elle est connue pour être contre intuitive et même les grands physiciens reconnaissent cela. *Richard Feynman*, Prix Nobel de physique en 1965, a ainsi écrit: "*I think I can safely say that nobody understands quantum mechanics.*" [Feynman 1965], c'est à dire que personne ne comprend vraiment la physique quantique. Cela est dû aux nombreux concepts inhabituels dans le monde macroscopique que nous apercevons tous les jours, des concepts comme le non déterminisme et l'incertitude, la dualité onde-corpuscule, la contractualité, l'intrication ou la non localité, etc.

L'avènement des phénomènes quantiques vu comme obstacle au développement de l'informatique est de nos jours perçu comme la clé du développement de nouvelles machines aux aptitudes extraordinaires. Des machines quantiques dotées d'un parallélisme intrinsèque fruit de la superposition d'états qu'un système

quantique peut maintenir. On n'utilisera plus des cases comportant soit "0" soit "1" mais des cases comportant les deux en même temps appelées qubits. D'après David Deutsch [Deutsch 1985], un ordinateur quantique de 100 qubits nous permettrait de simuler un cerveau humain et un ordinateur de 300 qubits serait capable de simuler l'évolution de tout l'univers. Les recherches sont en cours pour la construction de tels ordinateurs et cette tâche s'est révélée assez compliquée à cause surtout des phénomènes de décohérence.

Dans l'attente de réalisation de vraies machines quantiques, qui pourraient exécuter effectivement les différents algorithmes quantiques développés pour résoudre des problèmes complexes en des temps très réduits, on essaye actuellement de développer de nouvelles approches s'inspirant des principes de la théorie quantique [Nedjah et al. 2008] pour traiter plus efficacement les différents problèmes, notamment ceux portant sur l'optimisation.

Dans le cadre de cette thèse, je m'intéresse à l'hybridation entre les algorithmes évolutionnaires, qui ont déjà fait leur preuve dans le domaine de l'optimisation, et les principes de l'informatique quantique qui ont permis d'ouvrir de nouvelles perspectives. Cette hybridation vise à améliorer le comportement des algorithmes évolutionnaires en leur donnant plus de chances pour avoir un bon équilibre entre renforcement et diversification au cours du processus d'optimisation.

L'objectif de cette thèse est de:

- montrer l'utilité d'une telle hybridation pour garantir de meilleures solutions via une convergence plus rapide utilisant des populations de tailles nettement inférieures,
- définir de nouveaux opérateurs évolutionnaires quantiques et mettre l'accent sur l'importance du rôle qu'ils jouent dans le processus d'optimisation,
- démontrer la supériorité des algorithmes développés dans la résolution de problèmes célèbres tels que le problème du sac à dos, le problème du voyageur de commerce et le problème des N reines,
- concevoir des algorithmes évolutionnaires quantiques pour traiter deux problèmes cruciaux du domaine de la vision par ordinateur, à savoir le recalage et la

segmentation d'images avec prise en considération de l'aspect multiobjectif pour le deuxième.

Le travail présenté dans ce document est organisé en cinq chapitres:

Le chapitre 1 est une introduction à l'informatique quantique. Il présente quelques informations sur l'histoire et le développement de l'informatique quantique, la définition de quelques concepts fondamentaux tels que le qubit, le registre quantique, la superposition d'états, l'intrication et les opérateurs quantiques. Il présente également quelques algorithmes quantiques célèbres, les principales voies empruntées pour la réalisation de l'ordinateur quantique et les difficultés majeures rencontrées.

Le chapitre 2 est consacré aux problèmes de recalage et de segmentation d'images. On y trouve la définition du recalage et son utilité, les différents types de recalage et ses applications, la définition de la segmentation d'images et les différentes approches adoptées pour sa réalisation.

Le chapitre 3 aborde le domaine d'optimisation combinatoire et met l'accent sur l'optimisation multiobjectif. Il dresse notamment un état de l'art concernant les différentes approches évolutionnaires développées pour la résolution de problèmes d'optimisation multiobjectif.

Le chapitre 4 expose les algorithmes évolutionnaires quantiques. Il explique les lacunes que rencontrent les algorithmes existants et introduit les améliorations qu'on a apporté dans le cadre de cette thèse. Le rôle que jouent les différents paramètres et opérateurs a été mis en évidence sur un problème d'école, celui du sac à dos. Le chapitre décrit par la suite les algorithmes conçus pour la résolution du problème du voyageur de commerce et celui des N reines et rapporte les bons résultats obtenus.

Le chapitre 5 décrit en premier lieu la démarche suivie pour effectuer différents types de recalage d'images multimodales en se basant sur l'information mutuelle comme mesure de similarité. On y trouve les résultats remarquables obtenus pour des images de sources diverses et des transformations variées. Le chapitre expose ensuite l'approche adoptée pour la segmentation multiobjectif d'images qui est constituée de deux phases: une phase de décomposition et une phase de fusion qui

utilise un algorithme évolutionnaire quantique remodelé pour prendre en compte l'aspect multiobjectif.

Enfin, dans la conclusion générale est présenté un bilan des recherches entreprises avec la mise en évidence des objectifs atteints et des difficultés rencontrées avant de dégager quelques axes pour des futures recherches.

Chapitre 1

Informatique quantique

1.1. Introduction

Richard Feynman a observé au début des années 80 que certains effets de la mécanique quantique ne peuvent être simulés efficacement sur un ordinateur classique. Cette observation a mené à une idée intéressante : C'est la possibilité du traitement des problèmes informatiques d'une manière plus efficace en utilisant les principes de la mécanique quantique [Ekert et al. 2000] [Rieffel et Polak 2000].

Avec la théorie de la relativité, la mécanique quantique aura été la théorie scientifique la plus révolutionnaire du XXème siècle. Elle nous permet d'accéder au monde de l'infiniment petit peuplé d'atomes, de photons, de neutrinos et d'autres particules aux noms peu habituels. C'est un monde surprenant et étrange qui semble défier la logique et le bon sens.

Un bit classique ne peut comprendre qu'un des deux états: l'état haut et l'état bas. Dans beaucoup d'applications, cela suffit pour la modélisation et la réalisation de solutions convenables. Mais dans des domaines tels que l'optimisation combinatoire et la cryptographie, avec la décomposition en facteurs ou pour les applications actuelles et futures qui nécessitent des calculs parallèles, il montre ses limites au niveau du temps de réponse qui est de l'ordre, selon les algorithmes utilisés et la taille des nombres manipulés, de plusieurs centaines voire milliers d'années. Cela justifie les besoins incessants de puissance de calcul pour pouvoir traiter ces problèmes en un temps raisonnable. L'ordinateur quantique apparaît comme la réponse à ces attentes.

1.2. Fondements physiques de l'informatique quantique

La Physique quantique est la théorie qui décrit le monde microscopique des atomes et des particules. Développée au début du XXe siècle, cette théorie a ouvertement remis en question les lois de la physique classique, héritage d'Isaac Newton et de Galilée. Cette section décrit une célèbre expérience qui explique quelques propriétés de la physique quantique.

Expérience de polarisation des photons

Les photons sont les seules particules que nous pouvons directement observer. L'expérience suivante est simple et peut être exécutée avec l'équipement minimal : Une source lumineuse forte, comme un indicateur de laser, et trois polaroïds (filtres de polarisation).

Une poutre de brillance légère sur un écran de projection. Les filtres A, B et C sont polarisés respectivement : Horizontalement, à 45° et Verticalement, et peuvent être placés de façon à croiser la poutre de lumière. Cette expérience va se dérouler comme suit [Rieffel et Polak 2000] :

Premièrement : Insérer le filtre A : Supposant que la lumière arrivée est aléatoirement polarisée. Donc, l'intensité de la lumière sortante va être la moitié de l'intensité de celle entrante. Les photons sortants sont maintenant polarisés horizontalement.

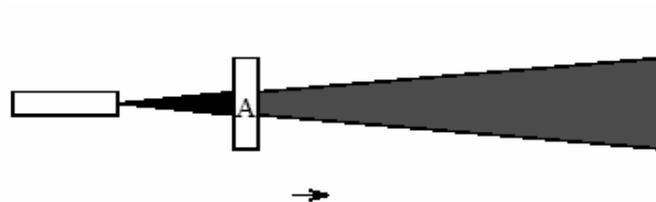


Figure 1.1. L'ajout du filtre A

La fonction du filtre A ne peut pas être expliquée telle que A est une passoire qui laisse passer seulement les photons déjà polarisés horizontalement.

Deuxièmement : quand le filtre C est inséré l'intensité de la sortie se lâche vers le zéro. Aucun des photons polarisés horizontalement ne peut passer à travers le filtre vertical.

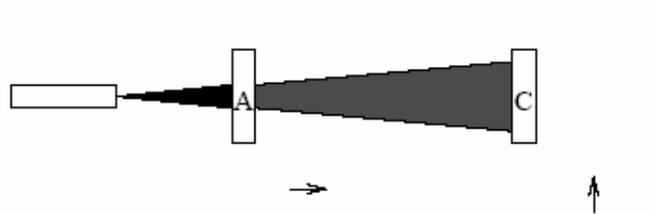


Figure 1.2. L'ajout du filtre C

Finalement : Après l'insertion du filtre B entre A et C, une petite quantité de la lumière sera visible sur l'écran, exactement un huitième (1/8) de la lumière originale.

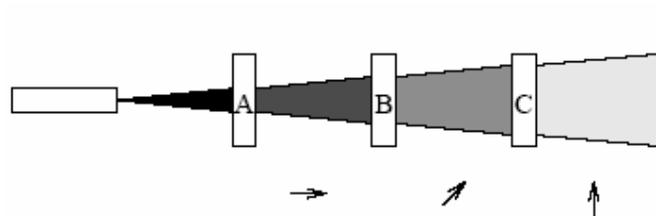


Figure 1.3. L'ajout du filtre B

Explication

L'état de polarisation du photon peut être modélisé par un vecteur pointé dans la direction appropriée. Toute polarisation arbitraire peut être exprimée comme une combinaison linéaire:

$a |\uparrow\rangle + b |\rightarrow\rangle$ des deux vecteurs de base :

$|\uparrow\rangle$: Polarisation horizontale.

$|\rightarrow\rangle$: Polarisation verticale.

Puisque nous sommes intéressés seulement à la direction de la polarisation, le vecteur d'état sera un vecteur unitaire :

$$|a|^2 + |b|^2 = 1 \quad (1.1)$$

Notons que : le choix de la base de représentation est complètement arbitraire: n'importe quelle paire de vecteurs unitaires orthogonaux peut constituer une base. Exemple: $|\uparrow\rangle$ et $|\rightarrow\rangle$.

Le principe de la mesure de la mécanique quantique énonce que tout dispositif de mesure d'un système à deux dimensions a un ensemble associé de vecteurs de base. Et la mesure d'un état transforme cet état à l'un des vecteurs de dispositif de mesure associé. La probabilité que l'état est mesuré selon un vecteur de base $|u\rangle$, est la mesure de l'amplitude du composant de l'état original dans la direction du vecteur de base $|u\rangle$.

Exemple: étant donné un dispositif de mesure de polarisation de photons avec la base associée $\{|\uparrow\rangle, |\rightarrow\rangle\}$. L'état $\psi = a|\uparrow\rangle + b|\rightarrow\rangle$ est mesuré selon $|\uparrow\rangle$ avec une probabilité $|a|^2$ et selon $|\rightarrow\rangle$ avec une probabilité $|b|^2$.

Sachant qu'il peut exister plusieurs dispositifs de mesure, la mesure par deux dispositifs différents va avoir des résultats différents. En outre, la mesure de l'état quantique va changer l'état au résultat de la mesure.

Donc, si la mesure de l'état quantique $\psi = a|\uparrow\rangle + b|\rightarrow\rangle$ résulte en $|\uparrow\rangle$, alors l'état ψ est changé à $|\uparrow\rangle$, et une deuxième mesure relativement à la même base va retourner $|\uparrow\rangle$ avec la probabilité 1. Ainsi que la mesure d'un état change cet état, et il ne sera pas possible de savoir quel est l'état original.

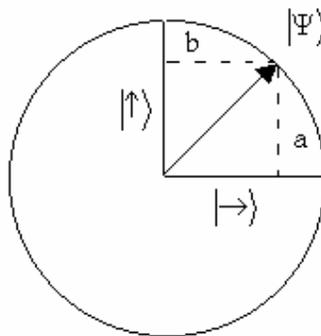


Figure 1.4. La mesure est une projection sur la base choisie.

La mécanique quantique peut expliquer l'expérience précédente de polarisation comme suit:

Un écran mesure l'état quantique des photons relativement à la base. Les filtres travaillent comme des dispositifs de mesure: Le filtre ne laisse passer que les photons qui ne sont pas orthogonaux à sa polarisation.

Exemple : Le filtre A mesure la polarisation des photons relativement au vecteur de base $|\rightarrow\rangle$, correspondant à sa polarisation. Les photons qui passent à travers le filtre A ont tous une polarisation $|\rightarrow\rangle$. Tous les photons reflétés par le filtre ont une polarisation $|\uparrow\rangle$.

Supposant que la source de lumière, produit des photons avec une polarisation aléatoire, le filtre A va mesurer 50 % de tous les photons comme des photons horizontalement polarisés. Ces photons vont passer à travers le filtre et leur état sera $|\rightarrow\rangle$.

Le filtre C va mesurer les photons relativement à $|\uparrow\rangle$. Or l'état $|\rightarrow\rangle = 0|\uparrow\rangle + 1|\rightarrow\rangle$, alors ceci va être projeté sur $|\uparrow\rangle$ avec une probabilité 0, et aucun photon ne va traverser le filtre.

Finalement, le filtre B va mesurer les photons relativement à la base :

$$\{ 1/\sqrt{2} (|\uparrow\rangle + |\rightarrow\rangle), 1/\sqrt{2} (|\uparrow\rangle - |\rightarrow\rangle) \}.$$

C'est ce qu'on écrit comme : $\{ |\nearrow\rangle, |\nwarrow\rangle \}$ tel que :

$$|\rightarrow\rangle = 1/\sqrt{2} (|\nearrow\rangle - |\nwarrow\rangle) \tag{1.2}$$

$$|\uparrow\rangle = 1/\sqrt{2} (|\nearrow\rangle + |\nwarrow\rangle) \tag{1.3}$$

Donc les photons qui sont mesurés selon $|\nearrow\rangle$ passent à travers le filtre.

Les photons qui passent par A avec l'état $|\rightarrow\rangle$ vont être mesurés par B comme $|\nearrow\rangle$ avec une probabilité de 1/2, et alors 50 % des photons passés par A vont passer par B et dans l'état $|\nearrow\rangle$. Et de la même façon les photons seront mesurés par le filtre C comme $|\uparrow\rangle$ avec une probabilité de 1/2. Donc seulement 1/8 (un huitième) des photons originaux vont passer à travers la séquence des filtres : A, B et C.

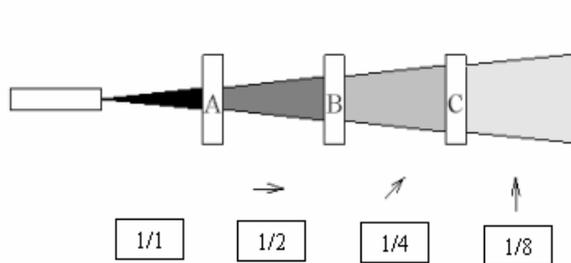


Figure 1.5. L'effet de chaque filtre sur la portion de lumière observée

1.3. Concepts de base

1.3.1. Espace d'état et notation de Dirac

Les physiciens, en théorie quantique, utilisent souvent une notation commode, inventée par *Dirac*, pour décrire les espaces d'états quantiques et les transformations agissant sur eux, cette notation est appelée *bra/ket*. Bra/ket vient du mot anglais "bracket", qui veut dire "crochet", Dirac baptisa dans [Dirac 1958] **bra** et **ket** respectivement les entités $|a\rangle$ et $\langle b|$.

ket : $|a\rangle$ devient $|\ \rangle$: vecteur colonne, typiquement utilisé pour décrire un état quantique.

Bra : $\langle b|$ devient $\langle \ |$: transposée conjuguée de $|\ \rangle$.

Par exemple, la base orthonormée $\{|0\rangle, |1\rangle\}$ peut être exprimé en tant que $\{(1,0)^T, (0,1)^T\}$. N'importe quelle combinaison linéaire complexe de $|0\rangle$ et $|1\rangle$, $a|0\rangle + b|1\rangle$, peut être écrite $(a, b)^T$.

Notons bien que le choix de l'ordre des vecteurs de base est arbitraire. C'est à dire qu'on peut très bien représenter $|0\rangle$ par $(0,1)^T$ et $|1\rangle$ par $(1,0)^T$.

La combinaison de $|x\rangle$ et de $|y\rangle$ en $\langle x| |y\rangle$, aussi écrite comme $\langle x|y\rangle$, dénote le produit *interne* des deux vecteurs.

NB: $|0\rangle$ est un vecteur unitaire ce qui implique que : $\langle 0|0\rangle=1$

$|0\rangle$ et $|1\rangle$ sont orthogonaux ce qui implique que : $\langle 0|1\rangle=0$

La notation $|x\rangle \langle y|$ dénote le produit externe des deux vecteurs. Par exemple, $|0\rangle \langle 1|$ est la transformation suivante:

$|0\rangle$ devient $(0, 0)^T$

$$|0\rangle \langle 1| |0\rangle = |0\rangle \langle 1|0\rangle = 0|0\rangle = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$|0\rangle$ devient $|1\rangle$

$$|0\rangle \langle 1| |1\rangle = |0\rangle \langle 1|1\rangle = |0\rangle$$

D'une manière équivalente, la transformation $|0\rangle \langle 1|$ peut être écrite dans une forme matricielle où :

$|0\rangle = (1, 0)^T$, $\langle 0| = (1, 0)$, $|1\rangle = (0, 1)^T$ et $\langle 1| = (0, 1)$ puis :

$$|0\rangle \langle 1| = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

Cette notation nous donne une manière commode pour spécifier des transformations sur des états quantiques en termes de ce qui arrive aux vecteurs de base. Par exemple la transformation qui échange $|0\rangle$ et $|1\rangle$ est donnée par la matrice :

$$X = |0\rangle \langle 1| + |1\rangle \langle 0|.$$

1.3.2. Bit quantique

L'élément de base de l'informatique quantique est le « bit quantique » ou qubit qui correspond à un système basé sur 2 niveaux correspondant aux états fondamentaux 0 et 1. Cependant, à la différence du bit classique qui ne peut être que dans un des deux états à la fois, le bit quantique peut être dans les deux états en même temps, ou plus précisément dans une superposition des 2 états de base 0 et 1.

Un qubit est un système physique basé sur deux valeurs logiques distinctes : 0 et 1, ou « oui » et « non ». Ces deux valeurs représentent les états de base notés par : $|0\rangle$ et $|1\rangle$. L'état du système, qui est une superposition des deux états fondamentaux, est donnée par:

$$|\varphi\rangle = a|0\rangle + b|1\rangle \quad (1.4)$$

L'observation de cet état va donner la valeur 0 avec une probabilité égale à $|a|^2$ et la valeur 1 avec une probabilité égale à $|b|^2$, sachant que a et b sont deux nombres complexes appelés amplitudes et vérifiant la contrainte: $|a|^2 + |b|^2 = 1$.

1.3.3. Registre quantique

Un registre quantique est un ensemble de qubits, une superposition arbitraire de n qubits. L'état d'un registre quantique peut être donné par:

$$\Psi = \sum_{x=0}^{2^n-1} C_x |X\rangle \quad (1.5)$$

Les amplitudes C_x gouvernant les probabilités des différents états satisfont la propriété:

$$\sum_{x=0}^{2^n-1} |C_x|^2 = 1 \quad (1.6)$$

Dans un système classique un registre de n bits représente une parmi les 2^n valeurs possibles, tandis qu'un registre quantique représente toutes les 2^n valeurs en même temps [Deutsch et Ekert 1998].

1.3.4. Les cinq principes de l'informatique quantique

L'informatique quantique est basée sur les cinq principes suivants qui sont inspirés de la mécanique quantique: la superposition d'états, l'interférence, l'enchevêtrement, le non déterminisme et la non duplication [Magniez 2002].

- Superposition d'états

Un bit quantique peut être dans l'état 0 ou 1, mais également les 2 en même temps. Donc un registre de n qubit pourra contenir 2^n valeurs distinctes. Une conséquence importante de cette superposition est qu'un système quantique de n qubit peut manipuler 2^n nombres en parallèle.

- Interférence

C'est l'action qui permet d'augmenter ou diminuer la probabilité d'avoir un état.

Il existe deux types d'interférence: Interférence Constructive et interférence destructive; l'interférence constructive permet d'augmenter la probabilité d'obtenir un état, alors que l'interférence destructive permet de diminuer la probabilité d'obtenir un état.

- Enchevêtrement ou intrication

Deux bits quantiques sont dits enchevêtrés si leur destins sont entièrement dépendants l'un de l'autre. Donc toute variation de l'un affecte simultanément l'autre.

- Non déterminisme et non duplication

La mécanique quantique est non déterministe : le hasard quantique est authentique et irréductible, c'est une loi fondamentale. La copie des qubits n'est pas toujours possible et un état inconnu ne peut pas être dupliqué.

- La réversibilité

L'évolution dans le temps d'un système quantique isolé, est décrite par une transformation unitaire. En d'autres termes, ça veut dire que l'aboutissement d'une condition à une action peut aussi mener à l'action inverse dans des circonstances inverse. Pour préserver la superposition et l'intrication nous avons besoin des évolutions unitaires. Une conséquence importante est que les opérations doivent être réversibles.

Les principes et les propriétés quantiques inspirés de la physique ne peuvent être exploités qu'à travers un formalisme et un ensemble de portes logiques universelles, qui permettent de réaliser des calculs et des programmes sur une vraie machine quantique. Ces portes quantiques donnent accès aux différents calculs qui exploitent le parallélisme espéré du monde quantique.

1.3.5. Calcul quantique

Il s'agit d'utiliser des processus quantiques pour réaliser des opérations logiques. Les unités de calcul sont des qubits évoluant dans des superpositions d'états [Rieffel et Polak 2000].

L'état d'un qubit (quantum bit) est un vecteur dans un espace vectoriel à deux dimensions.

Pour manipuler ces données quantiques (représentées sous forme de bits quantiques), on distingue deux types d'opérations: opérations simples dites portes quantiques simples et les successions d'opérations quantiques appelées circuits quantiques.

1.3.5.1. Portes quantiques

Portes quantiques simples

Les exemples suivants montrent certaines transformations de base, il s'agit de portes simples applicables sur un seul bit quantique. Ces portes peuvent être représentées par des matrices unitaires 2x2 [Rieffel et Polak 2000].

I : La transformation Identité

$$\begin{array}{l} |0\rangle \longrightarrow |0\rangle \\ |1\rangle \longrightarrow |1\rangle \end{array} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

X : La négation

$$\begin{array}{l} |0\rangle \longrightarrow |1\rangle \\ |1\rangle \longrightarrow |0\rangle \end{array} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Z : La transformation du décalage en phase

$$\begin{array}{l} |0\rangle \longrightarrow |0\rangle \\ |1\rangle \longrightarrow -|1\rangle \end{array} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Y : La combinaison de X et Z. On a $Y = Z X$

$$\begin{array}{l} |0\rangle \longrightarrow -|1\rangle \\ |1\rangle \longrightarrow |0\rangle \end{array} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

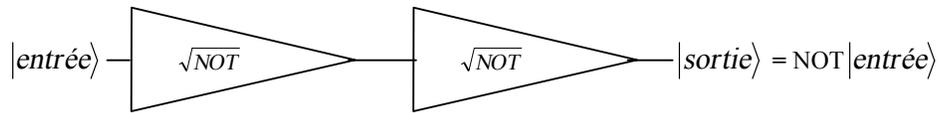
Porte Racine-carré-du-NOT (Opérateur \sqrt{NOT})

C'est la porte quantique la plus simple, notée \sqrt{NOT} . Elle constitue une opération unitaire (agissant sur un seul qubit), Deux portes successives décrivant l'opérateur classique NOT, notée U_{NOT} il agit sur les deux états observable d'un qubit.

$$U_{NOT} |0\rangle = |1\rangle \quad , \quad U_{NOT} |1\rangle = |0\rangle$$

Où

$$U_{\sqrt{NOT}} U_{\sqrt{NOT}} |0\rangle = |1\rangle \quad , \quad U_{\sqrt{NOT}} U_{\sqrt{NOT}} |1\rangle = |0\rangle$$



Cette porte n'a pas d'homologue dans le monde classique du fait qu'il n'y a pas un état entre $|0\rangle$ et $|1\rangle$

$$U_{NOT} |0\rangle = \left(\frac{1+i}{2}\right)|0\rangle + \left(\frac{1-i}{2}\right)|1\rangle \tag{1.7}$$

$$U_{NOT} |1\rangle = \left(\frac{1-i}{2}\right)|1\rangle + \left(\frac{1+i}{2}\right)|0\rangle \tag{1.8}$$

Controlled-Not

Cette porte provoque des opérations contrôlées. Le controlled-Not (CNot) opère sur deux qubits de la manière suivante : si le premier qubit est à 1, le deuxième qubit est modifié, sinon rien ne sera fait. Cette porte peut être représentée par :

$$\begin{array}{l}
 \text{CNot :} \quad |00\rangle \rightarrow |00\rangle \\
 \quad \quad \quad |01\rangle \rightarrow |01\rangle \\
 \quad \quad \quad |10\rangle \rightarrow |11\rangle \\
 \quad \quad \quad |11\rangle \rightarrow |10\rangle
 \end{array}
 \quad
 \begin{pmatrix}
 1 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 1 \\
 0 & 0 & 1 & 0
 \end{pmatrix}$$

Dans le cas général, on peut avoir plusieurs bits de contrôle. De la même manière le Non Contrôlé-Contrôlé (Controlled-Controlled-Not) qui fait la négation du dernier bit parmi trois si et seulement si les deux premiers sont tous les deux à 1.

La porte Walsh-Hadamard

Une autre transformation très importante à un seul qubit est la transformation de Hadamard. Cette porte effectue une rotation de l'état d'un qubit :

$$U_{WH}|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \quad (1.9)$$

$$U_{WH}|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad (1.10)$$

Cette porte est généralement utilisée pour générer des superpositions. Elle permet de créer, à partir de n qubits à l'état initial $|0\rangle$, une superposition équiprobable des 2^n valeurs valeur possible :

$$\begin{aligned} U_{WH}|0\rangle \otimes U_{WH}|0\rangle \otimes \dots \otimes U_{WH}|0\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \dots \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ &= \frac{1}{\sqrt{2^n}}(|00\dots0\rangle + |00\dots1\rangle + \dots + |11\dots1\rangle) \\ &= \frac{1}{\sqrt{2^n}}(|0\rangle + |1\rangle + |2\rangle + \dots + |2^n - 1\rangle) \end{aligned}$$

1.3.5.2. Circuits quantiques

Un circuit quantique est une combinaison de deux ou plusieurs portes quantiques permettant d'effectuer un traitement plus compliqué sur un système quantique.

Représentation graphique

Il est très utile d'avoir une représentation graphique des transformations d'états quantiques et spécialement quand plusieurs transformations sont combinées. Le Controlled-Not C_{Not} est représenté par un circuit de la forme [Rieffel at Polak 2000].

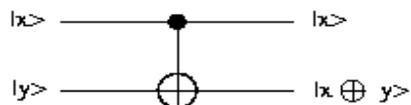


Figure 1.6. Circuit quantique représentant le Controlled-Not

Le cercle noir indique le bit de contrôle et le cercle ouvert indique la négation conditionnelle du bit sujet.

En général, on peut avoir plusieurs bits de contrôle. De la même manière le Non Contrôlé- Contrôlé (Controlled-Controlled-Not) qui fait la négation du bit dernier des trois si et seulement si les deux premiers sont tous les deux à 1. Le Controlled-Controlled-Not a la représentation graphique suivante :

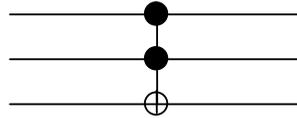


Figure 1.7. Controlled-Controlled-Not

Les opérations à un seul bit sont graphiquement représentées par une boîte étiquetée comme par exemple :

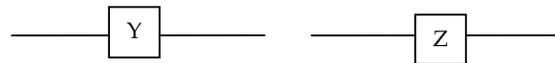


Figure 1.8. Opérations à une seule entrée

Exemple de circuit quantique

Le circuit quantique dans la figure 1.9 est un exemple où une succession de trois opérations basiques est appliquée sur un système à deux qubit pour réaliser une opération plus complexe. Les opérations quantiques simples sont notées 1, 2 et 3. $|\phi+\rangle$ est l'état final du système qui indique le résultat du calcul [Magniez 2002].

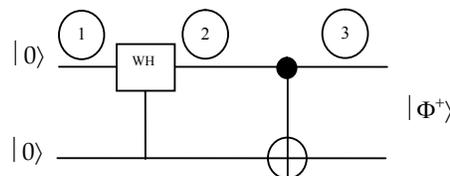


Figure 1.9. Circuit quantique

1. $|0\rangle \otimes |0\rangle = |00\rangle$
2. $(U_{WH} |0\rangle) \otimes |0\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}} \otimes |0\rangle = \frac{1}{\sqrt{2}} (|00\rangle+|10\rangle)$
3. $\frac{1}{\sqrt{2}} (|00\rangle+|11\rangle) = |\Psi\rangle$

1.3.6. Algorithmes quantiques

Un algorithme quantique est une succession d'application d'opérations quantiques sur des systèmes quantiques. Ces opérations quantiques sont soit des portes logiques quantiques simples, soit des circuits quantiques. Les algorithmes quantiques offrent une complexité algorithmique nettement inférieure à celle des algorithmes classiques, et cela grâce au concept de superposition d'états. Cette diminution de complexité a attiré l'attention de beaucoup de chercheurs qui ont fourni de grands efforts pour concevoir des algorithmes quantiques résolvant des problèmes connus du monde réel, citant l'algorithme de Shor (1994) pour la factorisation de grands nombres, l'algorithme de Grover (1996) pour la recherche dans une base de données non triée et l'algorithme de Kitaev pour la factorisation [Magniez 2002].

1.4. Algorithmes quantiques célèbres

Dans cette partie sont illustrés trois Algorithmes célèbres: le fameux algorithme de Shor, l'algorithme de Grover et celui de Deutsch.

1.4.1. Algorithme de Shor

Une fonction unidirectionnelle est une fonction qui peut être évaluée dans un sens mais le sens contraire de cette fonction est quasiment impossible. Un exemple des fonctions unidirectionnelles est l'opération de multiplication des nombres premiers. Les fonctions unidirectionnelles sont des fonctions clés pour la cryptographie. Le meilleur algorithme classique de factorisation d'un nombre de N bits requiert $O(\exp((64/9)^{1/3} N^{1/3} (\ln(N))^{2/3}))$ opérations. D'après Shor, un algorithme quantique ne nécessite que $O(N^2 \ln N \ln(\ln N))$ opérations [Shor 1994].

Problème: factoriser un nombre N arbitraire qui ne doit pas être une puissance de 2.

Soit x un nombre choisie au hasard , $x < N$.

Si x n'est pas premier avec N alors on a un diviseur de N .

Sinon (x est premier avec N), on définit l'ordre de x comme le plus petit entier r tel que : $x^r = 1[N]$, donc $f(a) = x^a [N]$ est une fonction périodique de période r , si on connaît r , alors on a environ 50% de chances pour que r soit pair et $x^{r/2} \neq -1 [N]$. Alors le PGCD ($N, x^{r/2} \pm 1$) est un facteur non trivial de N .

~ Petite application numérique :

Le premier nombre non premier, non paire, non puissance de 2 est 15. On va prendre $N = 15$, et $x = 7$ au hasard puis on va calculer $7^a [15]$

a	$7^a [15]$
1	7
2	4
3	13
4	1

Donc l'ordre r de 7 est 4; $7^4 = 1[15]$.

Le PGCD de 4+1 et 15 est un facteur de 15, PGCD(4+1, 15) = 5

Le PGCD de 4-1 et 15 est un facteur de 15, PGCD(4-1, 15) = 3

En résultats on a donc $15 = 5 * 3$.

~ La transformé de Fourier discrète

La transformée de Fourier est en général une application du domaine de temps vers le domaine des fréquences. C'est une application des fonctions périodiques de période r vers les fonctions qui ont des valeurs différentes de zéro seulement aux multiples de la fréquence $2\pi/r$. La transformé de Fourier discrète (TFD en anglais DFT) d'une fonction périodique d'une période r est une fonction concentrée près des multiples de N/r . Si la période r est un diviseur de N alors le résultat est une fonction qui a des valeurs différentes de zéro seulement aux multiples N/r , sinon le résultat rapprochera ce comportement et il y aura des limites non nulles aux entiers près des multiples de N/r .

Quand on applique une TFD sur un état $|a\rangle$ le résultat sera :

$$\text{TFD}(|a\rangle) = \frac{1}{\sqrt{r}} \sum_{c=0}^{r-1} |c\rangle \cdot e^{2\pi i ac/r} \quad (1.11)$$

~ **Les étapes de l'algorithme:**

Seules les étapes 4, 5, 6, 7, 8 et 9 sont sensées être effectuées sur un ordinateur quantique, les autres sont effectuées sur un ordinateur classique.

- *Etape 1*

Déterminer si le nombre N est premier, pair ou un exposant d'un nombre premier. Si c'est le cas on n'appliquera pas l'algorithme de Shor. Il existe des méthodes classiques efficaces pour résoudre le problème.

- *Etape 2*

Choisir un entier q qui est une puissance de 2 tel que $N^2 \leq q \leq 2N^2$.

- *Etape 3*

Choisir un nombre aléatoire x qui est co-premier avec N (deux nombre sont dit co-premiers si leur plus grand diviseur commun est 1). Il y a des méthodes classiques efficaces pour choisir un tel nombre x .

- *Etape 4*

Créer un registre quantique et le partitionner en deux parties, registre1 et registre2. Cela nous permet de représenter l'état de l'ordinateur quantique par Reg1 et Reg2. Reg1 doit avoir un nombre de qubits suffisant pour représenter les entiers $0 \dots q-1$. Reg2 doit avoir un nombre de qubits suffisant pour représenter les entiers $0 \dots N-1$.

- *Etape 5*

Créer une superposition des entiers de 0 à $q-1$ de même amplitude sur Reg1, et charger Reg2 par des 0 pour avoir l'état quantique :

$$\frac{1}{\sqrt{q}} \sum_{a=0}^{q-1} |a, 0\rangle \quad (1.12)$$

- *Etape 6*

Appliquer La transformation $x^a \text{ mod } N$ pour chaque nombre présent dans Reg1 et ranger le résultat dans Reg2. Grâce au parallélisme quantique ceci sera accompli en

une seule étape. L'ordinateur quantique va seulement calculer $x^{|a\rangle} \bmod N$, où $|a\rangle$ est la superposition de tous les états créés dans l'étape 5. L'état de la mémoire quantique après cette étape sera :

$$\frac{1}{\sqrt{q}} \sum_{a=0}^{q-1} |a, X^a \bmod N\rangle \quad (1.13)$$

- Etape 7

Mesurer Reg2 et observer une certaine valeur k . Ceci aura un effet secondaire de délimiter Reg1 en une superposition équiprobable de toutes les valeurs a comprises entre 0 et $q-1$ vérifiant la contrainte $x^a \bmod N = k$. L'état quantique après cette étape devient:

$$\frac{1}{\sqrt{\|A\|}} \sum_{a \in A} |a, k\rangle \quad (1.14)$$

Où A est l'ensemble des a tel que $x^a \bmod N = k$, et $\|A\|$ est le nombre d'éléments dans cet ensemble.

- Etape 8

Calculer la transformée de Fourier discrète sur Reg1. Cette étape est aussi effectuée en un seul coup. On aura l'état:

$$\frac{1}{\sqrt{\|A\|}} \sum_{a \in A} \frac{1}{\sqrt{q}} \sum_{c=0}^{q-1} |c, k\rangle \cdot e^{2\pi i ac/q} \quad (1.15)$$

- Etape 9

Mesurer l'état de Reg1. Soit m le résultat de cette mesure. Cet entier m a une grande probabilité d'être un multiple de q/r , où r est la période recherchée.

- Etape 10

En se basant sur la connaissance de m et q , il y a plusieurs manières pour calculer r .

- Etape 11

Une fois r est obtenu, un facteur de N peut être calculé par le calcul de $PGCD(x^{r/2} + 1, N)$ et $PGCD(x^{r/2} - 1, N)$. Si on trouve un facteur de N alors on arrête, sinon on reprend l'algorithme à partir de l'étape 4.

Remarque: Shor a montré que peu de répétitions de cet algorithme suffisent pour trouver un facteur de N . Il se peut que l'algorithme échoue pour les raisons suivantes:

1/ La transformée de Fourier est mesurée à 0 dans l'étape 9, ce qui implique l'impossibilité de l'étape 10.

2/ On peut trouver 1 ou N comme facteur ce qui est trivial.

1.4.2. Algorithme de Grover

Avec un ordinateur classique, si une liste non triée contient 10 000 objets il faudra en moyenne 5 000 étapes pour repérer un objet particulier et 10 000 au pire des cas. La complexité est donc de $O(N)$. Avec l'algorithme de recherche de Grover [Grover 1996], il faut seulement 100 étapes, une complexité de $O(\sqrt{N})$ seulement. Ceci est rendu possible en développant une série d'instructions à exécuter sur un ordinateur quantique

Malgré que cet algorithme n'offre qu'une accélération quadratique (moins significative que l'accélération exponentielle qu'offre l'algorithme de Shor), l'algorithme de Grover reste très utile car il fournit une technique qui peut être exploitée pour résoudre des problèmes NP complets.

~ Explication

Au départ il faut préparer un registre quantique dans une superposition de tous les états possibles (ou objets) de la liste. On y applique ensuite les instructions quantiques. Cet algorithme peut être décrit en terme d'ondes. Rappelons qu'un objet quantique est une particule et une onde. Comme toute onde elle peut interférer avec d'autres ondes. Ces interférences peuvent être constructives (les ondes se renforcent, comme lorsque deux crêtes de vagues se rencontrent) ou destructives. Tout le secret de l'algorithme de Grover se trouve là : les bonnes solutions de la recherche se renforcent et émergent de l'ensemble tandis que les autres s'annulent et sont rejetées.

~ L'oracle quantique

Définissons la fonction $f: \{0, \dots, N-1\} \rightarrow \{0, 1\}$ telle que:

$f(i)=1$ si i est l'élément recherché (i_0), sinon $f(i)=0$.

Cette fonction, utilisée dans l'algorithme classique, permet dans l'algorithme quantique d'assumer qu'il est possible de construire un opérateur (transformation) unitaire linéaire pour calculer f , U_f

$$U_f(|i\rangle|j\rangle) = |i\rangle|j \oplus f(i)\rangle \quad (1.16)$$

U_f est dite *oracle*. Dans l'équation précédente, $|i\rangle$ représente l'état du premier registre. Donc $i \in \{0, \dots, 2^n - 1\}$, $|j\rangle$ représente l'état du deuxième registre, $j \in \{0, 1\}$.

~ Procédure itérative

Pour la recherche dans une base de donnée de N éléments, Grover utilise l'opérateur D défini par :

$$D = \begin{pmatrix} \frac{2}{N} & \dots & \frac{2}{N} \\ \frac{2}{N} & \frac{2}{N} & \dots & \frac{2}{N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{2}{N} & \frac{2}{N} & \dots & \frac{2}{N} \end{pmatrix}$$

Cet opérateur possède deux propriétés :

- D est unitaire et peut être réalisé efficacement.
- Il peut être vu comme une inversion par rapport à la moyenne.

Quand on applique l'opérateur D sur un vecteur $|a\rangle$, les amplitudes supérieures à la moyenne seront réduites, respectivement celles qui sont inférieures à la moyenne seront augmentées.

Cet algorithme effectue deux opérations à chaque itération :

- a** . Consulter l'oracle U_f .
- b** . Appliquer l'inversion autour de la moyenne.

Après l'opération **a** on aura un résultat de N éléments. On prépare donc l'état initial comme une superposition d'égale probabilité ($\frac{1}{\sqrt{N}}$) de ces N éléments. Après, l'oracle quantique (U_f) reconnaît et marque la solution i_0 , $f(i_0) = 1$, avec un décalage

de phase. L'opérateur D augmente ensuite l'amplitude de l'élément i_0 et diminue celles des autres.

La répétition de ce processus avant de faire une mesure augmente la probabilité de mesurer i_0 .

Nb. La répétition de l'itération n'augmente pas toujours la chance de mesurer la bonne solution. L'amplitude de la solution marquée devient supérieure et inférieure par un cycle. Si on ne s'arrête pas au bon moment nous pouvons ne pas avoir une bonne chance de mesurer la bonne solution. Donc, pour obtenir des résultats utiles des algorithmes quantiques appliqués répétitivement, il faut savoir quand s'arrêter.

1.4.3. Algorithme de Deutsch - Jozsa

Nous avons une fonction $f : \{0,1\} \rightarrow \{0,1\}$. Il est ainsi possible de construire les quatre fonctions différentes agissant sur l'ensemble $\{0,1\}$.

$$f_1(x)=0$$

$$f_2(x)=1$$

$$f_3(x)=x$$

$$f_4(x)=\text{NOT}(x)$$

Problème:

On veut savoir si la fonction f est constante ($f_{1,2}$) ou balancée ($f_{3,4}$) ?

La réponse à cette question est le but de cet algorithme.

Pour un algorithme classique, on doit appliquer la fonction $f(x)$ deux fois (une fois avec 0 et une autre avec 1) pour avoir la réponse.

Sur une machine quantique, on ne fait intervenir la fonction $f(x)$ qu'une seule fois selon l'algorithme suivant : [Deutsch et Jozsa 1992]

Etape 1

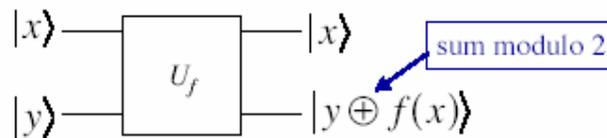
Préparation de l'état initial : $|0\rangle \otimes |0\rangle$

Etape 2

Rotation de chaque état initial livrant ainsi un état de superposition en appliquant la porte de Walsh-Hadamard.

Etape 3

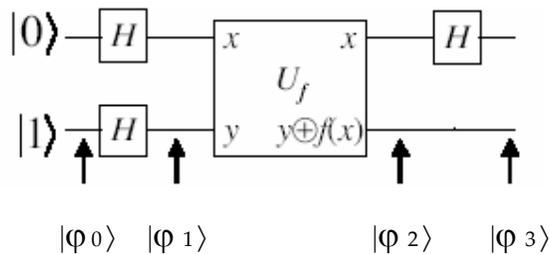
Appel de la fonction $f(x)$ en appliquant la transformation U_f qui a la représentation graphique suivante :



Etape 4

Rotation inverse de celle décrite à l'étape 2 pour trouver l'état final du système.

~Explication détaillée de l'algorithme



Etape 1

Nous avons en entrée l'état initial: $|\varphi_0\rangle = |01\rangle$

Etape 2

En appliquant la porte de Walsh-Hadamard sur chaque état, on obtient :

$$|\varphi_1\rangle = \left[\frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \right] \left[\frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \right] \tag{1.17}$$

$$|\varphi_1\rangle = \frac{1}{2} \sum_{x=0}^1 |x\rangle (|0\rangle - |1\rangle) \tag{1.18}$$

Etape 3

En faisant appelle à la transformation U_f qui agisse sur l'état $|\varphi_1\rangle$ Comme suit :

$$|\varphi_2\rangle = U_f |\varphi_1\rangle$$

$$\begin{aligned}
 |\varphi_2\rangle &= \mathbf{U}_f \left[\frac{1}{2} (\mathbf{C}|0\rangle - |1\rangle) + |1\rangle \otimes (|0\rangle - |1\rangle) \right] \\
 &= \frac{1}{2} [\mathbf{U}_f |0\rangle \otimes (|0\rangle - |1\rangle) + \mathbf{U}_f |1\rangle \otimes (|0\rangle - |1\rangle)] \\
 &= \frac{1}{2} [\mathbf{U}_f (|00\rangle - |01\rangle) + \mathbf{U}_f (|10\rangle - |11\rangle)] \\
 &= \frac{1}{2} [\mathbf{U}_f |00\rangle - \mathbf{U}_f |01\rangle + \mathbf{U}_f |10\rangle - \mathbf{U}_f |11\rangle] \\
 &= \frac{1}{2} [\mathbf{U}_f |0\rangle \otimes |0\rangle - \mathbf{U}_f |0\rangle \otimes |1\rangle - \mathbf{U}_f |1\rangle \otimes |0\rangle - \mathbf{U}_f |1\rangle \otimes |1\rangle] \\
 &= \frac{1}{2} [|0\rangle \otimes (|0 \oplus f(0)\rangle - |1 \oplus f(0)\rangle) + |1\rangle \otimes (|0 \oplus f(1)\rangle - |1 \oplus f(1)\rangle)] \\
 &= \frac{1}{2} \sum_{x=0}^1 |x\rangle \otimes (|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle) \tag{1.19}
 \end{aligned}$$

Remarquons que :

$$\bullet |0\rangle - |1\rangle \quad \text{si } f(x) = 0$$

$$|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle =$$

$$\bullet |1\rangle - |0\rangle \quad \text{si } f(x) = 1$$

On a donc :

$$|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle = (-1)^{f(x)} (|0\rangle - |1\rangle)$$

Alors:

$$|\varphi_2\rangle = \frac{1}{2} \sum_{x=0}^1 |x\rangle \otimes (-1)^{f(x)} (|0\rangle - |1\rangle)$$

On résume:

$$|\varphi_2\rangle = \mathbf{U}_f |\varphi_1\rangle = \frac{1}{2} \{ |0\rangle \otimes (|0\rangle - |1\rangle) (-1)^{f(0)} + |1\rangle \otimes (|0\rangle - |1\rangle) (-1)^{f(1)} \} \tag{1.20}$$

Nous avons deux cas :

$$f(0) = f(1) : |\varphi_2\rangle = \pm \frac{1}{2} (|0\rangle + |1\rangle) \otimes (|0\rangle - |1\rangle)$$

$$f(0) \neq f(1) : |\varphi_2\rangle = \pm \frac{1}{2} (|0\rangle - |1\rangle) \otimes (|0\rangle - |1\rangle)$$

Etape 4

Rotation inverse en appliquant la porte de Walsh-Hadamard sur le premier qubit.

$$f(0) = f(1) : |\varphi_3\rangle = \pm \frac{1}{\sqrt{2}} |0\rangle \otimes (|0\rangle - |1\rangle)$$

$$f(0) \neq f(1) : |\varphi_3\rangle = \pm \frac{1}{\sqrt{2}} |1\rangle \otimes (|0\rangle - |1\rangle)$$

Résultat:

Selon le premier qubit :

Si $|0\rangle$ alors f est constante

Si $|1\rangle$ alors f est balancée

1.5. Ordinateurs quantiques

1.5.1. Un peu d'histoire

Dans les années 1970 et 80, les premiers ordinateurs quantiques naissent par retournement dans l'esprit de physiciens tels que Richard Feynman, Paul Benioff, David Deutsch ou Charles H. Bennett. L'idée de Feynman était : « Au lieu de nous plaindre que la simulation des phénomènes quantiques demande des puissances énormes à nos ordinateurs actuels, utilisons la puissance de calcul des phénomènes quantiques pour faire plus puissant que nos ordinateurs actuels ».

Longtemps les physiciens ont douté que les ordinateurs quantiques utilisables puissent exister, et même qu'on puisse en faire quelque chose de viable s'ils existaient. Mais:

en 1994, Peter Shor, chercheur chez AT&T, montre qu'il est possible de factoriser des grands nombres dans un temps raisonnable à l'aide d'un ordinateur quantique. Cette découverte débloque brusquement des crédits ;

en 1996, Lov Grover, invente un algorithme basé sur les principes quantiques permettant de trouver une entrée dans une base de données non triée;

en 1998, IBM est le premier à présenter un ordinateur quantique à 2 qubits ;

en 1999, l'équipe d'IBM utilise l'algorithme de Grover sur un calculateur de 3 qubits et battent leur record l'année suivante avec un ordinateur de 5 qubits ;

le 19 décembre 2001, IBM crée un ordinateur quantique de 7 qubits et factorise le nombre 156 grâce à l'algorithme de Shor. Ces ordinateurs à 7 qubits sont bâtis autour

de molécules de chloroforme et leur durée de vie utile ne dépasse pas quelques minutes. On parle par dérision de wetware.

En 2006, Seth Lloyd, professeur au Massachusetts Institute of Technology (MIT), pionnier du calcul quantique et auteur du livre *Hacking the universe*, mentionne dans le numéro d'août 2006 de la revue *Technology Review* (page 24) l'existence d'ordinateurs quantiques à 12 qubits.

L'institut de traitement de l'information quantique de l'université d'Ulm en Allemagne présente en avril 2006 la première micro puce européenne linéaire tridimensionnelle qui piège plusieurs atomes ionisés Ca^+ de manière isolée.

Le 14 décembre 2007, L'Université du Queensland annonce travailler sur des circuits quantiques optiques

La société D-Wave a annoncé officiellement le 13 février 2007 avoir réalisé un ordinateur quantique à base solide de 16 qubits. Ces machines utiliseraient une puce nommée Europa qui fonctionne uniquement en milieu cryogénique. Reflétant le sentiment d'une partie de la communauté scientifique, *Scientific American* reste réservé. En décembre 2007 et d'après le site même du constructeur, les seules nouvelles concernant D-wave depuis février auront été sa participation à une conférence sur le calcul massif et la démonstration alléguée d'une machine à 28 qubits en novembre, commentée en détail par *Tom's Hardware* en juillet 2008. La compagnie affirme alors maintenir ses objectifs de 512 qubits au second trimestre 2008 et 1024 qubits fin 2008, et assuré que la commercialisation des calculateurs quantiques était bien "une question d'années et non de décennies"; elle a mentionné aussi son intention de rendre son calculateur et les capacités de corrélation très rapides de celui-ci accessibles à des chercheurs via l'Internet. Le 14 avril 2009, elle annonce en fin de compte une puce de 128 qubits.

1.5.2. Réalisation de l'ordinateur quantique

Les approches expérimentales proposées pour la réalisation d'un ordinateur quantique sont nombreuses et variées. Dans un premier temps, il s'agit de réaliser une porte logique élémentaire, c'est-à-dire d'arriver à coupler deux systèmes

quantiques de façon suffisamment forte pour que l'état de l'un puisse être modifié par l'état de l'autre avant que la décohérence n'entre en jeu. Le phénomène de décohérence, c'est-à-dire de perte des effets quantiques sur le long terme, est le principal frein au développement de l'ordinateur quantique. L'étape suivante consiste à augmenter le nombre de qubits en interaction pour construire un vrai ordinateur. Des propositions ont été faites pour cette dernière étape, mais la plupart soulèvent des difficultés technologiques considérables. Les principales propositions se rapportant à différents domaines de la physique sont décrites ci-dessous [Waldner 2006], [Stolze et Suter 2008].

~ **Les ions piégés** : Un ion piégé peut être vu comme une particule plongée dans un potentiel (problème hydrogéoïde). Il est alors possible de considérer deux configurations différentes de l'ion, donc deux niveaux d'énergie distincts. A l'aide d'impulsions laser, nous pouvons modifier l'état de chaque ion et l'intrication des états. Par cette technique, le groupe de Boulle (National Institute of Standards and Technology) a obtenu une intrication de quatre qubits.

~ **Les photons polarisés**: Dans le cas des photons, les deux états du qubit sont deux états de polarisation et les opérations sont réalisées par d'astucieuses techniques d'interférométrie. L'intrication à trois photons a été démontrée. Comme le support de l'information n'est pas solide, l'interférométrie de photons est plutôt envisagée pour la communication et la cryptographie quantique.

~ **La résonance magnétique nucléaire (RMN)** : A l'heure actuelle, le système ayant démontré expérimentalement le calcul quantique le plus sophistiqué (sept opérations logiques sur deux qubits) et qui a réussi à implémenter des algorithmes, notamment celui de Shor pour factoriser le nombre 15 en 2001 et celui de Deutsch-Jozsa avec un code de correction d'erreurs, est la résonance magnétique nucléaire (RMN). Les qubits sont les états de spin nucléaire d'atomes dans une molécule. Chaque spin a sa propre fréquence de résonance. Les spins des différents noyaux de la molécule sont couplés, et la fréquence de résonance d'un spin peut dépendre de l'état des spins voisins. On peut donc ainsi réaliser des transformations conditionnelles des spins, ce qui est l'ingrédient de base des portes quantiques.

~ **L'ordinateur quantique de Kane:** considéré le plus souvent comme une hybridation entre les boîtes quantiques et les ordinateurs fondés sur la résonance magnétique nucléaire. L'ordinateur de Kane est basé sur une grille d'atomes donneurs de phosphore embarqués dans un treillage de silicone pur. Les spins nucléaires et ceux des électrons participent tous au calcul [Kane 1998].

~ **Les qubits solides:** Les ordinateurs actuels étant fondés sur des puces en silicium, il est naturel d'envisager l'utilisation de techniques issues de la physique du solide. C'est le cas pour les circuits électriques supraconducteurs contenant des nanojonctions de Josephson (des barrières à travers lesquelles un courant peut passer par effet tunnel), ou pour les qubits utilisant des états électroniques ou des états de spin dans des boîtes quantiques semi-conductrices.

1.5.3. Principales difficultés

L'un des problèmes les plus sérieux pour le traitement de l'information sur un ordinateur quantique est la décohérence [Rieffel et Polak 2000]. L'interaction incontrôlée des particules représentant les qubits avec l'environnement externe perturbe le fragile état quantique intriqué ce qui engendre des erreurs dans le calcul. Le développement d'algorithmes de correction d'erreurs est devenu essentiel pour remédier à ce problème. Les méthodes de correction d'erreurs sont bien connues dans les systèmes de calcul conventionnels et ont été employées pendant des décennies. Ces méthodes utilisent divers types de redondances pour identifier puis corriger les erreurs. La situation pour les systèmes quantiques est nettement plus compliquée. La principale raison est que dans un système quantique il n'est pas possible de mesurer des valeurs de qubits sans détruire la superposition quantique. Il y avait donc un grand doute que la correction d'erreurs quantique ne serait jamais faisable, surtout en tenant compte de l'impossibilité de cloner ou de copier un état quantique.

Il existe 03 types d'erreurs de base :

- Erreurs de type **X** (Bit flip errors)

$$\alpha|0\rangle + \beta|1\rangle \rightarrow \alpha|1\rangle + \beta|0\rangle$$

- Erreurs de type **Z** (Phase errors)

$$\alpha|0\rangle + \beta|1\rangle \rightarrow \alpha|0\rangle - \beta|1\rangle$$

- Erreurs de type **Y** (Bit flip Phase errors)

$$\alpha|0\rangle + \beta|1\rangle \rightarrow \alpha|1\rangle - \beta|0\rangle$$

Remarquons que : $\mathbf{Y} = \mathbf{XZ}$, donc on doit uniquement corriger 2 des 3 types d'erreurs.

I, **X**, **Y** et **Z** sont appelés **les opérateurs unitaires d'erreurs**, tel que **I** est l'opérateur qui dénote sans-erreurs (No errors), tous ces opérateurs forment une base matricielle.

En général, une erreur **E** est exprimée comme une combinaison linéaire des opérateurs unitaires d'erreurs (**I,X,Y,Z**) comme suit :

$$E = c_1I + c_2X + c_3Y + c_4Z = \sum_i c_i E_i \quad (1.21)$$

c_i : coefficient de pondération. E_i : Opérateur d'erreur.

Un deuxième problème est celui de la gestion de la mémoire. La théorie actuelle envisage de gérer la mémoire d'un ordinateur quantique qubit par qubit ce qui entraîne un coût supplémentaire.

Un facteur très important dans la réalisation d'un ordinateur quantique est l'amélioration du taux d'erreur des portes quantiques. Pour la technique des ions piégés par exemple, le taux d'erreur d'une porte individuelle doit être amélioré au moins 100 fois pour que l'ordinateur quantique puisse fonctionner- ce qui est difficile mais pas inaccessible [Schmidt-Kaler et Grangier 2006].

Les calculs d'architecture estiment qu'un ordinateur quantique utiliserait plus de 99% de ses ressources juste pour gérer sa mémoire et corriger ses erreurs [Schmidt-Kaler et Grangier 2006]. De ce fait un ordinateur quantique universel ne sera pas composé de seulement quelques centaines de qubits mais de centaines de milliers de qubits. Selon les spécialistes, aucune objection physique fondamentale interdisant la réalisation d'un tel ordinateur n'a été énoncée, le défi est purement technologique.

1.6. Conclusion

Dans ce chapitre, on a essayé de dévoiler les principaux concepts liés au domaine de l'informatique quantique séduisant de part les nouvelles perspectives et opportunités qu'il ouvrirait aux membres de la communauté scientifique. Le défi est colossal, mais les chercheurs sont à pied d'œuvre pour franchir les obstacles rencontrés et ce ne sont pas les initiatives qui manquent pour construire l'ordinateur des rêves. Dans l'attente de l'arrivée d'un tel ordinateur, on peut déjà s'inspirer des concepts quantiques pour développer de nouvelles approches pouvant mieux traiter les différents problèmes.

Dans le prochain chapitre, on s'intéresse à deux problèmes de la vision par ordinateur qu'on va par la suite affronter en adoptant de nouvelles idées inspirées des principes de la physique quantique.

Chapitre 2

Recalage et segmentation d'images

2.1. Introduction

Malgré la vitesse exceptionnelle avec laquelle l'informatique s'est développée au cours des dernières décennies, la vision par ordinateur reste indiscutablement un des domaines où la totale satisfaction est loin d'être acquise. L'ordinateur n'arrive pas à voir comme le fait l'être humain qui ne trouve presque aucune difficulté à différencier des objets, les reconnaître et comprendre ce qui se passe dans une scène.

Le recalage d'images est une tâche fondamentale dans tout processus de vision par ordinateur. Il vise à trouver la meilleure transformation géométrique permettant la superposition des parties communes de deux ou plusieurs images ce qui est la plupart du temps nécessaire pour pouvoir comparer des images. Ce problème est considéré comme un problème d'optimisation pour la résolution duquel plusieurs méthodes ont été proposées.

La segmentation d'une image est une étape importante pour pouvoir interpréter son contenu. Elle vise à identifier des zones distinctes pouvant représenter des entités indépendantes. C'est une tâche difficile à accomplir car dans une image un même objet peut comporter des couleurs différentes alors que deux objets différents peuvent avoir des propriétés très semblables.

Dans ce chapitre on va présenter le recalage d'images, ses applications, ses types, les classes des techniques utilisées, les mesures de similarité entre images (en se concentrant sur l'information mutuelle) et les méthodes utilisées pour résoudre un tel problème. On présentera également la segmentation d'images, ses applications et les classes des méthodes utilisées pour son accomplissement.

2.2. Recalage d'image

2.2.1. Définition du recalage d'images

Définition 1 : Le recalage d'images, en anglais registration, fait référence à la procédure ayant pour but de trouver une transformation spatiale reliant deux ou plusieurs images [Brown 1992].

Définition 2 : Le recalage est la tâche qui cherche la meilleure transformation qui permet de superposer la plus grande partie commune possible des images à apparier compte tenu des variations de la scène [Brown 1992].

Définition 3 : Le recalage d'images est la tâche qui permet d'aligner deux images. Si on considère ces images comme deux tableaux (matrices) à deux dimensions notées I_1 et I_2 où $I_1(x,y)$ et $I_2(x,y)$ sont les intensités au point dont les coordonnées sont x et y , alors le recalage est l'opération qui permet d'estimer les fonctions f et g qui vérifient la relation [Brown 1992] :

$$I_2(x, y) = g(I_1(f(x, y))) \quad (2.1)$$

Où :

f : est une fonction à deux dimensions (2D) représentant la transformation géométrique.

g : est une fonction à une dimension (1D) représentant la transformation photométrique.

Définition 4 : Le recalage est la tâche qui consiste à estimer la transformation géométrique et photométrique optimale qui permet de superposer au mieux les deux images [Talbi et Layeb 2000].

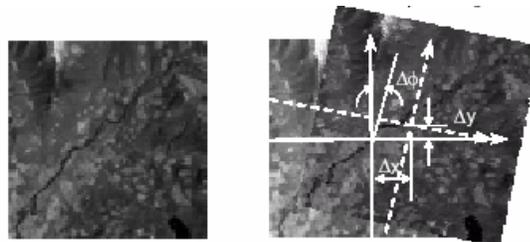


Figure 2.1. Recalage d'images

2.2.2. Classification des transformations géométriques

En prenant en considération la définition 3 ci-dessus, un but essentiel du recalage est de définir la transformation géométrique qui aligne au mieux les images à recalier. Il existe plusieurs types de transformations [Brown 1992].

2.2.2.1. Transformations rigides

Une transformation rigide est la composition d'une rotation et d'une translation. Ce type de transformations conserve les distances et les angles. L'hypothèse de rigidité convient au cas où l'on cherche à compenser la différence de positionnement d'un objet (patient par exemple) par rapport aux capteurs sans tenir compte d'éventuelles déformations des tissus imagés ou de distorsions géométriques créées par les procédés d'imagerie. Les transformations rigides peuvent être définies pour des images 2D par leur forme matricielle :

$$T(x) = Rx + t \quad (2.2)$$

t étant un vecteur colonne 2×1 représentant la translation, et R étant une matrice orthogonale directe 2×2 , c'est-à-dire vérifiant les conditions $RR^t = I_2$ et $\text{Det}(R) = 1$.

2.2.2.2. Similitudes

Parfois appelées homothéties, les similitudes sont des transformations à 4 degrés de liberté en 2D généralisant les transformations rigides par l'adjonction d'un facteur d'échelle global:

$$T(x) = sRx + t, \quad s > 0 \quad (2.3)$$

Bien que ces transformations ne conservent pas en général les distances, elles conservent les angles et les rapports entre les distances.

2.2.2.3. Transformations affines

De façon plus générale les transformations affines sont définies par :

$$T(x) = Ax + t \quad (2.4)$$

Où cette fois A est une matrice 2×2 quelconque. Les transformations affines conservent le parallélisme.

La matrice A étant quelconque, ses coefficients peuvent servir de paramètres libres. Toutefois on obtient une représentation géométriquement plus signifiante en exploitant la décomposition en valeurs singulières.

$$A = R \Delta Q \quad (2.5)$$

Où : Q et R sont des matrices orthogonales directes (matrices de rotation) et Δ une matrice diagonale dont les éléments sont positifs ou nuls.

Toute matrice carrée peut être écrite sous cette forme. Si Q est la matrice identité la transformation est une affinité c'est-à-dire une compression ou dilatation indépendante selon chaque axe suivie d'une transformation rigide. Ce type de transformations est utile lorsque la résolution spatiale de l'image source est méconnue. Plus généralement les variations de Q par rapport à l'identité autorisent des effets de cisaillement de la grille source [Roche 2001].

2.2.2.4. Transformations projectives

Les transformations projectives conviennent typiquement au cas où des images 3D sont recalées avec des images 2D acquises au moyen d'une caméra (par exemple des images vidéo ou des images radiologiques). Elles prennent en compte les effets de perspective liés à la formation de l'image 2D. Il existe autant de familles de transformations projectives que de modèles de caméra. Le plus utilisé étant le modèle à sténopé. Dans ce cas les droites restent des droites mais le parallélisme n'est pas conservé [Faugeras 1993].

2.2.2.5. Transformations locales paramétriques

Les classes de transformation vues précédemment sont dites globales car les variations qu'elles autorisent sont soit nulles (cas des transformations rigides), soit constantes dans l'espace. De ce fait, elles ne permettent pas de modéliser convenablement les déformations complexes d'un ensemble de tissus ou les distorsions géométriques inhérentes à certaines modalités d'acquisition.

Des transformations d'ordre plus général pourront être obtenues en perturbant localement la composante globale du mouvement. Les coordonnées de la déformation résiduelle peuvent être modélisées comme des combinaisons linéaires de fonctions élémentaires :

$$T(x) = Ax + t + \sum_{p=1}^P c_p G(x - x_p) \quad (2.6)$$

Les paramètres c_p sont des vecteurs 2×1 à déterminer (un jeu de paramètres pour chaque coordonnée) et les positions x_p désignent des points de contrôle qui peuvent être fixés ou non. La fonction G étant naturellement choisie de manière à ce que les fonctions $G(x - x_p)$ forment une famille libre. Une transformation de ce type possède $6 + 2P$ degrés de liberté en 2D. Des choix classiques pour G sont le produit tensoriel de fonctions B-splines ou des fonctions à base radiale telles que la fonction « plaque mince » ou Thin-plate en anglais : $G(x) = \|x\|^2 \log \|x\|$ en dimension 2 [Roche 2001].

2.2.3. Types et domaines d'application du recalage

Le recalage d'images est un domaine de recherche aux multiples applications. Le champ d'application le plus important au regard du nombre de publications est celui de l'imagerie médicale [Meshoul 2004].

2.2.3.1. Types de recalage

On distingue fondamentalement 4 types de recalage:

a- Recalage multimodalité : traite les problèmes où les images de la même scène sont prises par des modalités différentes ou capteurs différents.

b- Recalage de gabarit (template): recherche une forme de référence dans une image.

c- Recalage d'images prises de différents points de vue: Ce type de recalage requiert souvent des transformations locales afin d'éliminer les distorsions perspectives.

d- Recalage temporel : vise l'alignement de deux images d'une scène, prises à des moments différents. Ce type de recalage doit tolérer les dissemblances entre les deux images dues aux changements réels dans la scène.

2.2.3.1. Domaines d'application du recalage

Parmi les domaines d'application du recalage on distingue:

a. L'imagerie médicale

Actuellement, une multitude d'outils de diagnostic produisent des images contenant des informations de natures et d'ordres très divers. Dans ce domaine d'imagerie médicale on peut citer les applications suivantes [Sarrut 2000].

- Fusion d'informations:

Si l'on considère deux images d'un même patient, l'une fournissant des informations anatomiques (IRM) et l'autre fonctionnelles (TEP), une opération de recalages entre les deux images permet au praticien de localiser précisément les activités des organes. Donc il s'agit dans ce cas d'un recalage multimodalité.

Une étape de recalage est aussi nécessaire si on veut construire à partir d'une série de coupes 2D un modèle 3D exploitable.

- Comparaison:

Le recalage est utilisé pour suivre l'évolution d'une pathologie que présente un sujet. Il est essentiel pour pouvoir comparer des images prises à différentes périodes. D'autre part la création ou la consultation d'atlas électroniques nécessite une étape de recalage entre images de patients différents.

- Chirurgie assistée:

Ce domaine de recherche utilise le recalage d'images à différents niveaux. Il s'agit par exemple de proposer au médecin un modèle de réalité augmentée, en plus ou à la place, du véritable patient, ce qui nécessite l'utilisation de différentes techniques de recalage.

- Positionnement de patient:

Cette application est liée à la précédente par exemple en radiothérapie conformationnelle, le but est de mesurer et corriger la position d'un patient afin qu'elle soit conforme avec le plan d'irradiation prévu. La mesure du déplacement s'effectue grâce à des images de contrôle, comparées, à l'aide d'un recalage, à une image volumique précédemment acquise.

- *Biologie moléculaire:*

La reconnaissance de motifs 3D se correspondant entre protéines permet d'obtenir des informations sur les propriétés communes de molécules. Pour cela, l'appariement de succession d'atomes est effectué à l'aide de procédures de recalage. [Pennec 1996]

b. Mise en correspondance

La mise en correspondance est également désignée par appariement (matching) dont on trouve les applications suivantes [Sarrut 2000] :

- *Stéréo-vision:*

A l'aide d'images acquises avec un même appareil, mais à partir de points de vue différents, l'appariement stéréoscopique consiste à reconstruire la structure 3D de la scène observée. L'étape de recalage vise généralement à déterminer des points homologues dans chaque image, le correspondant de chaque point étant recherché dans un voisinage limité.

- *Reconnaissance d'objets:*

Etant donné un modèle d'un objet, l'objectif ici est de déterminer s'il est présent ou non dans une scène réelle, et si c'est le cas, d'estimer sa position et son orientation.

- *Suivi d'une cible:*

Dans une séquence d'images on cherche à connaître les positions successives d'un ou de plusieurs objets.

- *Image-mining :*

Il s'agit d'une recherche automatique d'images dans des bases de données visuelles.

- *Reconnaissance de visages:*

Le but est de déterminer si un visage est présent ou non dans une banque de visages, ou bien d'en retrouver l'orientation.

- *Reconnaissance basée sur l'empreinte digitale ou la rétine:*

Il s'agit d'identifier une personne en comparant son empreinte digitale ou sa rétine à celles de différentes personnes stockées sur une base de données.

2.2.4. Méthodes de recalage

Selon l'information prise en considération lors du recalage, on peut regrouper les techniques proposées en deux grandes classes. La première intègre les approches basées primitives (éléments caractéristiques extraits des images à recalcr). La deuxième en revanche englobe les techniques dites iconiques qui considèrent l'ensemble des intensités représentant le contenu des images.

2.2.4.1. Méthodes basées primitives (géométriques)

Les méthodes géométriques sont peut être les plus naturelles car elles semblent procéder de façon analogue à l'esprit humain conscient. Dans un premier temps, il s'agit d'identifier dans les images des entités géométriques communes qu'on appelle primitives et qui peuvent être des points, des lignes, des surfaces, des volumes, des repères orientés, voire des descriptions structurées mettant en jeu des relations entre éléments. Ayant ainsi réduit le signal visuel à une information géométriquement pertinente, on cherche la transformation spatiale qui apparie "au mieux" les différentes primitives [Mehoul 2004].

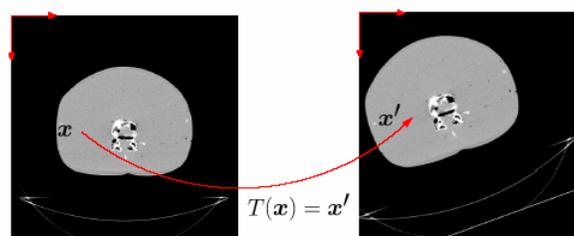


Figure 2.2. Transformations géométriques.

Les méthodes géométriques reposent donc sur deux étapes bien distinctes, qui sont d'ailleurs généralement menées de façon complètement indépendante. La première étape, dite de segmentation est souvent la plus problématique dès lors que l'on souhaite la réaliser automatiquement. Elle requiert d'extraire des primitives qui soient significatives, ce qui peut s'avérer difficile si les images sont fortement bruitées (par exemple, des images ultrasonores). Mais ce n'est pas tout car il faut encore que ces primitives correspondent aux mêmes réalités physiques dans les deux images; cette deuxième contrainte est particulièrement délicate dans un problème de recalage multimodal. Ainsi, la mise en oeuvre des méthodes géométriques nécessite une forte connaissance a priori sur la nature des objets imagés ce qui rend leur champ d'application très spécifique [Zhang 1993] [Meshoul 2004].

Ces méthodes ne tiennent compte que d'un sous-ensemble bien choisi parmi les masses d'informations de départ. Ce sous-ensemble est constitué de primitives élémentaires qui peuvent être des points, des lignes, des courbes, etc. Le point commun de ces méthodes est qu'elles passent toujours par trois étapes [Talbi et Layeb 2000] :

- Extraction de primitives.
- Recherche de la meilleure mise en correspondance.
- Estimation de la transformation optimale.

Extraction des primitives

Cette phase a une grande influence sur l'efficacité de l'appariement, c'est à dire qu'un mauvais choix des primitives donne un appariement biaisé et incorrect [Zhang 1993].

Méthodes d'extraction des primitives

IL y a deux façons pour extraire des primitives : [Talbi et Layeb 2000]

- Extraction manuelle

L'opérateur désigne manuellement les points caractéristiques un à un et peut même désigner les correspondances entre eux. Dans ce dernier cas, la tâche de

recalage se réduit à l'estimation de la transformation optimale.

- Avantage : On est sûr d'avoir un bon appariement.
- Inconvénient : Impraticable quand le volume de données est grand, surtout quand il s'agit d'applications en temps réel, comme c'est le cas en robotique par exemple.
- Extraction automatique

Elle se base sur les propriétés des points (ou primitives dans le cas général) et sur la théorie des invariants et la géométrie différentielle.

- Avantage : Cette méthode est plus pratique dans la majorité des problèmes qui nécessitent l'automatisation de la tâche de recalage.
- Inconvénient : Hormis l'augmentation du coût d'exécution du programme sur machine, l'inconvénient majeur de cette technique reste le problème des points aberrants, et surtout les outliers (les points qui n'apparaissent que dans l'une des images). Car, si un bon traitement de bas niveau minimise le risque d'extraire des points aberrants, la détection des outliers est plutôt difficile puisque ces derniers sont dus à la différence de point de vue duquel est prise l'image ou simplement dus à l'évolution de la scène qui a fait apparaître de nouveaux points caractéristiques.

Classes des primitives

Il existe deux classes de primitives, les primitives extrinsèques et les primitives intrinsèques.

a. Les primitives extrinsèques

Les primitives de cette classe sont des marqueurs placés dans la scène (Exemple : substance radio-opaque en imagerie médicale) ou sur le capteur lui-même.

b. Les primitives intrinsèques

Elles sont extraites à partir des données de la scène.

Contraintes sur le choix des primitives

Les primitives choisies pour être mises en correspondance doivent vérifier les contraintes suivantes :

- Elles doivent être invariantes par la classe de transformation utilisée.
- Elles doivent être facilement appariables, c'est à dire que chacune d'elles doit être de préférence unique de part ses propriétés géométriques et photométriques par rapport aux autres.

Il y a un compromis à vérifier, c'est que le nombre de points doit être suffisant pour que les paramètres puissent être estimés, et en même temps, il ne doit pas être très grand car cela provoque de l'ambiguïté dans la mise en correspondance et conduit à l'explosion combinatoire en matière de coût en temps et en espace mémoire.

Types de primitives

Le tableau suivant résume les types de primitives et leurs caractéristiques :
[Brown 1992]

Type	Caractéristiques
Forte intensité	- Beaucoup d'informations.
Courbures	- Structures intrinsèques. - Moins sensible au bruit.
Primitives saillantes	- Structure intrinsèque. - Positionnement précis
Primitives statistiques	- Utilise toute l'information. - Bonne pour les transformations rigides.
Primitives de haut niveau	- Utilise les relations et autres informations de haut niveau. - Bonne pour les appariements locaux.
Primitives selon modèle	- Structure intrinsèque précise. - Bruit dans une seule image.

Tableau 2.1. Types des primitives

Mise en correspondance des primitives et estimation de la transformation optimale

Il existe deux schémas de recalage utilisant les primitives qui sont les suivants [Talbi et Layeb 2000] :

Recalage avec retour en arrière (with feedback)

- Principe : caractérisé par les étapes suivantes :

- Extraire les traits caractéristiques (primitives) des images.
- Trouver une mise en correspondance entre les primitives de la première image et celles de la deuxième image.
- Estimer la transformation optimale à partir de l'ensemble d'appariements déterminés à l'étape 2.
- Tester la qualité du recalage en mesurant l'erreur entre l'image initiale et l'image résultat : Si l'erreur est suffisamment petite, alors conserver le résultat obtenu ; sinon revenir à l'étape 2 avec indication sur la direction de recherche de la mise en correspondance suivante.

- Avantage : Le recalage avec ce schéma est moins sensible aux distorsions locales incorrigibles, car à chaque fois la mise en correspondance est remise en cause.

- Inconvénient : L'implémentation de ce schéma requiert des algorithmes de recherche sophistiqués.

Recalage sans retour en arrière (without feedback)

Dans ce type de recalage, après l'extraction des primitives, la mise en correspondance est effectuée une fois pour toutes (c'est à dire qu'on y reviendra plus à cette étape), puis il y a l'estimation de la transformation optimale qui se base sur l'ensemble d'appariements établis dans l'étape précédente.

- Avantage : Diminution du temps de calcul (pas de retour en arrière).

- Inconvénients : Difficulté d'établir une bonne mise en correspondance dès le départ et s'il y a une erreur dans les appariements, il y aura certainement un mauvais recalage qu'on ne pourra plus améliorer.

2.2.4.2. Méthodes iconiques

Si les méthodes géométriques peuvent être considérées comme des approches haut-niveau, les méthodes iconiques sont des approches bas-niveau. Contrairement

aux premières, celles-ci ne nécessitent aucune segmentation préalable des images. Elles consistent essentiellement à optimiser un critère de ressemblance ou mesure de similarité fondé uniquement sur des comparaisons locales d'intensités. Dans ce cas, les primitives guidant le recalage sont les données brutes, c'est-à-dire des vecteurs 3D contenant la position et l'intensité des pixels. Pour faire un parallèle avec la vision naturelle, nous pouvons dire que les primitives iconiques sont de l'ordre de la sensation alors que les primitives géométriques sont de l'ordre de la perception, voire de la cognition dans le cas de primitives structurées comme les graphes relationnels (Figure 2.3) [Sarrut 2000] [Roche 2001].

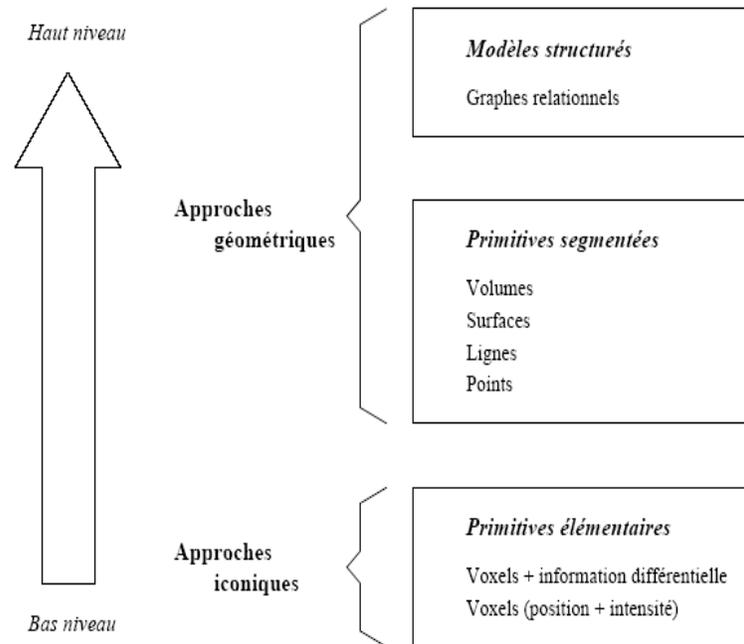


Figure 2.3. Classification générale des méthodes de recalage d'images selon le type de primitives qu'elles utilisent

On reconnaît une méthode iconique à deux propriétés essentielles. D'une part, le choix des primitives est complètement arbitraire : tous les pixels sont a priori des candidats valables. D'autre part, les primitives ne sont pas des entités géométriques; elles appartiennent à un espace figuratif différent du monde réel. Ainsi, le critère

utilisé pour comparer ces primitives est une mesure de similarité reflétant indirectement une distance géométrique.

Remarquons qu'il est possible d'enrichir l'information d'intensité par des informations différentielles d'ordre supérieur (par exemple, le gradient). De telles approches permettent de simuler un recalage géométrique fondé sur des appariements de contours, mais elles restent des méthodes purement iconiques. En revanche, rien n'empêche de choisir des primitives de types différents dans les images à recaler, notamment des primitives iconiques dans l'une et géométriques dans l'autre. Les méthodes de ce type peuvent être qualifiées d'hybrides.

Le canevas général des méthodes iconiques est le suivant. Tout d'abord une mesure de similarité est calculée à partir des intensités de tous les points des images et relative à une transformation est choisie. Ensuite, étant donnée une classe de transformation, une procédure d'optimisation est appliquée afin de maximiser cette mesure de similarité dans l'espace de transformation considéré.

Les méthodes iconiques ne font pas l'hypothèse que des structures identiques soient présentes dans les deux images, alors que c'est une nécessité pour les méthodes géométriques. Dans les méthodes géométriques la qualité du recalage est limitée par la précision de l'étape de segmentation qui est très difficile dans le cas des images multimodalités, cependant les méthodes iconiques ne demandent pas d'extraction des primitives et sont très adaptées pour les images multimodalités.

Mesures de similarité

Il existe plusieurs mesures de similarité dans la littérature du recalage d'images. La plupart des mesures existantes peuvent être définies à partir d'un histogramme conjoint [Horaud et Monga 1995]. D'autres mesures peuvent être appréhendées indépendamment de la notion d'histogramme conjoint. Celles-ci impliquent une certaine interprétation probabiliste, mais d'autres interprétations sont possibles et parfois plus naturelles. De plus en termes de mise en œuvre, le calcul effectif d'un histogramme conjoint peut accroître inutilement la complexité calculatoire d'une mesure. C'est le cas en particulier pour toutes les mesures basées sur des différences d'images. Il faut donc considérer ce formalisme comme unificateur d'un point de vue

conceptuel, mais pas nécessairement pertinent d'un point de vue algorithmique [Roche 2001].

Classification des mesures de similarité

Une mesure de similarité basée sur l'histogramme conjoint est une fonction à valeurs réelles dont l'argument est l'histogramme conjoint, lui-même fonction de la transformation spatiale. Le fondement commun aux nombreuses mesures proposées dans la littérature est l'idée que les intensités de deux images manifestent une cohérence d'autant plus forte que les images sont bien alignées. Le rôle de la mesure de similarité est précisément de donner une signification quantitative à cette notion de cohérence [Roche 2001].

Les figures 2.4 et 2.5 justifient ce principe à travers deux exemples concrets de recalage rigide intra-sujet. La figure 2.4 correspond au recalage monomodal de deux images IRM : on représente leur histogramme conjoint calculé pour deux transformations différentes, l'une traduisant un décalage visible (a), l'autre un recalage quasiment parfait (b). Dans la première situation, les couples d'intensité sont répartis dans le plan de façon relativement désordonnée. La cohérence est manifestement plus forte dans la deuxième situation, où on a la très nette impression que les intensités sont regroupées le long d'une droite [Roche 2001].

La figure 2.5 représente de façon analogue le cas du recalage multimodal d'une image scanner et d'une image IRM. L'évolution de l'histogramme conjoint entre la situation de décalage (a) et la situation de recalage (b) donne encore l'impression visuelle d'un gain de cohérence. Mais, cette fois-ci la relation entre les intensités n'est clairement plus de type affine lorsque les images sont recalées. On observe un amas secondaire, nettement plus dispersé, autour de l'intensité 50 dans l'IRM et 150 dans le scanner. Il n'est pas évident de caractériser une telle relation.

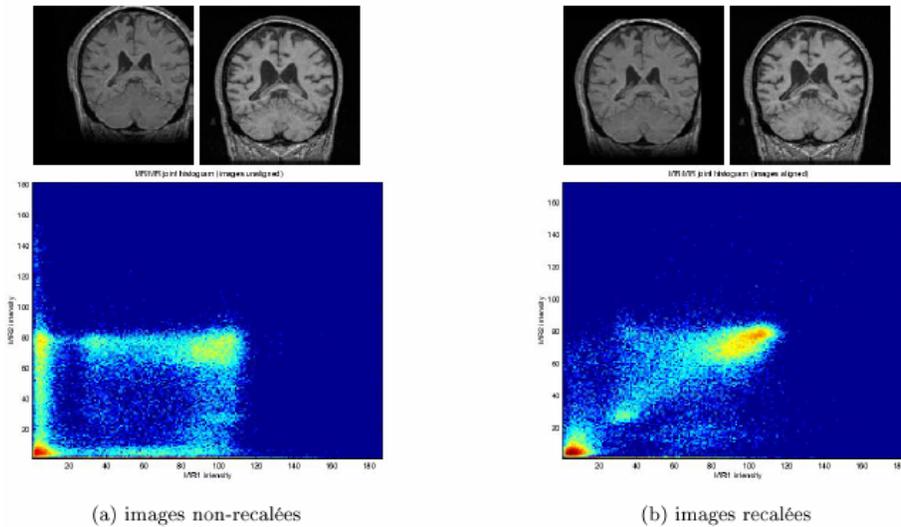


Figure 2.4. Histogramme conjoint de deux IRM cérébrales (coupes coronales représentées) calculé pour deux transformations différentes [Roche 2001]

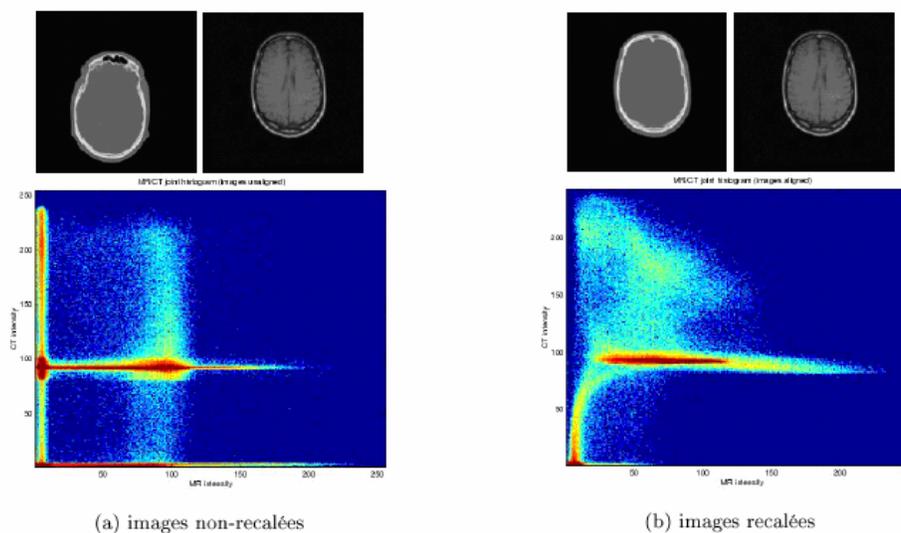


Figure 2.5. Histogramme conjoint d'un couple scanner/IRM (coupes axiales représentées) calculé pour deux transformations différentes [Roche 2001]

Le choix d'une mesure adaptée à un problème de recalage particulier n'est pas toujours clair et ce pour au moins deux raisons. D'une part, établir la relation théorique qui existe entre les intensités des images nécessiterait de faire appel à des modèles physiques d'acquisition qui sont le plus souvent soit indisponibles, soit très complexes et difficiles à synthétiser. D'autre part, la plupart des mesures de similarité se fondent sur des hypothèses de dépendance qui elles-mêmes ne sont pas

entièrement explicites. La classification suivante regroupe les mesures selon le type d'hypothèse auquel elles correspondent [Meshoul 2004].

a. Conservation de l'intensité

Les mesures de cette classe considèrent comme idéale la situation où les images sont identiques. Elles sont le plus souvent calculées à partir d'une image de différence.

Les plus usitées [Brown 1992] sont la somme des différences au carré (SDC) et la somme des différences en valeurs absolues (SDA). Plus récemment différents critères d'entropie calculés à partir de l'histogramme simple de l'image de différence ont été proposés [Buzug et Weese 1998]. Toutes ces mesures peuvent être calculées à partir de l'histogramme conjoint même si, ce n'est pas la manière la plus efficace de procéder.

b. Dépendance linéaire ou affine

On suppose que les intensités sont reliées par une fonction affine c'est-à-dire : $I \approx \alpha J + \beta$. Le cas où le paramètre β est supposé nul correspond à une hypothèse de dépendance linéaire. Ce type d'hypothèse permet de modéliser des contrastes différents entre les images. Les mesures qui lui sont adaptées sont pour la plupart des variantes du coefficient de corrélation [Brown 1992]. Nous pouvons également signaler une variante de la SDC [Alpert et al. 1996] ainsi que la variance du rapport des intensités [Woods et al. 1992].

c. Dépendance fonctionnelle

Pour des images de modalités différentes, en revanche, l'hypothèse de dépendance affine est souvent grossièrement fautive. On peut envisager de s'y ramener par un certain nombre de prétraitements [Meshoul 2004] ou alors supposer l'existence d'une relation fonctionnelle plus générale (non-linéaire, non-monotone) entre les intensités.

d. Dépendance statistique

L'hypothèse d'une dépendance fonctionnelle, même très générale, reste éminemment discutable. Un exemple typique est celui du recalage d'images

cérébrales scanner et IRM. En IRM T1, le liquide céphalo-rachidien et l'os donnent sensiblement les mêmes réponses alors qu'ils sont nettement différenciés dans le scanner ; c'est d'ailleurs ce qui explique la présence d'un amas minoritaire, que nous avons signalé plus haut, dans l'histogramme conjoint de la figure 2.4. A l'inverse les tissus mous sont généralement différenciés en IRM T1 mais confondus en scanner. En toute rigueur, il n'est donc pas possible de relier fonctionnellement les intensités d'une IRM T1 et d'un scanner [Wells et al. 1996].

C'est ici que la notion d'histogramme conjoint prend toute sa dimension. Puisque l'histogramme conjoint peut être interprété comme la densité de probabilité d'un couple de variables aléatoires, il devient naturel d'utiliser une mesure du degré de dépendance statistique entre ces variables comme mesure de similarité entre les images. La théorie de l'information [Blahut 1987][Cover et Thomas 1991] fournit une multitude de critères de liaison statistique, parmi lesquels l'entropie conjointe [Studholme et al. 1995], le coefficient de corrélation entropique [Studholme et al. 1998] mais surtout l'information mutuelle [Collignon et al. 1995][Viola 1995], sont devenus des mesures de similarité standard pour le recalage d'images. La notion de dépendance statistique étant plus générale que la notion de dépendance fonctionnelle, les mesures de cette classe sont celles qui reposent sur les hypothèses les plus faibles. Le tableau 2.2 résume les principales mesures de similarité qui sont abordées précédemment.

Nature de la dépendance	Mesure	Formule
Conservation de l'intensité	SDC*	$\sum_{i,j} p_{i,j}(i-j)^2$
	SDA*	$\sum_{i,j} p_{i,j} i-j $
Linéaire	Inter-corrélation normalisée	$\sum_{i,j} p_{i,j} \frac{ij}{e_I e_J}$
	Critère d'Alpert*	$\sum_{i,j} p_{i,j} \left(\frac{\bar{J}}{\bar{I}} i - j \right)^2$
Affine	Coefficient de corrélation	$\sum_{i,j} p_{i,j} \frac{(i - \bar{I})(j - \bar{J})}{\sigma_I \sigma_J}$
Fonctionnelle	Critère de Woods*	$\sum_j p_j \frac{\sigma_j}{\mu_j}$
	Rapport de corrélation	$1 - \frac{1}{\sigma_I^2} \sum_j p_j \sigma_j^2$
Statistique	Entropie conjointe*	$-\sum_{i,j} p_{i,j} \log p_{i,j}$
	Information mutuelle	$\sum_{i,j} p_{i,j} \log \frac{p_{i,j}}{p_i p_j}$
Notations	$p_i = \sum_j p_{i,j} \quad p_j = \sum_i p_{i,j} \quad p_{i j} = \frac{p_{i,j}}{p_j} \quad (= 0 \text{ si } p_j = 0)$ $\bar{I} = \sum_{i,j} p_{i,j} i = \sum_i p_i i \quad e_I^2 = \sum_{i,j} p_{i,j} i^2 \quad \sigma_I^2 = e_I^2 - \bar{I}^2$ $\bar{J} = \sum_{i,j} p_{i,j} j = \sum_i p_i j \quad e_J^2 = \sum_{i,j} p_{i,j} j^2 \quad \sigma_J^2 = e_J^2 - \bar{J}^2$ $\mu_j = \sum_i p_{i j} i \quad \sigma_j^2 = \sum_i p_{i j} i^2 - \mu_j^2$	

Tableau 2.2. Mesures de similarité utiles : Les mesures marquées d'un astérisque (*) sont à minimiser, tandis que les autres sont à maximiser

- Méthodes approchées pour le recalage

Après la définition du type de recalage (rigide, affine, ...) et la définition d'une mesure de similarité, le troisième aspect fondamental d'un algorithme de recalage est le choix de la méthode d'optimisation qui vise à maximiser le critère de similarité entre les images à mettre en correspondance. Pour les méthodes iconiques, ce choix est rendu critique par le fait qu'aucune mesure de similarité connue n'est convexe en tant que fonction des paramètres de transformation, et ce quel que soit l'espace de recherche considéré. Il est d'ailleurs permis de douter sérieusement de la possibilité même d'une mesure convexe qui soit fondée uniquement sur des comparaisons d'intensités. Cette lacune est sans doute la principale contrepartie des approches iconiques par rapport aux approches géométriques.

Une méthode exhaustive consiste à essayer tous les paramètres possibles et calculer la qualité des transformations associées. Une telle méthode apparaît très difficile voire impossible à cause de la taille grande de l'espace de recherche de ces paramètres. Donc l'utilisation d'autres méthodes d'optimisation dites méthodes approchées est plus pratique.

Le principe des méthodes approchées est de trouver une solution acceptée (une bonne solution) qui n'est pas nécessairement l'optimale. Parmi ces méthodes on trouve les réseaux de neurones artificiels [Talbi et Layeb 2000], le recuit simulé [Van et Aarts 1987], la recherche tabou [Laburthe 1998], les algorithmes génétiques [Meshoul et al. 2001], les colonies de Fourmies et les systèmes immunitaires artificiels [Bendiab et al. 2003].

2.3. Segmentation d'images

2.3.1. Définition

La segmentation d'une image est le processus qui la découpe en éléments (ensembles de pixels). Le but de ce processus est de simplifier la représentation de l'image pour la rendre plus facile à analyser et à interpréter.

La segmentation d'images est typiquement utilisée pour localiser des objets ou reconnaître des contours d'objets. Techniquement, la segmentation d'images consiste à étiqueter chaque pixel de telle sorte que les pixels partageant certaines propriétés visuelles auront la même étiquette.

Le résultat d'une segmentation est donc un ensemble de pixels couvrant la totalité de l'image ou un ensemble de contours délimitant des régions représentant une certaine homogénéité de couleur, intensité, texture, etc. Deux régions adjacentes doivent avoir une différence significative en considérant la propriété choisie pour le découpage de l'image.

Formellement, la segmentation d'images est définie selon Horowitz et Pavlidis [Horowitz et Pavlidis 1974] comme suit:

Soient:

- X le domaine de l'image.
- I la fonction qui associe à chaque pixel une valeur $I(x,y)$.
- P un prédicat d'homogénéité défini sur l'ensemble des parties de X .

La segmentation S de X est définie comme une partition de X en n sous-ensembles $\{R_1, R_2, \dots, R_n\}$ tels que:

1. $X = \bigcup_{i=1}^n R_i$
2. $\forall i \in \{1, \dots, n\}$, R_i est connexe.
3. $\forall i \in \{1, \dots, n\}$, $P(R_i)$ est vrai.
4. $\forall i, j \in \{1, \dots, n\}$, $i \neq j$ / R_i est adjacente à $R_j \Rightarrow P(R_i \cup R_j)$ est faux.

2.3.2. Applications de la segmentation d'images

LA segmentation d'images est une phase d'importance capitale dans tout processus de vision par ordinateur. Il y a donc une multitude de champs où son utilisation offrira une aide appréciable. Parmi les domaines d'application on cite:

- L'imagerie médicale: localisation de tumeurs et d'autres pathologies, mesure du volume d'une tissu, chirurgie guidée ou assistée par ordinateur, étude de structures anatomiques, etc.
- L'imagerie satellitaire: localisation d'objets (routes, forêts, habitations), calcul d'itinéraires, etc.
- Reconnaissance de visages: identification ou vérification de personnes à partir d'images ou de séquences vidéo.
- Reconnaissance basée sur l'empreinte digitale, la rétine ou l'iris.
- Systèmes de contrôle de circulation.
- Localisation dans les chaînes de montage, lecture de code-barres, reconnaissance de caractères, vision robotique, analyse de vidéos, etc.

2.3.3. Méthodes de segmentation d'images

Plusieurs algorithmes généraux ont été développés pour la segmentation d'images. Cependant, un algorithme donné n'est pas approprié pour toute application faisant appel à la segmentation. Cela explique en partie la multitude de méthodes et techniques développées à cette fin [Fu et Mei 1981][Pal et Pal 1993]. Beaucoup parmi ces méthodes et techniques sont associées aux connaissances liées au domaine d'application pour un meilleur rendement.

Les méthodes de segmentation d'images peuvent être classées en catégories [Ho et Lee 2001]:

- **Les méthodes basées sur l'histogramme**

Dans ces méthodes, l'histogramme de l'image est utilisé pour classer les pixels selon leurs valeurs colorimétriques. La méthode des *k-means* [Tou et Gonzalez 1974] [Pappas 1992] est la plus populaire dans cette catégorie.

- **Les méthodes basées sur l'extraction de contours**

Dans ces méthodes, les pixels représentant des changements marqués d'intensité sont extraits et liés pour former des frontières qui représenteront les contours d'objets ou de régions que inclut l'image. Ces méthodes exigent un bas coût de calcul mais

présentent en contrepartie de sérieuses difficultés pour choisir les bons seuils et produire des contours continus d'une largeur de un pixel [Sahoo et al. 1988] [Helterbrand 1996].

- **Les méthodes basées sur la définition de régions**

Ces méthodes cherchent à détecter des régions qui satisfont un certain critère d'homogénéité. Cette catégorie renferme les méthodes d'accroissement de régions et les méthodes pyramidales. Les méthodes d'accroissement de régions [Chang et Lee 1994] [Adams et Bischof 1994] [Hojjatoleslami et Kittler 1998] partent d'un ensemble initiale de régions qui vont par la suite intégrer progressivement de nouveaux pixels vérifiant une propriété donnée. Les méthodes pyramidales [Rezaee et al. 2000] considèrent différents niveaux de résolution. Ces méthodes sont puissantes mais peuvent résulter en une sur-segmentation.

- **Les méthodes de décomposition/fusion (split/merge)**

Ces méthodes visent à corriger le problème de sur-segmentation en adoptant un processus à deux phases. La première phase effectue une segmentation préliminaire de l'image en la découpant en zones homogènes tandis que la deuxième phase essaye d'avoir une segmentation de meilleure qualité en fusionnant des régions voisines jugées suffisamment similaires [Ho et Lee 2001] [Chun et Yang 1996] [Bhandarkar et Zhang 1999].

2.4. Conclusion

Dans ce chapitre on a présenté le problème de recalage d'images : ses principes, ses types et les méthodes utilisées pour la résolution d'un tel problème. On s'est aussi intéressé aux différentes mesures de similarité. Le domaine de la segmentation d'images a également été abordé avec une présentation sommaire de ses applications et une classification des différentes techniques utilisées. Ces deux problèmes peuvent être formulés sous forme de problèmes d'optimisation. La recherche de la solution optimale est une tâche très coûteuse voire impossible à effectuer par les méthodes exhaustives. Ceci a favorisé le développement de méthodes approchées dont le

principe est de trouver, en utilisant des processus stochastiques, une solution acceptable qui n'est pas nécessairement l'optimale mais en un temps raisonnable.

Dans le prochain chapitre seront présentés les sujets d'optimisation combinatoire et d'optimisation multiobjectif. Des sujets qui vont être affrontés dans les autres chapitres par les algorithmes évolutionnaires quantiques développés dans le cadre de cette thèse.

Chapitre 3

Optimisation combinatoire et optimisation multiobjectif

3.1. Introduction

Les problèmes de la vie réelle sont difficiles à résoudre. L'espace de recherche peut être très vaste et une recherche exhaustive de la meilleure solution peut ne pas être possible, même en utilisant les ordinateurs les plus puissants qui explorent plusieurs milliards de solutions en une seconde. Pour certains problèmes, il y a des contraintes fortes qui rendent même la construction d'une solution réalisable difficile. Pour d'autres problèmes, la fonction d'évaluation qui décrit la qualité des solutions est bruitée ou change dans le temps ce qui nécessite plusieurs recherches au lieu d'une seule. Si on dispose en plus de plusieurs critères, surtout s'ils sont contradictoires, notre tâche sera encore plus compliquée. Dans tous les cas cités ci-dessus il s'agit de problèmes d'optimisation.

Dans ce chapitre, on s'intéresse au domaine de l'optimisation. On commence par les problèmes d'optimisation combinatoire mono objectif, les différentes classes de complexité et les méthodes utilisées pour les résoudre. On passe par la suite au thème de l'optimisation multiobjectif où on va notamment dresser un état de l'art concernant les approches évolutionnaires destinées à traiter les problèmes d'optimisation multicritères.

3.2. Optimisation combinatoire

3.2.1. Définition

Un problème d'optimisation combinatoire consiste à trouver une valeur minimale ou maximale d'une fonction à plusieurs variables indépendantes. Cette fonction, appelée la fonction objectif ou la fonction de coût, représente une mesure quantitative de la qualité de certains systèmes complexes.

Un exemple très connu est le problème de voyageur de commerce (en Anglais TSP pour *Travelling Salesman Problem*) [Lawler et al. 1985] qui consiste à trouver le tour minimal que peut traverser un voyageur en parcourant toutes les villes. L'une des applications la plus importante de ce problème d'optimisation est la conception de circuits des systèmes informatiques. La fonction de coût dans ce cas est la longueur du câblage. Les longues lignes diminuent la vitesse du circuit électrique, ce qui ralenti la vitesse de l'ordinateur.

Jusqu'aujourd'hui, Il n'y a pas encore un algorithme qui trouve le tour le plus court du TSP ou la longueur du câblage dans la conception d'un circuit dans un temps polynomial de N , où N est la taille du problème. Quand N augmente, la complexité du calcul augmente exponentiellement et elle sera supérieure à la limite de la puissance du calcul de l'ordinateur. Ce type de problèmes s'appellent «les problèmes difficiles ou complets et sont non déterministes en temps Polynomial (NP-difficile ou NP-complet)». Les chercheurs dans le domaine affirment qu'il n'existe pas d'algorithme qui donne une solution à un problème NP-difficile ou NP-complet dans un temps polynomial. Par conséquent, on dispose d'algorithmes approximatifs qui donnent une solution approximative ou une solution exacte avec une certaine probabilité dans un temps polynomial.

3.2.2. Les problèmes NP-difficile et les problèmes NP-complet

Les classes des problèmes NP-difficile et NP-complet regroupent des problèmes qui sont tout à fait difficile à résoudre. La différence entre ces deux classes n'est pas en leur degré de difficulté mais se manifeste plutôt par le type du résultat attendu de

la résolution du problème: «oui ou non» pour NP-complet et en général des propositions pour NP-difficile (par exemple la tour minimal dans le TSP).

D'abord, on considère la classe NP avant de définir les classes NP-difficile et NP-complet. La classe NP est une sous-classe de la classe décidable des problèmes de décision dont la réponse est exprimée par «oui» ou «non». Par exemple, le problème du circuit de Hamilton est un problème qui vérifie l'existence d'un circuit de Hamilton qui est une boucle fermée construite par certaines arêtes et par tous les sommets dans un graphe donné. Un exemple de ce problème du circuit de Hamilton est montré par la figure 3.1.

En général, un problème de la classe NP est un problème qui peut être vérifié dans un temps polynomial pour prouver l'existence d'une solution ou non. Par exemple, on peut répondre «oui» ou «non» pour le problème du circuit de Hamilton. Le temps de la vérification de l'existence de la boucle fermée qui contient tous les sommets est proportionnel à N .

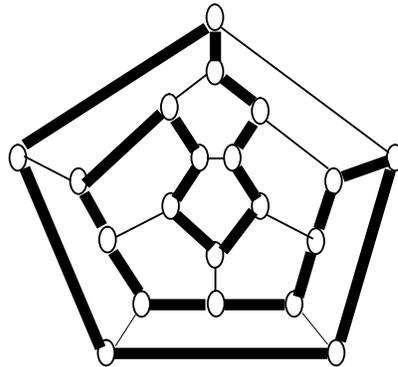


Figure 3.1. Le circuit de Hamilton représenté par la ligne grasse

Pour être précis, un problème de la classe NP est défini comme un problème qui peut être résolu par la machine de Turing non déterministe dans un temps polynomial de N .

Avant de présenter la définition de la machine de Turing non déterministe, on donne la définition de la machine de Turing (déterministe). La machine de Turing non déterministe est le modèle étendu de la machine de Turing déterministe. La machine de Turing déterministe est construite avec :

- La bande qui inclut des éléments appartenant à un alphabet (les instructions et les données).
- L'état de la machine.
- La fonction de transition d'états en utilisant l'élément lu de la bande.

La machine de Turing est un modèle simple des ordinateurs actuels. Pour la machine de Turing déterministe, la fonction de transition est simple à évaluer. La machine de Turing lit un élément de la bande et modifie sa configuration à partir de la configuration précédente selon l'élément lu. Un seul état qui sera modifié est donné par cette transition.

D'une autre part, la fonction de transition de la machine de Turing non déterministe est multiple. Le dispositif de la machine de Turing non déterministe est différent par rapport aux ordinateurs conventionnels. La machine de Turing non déterministe fait des reproductions de toutes les configurations possibles par une étape et le nombre de reproductions augmente exponentiellement en fonction de l'étape. Ce dispositif est tout à fait utile pour résoudre des problèmes complexes.

Tous les problèmes de la classe NP peuvent être vérifiés en un temps polynomial même sur une machine de Turing déterministe, mais la difficulté de résoudre ces problèmes n'est pas la même. Les problèmes sont divisés en quelques sous-classes, la sous-classe P pour des problèmes faciles et la sous-classe NP -complet pour des problèmes difficiles. Un problème qui est résolu dans un temps polynomial sur la machine de Turing déterministe appartient à la classe P . Il est clair que NP contient P parce que la machine de Turing non déterministe inclut dans sa définition la machine de Turing déterministe.

Pour un exemple d'un problème de la classe P , on explique le problème du circuit de Hamilton pour un graphe dont le degré maximum de chaque sommet est 2 (pour qu'il appartienne à P). Les graphes typiques sont montrés dans la figure 3.2 ci-dessous. Les réponses des problèmes de gauche et de droite sont respectivement «oui» et «non». On peut vérifier l'existence d'un circuit de Hamilton par la méthode suivante :

1. Commencer à partir d'un sommet et aller au sommet voisin relié à lui par une arête. (On ne peut pas aller au voisin qui a déjà été visité).
2. Répéter 1 jusqu'à ce qu'il n'y ait pas de sommet à visiter.
3. Si le dernier sommet est relié au sommet de départ et tous les sommets ont été visités, la réponse est «oui». La réponse est «non» autrement.

Cette procédure s'exécute dans un temps polynomial. Si le degré maximum d'un sommet n'est pas limité à deux, cette procédure ne convient pas.



Figure 3.2. Le problème du circuit de Hamilton avec les réponses «oui» et «non»

Le graphe de gauche est une simple boucle (circuit de Hamilton) et le graphe de droite n'est pas une simple boucle

On considère les classes difficiles NP-difficile et NP-complet. Si un problème de la classe *NP* peut être résolu en utilisant un algorithme particulier du problème répété en temps polynomial, ce problème s'appelle «NP-difficile». En d'autres mots, on peut résoudre n'importe quel problème *NP* en appelant le sous-programme pour un problème NP-difficile particulier. Un problème NP-difficile est plus difficile que tous les problèmes de la classe *NP* ou aussi difficile comme les autres problèmes NP-difficile dans l'ordre polynomial. L'algorithme du problème NP-difficile est l'algorithme le plus général dans la classe *NP*. Les problèmes NP-difficile par définition ne sont pas limités pour appartenir à la classe *NP*. Par exemple, le TSP n'est pas un problème de la classe *NP* mais un problème de la classe NP-difficile parce que la réponse à ce problème n'est pas «oui» ou «non» mais c'est la longueur du tour. La sous-classe «NP-complet» est l'ensemble des problèmes NP-difficile qui appartiennent à la classe *NP*. Les problèmes NP-complet sont plus difficiles que tous les autres problèmes de la classe *NP* et difficile comme tous les problèmes NP-

difficile. Le problème du circuit de Hamilton est connu comme un problème NP-complet.

La difficulté des problèmes peut être exprimée symboliquement comme $P \leq (NP - P - (NP\text{-complet})) \leq NP\text{-complet} \approx NP\text{-difficile}$. C'est une conséquence insignifiante de leur définition que si l'un des problèmes NP-complet peut être résolu dans un temps polynomial, puis tous les problèmes de la classe NP sont également résolus en temps polynomial. Dans ce cas-ci on aurait $P = NP$. Jusqu'à présent, il n'existe pas d'algorithme polynomial pour résoudre un problème NP-complet, et la question de si P est égal à NP est encore ouverte, bien que la majorité supposent que les problèmes NP-complet ne sont pas polynomiaux. La situation la plus probable est indiquée dans la figure ci-dessous. Les problèmes NP peuvent être divisés en trois sous-classes : NP-complet, P et la sous-classe qui n'appartient pas aux deux sous-classes précédentes. La classe NP-difficile est divisée en deux sous-classes : NP-complet et une autre partie qui indique si le problème est un problème de décision ou non. Expérimentalement, le temps nécessaire pour résoudre les problèmes NP-difficile et les problèmes NP-complet augmente exponentiellement avec N .

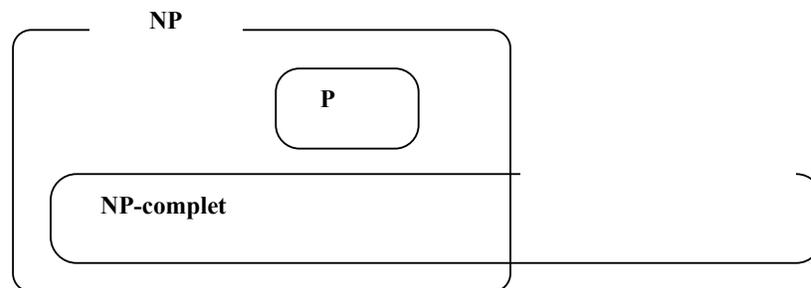


Figure 3.3. La classe des problèmes NP suppose que $P \neq NP$.

Sous cette supposition on peut voir qu'il y a des problèmes NP qui ne sont ni de la classe NP-complet ni de la classe P

3.2.3. Méthodes approchées pour la résolution des problèmes d'optimisation combinatoire

Vue la difficulté, voire l'impossibilité de résoudre les problèmes d'optimisation combinatoire en un temps raisonnable avec les méthodes exactes, d'autres méthodes dites approchées ont été développées et utilisées pour parvenir à des solutions acceptables (pas toujours optimales) en un temps beaucoup plus court. Ces méthodes approchées peuvent être classées en différentes catégories [Freville 2000] [Blum et Roli 2003]:

- **Constructives** : algorithmes glouton, méthode Pilote, GRASP.
- **Recherche locale** : algorithmes de descente, multi-départs, recuit simulé [Aarts et al. 2005], algorithme à seuil, recherche Tabou [Osman 1993], méthode de bruitage.
- **Évolutionnistes** : algorithmes génétiques, algorithmes évolutionnaires, recherche dispersée, méthode des chemins, systèmes de fourmis [Colomi et al. 1992] [Dorigo et Stützle 2004], essaims de particules [Al-Kazemi et Mohan 2002], Boids de Reynolds.
- **Réseaux de neurones** : Modèle de Hopfield-Tank, machine de Boltzmann [Aarts et Korst 1988], réseau auto-adaptatif, réseau élastique.
- **Heuristiques Bayésiennes** : optimisation globale, optimisation discrète.
- **Superposition** : perturbation des données, perturbation des paramètres d'une heuristique.

3.3. Optimisation multiobjectif

3.3.1. Définition

La plupart des problèmes d'optimisation ont, de nature, plusieurs objectifs à considérer. Un problème d'optimisation multiobjectif, ou multi-critère, consiste à optimiser simultanément plusieurs fonctions objectifs. Les objectifs sont le plus souvent contradictoires [Osyczka, 1985]. C'est-à-dire que l'amélioration d'une

solution par rapport à un objectif va très probablement impliquer une détérioration par rapport à d'autres objectifs.

3.3.2. Formulation du problème

On considère le vecteur $x = [x_1, x_2, \dots, x_n]^T$ composé de variables dites variables de décision. Chaque variable x_i appartient à un intervalle $[x_i^l, x_i^u]$. Les intervalles de définition de tous les x_i constituent l'espace de décision D . L'image de l'espace de décision dans l'espace de critères est dite espace de recherche.

Un problème d'optimisation multiobjectif consiste à trouver le vecteur $x^* = [x_1^*, x_2^*, \dots, x_n^*]^T$ qui satisfait les m contraintes d'inégalité:

$$g_i(x) \geq 0 \quad i=1,2,\dots,m \quad (3.1)$$

les p contraintes d'égalité:

$$h_i(x) = 0 \quad i=1,2,\dots,p \quad (3.2)$$

Et optimise la fonction vectrice:

$$f(x) = [f_1(x), f_2(x), \dots, f_k(x)]^T \quad (3.3)$$

Une solution est faisable si elle vérifie toutes les contraintes d'égalité et toutes les contraintes d'inégalité.

Si une solution ne satisfait pas une des contraintes, elle est dite solution infaisable.

3.3.3. Optimalité de Pareto et dominance

3.3.3.1. Notion d'optimum en optimisation multiobjectif

Ayant plusieurs fonctions objectifs, la notion d'optimum change parce qu'en réalité on cherche à trouver de bons compromis, et non pas une solution unique. La notion d'optimum qui est le plus souvent adoptée est celle proposée par Francis Ysidro Edgeworth en 1881. Cette notion a été plus tard généralisée par Wilfredo Pareto [Pareto, 1886]. Certains désignent cette notion par l'optimum d'Edgeworth-

Pareto. Dans la suite de cette thèse, on adoptera le terme le plus communément utilisé: l'optimum de Pareto.

Une solution est dite Pareto optimale s'il n'existe pas une autre solution faisable qui optimise certains critères sans causer une dégradation d'au moins un autre critère.

3.3.3.2. Définition de la dominance

Ayant un problème de minimisation multiobjectif, une solution $x(i)$ est dite dominer une autre solution $x(j)$ si et seulement si:

$$\forall n \in \{1, 2, \dots, k\}, f_n(x(i)) \leq f_n(x(j)) \quad (3.4)$$

$$\exists n \in \{1, 2, \dots, k\} \text{ pour lequel } f_n(x(i)) < f_n(x(j)) \quad (3.5)$$

3.3.3.3. Ensemble de Pareto

Soit P un ensemble de solutions candidates d'un problème d'optimisation multiobjectif. L'ensemble $P' \subseteq P$, composé des éléments de P qui ne sont dominées par aucun élément de P , est dit sous-ensemble non dominé de l'ensemble P .

L'ensemble de Pareto global d'un problème d'optimisation multiobjectif (appelé couramment ensemble de Pareto) est l'ensemble des solutions non dominées par aucun élément de l'ensemble des solutions faisables.

L'image de l'ensemble de Pareto dans l'espace des critères est dite surface de Pareto ou front de Pareto dans le cas d'un problème biobjectif.

3.3.4. Un peu d'histoire

Les origines des fondements mathématiques de l'optimisation multiobjectif remontent à la période allant de 1895 à 1906 [Coello et al. 2002]. Durant cette période, Georg Cantor et Felix Hausdorff posèrent les bases des espaces ordonnés à dimension infinie. Cantor introduisit également les classes d'équivalence et formula les premières conditions suffisantes pour l'existence d'une fonction d'utilité. Hausdorff donna également le premier exemple d'un ordre total.

Toutefois, c'était le concept du Vector maximum problem, introduit par Harold W. Kuhn et Albert W. Tucker en 1951, qui a fait de l'optimisation multiobjectif une discipline mathématique à part.

Cependant la théorie de l'optimisation multiobjectif est restée relativement sous-développée durant les années 1950. Ce n'était que durant les années 1960 que les fondements de l'optimisation multiobjectif furent consolidés et tenus sérieusement par des purs mathématiciens quand Leonard Hurwicz généralisa les résultats de Kuhn et Tucker aux espaces vectoriels topologiques.

L'application de l'optimisation multiobjectif aux domaines autres que l'économie commença avec le travail de Koopmans en 1951 dans la théorie de la production et le travail de Marglin en 1967 dans la planification des ressources d'eau.

La première application en ingénierie, citée dans la littérature, fût un papier de Zadeh au début des années 1960 [Zadeh 1963]. Cependant, l'utilisation de l'optimisation multiobjectif ne s'est généralisée qu'au début des années 1970.

3.3.5. Optimisation multiobjectif et algorithmes

évolutionnaires

3.3.5.1. Pourquoi les algorithmes évolutionnaires?

Actuellement, il y a plus de 30 techniques mathématiques de programmation pour l'optimisation multiobjectif. Néanmoins, ces techniques tendent à générer des éléments de l'ensemble de Pareto, un à chaque fois. En plus, la plus part de ces techniques sont sensibles à la forme du front ou de la surface de Pareto. Certaines techniques ne sont pas applicables lorsque le front de Pareto est concave ou discontinu.

Les algorithmes évolutionnaires semblent être particulièrement adéquats pour la résolution de problèmes d'optimisation multiobjectif. Cela revient essentiellement à leur capacité de travailler simultanément sur un ensemble de solutions possibles (appelé population) ce qui nous permettra de trouver plusieurs éléments de l'ensemble de Pareto en une seule exécution de l'algorithme. En plus, les algorithmes

évolutionnaires sont moins sensibles à la forme de la surface de Pareto. Ils peuvent être facilement appliqués sur des problèmes pour lesquels le front de Pareto est concave ou discontinu.

3.3.5.2. Vue d'horizon des algorithmes évolutionnaires pour l'optimisation multiobjectif

Le potentiel des algorithmes évolutionnaires dans l'optimisation multiobjectif était pressenti par Rosenberg dans les années 1960, mais la première implémentation réelle fût celle de Schaffer [Schaffer 1985]. Durant la décennie qui suivit, le domaine resta pratiquement inactif. Il commença à croître spectaculairement au milieu des années 1990 avec le développement de plusieurs techniques et applications.

Les approches évolutionnaires pour l'optimisation multiobjectif peuvent être classées en techniques de première génération et techniques de deuxième génération.

a. Techniques de première génération

Certaines techniques de première génération ne considèrent pas directement l'optimalité de Pareto. Dans ce groupe on trouve les approches agrégatives, l'algorithme VEGA, l'ordonnancement lexicographique, la méthode ε -contrainte et les approches à vecteur cible. Le tableau suivant résume le principe, les avantages et les inconvénients de chaque approche citée ci-dessus.

Approche	Principe	Avantages	Inconvénients
Approches agrégatives	Combinent tous les objectifs en utilisant des opérations arithmétiques pour former un seul objectif.	- Faciles à implémenter. - Efficaces.	- Difficulté de choisir les bons opérateurs et les bons poids.
VEGA (Vector Evaluated Genetic)	Utilise des sous-populations qui optimisent chaque	- Facile à implémenter.	- N'a pas un mécanisme explicite pour maintenir la

Approche	Principe	Avantages	Inconvénients
Algorithm) [Schaffer 1985]	critère séparément.	- Efficace.	diversité. - Ne produit pas nécessairement des vecteurs non dominés.
Ordonnancement lexicographique [Fourman 1985]	L'utilisateur choisit un ordre d'importance des objectifs. L'optimisation concerne les objectifs un par un selon l'ordre choisi.	- Facile à implémenter. - Efficace.	- La performance dépend de l'ordre choisi. - inapproprié pour un grand nombre de critères.
Méthode ϵ -contrainte	Optimisation du principal critère avec considération des autres comme contraintes liés à des seuils ϵ_i ajustés à chaque fois.	- Facile à implémenter.	- Coût de calcul élevé (plusieurs exécutions peuvent être nécessaires).

Approche	Principe	Avantages	Inconvénients
Approches à vecteurs cibles	Définition d'une valeur cible pour chaque objectif puis minimisation des différences entre la solution actuelle et ces valeurs cibles.	<ul style="list-style-type: none"> - Facile à implémenter. - Efficace. 	<ul style="list-style-type: none"> - les valeurs cibles doivent être dans la région faisable. - La définition des valeurs cibles peut être difficile et peut entraîner un coût additionnel. - Certaines valeurs peuvent mener à une fausse piste.

Tableau 3.1. Techniques de première génération ne considérant pas l'optimalité au sens de Pareto

D'autres techniques de la première génération sont basées sur l'optimalité de Pareto. Parmi ces techniques on cite: le pur classement basé Pareto, MOGA, NSGA, NPGA et NPGA2. Ces techniques requièrent une procédure de classement et une technique pour maintenir la diversité de la population. Le tableau suivant résume le principe, les avantages et les inconvénients de chacune de ces techniques.

Approche	Principe	Avantages	Inconvénients
Le pur classement basé Pareto	Suit la proposition de Goldberg dans son livre <i>Genetic Algorithms in Search</i> [Goldberg 1989] pour trier les solutions selon le principe de dominance au sens de Pareto.	<ul style="list-style-type: none"> - Implémentation relativement facile. - N'est pas sensible à la forme et à la continuité du front. 	- La vérification de la dominance est $O(kM^2)$, k étant le nombre de critères et M la taille de la population.
MOGA (MultiObjective Genetic Algorithm) [Fonseca & Fleming 1993]	Le rang d'un individu correspond au nombre d'individus dans la population qui le dominant. L'algorithme utilise un partage de fitness et des restrictions de croisement.	<ul style="list-style-type: none"> - Implémentation relativement facile. - Efficace. - MOGA était le plus populaire et le plus performant parmi les algorithmes de la première génération. 	- Sa performance dépend du bon choix du facteur commun.

Approche	Principe	Avantages	Inconvénients
NSGA (Non-dominated Sorting Genetic Algorithm) [Srinivas et Deb 1994]	Basé sur la séparation des individus en plusieurs groupes en fonction du degré de domination de chaque individu.	<ul style="list-style-type: none"> - Relativement facile à implémenter. - Applicable à des problèmes avec un nombre quelconque d'objectifs. 	<ul style="list-style-type: none"> - La complexité est $O(kM^3)$. - Nécessite la spécification d'un paramètre de partage (sharing) auquel l'algorithme est très sensible.
NPGA (Niche-Pareto Genetic Algorithm) [Horn et al. 1994] et NPGA2 [Erickson et al. 2001]	Utilise un schéma de tournoi de sélection basé sur la dominance de Pareto. À chaque fois, deux individus choisis aléatoirement sont comparés à un sous-ensemble de la population pour que l'un soit sélectionné.	<ul style="list-style-type: none"> - Facile à implémenter. - Efficace puisque le classement Pareto n'est pas appliqué à toute la population. - Une bonne performance. 	<ul style="list-style-type: none"> - Nécessite en plus du facteur de partage un autre paramètre (la taille du tournoi).

Tableau 3.2. Techniques de première génération considérant l'optimalité Pareto

b. Techniques de deuxième génération

La deuxième génération des techniques d'optimisation multiobjectif évolutionnaire donne plus d'importance à l'efficacité. L'utilisation de populations

secondaires (externes) devient chose commune et d'intelligentes manières sont fournies pour générer des solutions non dominées et uniformément distribuées.

Strength Pareto Evolution Algorithm (SPEA)

Proposé par Zitzler et Thiele [Zitzler et Thiele 1999], cet algorithme utilise une archive (ou population externe) pour le stockage des solutions non dominées. Il utilise une valeur de force similaire à la valeur de classement de l'algorithme MOGA pour calculer la fitness des individus. Une technique de clustering, appelée "average linkage method" ou méthode de liaison moyenne, est utilisée pour réduire l'ensemble de Pareto tout en maintenant la diversité dans la population externe.

Avantages et inconvénients

- Facile à implémenter.
- n'exige pas de réglage de paramètres de partage.
- permet de bien échantillonner les individus dans l'espace et de bien distribuer les solutions sur le front de Pareto.
- la valeur de fitness des individus dépend de la taille choisie pour la population externe.

Exemples d'application

- Exploration de trade-offs des implémentations software pour les algorithmes DSP [Zitzler et al. 1999].
- Planification des traitements [Petrovski et McCall 2001].
- Affectation dans les services de radiologie [Lahanas et al. 2001].

SPEA2

C'est une version révisée de SPEA proposée dans [Zitzler et al. 2001]. Elle a trois différences essentielles par rapport à SPEA:

- Elle utilise une stratégie d'affectation de fitness qui prend en compte, pour chaque individu, le nombre d'individus qui le dominant et le nombre d'individus qu'il domine.

- Elle utilise une technique du plus proche voisin pour l'estimation de la densité ce qui permet de guider la recherche plus efficacement.
- Elle adopte une méthode améliorée pour le découpage de l'archive qui garantit la préservation des solutions frontalières.

Pareto Archived Evolution Strategy (PAES)

Cet algorithme, proposé par Knowles et Corne [Knowles et Corne 2000], utilise un seul individu avec une stratégie d'évolution (1+1) et un archive pour stocker les solutions non dominées. Il utilise une grille adaptative pour maintenir la diversité.

L'algorithme se présente comme suit:

1. Génération aléatoire d'un individu c et ajout de c à l'archive.
2. Production d'un nouvel individu m par mutation de c .
3. Evaluer m :
 - Si m domine c , remplacer c par m dans l'archive et considérer m comme solution courante.
 - Si m est dominé par un membre de l'archive, écarter m .
 - Sinon
 - Si m domine un individu ou plus de l'archive, ajouter m à l'archive et supprimer les individus dominés.
 - Sinon si m appartient à une zone moins encombrée qu'un individu de l'archive, remplacer l'individu par m .
4. Considérer comme solution courante l'individu, parmi c et m , qui appartient à la zone la moins encombrée.
5. Aller à 2.

Une technique de Crowding, basée sur le découpage en hypercubes de l'espace des objectifs, est utilisée pour déterminer l'encombrement d'une zone.

Avantages et inconvénients

- Facile à implémenter.
- Nombre réduit des paramètres de l'algorithme évolutionnaire dû à l'utilisation d'un seul individu par génération.
- Mise à jour rapide de l'archive grâce à la technique de Crowding.
- Son efficacité dépend du paramètre l de découpage de l'espace des paramètres.

Exemples d'application

- Problème de routage hors-ligne [Knowles et Corne 1999].
- Problème de gestion distribuée adaptative d'une base de données [Knowles et al. 2000].

Pareto Envelope-based Selection Algorithm (PESA)

C'est un algorithme génétique proposé par les mêmes auteurs de l'algorithme PAES. L'algorithme utilise une population interne PI et une population externe PE. Il reprend également le crowding utilisé dans PAES avec définition d'un paramètre appelé `squeeze_factor` pour mesurer l'encombrement d'une zone [Knowles et al. 2000].

L'algorithme se présente comme suit:

1. Génération aléatoire des individus de PI et initialisation de PE avec l'ensemble vide.
2. Evaluation des individus de PI et transfert des éléments non dominés dans PE.
3. Si le critère d'arrêt est vérifié, on retourne PE comme ensemble de solution. Sinon, les éléments de PI sont reconstitués de la manière suivante:
 - avec une probabilité p_c , deux individus de PE sont choisis pour produire un nouvel individu par croisement puis mutation.
 - avec une probabilité $1-p_c$, un individu est choisi pour produire un nouvel individu par mutation seulement.
4. Aller à 2.

Lors de l'évaluation, un individu est ajouté à la population externe s'il n'est pas dominé par un autre individu de PI ou de PE. Et tout individu de PE dominé par un autre individu est supprimé.

Si l'ajout d'un nouvel individu provoque un dépassement de capacité de PE, l'individu ayant le plus élevé `squeeze_factor` est supprimé.

Le `squeeze_factor` est égal au nombre d'individus appartenant au même hypercube.

Lors de la sélection, les individus ayant le plus petit `squeeze_factor` sont favorisés. Ceci permettrait de guider la recherche vers les zones les moins encombrées.

Avantages et inconvénients

- Facile à implémenter.
- La sélection basée sur le `squeeze_factor` offre une meilleure répartition des individus dans l'espace de recherche.
- Plus efficace que SPEA et PAES puisqu'il donne un plus grand pourcentage de solutions sur le front de Pareto en un nombre réduit d'itérations [Berro 2001].
- Son efficacité dépend du paramètre l de découpage de l'espace des paramètres.

Exemple d'application

- Problèmes de télécommunications [Knowles et al. 2000].

PESA-II

C'est une version révisée de PESA. L'algorithme PESA-II [Corne et al. 2001] adopte une sélection basée sur la région et non pas sur l'individu. A chaque fois, la région (l'hypercube) ayant le plus petit `squeeze_factor` est choisie pour générer aléatoirement un nouvel individu appartenant à cette région.

NSGA-II

C'est une technique proposée dans [Deb et al. 2000] qui est plus efficace que NSGA en terme de temps de calcul. C'est une méthode élitiste utilisant un opérateur

de comparaison basé sur la mesure d'encombrement qui maintient la diversité sans spécification de paramètres additionnels (le paramètre de partage n'est plus nécessaire).

Exemples d'application

- Gestion des forêts [Ducheyne et al. 2001].
- Optimisation de formes [Deb et Goel 2001].

MOMGA (MultiObjective Messy Genetic Algorithm)

Cet algorithme, proposé par Van Veldhuizen et Lamont [Van Veldhuizen et Lamont 2000], est une tentative d'étendre l'algorithme génétique désarticulé (Messy Genetic Algorithm [Goldberg et al. 1989]) pour résoudre des problèmes d'optimisation multiobjectif.

MOMGA est constitué de trois phases:

1. Phase d'initialisation: Dans cette phase MOMGA produit tous les éléments de base (blocs constructifs) d'une taille donnée à travers un processus déterministe appelé énumération partiellement énumérative.

2. Phase primordiale: Elle effectue une sélection par tournoi sur la population. A chaque fois, deux candidats sont aléatoirement choisis pour être comparés à un ensemble d'autres individus. Si seulement l'un d'entre eux est non dominé, il est alors sélectionné pour la reproduction. Si les deux sont non dominés ou les deux sont dominés une stratégie de partage (sharing) est alors adoptée.

3. Phase juxtapositionnelle: Dans cette phase, l'algorithme procède à l'accroissement de la population à travers l'utilisation de l'opérateur de recombinaison couper-coller.

Exemples d'application

- Coception de contrôleurs [Herreros et al. 1999].
- Benchmarks traditionnels [Van Veldhuizen et Lamont 2000].

MOMGA-II

Proposé dans [Zydallis et al. 2001], cet algorithme diffère de l'algorithme MOMGA original dans les deux premières phases. La phase d'initialisation est complètement probabiliste. Elle crée un nombre contrôlé d'éléments de base de taille bien déterminée. La deuxième phase procède à un filtrage pour réduire le nombre d'éléments de base. Les meilleurs éléments sont alors stockés. La phase juxtapositionnelle est la même adoptée dans MOMGA.

Exemples d'application

- Benchmarks traditionnels [Zydallis et al. 2001].

Micro Genetic Algorithm (MGA)

Cet Algorithme, proposé dans [Coello et Toscano 2001], utilise les résultats obtenus par Goldberg selon lesquels une population de petite taille suffit pour avoir une convergence. Le mécanisme proposé consiste à appliquer des opérateurs génétiques sur une population composée d'un très petit nombre d'individus jusqu'à perte de diversité, puis sélectionner les meilleurs individus et générer aléatoirement de nouveaux individus pour recommencer.

Coello utilise dans son algorithme une archive constituée de deux parties: l'une remplaçable et l'autre irremplaçable. La partie irremplaçable n'est mise à jour que lorsque le micro algorithme converge. Elle sert à maintenir la diversité. La partie remplaçable est modifiée régulièrement au cours du déroulement du micro algorithme.

Avantages et inconvénients

- Facile à implémenter.
- Un temps de calcul réduit.
- Beaucoup de paramètres à régler.

Exemple d'application

- Benchmarks traditionnels [Coello et Toscano 2001].

3.4. Conclusion

Dans ce chapitre, on a donné les principaux concepts liés aux problèmes d'optimisation combinatoire et d'optimisation multiobjectif. Les méthodes approchées représentent l'issue la plus en vue pour traiter les différents problèmes en un temps raisonnable malgré le fait que ces méthodes ne garantissent pas l'optimalité des solutions trouvées. Parmi les méthodes approchées, les méthodes basées sur les algorithmes évolutionnaires ont été choisies par la majorité des chercheurs pour traiter les problèmes d'optimisation multiobjectif.

Dans le prochain chapitre, je vais présenter en détails les algorithmes évolutionnaires quantiques originales et les améliorations qui leur ont été apportées dans le cadre de cette thèse. Différents problèmes ont été utilisés pour valider notre travail.

Chapitre 4

Algorithmes évolutionnaires quantiques (AEQ)

4.1. Introduction

L'exploitation de toutes les opportunités qu'offre le monde merveilleux de l'informatique quantique ne sera possible qu'après la mise en place de vrais ordinateurs quantiques. Des ordinateurs capables de stocker l'information sur des bits quantiques, capables d'appliquer des opérations quantiques sur l'information, capables d'implémenter le principe d'enchevêtrement et capables surtout de faire face au problème de décohérence qui peut souvent se manifester dans un système quantique.

A ce jour, le meilleur des ordinateurs quantiques ne dispose pas de suffisamment de qubits et ne peut pas préserver sa cohérence assez longtemps pour accomplir des algorithmes quantiques. C'est ce qui a alimenté une nouvelle tendance visant à tirer profit des principes de l'informatique quantique sans exiger la présence d'une machine quantique. Plusieurs techniques implémentées sur des machines conventionnelles ont été donc enrichies par des concepts provenant du domaine de la physique quantique et les résultats obtenus ont été indiscutablement remarquables, voire meilleurs dans la plupart des cas. On a ainsi conçu des réseaux de neurones, des colonies de fourmis, des essaims de particules et des algorithmes génétiques, tous augmentés par les bases de l'informatique quantique [Nedjah et al. 2008].

Les algorithmes évolutionnaires quantiques (AEQ), ou plus précisément les algorithmes évolutionnaires inspirés du quantique (Quantum-Inspired Evolutionary Algorithms), ont marqué avec force leur présence ces dernières années dans le domaine de l'optimisation combinatoire. Le premier travail dans ce sens a été celui de *Narayanan* et *Moore* [Narayanan et Moore 1996] qui ont proposé un algorithme génétique s'inspirant des principes de l'informatique quantique. L'algorithme adopte une vision multi-univers où chaque univers possède sa propre population de chromosomes. Les populations dans chaque univers évoluent en parallèle selon les règles des algorithmes génétiques conventionnels. Cependant, juste après le croisement classique dans chaque univers, les univers peuvent interférer les uns avec les autres pour éventuellement générer de meilleures solutions. Ce qu'on reproche à cette méthode est son interprétation différente de l'interprétation standard de la mécanique quantique, son pauvre formalisme et la non évidence de son apport par rapport aux algorithmes génétiques classiques [Giraldi et al. 2004] [Sofge 2008]. C'est ce qui a poussé à l'abandon de cette voie.

Dans [Han et Kim 2000], *K-H. Han* et *J-H. Kim* ont proposé un nouveau modèle d'algorithme génétique s'inspirant des concepts de l'informatique quantique tels que le bit quantique (qubit), la superposition d'états et un opérateur quantique (la porte D). Ils ont appliqué l'algorithme proposé sur un problème d'optimisation combinatoire, celui du *sac-à-dos binaire (0/1knapsack)* et ont surtout conclu que l'hybridation entre algorithmes génétique et l'informatique quantique est une voie très prometteuse. Ce qui doit être noté est que l'algorithme de *K-H. Han* et *J-H. Kim* n'utilise ni mutation ni croisement. Selon les auteurs, l'utilisation de ces deux opérations conduit vers une détérioration des performances.

Dans la suite de ce chapitre, je vais présenter l'algorithme original de *K-H. Han* et *J-H. Kim* suivi de nouvelles versions que nous avons développé pour résoudre différents problèmes et qui ont montré de meilleurs comportements.

4.2. Algorithmes évolutionnaires quantiques

Présentation

Un algorithme génétique quantique, tel que décrit par *K-H. Han* et *J-H. Kim* dans [Han et Kim 2000], est basé sur l'évolution d'une population constituée de chromosomes quantiques. Un chromosome quantique n'est rien d'autre qu'un registre quantique, c'est-à-dire une chaîne de bits quantiques (ou qubits).

Un chromosome de m gènes peut être représenté comme suit:

$$\begin{pmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_m \\ \beta_1 & \beta_2 & \dots & \beta_m \end{pmatrix}$$

Où, pour $i=1..m$, $\begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix}$ représente un qubit vérifiant la condition : $|\alpha_i|^2 + |\beta_i|^2 = 1$.

Un tel chromosome ne représente pas une simple solution du problème à résoudre, mais plutôt une superposition de toutes les solutions possibles du problème. Ceci est dû au fait que la mesure d'un qubit peut donner la valeur binaire "0" avec la probabilité $|\alpha_i|^2$ et peut aussi donner la valeur binaire "1" avec la probabilité $|\beta_i|^2$. Donc, toute chaîne binaire a potentiellement la possibilité d'être le résultat d'une opération de mesure.

Grâce à ce principe de superposition, il n'est pas nécessaire que la population soit de grande taille; un petit nombre d'individus peut facilement couvrir tout l'espace de recherche et le problème de diversification ne sera pas posé. Théoriquement, un seul chromosome suffirait puisque sa mesure peut engendrer n'importe quel état de l'espace de recherche. Cependant, l'expérience a montré qu'un meilleur comportement de l'algorithme est observé pour des tailles un peu supérieures [Han 2003].

L'algorithme consiste à faire évoluer l'ensemble des individus tout en gardant continuellement en mémoire la meilleure solution trouvée au cours des précédentes itérations. A chaque itération, l'actuelle meilleure solution sert comme guide pour rechercher de nouvelles solutions qui peuvent s'avérer meilleures. Cela est implémenté via une porte quantique, appelée la porte D, qui effectue une rotation du

qubit de façon à augmenter la probabilité d'avoir la valeur binaire du bit correspondant dans la meilleure solution actuelle (figure 4.1).

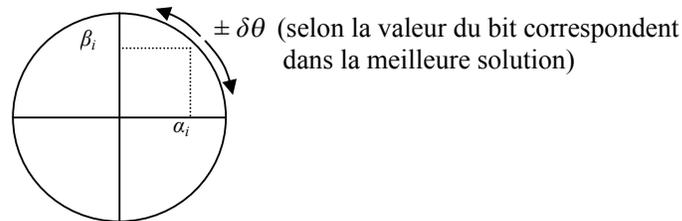


Figure 4.1. Effet de la porte D sur le qubit

La porte D prend la forme matricielle suivante:

$$\begin{pmatrix} \cos \delta\theta & -\sin \delta\theta \\ \sin \delta\theta & \cos \delta\theta \end{pmatrix}$$

Le tableau ci-dessous donne la direction de rotation en fonction de α_i , β_i et de la valeur du bit correspondant dans la meilleure solution actuelle.

α	β	Valeur du bit de référence	Angle de rotation
>0	>0	1	$+\delta\theta$
>0	>0	0	$-\delta\theta$
>0	<0	1	$-\delta\theta$
>0	<0	0	$+\delta\theta$
<0	>0	1	$-\delta\theta$
<0	>0	0	$+\delta\theta$
<0	<0	1	$+\delta\theta$
<0	<0	0	$-\delta\theta$

Tableau 4.1. Direction de la rotation

L'algorithme génétique quantique de *K-H. Han* et *J-H. Kim* est décrit comme suit:

Soient $Q(t)$ la population composée de chromosomes quantiques à l'instant t , $B(t)$ la meilleure solution jusqu'à l'instant t , et $P(t)$ l'ensemble de solutions binaires issues de la mesure de $Q(t)$.

```
t ← 0  
Produire P(t) par observation de Q(t)  
Evaluer P(t)  
Stocker la meilleure solution parmi P(t) dans B(t)  
Tant que (critère d'arrêt non satisfait) faire  
    t ← t+1  
    Produire P(t) par observation de Q(t-1)  
    Evaluer P(t)  
    Générer Q(t) en appliquant la porte D sur Q(t-1)  
    Stocker la meilleure solution parmi B(t-1) et P(t) dans B(t)  
Fin Tant que
```

Discussion

Le meilleur comportement d'un algorithme génétique, dans le cas général, résulte d'un bon équilibre entre deux phénomènes: diversification et renforcement. La diversification permet à l'algorithme d'atteindre des zones non encore explorées de l'espace de recherche et qui peuvent renfermer des solutions optimales du problème en question; tandis que le renforcement permet de stabiliser et accentuer la recherche dans des sites particuliers autour de certaines solutions. Un mauvais équilibre entre ces deux phénomènes va très probablement compromettre les chances de l'algorithme à atteindre la ou les solutions optimales du problème posé. Ainsi une diversification accrue tendrait à une recherche purement aléatoire et diminuerait donc la probabilité de trouver une solution optimale du problème. À l'inverse, un renforcement excessif de la recherche pourrait mal explorer l'espace de recherche et donc ne jamais atteindre les zones où se situent les meilleures solutions.

Dans l'algorithme génétique quantique décrit ci-dessus le codage quantique des chromosomes constitue le moyen employé par les auteurs pour garantir la diversification de la recherche. Un qubit pourrait, dans la majorité des situations,

produire lors de sa mesure la valeur "0" ou la valeur "1". Donc, un chromosome pourrait potentiellement couvrir tout l'espace de recherche.

L'opérateur de rotation, ou la porte D, est l'outil garantissant le renforcement de la recherche autour de la meilleure solution actuelle. Au cours des itérations, cet opérateur augmente, pour chaque qubit, la probabilité d'avoir la valeur du bit correspondant dans la meilleure solution. Ceci en gardant, dans la majorité des situations, la possibilité, même affaiblie, d'avoir la valeur inverse qui pourrait contribuer à la découverte d'une nouvelle meilleure solution.

Cette stratégie n'est pas tout le temps gagnante. Le problème majeur est qu'en cas de non découverte d'une nouvelle meilleure solution après un certain nombre d'itérations, la probabilité pour un qubit de générer la valeur binaire inverse de la valeur du bit correspondant dans la meilleure solution sera amplement réduite, voir même mise à 0. La généralisation de ce phénomène sur l'ensemble des qubits entraînera l'effondrement de la superposition d'états et le blocage des chromosomes dans un optimum local.

Exemple:

Supposant que la meilleure solution actuelle est:

1 0 1 1 0 0

Après un certain nombre d'itérations sans changement de la meilleure solution, un chromosome quantique peut, via l'application de la porte D, perdre de son potentiel de générer de nouvelles solutions et prendre des valeurs comme:

-0.0002 0.9997 -0.0000 0.0001 0.9999 1.0000

0.9999 0.0245 -1.0000 -1.0000 -0.0141 0.0000

La mesure de ce chromosome va donner très probablement:

1 0 1 1 0 0

et va donc être piégé dans un optimum local.

Pour faire face à ce problème, les différents paramètres de l'algorithme doivent être minutieusement choisis. Une grande valeur pour l'angle de rotation entraînerait

très souvent une perte de diversité et une convergence prématurée vers un optimum local. Tandis qu'une petite valeur va diminuer considérablement la vitesse de convergence de l'algorithme. L'autre paramètre à définir est le nombre adéquat de chromosomes. Une population constituée d'un seul chromosome a théoriquement la possibilité d'atteindre la solution optimale du problème posé grâce au principe de superposition d'états. Mais, en pratique, cela est rarement réalisé car la mesure a tendance à produire le même résultat après un certain nombre d'itérations avec la même meilleure solution comme guide de la recherche. D'autre part, une population composée d'un grand nombre d'individus diminuerait la probabilité de convergence vers un optimum local, mais entraînerait une élévation du coût de calcul. On doit donc attacher une attention particulière au choix de la taille de la population.

4.3. Nouveaux algorithmes évolutionnaires quantiques

Le problème principal des algorithmes décrits ci-dessus est celui de la perte de diversité. Pour rendre un algorithme génétique quantique moins exposé à ce problème nous avons proposé dans [Talbi et al. 2004a] l'incorporation de nouvelles opérations génétiques quantiques. Il s'agit de la mutation quantique, de la permutation quantique, du croisement quantique et de la sélection quantique.

Selon le taux de mutation, la mutation quantique s'applique sur un nombre de qubits de chaque chromosome quantique. La mutation quantique d'un qubit est réalisée via l'application de l'opérateur de négation quantique sur ce qubit. Le résultat de cette mutation est la permutation entre les coefficients α_i et β_i et donc l'inversion des probabilités d'avoir les valeurs binaires "0" et "1" suite à une opération de mesure (figure 4.2).

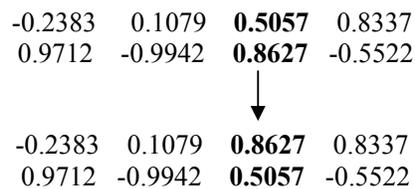


Figure 4.2. Mutation quantique

Cette opération augmenterait la probabilité d'avoir de nouvelles solutions suite à une opération de mesure et permettrait ainsi de sortir des minima locaux et explorer de nouvelles portions de l'espace de recherche.

Dans certaines catégories de problèmes, la permutation quantique pourrait jouer un rôle décisif dans la découverte de la solution optimale. Il s'agit là d'une interversion tenant lieu entre deux qubits (figure 4.3).

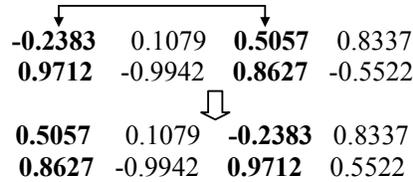


Figure 4.3. Permutation quantique

Le croisement quantique est une opération qui peut être appliquée sur chaque paire de chromosome. Cela est rendu possible et pratique grâce à la petite taille de la population quantique (3 croisements dans une population composée de 3 chromosomes, 6 croisements dans une population composée de 4 chromosomes et 10 croisements pour une population composée de 5 chromosomes).

Le croisement quantique entre deux individus (parents) en un point donné permet de générer deux nouveaux individus (descendants) dont les gènes proviennent de leurs deux parents (figure 4.4).

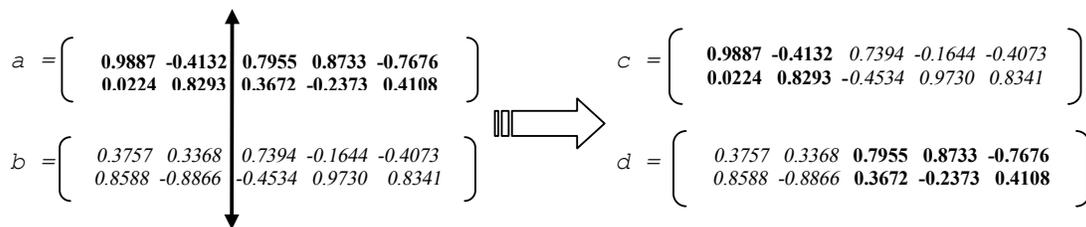


Figure 4.4. Croisement quantique

On peut aussi procéder à des croisements quantiques multipoint de la même manière.

Cette opération de croisement peut permettre de se rapprocher rapidement de l'optimum via la combinaison de fragments provenant des solutions actuelles.

Après une opération de croisement quantique, le nombre d'individus augmente. Donc, pour maintenir la taille initiale de la population, une opération de sélection quantique s'impose.

Il est difficile, bien que faisable, de considérer les caractéristiques des chromosomes comme formuler un critère de sélection. Ceci est dû au fait qu'un chromosome représente potentiellement une multitude de solutions éparpillées dans l'espace de recherche. On a donc suggéré de se baser sur la qualité de la solution binaire issue d'une opération de mesure pour évaluer un chromosome.

Une opération de mesure est appliquée sur l'ensemble des chromosomes. Une solution est obtenue de chaque chromosome et les chromosomes desquels émanent les meilleures solutions sont choisis pour former la nouvelle génération. On peut aussi garder un ou deux chromosomes qui génèrent de faibles solutions mais qui pourraient servir dans les prochaines étapes de la recherche.

4.4. Application à la résolution de problèmes d'optimisation combinatoire

4.4.1. Problème du sac à dos

Le problème du sac à dos (Knapsack Problem ou KSP) est l'un des problèmes d'optimisation combinatoire les plus connus. L'énoncé du problème est le suivant:

Ayant un sac à dos de capacité $C > 0$ et un ensemble de N objets possédant chacun un poids $w_i > 0$ et une valeur $v_i > 0$, on cherche la sélection d'objets ($x_i = 1$ si l'objet i est choisi, 0 sinon) telle que $\sum_{i=1}^N x_i w_i \leq C$ et $\sum_{i=1}^N x_i v_i$ soit maximale.

Pour appliquer les différentes variantes d'algorithmes génétiques quantiques (sans mutation ni permutation ni croisement, avec mutation seulement, avec permutation seulement, avec croisement seulement, avec mutation et croisement quantiques, etc.) à la résolution du problème du sac à dos et pour que les différentes variantes soient comparables, on a considéré une même instance de taille $N=250$. C'est une instance fortement corrélée générée de la manière suivante:

w_i : uniformément distribués sur $]0,10[$;

$$v_i = w_i + 5 ;$$

$$C = \frac{1}{2} \sum_{i=1}^N w_i .$$

On a exécuté chacune des variantes de l'algorithme 30 fois, avec 2000 générations à chaque exécution et cela pour différentes valeurs des paramètres.

Le but est de pouvoir comparer les différentes variantes et de trouver les meilleures valeurs à utiliser pour les différents paramètres.

Dans une première série de tests, on a considéré les algorithmes génétiques quantiques tels que décrits par *K-H. Han* et *J-H. Kim*. Les seuls paramètres de ces algorithmes sont le nombre de chromosomes et l'angle de rotation utilisé par la porte D.

- Effet de la valeur de l'angle de rotation

Dans les tableaux suivants sont rapportés la moyenne des résultats obtenus, le meilleur résultat et l'écart type entre les 30 résultats obtenus pour chaque valeur du nombre de chromosomes et chaque valeur de l'angle de rotation:

Nombre de chromosomes NC=1

Angle de rotation	$\pi/10$	$\pi/15$	$\pi/20$	$\pi/30$	$\pi/60$
Moyenne	1374,34	1371,57	1347,30	1272,21	1214,57
Meilleur résultat	1400,10	1400,50	1379,70	1305,80	1265,50
Ecart type	12,89	12,65	16,45	18,21	19,25

Tableau 4.2. Effet du changement de la valeur de l'angle de rotation dans KSP pour NC=1

Nombre de chromosomes NC=2

Angle de rotation	$\pi/10$	$\pi/15$	$\pi/20$	$\pi/30$	$\pi/60$
Moyenne	1396,44	1403,77	1378,94	1305,69	1208,39
Meilleur résultat	1420,80	1425,70	1400,80	1355,70	1245,80
Ecart type	9,06	12,14	15,43	19,92	24,46

Tableau 4.3. Effet du changement de la valeur de l'angle de rotation dans KSP pour NC=2

Nombre de chromosomes NC=3

Angle de rotation	$\pi/10$	$\pi/15$	$\pi/20$	$\pi/30$	$\pi/60$
Moyenne	1412,17	1420,78	1399,19	1324,95	1218,85
Meilleur résultat	1434,80	1440,80	1420,60	1356,80	1255,70
Ecart type	10,68	10,74	10,23	18,84	19,53

Tableau 4.4. Effet du changement de la valeur de l'angle de rotation dans KSP pour NC=3

Nombre de chromosomes NC=4

Angle de rotation	$\pi/10$	$\pi/15$	$\pi/20$	$\pi/30$	$\pi/60$
Moyenne	1419,31	1428,94	1408,62	1340,83	1208,43
Meilleur résultat	1435,30	1440,80	1430,10	1370,70	1250,60
Ecart type	6,90	7,71	11,40	18,58	23,93

Tableau 4.5. Effet du changement de la valeur de l'angle de rotation dans KSP pour NC=4

Nombre de chromosomes NC=5

Angle de rotation	$\pi/10$	$\pi/15$	$\pi/20$	$\pi/30$	$\pi/60$
Moyenne	1423,68	1432,81	1416,03	1349,36	1211,74
Meilleur résultat	1450,60	1450,80	1430,80	1379,90	1260,70
Ecart type	9,71	9,65	10,71	17,70	23,89

Le tableau suivant résume les moyennes des résultats obtenus:

Angle de rotation	$\pi/10$	$\pi/15$	$\pi/20$	$\pi/30$	$\pi/60$
NC=1	1374,34	1371,57	1347,30	1272,21	1214,57
NC=2	1396,44	1403,77	1378,94	1305,69	1208,39
NC=3	1412,17	1420,78	1399,19	1324,95	1218,85
NC=4	1419,31	1428,94	1408,62	1340,83	1208,43
NC=5	1423,68	1432,81	1416,03	1349,36	1211,74

Tableau 4.6. Effet du changement de la valeur de l'angle de rotation dans KSP-
Récapitulation

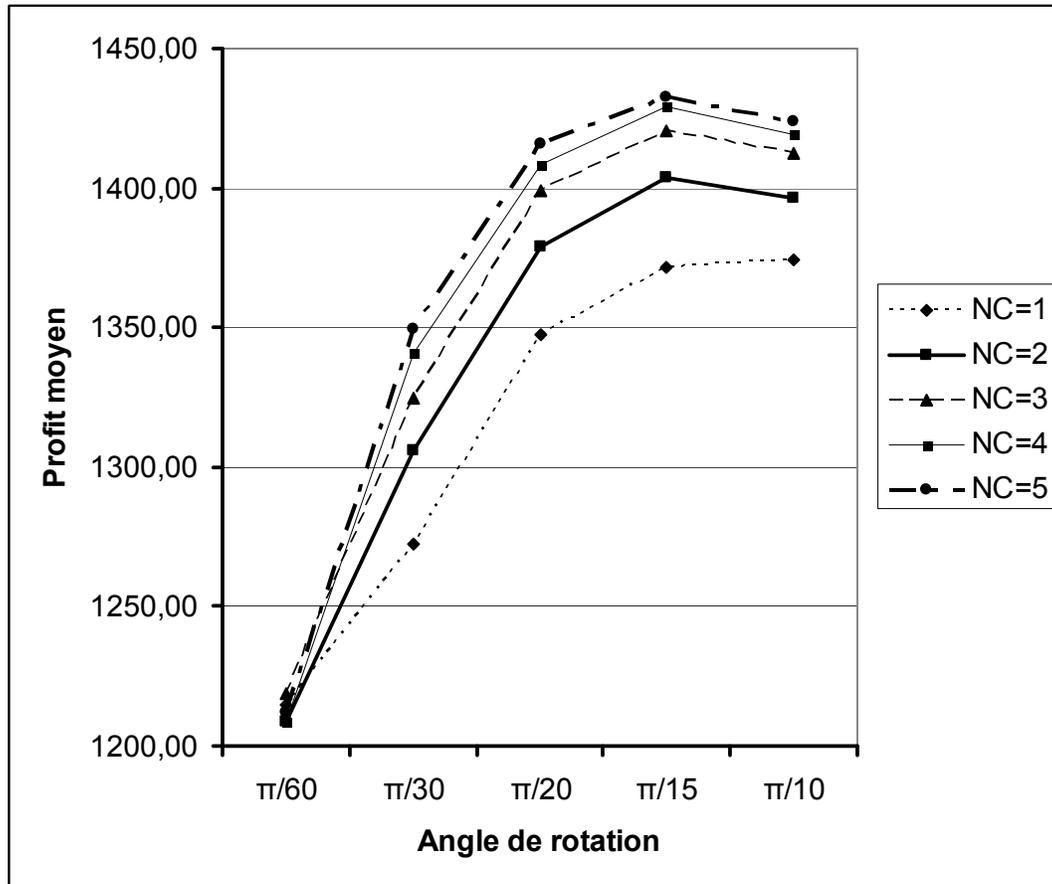


Figure 4.5. Performance moyenne d'un AGQ en fonction de l'angle de rotation

On remarque que la performance moyenne augmente avec le nombre de chromosomes et qu'un angle de rotation de $\pi/15$ radians offre un meilleur comportement par rapport à un angle de $\pi/10$, $\pi/20$ et surtout par rapport à $\pi/30$ et $\pi/60$.

On remarque aussi que l'écart type entre les valeurs obtenues pour les mêmes paramètres n'est pas très grand. On note toutefois qu'il est plus important pour les paramètres ayant donné de mauvais résultats. Cela est en grande partie causé par la convergence prématurée vers des minima locaux.

Théorème 1

Soit un algorithme génétique quantique avec un angle de rotation égal à $\delta\theta$. Si aucune amélioration du meilleur résultat n'est observée au bout d'au plus $\pi/(2.\delta\theta)$ itérations, tous les qubits seront entraînés soit à $(\sin \phi) |0\rangle + (\cos \phi) |1\rangle$ soit à $(\cos \phi) |0\rangle + (\sin \phi) |1\rangle$ sachant que $\phi \in]k.\pi - \delta\theta, k.\pi + \delta\theta[$ et $k \in \{0,1\}$.

Démonstration

Soit un qubit q_i . Le $i^{\text{ème}}$ bit dans la meilleure solution actuelle peut avoir une des deux valeurs: 0 ou 1.

Si le $i^{\text{ème}}$ bit est égal à 0, l'opération d'interférence va à chaque itération remuer le qubit q_i d'un angle $\delta\theta$ vers la valeur représentée par l'angle 0 ou π . Et comme le chemin vers cet angle ne peut excéder $\pi/2$ (les valeurs les plus loin étant $\pi/2$ et $3\pi/2$), il nous faut au plus $(\pi/2)/\delta\theta$ itérations pour le parcourir.

Si le $i^{\text{ème}}$ bit est égal à 1, l'opération d'interférence va à chaque itération remuer le qubit q_i d'un angle $\delta\theta$ vers la valeur représentée par l'angle $\pi/2$ ou $3\pi/2$. Et comme le chemin vers cet angle ne peut excéder $\pi/2$ (les valeurs les plus loin étant 0 et π), il nous faut au plus $(\pi/2)/\delta\theta$ itérations pour le parcourir.

Résultat

Un résultat du théorème 1 est qu'au bout d'au plus $\pi/(2.\delta\theta)$ itérations, la probabilité pour un qubit d'avoir une valeur différente de la valeur du bit correspondant dans la meilleure solution en cours comme produit de l'opération de mesure est au plus égale à $\sin^2 \delta\theta$. Il s'agit là d'un tirage de Bernoulli de paramètre $\sin^2 \phi$.

On aura donc pour N qubits un tirage suivant une loi proche d'une binomiale de paramètres N et $\sin^2 \delta\theta$. Soit X la variable aléatoire qui suit cette loi binomiale. $p(X=P) = C_N^P . (\sin^2 \delta\theta)^P . (\cos^2 \delta\theta)^{N-P}$, P donne le nombre de qubits donnant une valeur différente de la valeur correspondante dans la solution en cours.

Le graphique ci-dessous donne la distribution de X pour des valeurs de l'angle de rotation variant entre $\pi/180$ et $\pi/6$ avec un pas égal à $\pi/180$ (c'est-à-dire les valeurs comprises entre 1° et 30°), et P variant entre 1 et 100.

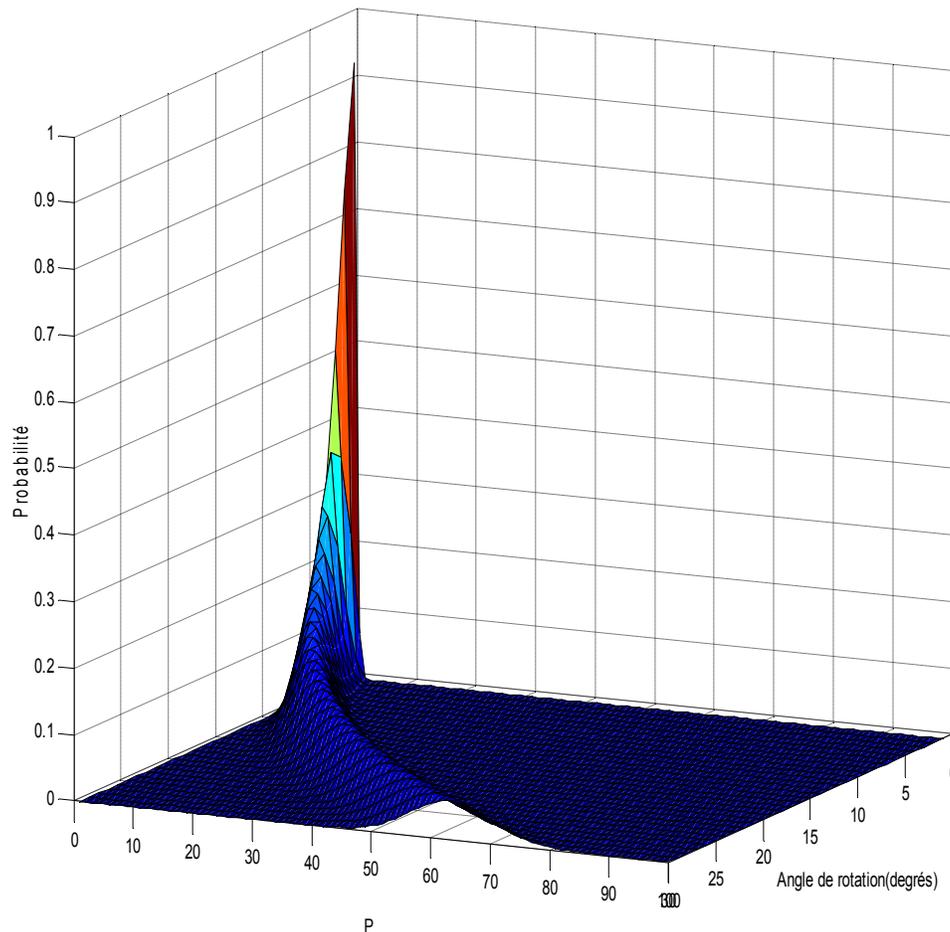


Figure 4.6. Distribution de X pour des valeurs de l'angle de rotation variant 1° et 30° et P variant entre 1 et 100

On remarque que la probabilité d'avoir un résultat de mesure plus loin de la solution en cours augmente avec la croissance de l'angle de rotation.

Dans le graphique ci-dessous (Figure 4.7) nous rapportons l'évolution sur les 2000 générations de la valeur du meilleur résultat dans la meilleure exécution pour chaque valeur de NC en prenant $\pi/15$ comme valeur de l'angle de rotation.

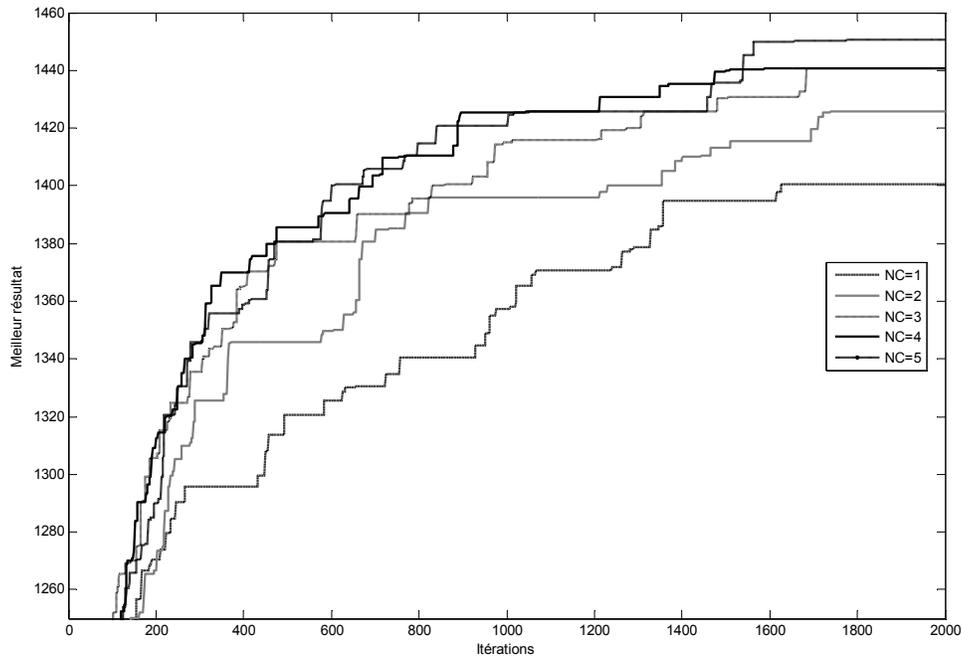


Figure 4.7. Evolution de la valeur du meilleur résultat dans la meilleure exécution

- Effet de l'opération de mutation

Dans les tableaux qui suivent, on compare le comportement, sur 30 exécutions à 2000 itérations chacune, des algorithmes génétiques quantiques avec et sans mutation pour les valeurs de l'angle de rotation $\pi/10$, $\pi/15$ et $\pi/20$. On a testé deux valeurs du taux de mutation: 0.3% et 0.5%. Ce sont les valeurs ayant donné après un nombre d'essais un meilleur comportement par rapport à d'autres valeurs prises dans l'intervalle [0.1%,5%].

$\theta=\pi/10$:

Nombre de chromosomes	Taux de mutation	Moyenne	Meilleur résultat	Ecart type
1	0,0%	1374,34	1400,10	12,89
	0,3%	1374,03	1400,70	13,27
	0,5%	1369,13	1393,90	11,54
2	0,0%	1396,44	1420,80	9,06
	0,3%	1397,69	1420,70	11,13
	0,5%	1389,26	1405,50	9,55
3	0,0%	1412,17	1434,80	10,68
	0,3%	1409,65	1425,70	7,55
	0,5%	1400,17	1420,20	11,00

Nombre de chromosomes	Taux de mutation	Moyenne	Meilleur résultat	Ecart type
4	0,0%	1419,31	1435,30	6,90
	0,3%	1413,49	1435,60	9,57
	0,5%	1409,95	1430,50	10,51
5	0,0%	1423,68	1450,60	9,71
	0,3%	1420,15	1440,80	9,18
	0,5%	1413,39	1435,50	8,57

Tableau 4.7. Effet de l'opération de mutation dans KSP pour $\theta=\pi/10$ $\theta=\pi/15$:

Nombre de chromosomes	Taux de mutation	Moyenne	Meilleur résultat	Ecart type
1	0,0%	1371,57	1400,50	12,65
	0,3%	1377,35	1400,70	14,26
	0,5%	1376,51	1404,70	13,12
2	0,0%	1403,77	1425,70	12,14
	0,3%	1405,61	1420,70	9,95
	0,5%	1397,30	1410,10	8,31
3	0,0%	1420,78	1440,80	10,74
	0,3%	1422,87	1440,80	11,43
	0,5%	1411,96	1430,50	9,77
4	0,0%	1428,94	1440,80	7,71
	0,3%	1428,37	1440,80	8,26
	0,5%	1421,78	1435,50	10,21
5	0,0%	1432,81	1450,80	9,65
	0,3%	1433,29	1455,70	10,48
	0,5%	1427,48	1445,60	9,09

Tableau 4.8. Effet de l'opération de mutation dans KSP pour $\theta=\pi/15$ $\theta=\pi/20$:

Nombre de chromosomes	Taux de mutation	Moyenne	Meilleur résultat	Ecart type
1	0,0%	1347,30	1379,70	16,45
	0,3%	1363,77	1385,10	12,31
	0,5%	1371,97	1395,20	9,86
2	0,0%	1378,94	1400,80	15,43
	0,3%	1398,27	1430,60	12,35
	0,5%	1398,61	1415,60	10,59
3	0,0%	1399,19	1420,60	10,23
	0,3%	1413,13	1425,80	9,86
	0,5%	1411,54	1430,30	8,99
4	0,0%	1408,62	1430,10	11,40
	0,3%	1421,69	1445,70	9,96
	0,5%	1418,61	1434,00	8,78
5	0,0%	1416,03	1430,80	10,71
	0,3%	1429,98	1445,70	10,33
	0,5%	1423,73	1440,80	9,59

Tableau 4.9. Effet de l'opération de mutation dans KSP pour $\theta=\pi/20$

On remarque que pour le problème du sac à dos l'opération de mutation n'apporte pas grand-chose sinon n'apporte rien pour de grands angles de rotation (pour un taux de mutation à 0.3% une seule amélioration de la moyenne des 30 exécutions sur les 5 valeurs du nombre de chromosomes pour l'angle $\pi/10$, et pour un taux de 0.5%, aucune amélioration n'a été observée). Pour des angles plus petits, l'opération de mutation donne apparemment de meilleurs résultats (pour un angle de $\pi/20$, toutes les moyennes ont été améliorées pour les deux valeurs du taux de mutation).

L'opération de mutation sert donc à faire sortir l'algorithme d'un minimum local qui peut résulter d'un petit angle de rotation.

- Effet de l'opération de permutation

Pour étudier l'effet de la permutation sur le comportement des algorithmes génétiques quantiques lors de la résolution du problème du sac à dos, on a mis en place une série de tests en prenant les angles de rotation $\pi/10$, $\pi/15$ et $\pi/20$ et deux taux de permutation, à savoir 0.5% et 1%. Les tableaux qui suivent résument les résultats de 30 exécutions pour chaque configuration avec 2000 itérations à chaque exécution.

$\theta=\pi/10$:

Nombre de chromosomes	Taux de permutation	Moyenne	Meilleur résultat	Ecart type
1	0,0%	1374,34	1400,10	12,89
	0,5%	1373,03	1390,10	10,40
	1,0%	1371,67	1395,40	10,95
2	0,0%	1396,44	1420,80	9,06
	0,5%	1398,52	1418,40	11,06
	1,0%	1396,74	1420,50	9,32
3	0,0%	1412,17	1434,80	10,68
	0,5%	1409,13	1425,50	8,23
	1,0%	1409,88	1429,00	8,77
4	0,0%	1419,31	1435,30	6,90
	0,5%	1416,58	1435,50	9,17
	1,0%	1419,43	1435,70	7,06
5	0,0%	1423,68	1450,60	9,71
	0,5%	1419,33	1435,10	8,07
	1,0%	1423,08	1440,10	8,19

Tableau 4.10. Effet de l'opération de permutation dans KSP pour $\theta=\pi/10$

$\theta=\pi/15$:

Nombre de chromosomes	Taux de permutation	Moyenne	Meilleur résultat	Ecart type
1	0,0%	1371,57	1400,50	12,65
	0,5%	1387,47	1420,80	13,83
	1,0%	1387,70	1420,70	16,09
2	0,0%	1403,77	1425,70	12,14
	0,5%	1410,94	1425,40	9,79
	1,0%	1413,75	1435,10	9,44
3	0,0%	1420,78	1440,80	10,74
	0,5%	1423,66	1440,70	9,93
	1,0%	1422,27	1440,30	9,05
4	0,0%	1428,94	1440,80	7,71
	0,5%	1431,66	1445,70	6,43
	1,0%	1429,98	1445,60	8,38
5	0,0%	1432,81	1450,80	9,65
	0,5%	1433,89	1450,40	7,49
	1,0%	1437,04	1450,60	6,76

Tableau 4.11. Effet de l'opération de permutation dans KSP pour $\theta=\pi/15$ $\theta=\pi/20$:

Nombre de chromosomes	Taux de permutation	Moyenne	Meilleur résultat	Ecart type
1	0,0%	1347,30	1379,70	16,45
	0,5%	1385,31	1415,40	14,02
	1,0%	1379,98	1405,30	12,79
2	0,0%	1378,94	1400,80	15,43
	0,5%	1412,14	1440,50	11,94
	1,0%	1410,72	1425,00	10,12
3	0,0%	1399,19	1420,60	10,23
	0,5%	1426,41	1445,50	7,84
	1,0%	1424,93	1445,80	8,72
4	0,0%	1408,62	1430,10	11,40
	0,5%	1433,03	1450,80	7,95
	1,0%	1434,74	1450,70	8,96
5	0,0%	1416,03	1430,80	10,71
	0,5%	1434,26	1455,50	8,05
	1,0%	1438,22	1460,80	8,50

Tableau 4.12. Effet de l'opération de permutation dans KSP pour $\theta=\pi/20$

On remarque qu'à l'exception de l'angle de rotation $\pi/10$ où l'algorithme se comporte mieux sans permutation, il y a une amélioration de la moyenne des résultats obtenus pour les angles $\pi/15$ et $\pi/20$ et pour les deux valeurs prises comme taux de permutation.

- Effet de l'opération de croisement

Les tableaux qui suivent résument l'effet de l'opération du croisement quantique sur le comportement des algorithmes génétiques quantiques lors de la résolution du problème du sac à dos. On a pris les angles de rotation $\pi/10$, $\pi/15$ et $\pi/20$ (avec, pour chaque configuration, 30 exécutions de 2000 itérations chacune).

$\theta=\pi/10$:

Nombre de chromosomes	Croisement?	Moyenne	Meilleur résultat	Ecart type
2	Non	1396,44	1420,80	9,06
	Oui	1415,53	1434,20	11,54
3	Non	1412,17	1434,80	10,68
	Oui	1434,27	1455,80	9,30
4	Non	1419,31	1435,30	6,90
	Oui	1447,06	1460,70	5,88
5	Non	1423,68	1450,60	9,71
	Oui	1454,50	1465,50	4,31

Tableau 4.13. Effet de l'opération de croisement dans KSP pour $\theta=\pi/10$

$\theta=\pi/15$:

Nombre de chromosomes	Croisement?	Moyenne	Meilleur résultat	Ecart type
2	Non	1403,77	1425,70	12,14
	Oui	1424,77	1435,80	7,67
3	Non	1420,78	1440,80	10,74
	Oui	1444,22	1460,80	9,01
4	Non	1428,94	1440,80	7,71
	Oui	1453,56	1470,80	6,53
5	Non	1432,81	1450,80	9,65
	Oui	1461,12	1470,80	5,86

Tableau 4.14. Effet de l'opération de croisement dans KSP pour $\theta=\pi/15$

$\theta=\pi/20$:

Nombre de chromosomes	Croisement?	Moyenne	Meilleur résultat	Ecart type
2	Non	1378,94	1400,80	15,43
	Oui	1410,61	1435,70	9,57
3	Non	1399,19	1420,60	10,23
	Oui	1434,07	1450,80	8,89
4	Non	1408,62	1430,10	11,40
	Oui	1443,83	1460,80	8,00
5	Non	1416,03	1430,80	10,71
	Oui	1452,27	1465,80	5,75

Tableau 4.15. Effet de l'opération de croisement dans KSP pour $\theta=\pi/20$

On remarque que pour tous les tests, l'opération de croisement quantique améliore considérablement les résultats obtenus. L'apport de cette opération est meilleur que celui des deux opérations précédentes (mutation et permutation quantiques).

- Effet de la combinaison de la permutation avec le croisement quantique

$\theta=\pi/10$:

Nombre de chromosomes	Taux de permutation	Croisement?	Moyenne	Meilleur résultat	Ecart type
2	0,0%	Oui	1415,53	1434,20	11,54
	0,5%	Oui	1416,11	1429,90	8,34
	1,0%	Oui	1407,85	1433,30	9,43
3	0,0%	Oui	1434,27	1455,80	9,30
	0,5%	Oui	1435,43	1450,60	8,19
	1,0%	Oui	1428,79	1440,70	8,23
4	0,0%	Oui	1447,06	1460,70	5,88
	0,5%	Oui	1447,25	1465,30	7,99
	1,0%	Oui	1441,25	1450,60	5,78
5	0,0%	Oui	1454,50	1465,50	4,31
	0,5%	Oui	1451,13	1460,80	6,62
	1,0%	Oui	1450,41	1470,20	6,84

Tableau 4.16. Effet de la combinaison de la permutation avec le croisement dans KSP pour $\theta=\pi/10$

$\theta=\pi/15$:

Nombre de chromosomes	Taux de permutation	Croisement?	Moyenne	Meilleur résultat	Ecart type
2	0,0%	Oui	1424,77	1435,80	7,67
	0,5%	Oui	1433,57	1450,70	7,58
	1,0%	Oui	1430,19	1445,30	9,46
3	0,0%	Oui	1444,22	1460,80	9,01
	0,5%	Oui	1452,66	1470,80	6,89
	1,0%	Oui	1448,33	1460,60	5,34
4	0,0%	Oui	1453,56	1470,80	6,53
	0,5%	Oui	1462,24	1470,80	4,58
	1,0%	Oui	1459,60	1470,40	4,55
5	0,0%	Oui	1461,12	1470,80	5,86
	0,5%	Oui	1467,92	1475,80	5,68
	1,0%	Oui	1467,04	1480,60	5,68

Tableau 4.17. Effet de la combinaison de la permutation avec le croisement dans KSP pour $\theta=\pi/15$

$\theta=\pi/20$:

Nombre de chromosomes	Taux de permutation	Croisement?	Moyenne	Meilleur résultat	Ecart type
2	0,0%	Oui	1410,61	1435,70	9,57
	0,5%	Oui	1438,97	1450,50	7,02
	1,0%	Oui	1429,78	1450,70	8,87
3	0,0%	Oui	1434,07	1450,80	8,89
	0,5%	Oui	1460,39	1470,80	5,85
	1,0%	Oui	1451,91	1465,70	6,47
4	0,0%	Oui	1443,83	1460,80	8,00
	0,5%	Oui	1468,95	1475,80	5,17
	1,0%	Oui	1466,17	1480,70	5,61
5	0,0%	Oui	1452,27	1465,80	5,75
	0,5%	Oui	1473,79	1485,80	5,65
	1,0%	Oui	1473,54	1480,80	5,04

Tableau 4.18. Effet de la combinaison de la permutation avec le croisement dans KSP pour $\theta=\pi/20$

On remarque que la l'intégration aux algorithmes génétiques quantiques des deux opérations de permutation et de croisement quantiques améliore encore plus les performances de ces algorithmes, surtout avec le taux de permutation 0.5%.

4.4.2. Problème du voyageur de commerce

Le problème du voyageur de commerce (Travelling Salesman Problem ou TSP) est un problème célèbre utilisé très fréquemment pour évaluer les techniques d'optimisation combinatoire. Ce problème consiste à trouver le chemin Hamiltonien le plus court que doit suivre un voyageur de commerce pour visiter N villes (figure 4.8).

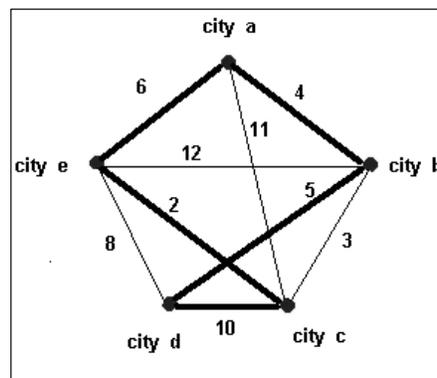


Figure 4.8. Exemple du problème de voyageur de commerce

Les applications de ce problème sont nombreuses [Gutin et Punnen 2006]: optimisation de parcours d'un robot, conception de circuits électroniques, passage de trains sur une voie, atterrissage d'avions, processus de fabrication en industrie chimique, etc.

Le problème du voyageur de commerce est dit symétrique quand le coût pour aller de la ville i à la ville j est le même pour aller de j vers i . Autrement, le problème est dit asymétrique.

Quand le nombre de villes augmente, la recherche exhaustive de la meilleure solution devient hors de portée des capacités de calcul actuelles. Pour un problème à $N=24$ villes par exemple, une recherche exhaustive doit évaluer $(N-1)! / 2 = 12.926.008.369.442.488.320.000$ chemins possibles si le problème est symétrique. Ceci nécessiterait plus de 400 000 années sur une machine capable d'évaluer un milliard chemins par seconde. Ce temps est multiplié par 2 si le problème est asymétrique.

On a testé de nouveaux algorithmes génétiques quantiques sur des instances de différentes tailles du problème du voyageur de commerce [Talbi et al. 2004c]. La symétrie de l'instance n'a pas d'importance dans la solution développée qui traitent aussi bien les instances symétriques que celles asymétriques.

L'entrée de l'algorithme est une matrice D constituée de $N \times N$ éléments. L'élément $D[i,j]$ indique la distance entre la ville i et la ville j . La matrice de la figure 4.9 donne les distances de l'instance "gr24.tsp" [TSPLIB95].

La solution d'une instance de TSP de N villes peut être représentée par une matrice binaire A de taille $N \times N$ associant à chaque ville son rang dans l'ordre de visite. Autrement dit, $A[i,j]$ prendra la valeur 1 si j est la $i^{\text{ème}}$ ville à visiter, sinon $A[i,j]$ prendra la valeur 0.

1	257	187	91	150	80	130	134	243	185	214	70	272	219	293	54	211	290	268	261	175	250	192	121
257	0	196	228	112	196	167	154	209	86	223	191	180	83	50	219	74	139	53	43	128	99	228	142
187	196	0	158	96	88	59	63	286	124	49	121	315	172	232	92	81	98	138	200	76	89	235	99
91	228	158	0	120	77	101	105	159	156	185	27	188	149	264	82	182	261	239	232	146	221	108	84
150	112	96	120	0	63	56	34	190	40	123	83	193	79	148	119	105	144	123	98	32	105	119	35
80	196	88	77	63	0	25	29	216	124	115	47	245	139	232	31	150	176	207	200	76	189	165	29
130	167	59	101	56	25	0	22	229	95	86	64	258	134	203	43	121	164	178	171	47	160	178	42
134	154	63	105	34	29	22	0	225	82	90	68	228	112	190	58	108	136	165	131	30	147	154	36
243	209	286	159	190	216	229	225	0	207	313	173	29	126	248	238	310	389	367	166	222	349	71	220
185	86	124	156	40	124	95	82	207	0	151	119	159	62	122	147	37	116	86	90	56	76	136	70
214	223	49	185	123	115	86	90	313	151	0	148	342	199	259	84	160	147	187	227	103	138	262	126
70	191	121	27	83	47	64	68	173	119	148	0	209	153	227	53	145	224	202	195	109	184	110	55
272	180	315	188	193	245	258	228	29	159	342	209	0	97	219	267	196	275	227	137	225	235	74	249
219	83	172	149	79	139	134	112	126	62	199	153	97	0	134	170	99	178	130	69	104	138	96	104
293	50	232	264	148	232	203	190	248	122	259	227	219	134	0	255	125	154	68	82	164	114	264	178
54	219	92	82	119	31	43	58	238	147	84	53	267	170	255	0	173	190	230	223	99	212	187	60
211	74	81	182	105	150	121	108	310	37	160	145	196	99	125	173	0	79	57	90	57	39	182	96
290	139	98	261	144	176	164	136	389	116	147	224	275	178	154	190	79	0	86	176	112	40	261	175
268	53	138	239	123	207	178	165	367	86	187	202	227	130	68	230	57	86	0	90	114	46	239	153
261	43	200	232	98	200	171	131	166	90	227	195	137	69	82	223	90	176	90	0	134	136	165	146
175	128	76	146	32	76	47	30	222	56	103	109	225	104	164	99	57	112	114	134	0	96	151	47
250	99	89	221	105	189	160	147	349	76	138	184	235	138	114	212	39	40	46	136	96	0	221	135
192	228	235	108	119	165	178	154	71	136	262	110	74	96	264	187	182	261	239	165	151	221	0	169
121	142	99	84	35	29	42	36	220	70	126	55	249	104	178	60	96	175	153	146	47	135	169	0

Figure 4.9. La matrice des distances de l'instance "gr24.tsp"

La solution de l'exemple de la figure 4.8 est représentée comme suit:

1	0	0	0	0
0	1	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	0	0	1

Figure 4.10. La représentation de la solution de l'instance de la figure 4.8

Pour traiter le problème du voyageur de commerce, on a apporté quelques modifications aux algorithmes génétiques quantiques décrits dans les sections précédentes dans le but de générer toujours des solutions structurellement acceptables, c'est-à-dire des chemins Hamiltoniens.

On commence par générer aléatoirement une population composée de NC chromosomes quantiques, constitué chacun de NxN qubits (N est le nombre de villes à visiter) (figure 4.11).

α_{11}	α_{12}	α_{13}	α_{14}	α_{1N}
β_{11}	β_{12}	β_{13}	β_{14}	β_{1N}
α_{21}	α_{22}	α_{23}	α_{24}	α_{2N}
β_{21}	β_{22}	β_{23}	β_{24}	β_{2N}
α_{31}	α_{32}	α_{33}	α_{34}	α_{3N}
β_{31}	β_{32}	β_{33}	β_{34}	β_{3N}
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
α_{N1}	α_{N2}	α_{N3}	α_{N4}	α_{NN}
β_{N1}	β_{N2}	β_{N3}	β_{N4}	β_{NN}

Figure 4.11. Codage génétique quantique

A partir de la population initiale, on applique cycliquement quatre opérations suivies d'une mesure (figure 4.12).

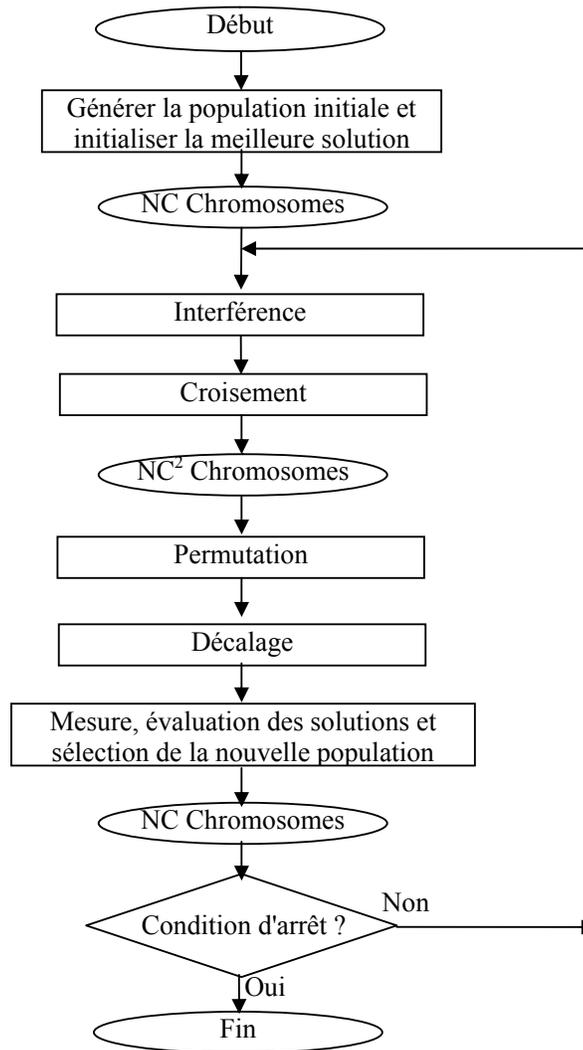


Figure 4.12. Schéma général de l'algorithme

La figure 4.13 donne un exemple de croisement entre deux chromosomes de taille 4x4.

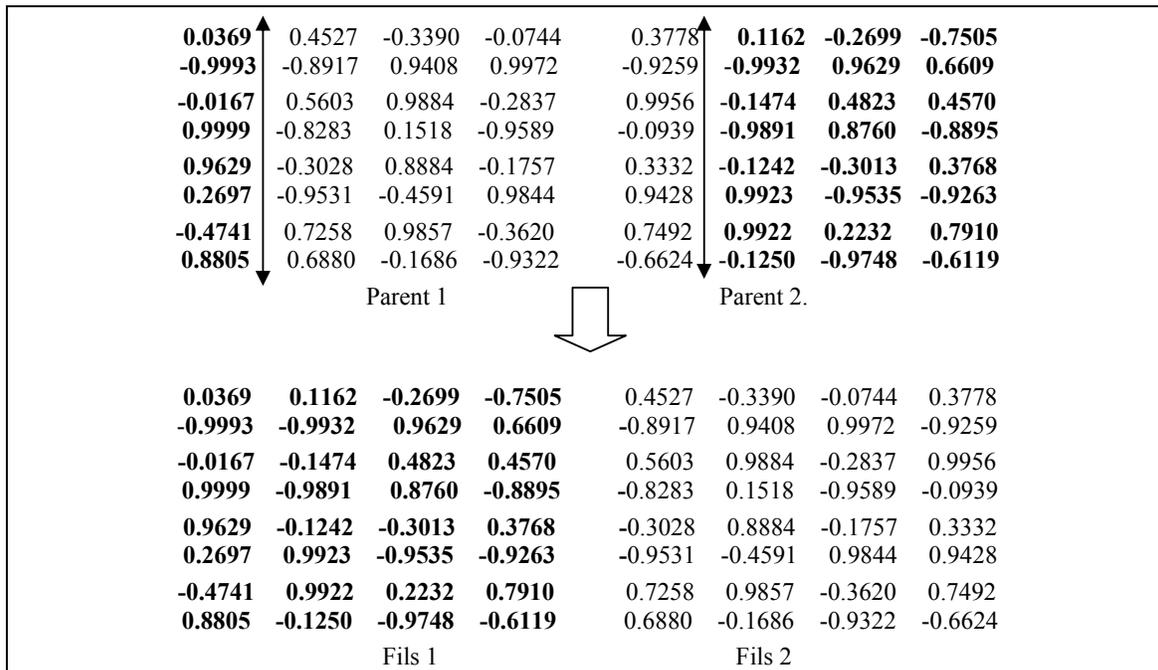


Figure 4.13. Opération de croisement

L'opération de permutation utilisée dans cet algorithme touche quelques chromosomes et s'effectue entre deux lignes (et non pas deux qubits). Cette opération obéit à deux paramètres: *Seuil_de_Permutation*, qui détermine la probabilité selon laquelle est prise la décision de faire des permutations au sein d'un chromosome, et *Probabilité_de_Permutation*, qui détermine pour chaque ligne si elle doit être permuée avec une autre ou pas. Durant les tests, on a choisi ce dernier paramètre de telle sorte à avoir en moyenne une ou deux permutations dans un chromosome.

```
Procédure Permuter
{Pour chaque chromosome chr
  Générer un nombre aléatoire réel  $r \in [0,1]$ 
  Si ( $r \leq \text{Seuil\_de\_Permutation}$ )
    Pour chaque ligne  $i$ 
      Générer un nombre aléatoire réel  $r \in [0,1]$ 
      Si ( $r \leq \text{Probabilité\_de\_Permutation}$ )
        Générer un nombre aléatoire  $j \in [1,N]$ 
        Permuter les lignes  $i$  et  $j$  de chr
      FinSi
    FinPour
  FinSi
FinPour }
```

L'opération de "décalage" est utilisée pour déplacer en bloc un ensemble de lignes vers une nouvelle position. Elle est effectuée comme suit:

```
Procédure Décalage
{Pour chaque chromosome
  Générer un nombre aléatoire réel  $r \in [0,1]$ 
  Si ( $r \leq \text{Seuil\_de\_décalage}$ )
    Générer un nombre aléatoire Nbr  $\in [1,N/3]$ 
    Générer une position aléatoire  $i \in [1,N-Nbr]$ 
    Générer un nombre aléatoire step  $\in [1,N-Nbr-i]$ 
    Décaler Nbr lignes à partir de la position  $i$  selon step.
  FinSi
FinPour }
```

Pour extraire une solution cohérente parmi celles présentes en superposition dans un chromosome quantique, on a besoin d'un type spécial de mesure. Cette mesure doit générer une solution ayant exactement une valeur à 1 par ligne et exactement une valeur à 1 par colonne (pour que le circuit soit Hamiltonien). Ceci est fait de la manière suivante:

```
Procédure Mesure
{Initialiser la matrice solution sol avec des zéros
 créer une copie ch du chromosome chr
      //pour préserver les valeurs originales
Générer un nombre aléatoire  $r \in [1, N]$ 
Pour  $i \leftarrow r$  à  $[(r+N-1) \text{ modulo } N]$ 
  Générer un nombre aléatoire  $j \in [1, N]$ 
  Générer un nombre aléatoire  $p \in [0, 1]$ 
  TantQue ( $p > |\beta_{ij}|^2$ )
     $j \leftarrow j+1 \text{ modulo } N$ 
    Générer un nombre aléatoire  $p \in [0, 1]$ 
  FinTantQue
   $\text{sol}[i, j] \leftarrow 1$ 
  Pour  $k \leftarrow 1$  to  $N$ 
     $\beta_{kj} \leftarrow 0$ 
     $\alpha_{kj} \leftarrow 1$ 
     $\beta_{ik} \leftarrow 0$ 
     $\alpha_{ik} \leftarrow 1$ 
  FinPour
FinPour }
```

Après l'opération de mesure, on garde lors de l'opération de la sélection, parmi les NC^2 chromosomes, les $(NC-1)$ chromosomes d'où proviennent les $(NC-1)$ meilleures solutions (donnant les plus courts chemins), en plus d'un chromosome choisi aléatoirement parmi les autres (pour garder plus de divergence).

L'application de la démarche ci-dessus à la résolution de certaines instances obtenues du site de référence du TSP [TSPLIB95] a donné les résultats résumés dans le tableau suivant:

N° Test	"Burma14"		"Gr17"		"Gr21"		"Gr24"	
	Résultat	Déviatio	Résultat	Déviatio	Résultat	Déviatio	Résultat	Déviatio
1	3087	0 %	2103	0,86%	2782	2,77%	1272	0,00%
2	3087	0 %	2120	1,68%	2801	3,47%	1289	1,34%
3	3087	0 %	2120	1,68%	2851	5,32%	1314	3,30%
4	3087	0 %	2085	0,00%	2707	0,00%	1289	1,34%
5	3087	0 %	2103	0,86%	2801	3,47%	1304	2,52%
6	3087	0 %	2085	0,00%	2707	0,00%	1289	1,34%
7	3087	0 %	2085	0,00%	2782	2,77%	1289	1,34%
8	3087	0 %	2120	1,68%	2707	0,00%	1272	0,00%
9	3087	0 %	2085	0,00%	2707	0,00%	1304	2,52%
10	3087	0 %	2090	0,24%	2707	0,00%	1278	0,47%
Sol. Opt.	3087		2085		2707		1272	
Min	3087		2085		2707		1272	
μ	3087		2099,6		2755,2		1290	
σ	0		15,69		54,18		14	
Max	3087		2120		2851		1314	
Max. Iter.	400		1300		1800		4000	
NC	4		4		4		4	
Δθ	$\pi/25$		$\pi/25$		$\pi/25$		$\pi/25$	
PT	0.1		0.1		0.1		0.1	
PP	0.10		0.08		0.08		0.06	
ST	0.25		0.25		0.25		0.25	
Sol. Opt.: la solution optimale; μ: moyenne des résultats obtenus; σ: écart type; Max. Iter.: nombre maximum d'itérations; Δθ: Angle de rotation; PT: Seuil_de_Permutation; PP: Probabilité_de_Permutation; ST: Seuil_de_Décalage.								

Tableau 4.19. Résumé des résultats de l'application des AEQ pour la résolution du TSP

On remarque que les résultats obtenus sont remarquablement bons, surtout en comparaison par rapport aux algorithmes génétiques standard qui nécessitent un grand nombre de chromosomes et des dizaines de milliers d'itérations [Reeves et Rowe 2003]. Les paramètres qui figurent dans le tableau ont été choisis expérimentalement après une longue série de tests que j'ai jugé peu utile de les présenter ici pour ne pas alourdir le rapport.

4.4.3. Problème des N-Reines

Le problème des N reines (N-Queens Problem ou NQP) est une généralisation du problème des 8 reines, un problème d'optimisation combinatoire introduit en

premier lieu par *Max Bezzel*, un joueur d'échecs, en 1848. Depuis 1850, ce problème a attiré l'attention de fameux mathématiciens, notamment *Gauss*, *Polya* et *Lucas*. Il a été utilisé au début des années 1990, dans le jeu sur ordinateur "*The 7th Guest*" (Le Septième Invité).

Le NQP consiste à placer N reines sur un échiquier de dimension $N \times N$ de telle façon qu'aucune reine ne puisse être capturée par une autre. Ceci dit qu'on ne doit pas avoir plus qu'une reine sur une ligne, une colonne ou une diagonale. Une solution du problème des 8 reines est illustrée dans la figure 4.14.

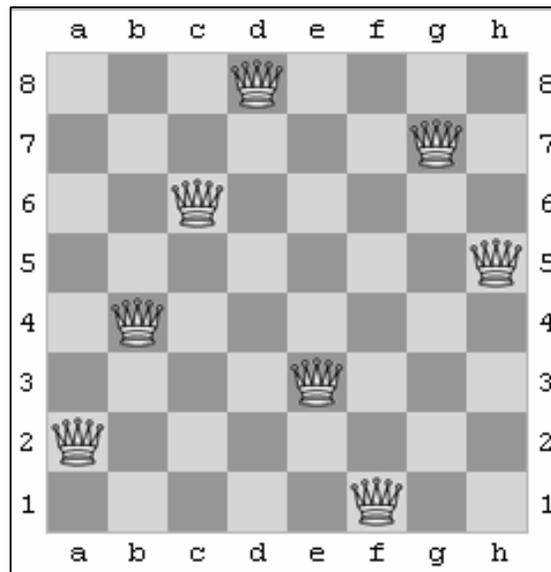


Figure 4.14. Une solution du problème des 8 reines

On a appliqué les différentes variantes des algorithmes génétiques quantiques pour la résolution de problèmes de tailles allant de 4 à 16 en adoptant le même codage que celui utilisé pour résoudre le TSP [Draa et al. 2005].

On a exécuté l'algorithme 30 fois pour chaque configuration, avec seulement 3 chromosomes et un nombre maximum d'itérations de seulement 500.

Dans toutes les exécutions l'angle de rotation choisi est égal à $\pi/20$.

La fonction objectif utilisée pour évaluer un résultat est égale au nombre de paires de reines pouvant capturer l'une l'autre.

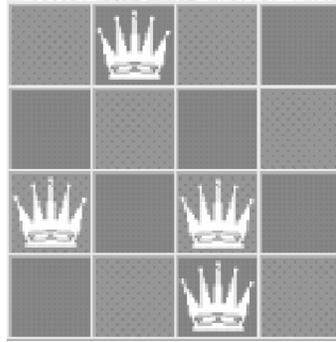


Figure 4.15. Exemple d'une configuration à 4 reines

Dans la solution de la figure 4.15, la fonction objectif est égale à deux.

Les tableaux suivants résument les résultats obtenus en fonction de différentes valeurs des probabilités de mutation et de permutation, et de l'application ou non de l'opérateur de croisement.

Pour chaque configuration, on donne le nombre de succès (NS), la moyenne des nombres d'itérations pour les exécutions qui ont abouti à un succès (MoyIt) et la moyenne des meilleures valeurs qu'a atteint la fonction objectif pour les exécutions n'ayant pas converger vers un succès au bout des 500 itérations (MoyFit).

N			4			5			6			7		
Mutation	Croisement?	Permutation	NS	MoyIt	MoyFit									
0%	Non	0%	30	43,9	/	30	57,6	/	10	96,2	1,0	19	204,4	1,1
1%	Non	0%	30	18,9	/	30	23,4	/	19	127,2	1,0	29	84,1	1,0
3%	Non	0%	30	10,7	/	30	16,9	/	26	116,6	1,0	29	90,7	1,0
5%	Non	0%	30	10,1	/	30	9,9	/	28	105,4	1,0	30	49,0	/
0%	Non	1%	30	27,2	/	30	37,8	/	15	186,3	1,0	25	100,3	1,0
0%	Non	3%	30	15,0	/	30	14,9	/	19	149,7	1,0	30	71,2	/
0%	Non	5%	30	12,1	/	30	19,4	/	25	161,4	1,0	29	115,7	1,0
1%	Non	3%	30	12,7	/	30	16,6	/	26	117,5	1,0	30	98,1	/
1%	Non	1%	30	11,3	/	30	12,4	/	18	136,1	1,0	28	88,5	1,0
0%	Oui	0%	30	18,5	/	30	47,8	/	11	121,0	1,0	21	177,0	1,0
0%	Oui	1%	30	32,8	/	30	27,3	/	15	175,6	1,0	25	112,8	1,0
0%	Oui	3%	30	15,9	/	30	19,6	/	25	174,6	1,0	30	103,0	/
0%	Oui	5%	30	10,1	/	30	15,8	/	24	136,7	1,0	30	110,4	/
1%	Oui	3%	30	11,4	/	30	15,8	/	23	193,5	1,0	30	65,8	/
2%	Oui	2%	30	10,3	/	30	13,1	/	26	146,7	1,0	29	65,3	1,0

Tableau 4.20. Résumé des résultats de l'application des AEQ pour la résolution du problème des N-reines pour N=4 à 7

N			8			9			10		
Mutation	Croisement?	Permutation	NS	Moylt	MoyFit	NS	Moylt	MoyFit	NS	Moylt	MoyFit
0%	Non	0%	18	153,6	1,1	14	163,1	1,0	14	256,4	1,0
1%	Non	0%	24	92,3	1,0	20	118,7	1,0	18	178,1	1,0
3%	Non	0%	26	101,5	1,0	21	190,2	1,0	18	217,4	1,0
5%	Non	0%	26	97,2	1,0	24	192,0	1,0	17	181,3	1,0
0%	Non	1%	21	178,3	1,0	24	188,9	1,2	11	209,8	1,1
0%	Non	3%	22	98,4	1,0	16	163,6	1,0	14	212,9	1,0
0%	Non	5%	23	108,2	1,0	23	176,5	1,0	14	217,1	1,0
1%	Non	3%	28	117,8	1,0	20	173,9	1,0	19	247,4	1,0
1%	Non	1%	21	131,7	1,0	23	169,0	1,0	13	181,8	1,0
0%	Oui	0%	22	174,0	1,0	18	182,8	1,1	12	227,7	1,0
0%	Oui	1%	23	160,7	1,0	21	191,5	1,0	10	231,0	1,0
0%	Oui	3%	27	118,7	1,0	19	152,0	1,0	14	210,0	1,0
0%	Oui	5%	23	188,7	1,0	22	164,7	1,0	15	183,7	1,1
1%	Oui	3%	19	105,3	1,0	25	91,5	1,0	10	218,9	1,0
2%	Oui	2%	27	150,0	1,0	22	138,0	1,0	16	153,8	1,0

Tableau 4.21. Résumé des résultats de l'application des AEQ pour la résolution du problème des N-reines pour N=8 à 10

N			11			12			13		
Mutation	Croisement?	Permutation	NS	Moylt	MoyFit	NS	Moylt	MoyFit	NS	Moylt	MoyFit
0%	Non	0%	9	250,6	1,0	7	267,3	1,2	9	222,9	1,2
1%	Non	0%	14	223,6	1,0	10	233,4	1,0	6	228,8	1,1
3%	Non	0%	7	156,1	1,0	10	277,0	1,1	4	212,0	1,1
5%	Non	0%	11	308,2	1,1	5	247,6	1,1	4	380,8	1,3
0%	Non	1%	10	257,8	1,1	8	253,6	1,0	7	214,0	1,1
0%	Non	3%	12	201,1	1,0	9	212,3	1,0	11	227,1	1,1
0%	Non	5%	13	198,4	1,0	10	195,4	1,0	8	279,8	1,0
1%	Non	3%	11	194,8	1,1	8	180,9	1,0	6	300,3	1,1
1%	Non	1%	12	200,8	1,1	11	185,2	1,0	6	356,5	1,1
0%	Oui	0%	6	247,3	1,1	8	210,0	1,0	6	254,8	1,1
0%	Oui	1%	10	284,1	1,0	11	260,2	1,1	12	235,3	1,2
0%	Oui	3%	11	209,1	1,1	15	184,4	1,0	7	311,1	1,1
0%	Oui	5%	9	172,3	1,0	12	288,3	1,0	7	282,9	1,2
1%	Oui	3%	13	222,1	1,0	3	180,3	1,0	9	226,2	1,1
2%	Oui	2%	6	265,8	1,1	4	221,0	1,0	8	320,6	1,0

Tableau 4.22. Résumé des résultats de l'application des AEQ pour la résolution du problème des N-reines pour N=11 à 13

N			14			15			16		
Mutation	Croisement?	Permutation	NS	Moylt	MoyFit	NS	Moylt	MoyFit	NS	Moylt	MoyFit
0%	Non	0%	4	207,0	1,2	7	286,7	1,3	4	435,8	1,2
1%	Non	0%	6	315,3	1,3	5	279,6	1,3	5	286,0	1,3
3%	Non	0%	4	331,0	1,3	2	332,0	1,6	4	258,5	1,7
5%	Non	0%	1	320,0	1,4	0	/	1,8	0	/	1,9
0%	Non	1%	11	221,2	1,3	9	256,8	1,1	7	237,7	1,2
0%	Non	3%	8	338,9	1,2	7	281,0	1,2	11	344,9	1,2
0%	Non	5%	7	380,9	1,1	6	241,7	1,3	5	279,0	1,4
1%	Non	3%	6	197,2	1,3	5	217,0	1,4	1	326,0	1,5
1%	Non	1%	7	183,1	1,2	4	398,8	1,2	2	232,5	1,3
0%	Oui	0%	10	272,7	1,2	5	269,0	1,2	5	433,6	1,4
0%	Oui	1%	5	320,2	1,1	8	280,8	1,2	3	132,3	1,2
0%	Oui	3%	7	202,3	1,1	5	264,6	1,1	4	230,5	1,3
0%	Oui	5%	4	165,0	1,5	3	395,7	1,3	2	292,0	1,5
1%	Oui	3%	3	402,7	1,2	2	264,5	1,2	1	165,0	1,4
2%	Oui	2%	1	352,0	1,2	2	396,5	1,4	4	332,5	1,7

Tableau 4.23. Résumé des résultats de l'application des AEQ pour la résolution du problème des N-reines pour N=14 à 16

Les résultats obtenus montrent l'efficacité des algorithmes génétiques quantiques dans la résolution du NQP, surtout avec les nouveaux opérateurs que nous avons introduit. Trois chromosomes et quelques centaines d'itérations sont capables dans une bonne partie des cas d'aboutir à la solution optimale d'un problème de taille importante.

4.5. Conclusion

Dans ce chapitre, on a exposé les algorithmes évolutionnaires quantiques avec une analyse en profondeur de leur comportement. On a aussi prouvé théoriquement et empiriquement l'utilité des opérateurs que nous avons intégré au modèle de base. Un meilleur équilibre entre intensification et diversification de la recherche est obtenu avec les nouveaux opérateurs. Les algorithmes développés pour traiter différents problèmes connus d'optimisation combinatoire ont été présentés et les résultats obtenus ont été très satisfaisants aussi bien en terme de qualité qu'en terme de nombre de chromosomes et nombre de générations nécessaires.

Dans le prochain chapitre, les algorithmes évolutionnaires quantiques seront exploités dans le domaine de la vision par ordinateur. Il sera question en premier lieu d'opérer différents types de recalage avant de traiter le problème de la segmentation d'images avec prise en considération de l'aspect multiobjectif.

Chapitre 5

Algorithmes évolutionnaires quantiques pour le recalage et la segmentation multiobjectif d'images

5.1. Introduction

Les problèmes liés à la vision par ordinateur sont d'un niveau de difficulté considérable car à la différence des êtres humains et des animaux dotés d'un système visuel performant, la machine acquiert l'image initialement comme simple matrice de points n'ayant aucune relation apparente entre leurs valeurs colorimétriques. La complexité de la perception de l'image réside en premier lieu dans la difficulté de déterminer toutes les relations entre ses éléments pour en extraire le maximum d'informations.

Plusieurs approches et techniques ont été développées et exploitées pour traiter la complexité relative aux problèmes de l'imagerie. On en trouve des approches exactes purement mathématiques qui sont très souvent confrontées au problème de faisabilité en terme de temps de calcul. Le développement de la puissance des machines et des opportunités de calcul distribué a contribué à la levée de cet obstacle pour certaines tâches. D'autres tâches à forte combinatoire restent cependant hors de portée des techniques de cette catégorie. Les méthodes heuristiques ont fréquemment été utilisées pour diminuer le coût de calcul tout en fournissant des résultats acceptables ou assez proches de l'optimum recherché.

On a vu dans le chapitre précédent que les algorithmes évolutionnaires quantiques sont très efficaces dans la résolution de problèmes d'optimisation. Par

conséquent, leur utilisation dans le domaine de la vision par ordinateur, où les problèmes confrontés peuvent souvent être formulés comme problèmes d'optimisation, semble être une voie prometteuse.

Dans ce chapitre, deux problèmes d'importance majeure dans le domaine de la vision par ordinateur sont considérés. Il s'agit du problème de recalage d'images et du problème de segmentation d'images. Pour le premier problème, des algorithmes évolutionnaires quantiques sont utilisés pour effectuer différents types de recalage: recalage rigide/ rigide avec facteur de zoom/ affine, monomodal/ multimodal. Pour le problème de segmentation, les algorithmes évolutionnaires quantiques ont été remodelés pour prendre en compte l'aspect multiobjectif du problème jusqu'alors négligé ou mal investigué.

5.2. Algorithmes évolutionnaires quantiques pour le recalage d'images

5.2.1. Description

Ayant deux images I_1 et I_2 obtenues de capteurs identiques (recalage monomodal) ou différents (recalage multimodal), l'algorithme proposé permet l'estimation de la transformation géométrique qui aligne les deux images. Il s'agit d'une approche iconique puisque l'algorithme considère la totalité de l'information photométrique contenue dans les images et non pas des primitives extraites des images.

Une population initiale composée de N chromosomes quantiques est générée de manière aléatoire. La longueur de chaque chromosome quantique dépend du type de la transformation recherchée.

Pour une transformation rigide [Talbi et al. 2004a], qui est la composition d'une translation et d'une rotation, on a trois paramètres: dx , dy et θ . La nouvelle position (x', y') d'un point (x_2, y_2) de la deuxième image est calculée comme suit:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} dx \\ dy \end{pmatrix} + \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} \quad (5.1)$$

Chaque paramètre est codé sur 8 bits quantiques. Les paramètres dx et dy appartiennent à l'intervalle $[-128, +127]$ tandis que $d\theta$ appartient à l'intervalle $[-2, +2]$ radians (cet intervalle est subdivisé en $2^8=256$ valeurs réelles). Donc, la longueur d'un chromosome quantique pour ce type de transformation est de 24 qubits.

Pour une transformation rigide avec zoom, on intègre un autre paramètre représentant le facteur d'agrandissement s et l'équation précédente devient:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} dx \\ dy \end{pmatrix} + s \cdot \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} \quad (5.2)$$

s appartient à l'intervalle $[0.5, 2]$ et est représenté aussi sur 8 bits ce qui porte la longueur du chromosome à 32 qubits.

Pour une transformation affine, on considère 6 paramètres: dx , dy , a_{11} , a_{12} , a_{21} et a_{22} . La nouvelle position d'un point de la deuxième image est calculée selon l'équation:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} dx \\ dy \end{pmatrix} + \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \cdot \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} \quad (5.3)$$

Chaque paramètre est codé sur 8 qubits. On aura donc des chromosomes à 48 qubits. Les paramètres a_{11} , a_{12} , a_{21} et a_{22} appartiennent à l'intervalle $[-2,+2]$.

Pour l'évaluation d'une transformation, on a besoin d'une mesure de similarité. Il existe plusieurs mesures parmi lesquelles je cite: le critère quadratique, la corrélation croisée normalisée, le coefficient de corrélation, etc. Ces mesures sont simple et faciles à programmer mais ne sont pas utilisables dans le cas d'un recalage multimodal où les images ont des structures similaires mais des caractéristiques différentes. L'idée est d'utiliser la relation entre les éléments constitutifs de l'image au lieu d'exploiter seulement leurs valeurs. Autrement dit, au lieu d'exprimer l'image acquise en fonction de l'image de référence, on l'exprime comme étant prévisible de l'image de référence. En statistiques, cette notion de prévisibilité est liée à un vieux concept introduit à la théorie de l'information par *Shannon* connu sous le nom d'*entropie* [Shannon 1951].

L'entropie est une statistique qui résume l'aspect aléatoire d'une certaine variable aléatoire. Cette mesure nous dit à quel degré une variable est aléatoire. L'entropie la plus connue et la plus utilisée est celle définie par Shannon en 1948.

Etant donnée une certaine variable aléatoire représentée par une distribution X c'est-à-dire un ensemble de couples (x_i, p_i) où x_i est une valeur et $p_i = p(X = x_i)$ est la probabilité d'occurrence de la valeur x_i . L'entropie de X dénotée par $H(X)$ est définie comme une espérance :

$$H(X) = -\sum p_i \log_2 p_i \quad (5.4)$$

L'entropie est définie en terme du logarithme de base 2. Intuitivement, l'entropie mesure l'information moyenne fournie par une certaine distribution.

Lorsqu'il s'agit d'un problème de dépendance entre deux variables aléatoires X et Y définies par des distributions de probabilités, on s'intéresse à la distribution conjointe qui donne le nombre d'occurrence des couples de valeurs (x,y) . Dans ce cas, on définit l'entropie conjointe $H(X,Y)$ comme :

$$H(X,Y) = -\sum \sum p(x,y) \log_2 p(x,y) \quad (5.5)$$

Dans le cas d'une dépendance entre X et Y , la connaissance de l'une permet de prédire aisément l'autre. Dans le cas d'une indépendance totale, on peut facilement vérifier que l'entropie conjointe $H(X,Y)$ n'est rien d'autre que la somme des entropies marginales $H(X)$ et $H(Y)$. Partant de ces notions de base, les chercheurs en vision et traitement d'images ont développé des mesures adaptées à la comparaison d'images multi sources. L'*information mutuelle* (IM) est l'une de ces mesures [Viola 1995]. Elle est définie par l'expression :

$$IM(X,Y) = H(X) + H(Y) - H(X,Y) \quad (5.6)$$

C'est en fait une mesure de réduction de l'entropie conjointe qui s'annule dans le cas d'une totale indépendance et qui est maximale dans le cas contraire.

Le schéma général de l'algorithme utilisé est le suivant:

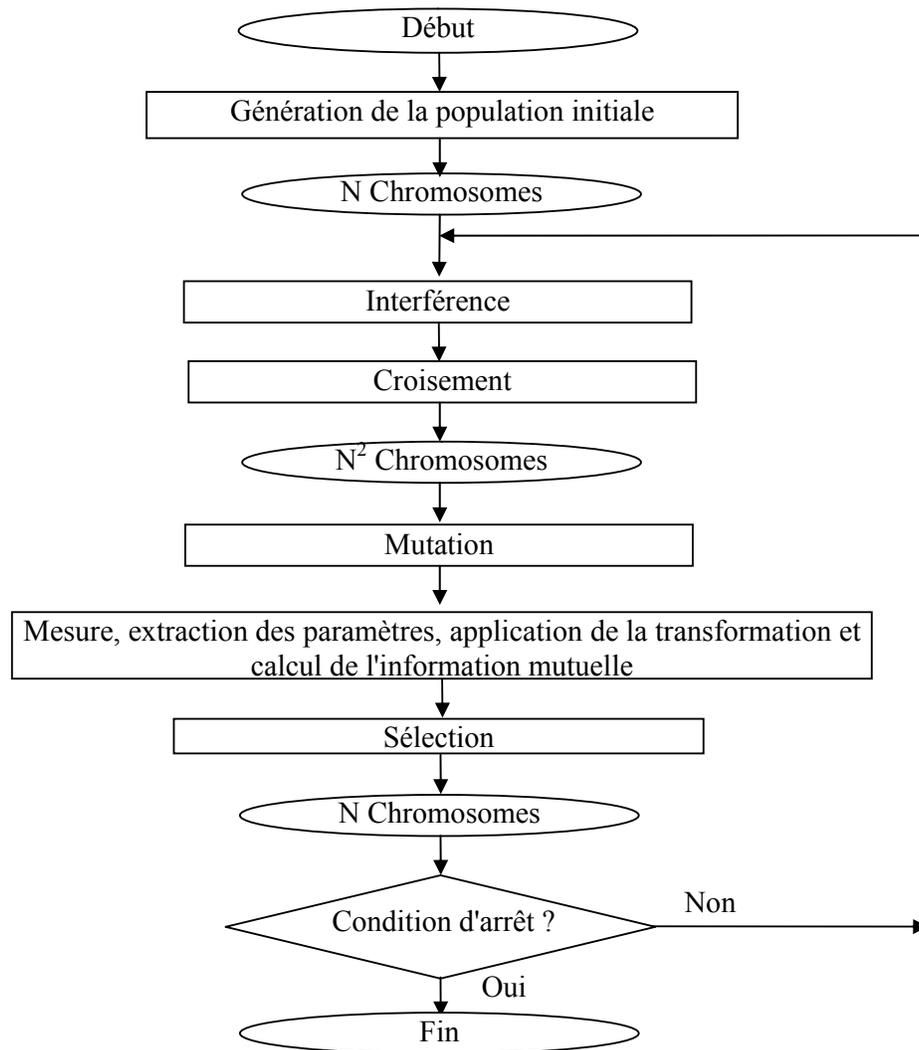


Figure 5.1. Algorithme évolutionnaire quantique pour le recalage d'images

5.2.2. Résultats expérimentaux

On présente dans cette sous-section quelques résultats expérimentaux pour les trois types de recalage cités ci-dessus. Pour chaque paire d'images, l'algorithme évolutionnaire quantique est exécuté 10 fois. On donne l'image obtenue dans la meilleure exécution ainsi que la moyenne et l'écart type de la valeur de l'information mutuelle sur l'ensemble des 10 exécutions.

5.2.2.1. Recalage rigide

L'algorithme évolutionnaire quantique est exécuté avec deux puis trois chromosomes dans la population initiale et ce pour trois différentes valeurs de l'angle de rotation, à savoir $\pi/12$, $\pi/20$ et $\pi/30$.

a. Images monomodales

Dans l'exemple de la figure 5.2, la deuxième image est obtenue en appliquant une translation puis une rotation sur la première

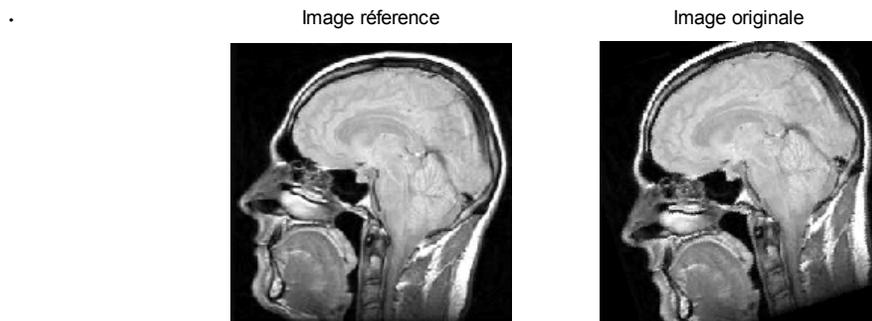


Figure 5.2. Images IRM

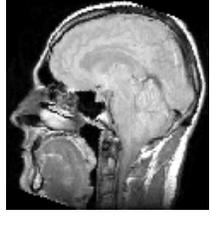
Θ	2 chromosomes	3 chromosomes
$\pi/12$	<p data-bbox="634 369 740 394">IM=2.1962</p> 	<p data-bbox="1065 369 1170 394">IM=2.4991</p> 
$\pi/20$	<p data-bbox="634 695 740 720">IM=2.1851</p> 	<p data-bbox="1065 695 1170 720">IM=2.1851</p> 
$\pi/30$	<p data-bbox="634 999 740 1024">IM=2.4925</p> 	<p data-bbox="1065 999 1170 1024">IM=2.4925</p> 

Tableau 5.1. Recalage rigide des images de la figure 5.2

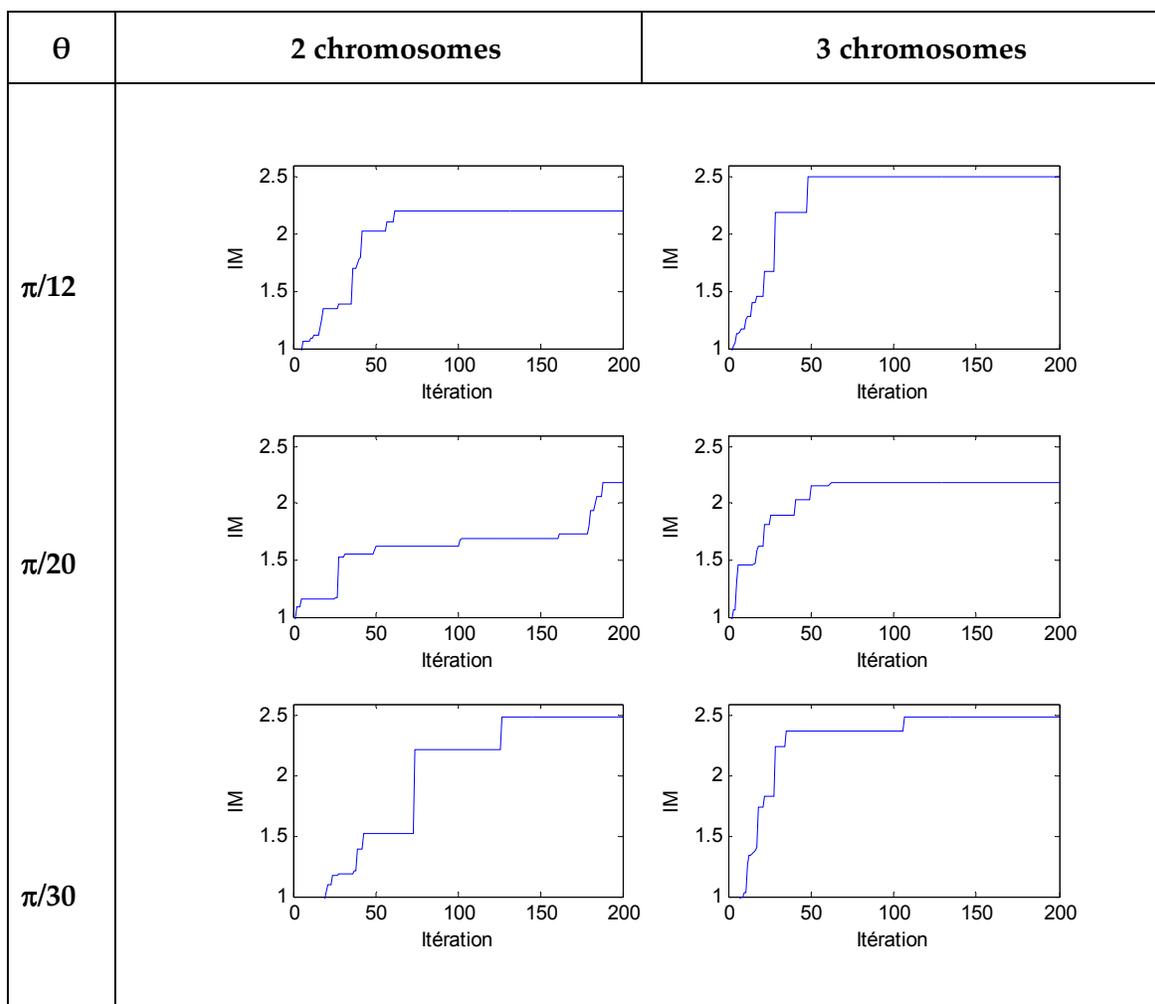


Tableau 5.2. Evolution de la meilleure valeur de l'information mutuelle au cours du recalage rigide des images de la figure 5.2

θ	2 chromosomes	3 chromosomes
$\pi/12$	2,0852	2,2347
$\pi/20$	1,9904	2,0596
$\pi/30$	1,9083	2,1851

Tableau 5.3. Valeur moyenne de l'information mutuelle sur les 10 exécutions

θ	2 chromosomes	3 chromosomes
$\pi/12$	0,0329	0,1558
$\pi/20$	0,0472	0,1114
$\pi/30$	0,2046	0,1507

Tableau 5.4. Ecart type de l'information mutuelle sur les 10 exécutions

b. Images multimodales

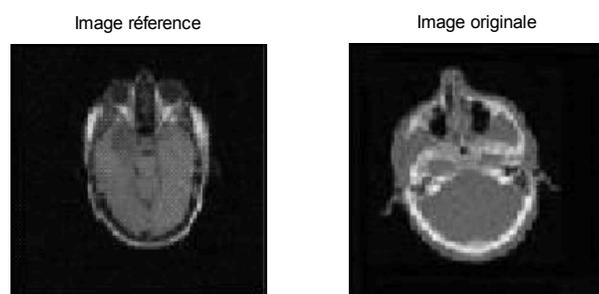


Figure 5.3. Paire d'images: IRM-Tomographie calculée à rayon X (CTI)

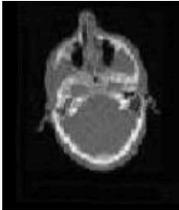
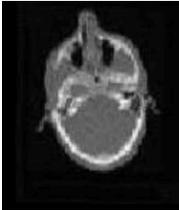
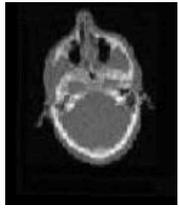
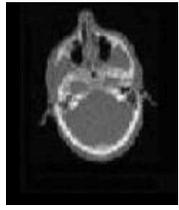
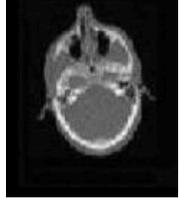
θ	2 chromosomes	3 chromosomes
$\pi/12$	IM=1.5467 	IM=1.5467 
$\pi/20$	IM=1.5425 	IM=1.5597 
$\pi/30$	IM=1.566 	IM=1.5587 

Tableau 5.5. Recalage rigide des images de la figure 5.3

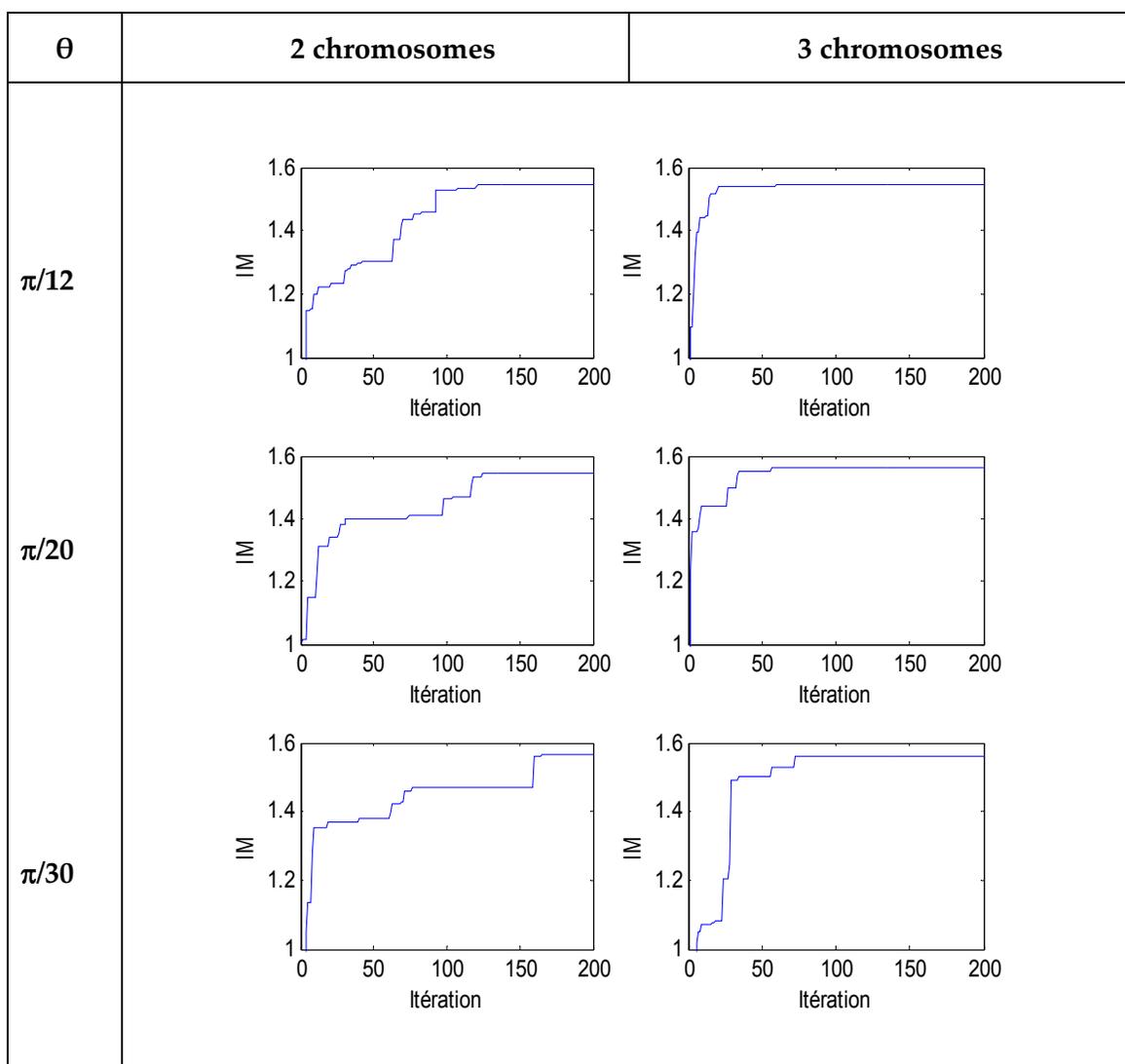


Tableau 5.6. Evolution de la meilleure valeur de l'information mutuelle au cours du recalage rigide des images de la figure 5.3

θ	2 chromosomes	3 chromosomes
$\pi/12$	1,4178	1,4034
$\pi/20$	1,4327	1,4166
$\pi/30$	1,4300	1,3866

Tableau 5.7. Valeur moyenne de l'information mutuelle sur les 10 exécutions

θ	2 chromosomes	3 chromosomes
$\pi/12$	0,0836	0,0598
$\pi/20$	0,0923	0,0799
$\pi/30$	0,0965	0,0789

Tableau 5.8. Ecart type de l'information mutuelle sur les 10 exécutions

On remarque que les performances de l'algorithme ne changent pas beaucoup avec la variation de l'angle de rotation dans l'opération d'interférence. Les résultats sont très satisfaisants et 200 itérations seulement ont été suffisantes pour recalibrer correctement les deux images dans plusieurs exécutions.

5.2.2.2. Recalage rigide avec facteur d'échelle

a. Images monomodales

Dans cette série de tests, on a gardé seulement la valeur $\pi/20$ pour l'angle de rotation.

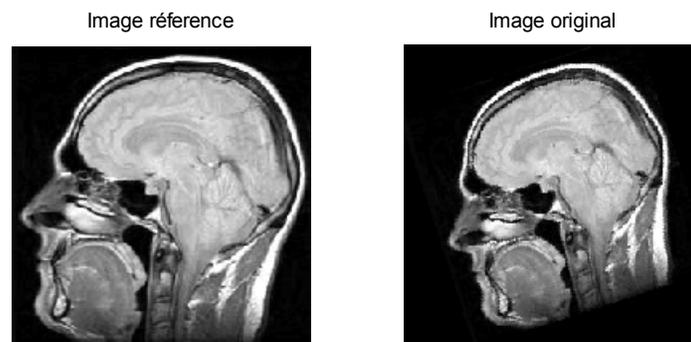


Figure 5.4. Images IRM

θ	2 chromosomes	3 chromosomes
$\pi/20$	<p>IM=2.2455</p>	<p>IM=2.0495</p>

Tableau 5.9. Recalage rigide des images de la figure 5.4

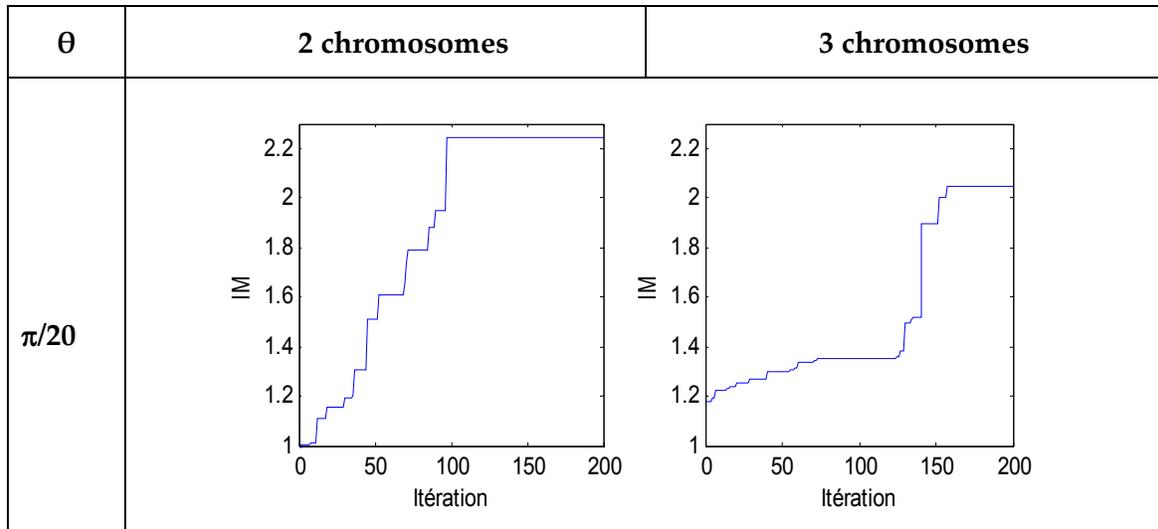


Tableau 5.10. Evolution de la meilleure valeur de l'information mutuelle au cours du recalage rigide des images de la figure 5.4

θ	2 chromosomes	3 chromosomes
Moyenne	1.8512	1.7911
Ecart type	0.1860	0.2613

Tableau 5.11. Valeur moyenne et écart type de l'information mutuelle sur les 10 exécutions

b. Images multimodales

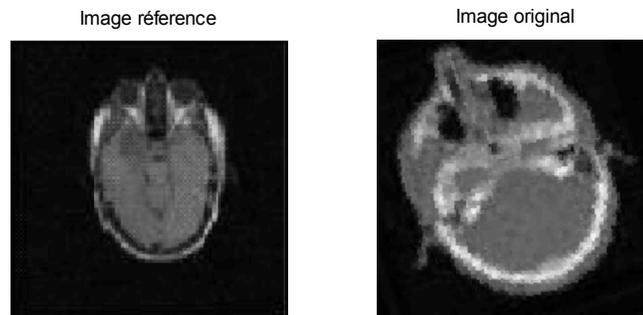


Figure 5.5. Paire d'images: IRM-Tomographie calculée à rayon X (CTI)

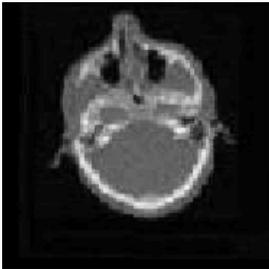
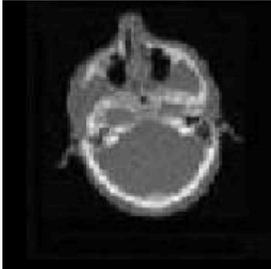
θ	2 chromosomes	3 chromosomes
$\pi/20$	IM=1.5425 	IM=1.566 

Tableau 5.12. Recalage rigide des images de la figure 5.5

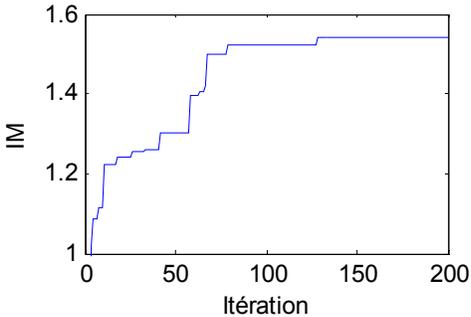
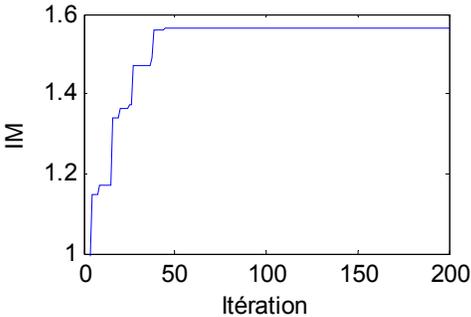
θ	2 chromosomes	3 chromosomes
$\pi/20$		

Tableau 5.13. Evolution de la meilleure valeur de l'information mutuelle au cours du recalage rigide des images de la figure 5.5

θ	2 chromosomes	3 chromosomes
Moyenne	1.4402	1.4317
Ecart type	0.1868	0.1682

Tableau 5.14. Valeur moyenne et écart type de l'information mutuelle sur les 10 exécutions

Vu les valeurs de l'écart type, on remarque que les exécutions diffèrent plus les unes des autres. Cependant, la qualité du recalage dans les meilleures exécutions reste très bonne et la moyenne des résultats de l'information mutuelle reste assez acceptable.

5.2.2.2. Recalage affine

Dans ce type de recalage, la longueur du chromosome quantique double par rapport premier type traité. C'est la raison pour laquelle la conservation des paramètres précédents concernant le nombre de chromosomes et le nombre de générations n'a pas permis d'avoir de très bons résultats. Pour cela, on a opté pour l'ajout d'un chromosome à la population initiale portant son effectif à 4 et l'évaluation des performances au bout de 500 itérations au lieu de 200.

a. Images monomodales

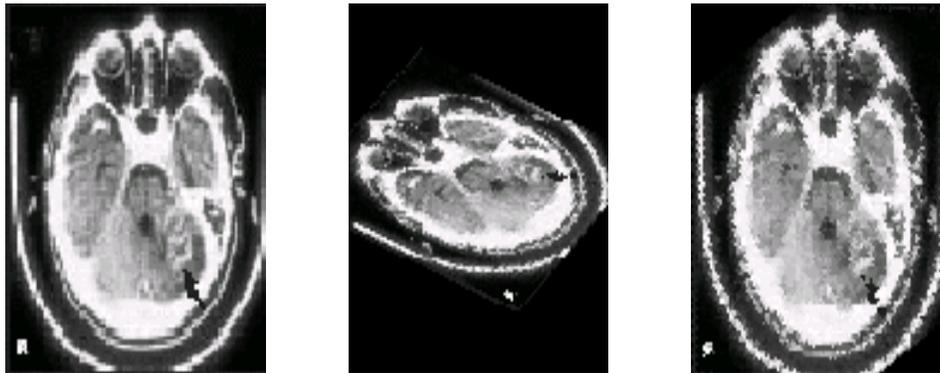


Figure 5.6. Recalage affine d'images monomodales

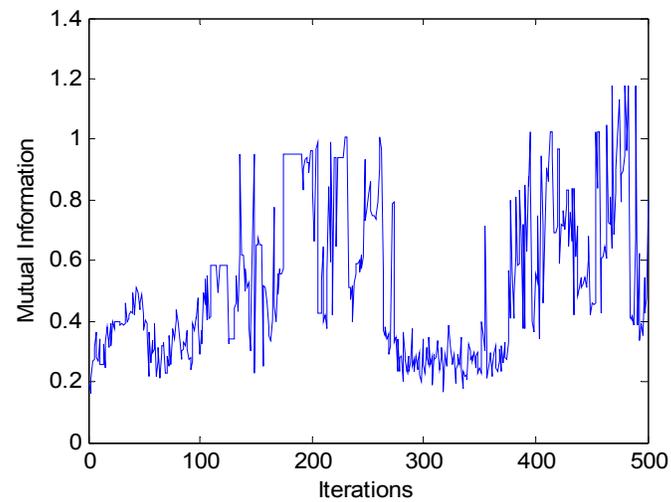


Figure 5.7. Evolution de la meilleure valeur de l'information mutuelle

La moyenne des valeurs prises par l'information mutuelle sur les 10 exécutions était de 0.9960 tandis que l'écart type était de 0.0912.

b. Images multimodales



Figure 5.8. Recalage affine d'images multimodales (IRM-CTI)

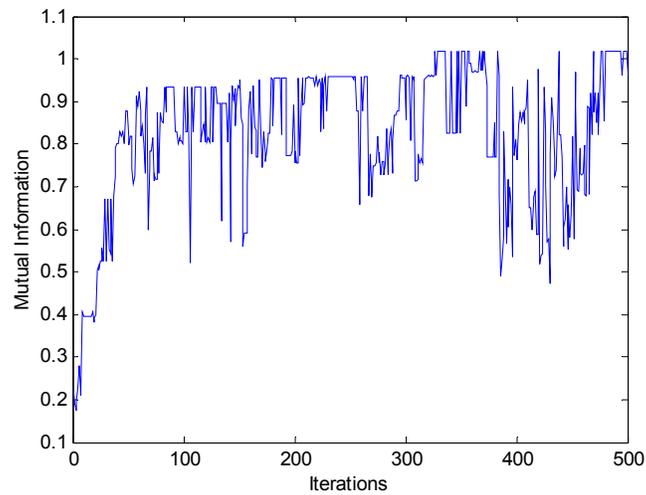


Figure 5.9. Evolution de la meilleure valeur de l'information mutuelle

La moyenne des valeurs prises par l'information mutuelle sur les 10 exécutions était de 0.8515 alors que l'écart type était de 0.1346.

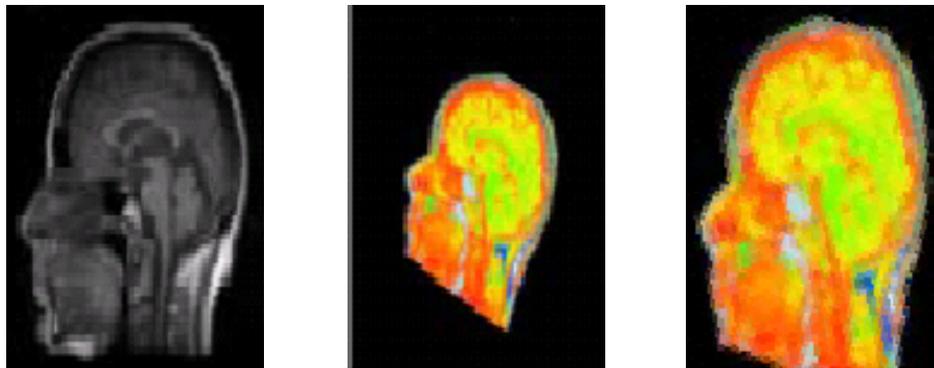


Figure 5.10. Recalage affine d'images multimodales (IRM- scintigraphie)

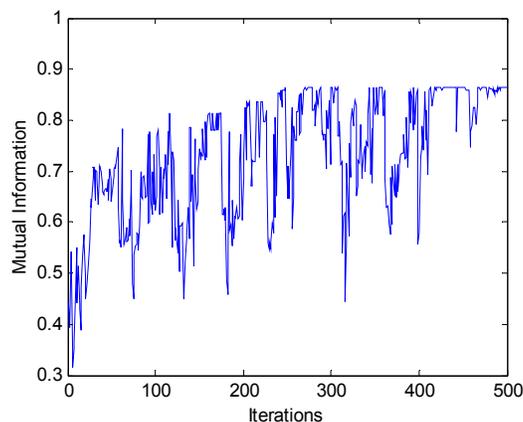


Figure 5.11. Evolution de la meilleure valeur de l'information mutuelle

La moyenne des valeurs prises par l'information mutuelle sur les 10 exécutions était de 0.7032 tandis que l'écart type était de 0.1258.

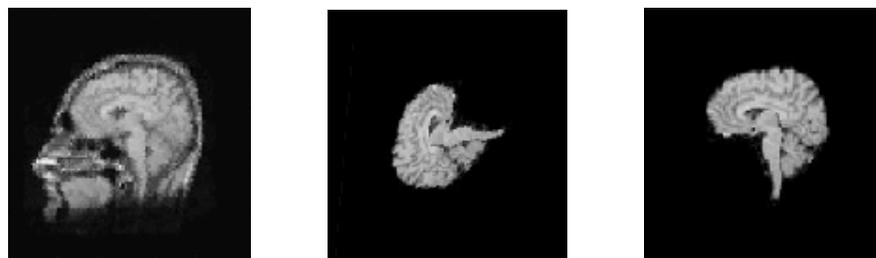


Figure 5.12. Un autre exemple de recalage affine d'images multimodales

5.2.3. Comparaison aux algorithmes génétiques classiques

Deux qualités évidentes de nos algorithmes évolutionnaires quantiques sont la petite taille de la population et le petit nombre de générations requises pour la convergence. Les algorithmes génétiques classiques utilisent souvent plusieurs dizaines de chromosomes et requièrent en général plusieurs milliers de générations pour converger.

5.2.3.1. Temps de calcul

Le graphique suivant montre la différence considérable en temps de calcul entre notre algorithme (AEQ), un algorithme génétique classique utilisant 15 chromosomes

(AGC-15) et un algorithme génétique classique utilisant 50 chromosomes (AGC-50) et ce pour 8 paires d'images différentes (différents types de transformation).

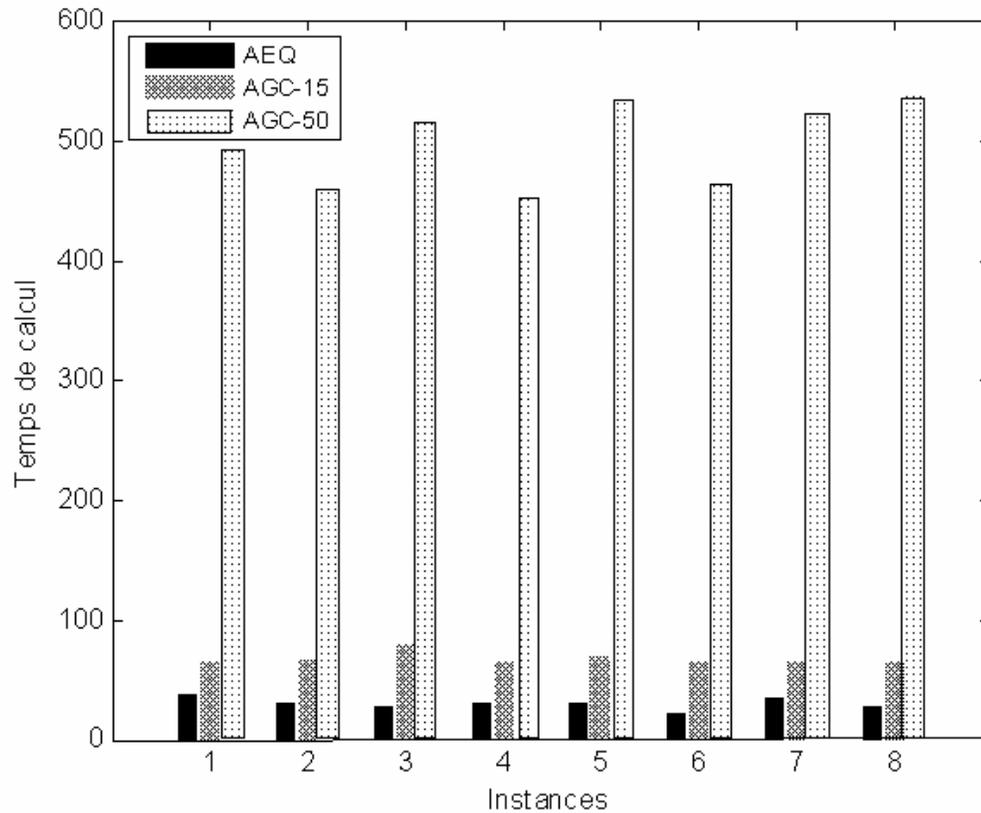


Figure 5.13. Comparaison des temps de calculs de AEQ, AGC-15 et AGC-50

5.2.3.1. Qualité du recalage

La figure 5.14 donne une comparaison entre l'évolution de la meilleure valeur de l'information mutuelle dans la meilleure exécution pour les 8 instances et ce entre l'AEQ et l'AGC-50 avec un nombre de générations variant entre 300 et 800 selon le type de la transformation recherchée. On note une nette supériorité de la qualité de recalage obtenue par les algorithmes évolutionnaires quantiques.

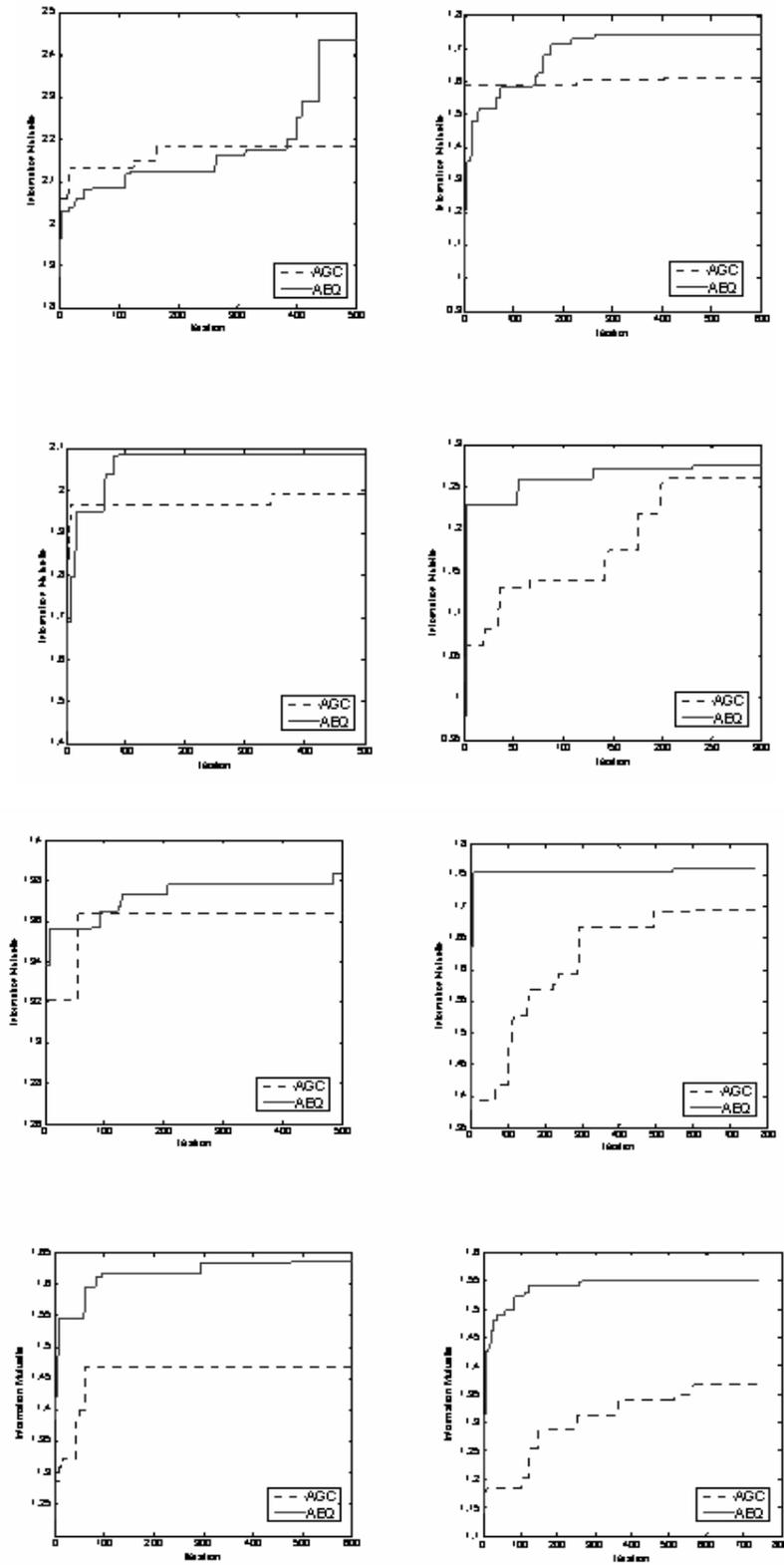


Figure 5.14. Comparaison de l'évolution de la meilleure valeur de l'information mutuelle de AEQ et AGC-50

5.3. Algorithmes évolutionnaires quantiques pour la segmentation multiobjectif d'images

5.3.1. Motivation

Le problème de segmentation d'images a été très abondamment étudié dans la littérature. Ceci n'a rien d'étrange quand on connaît l'importance de la phase de segmentation pour tout traitement de haut niveau comme la reconnaissance, le suivi d'objet, l'interprétation de scènes, etc. Il est à noter cependant que sont rares les cas où on pourrait extraire à partir d'un simple résultat de segmentation toute l'information que comporte une image. Cela est la conséquence de la formulation du problème qui est faite sur la base d'une fonction objectif qui détermine un partitionnement unique de l'image. On ne peut par exemple avoir en même temps une spécification d'objet et de détails inclus dans cet objet car un point ne doit appartenir qu'à une région.

La segmentation d'une image avec la tenue en compte de l'aspect multiobjectif de ce problème ouvre de nouvelles perspectives car elle donne comme résultat un ensemble de solutions dites solutions non dominées et non pas une seule solution. Ceci dit que dans cet ensemble, aucune solution n'est meilleure qu'une autre par rapport à tous les objectifs considérés. Un processus de haut niveau peut alors considérer l'ensemble de ces solutions pour donner lieu à une description plus riche et plus complète de l'image qui est ainsi vue de manières différentes par un tel processus. Cette démarche aurait un meilleur rendement si les possibilités modernes de calcul parallèle sont exploitées.

5.3.2. Description de l'algorithme

Notre tâche de segmentation multiobjectif d'images est réalisée à travers une approche de décomposition/fusion (figure 5.15).

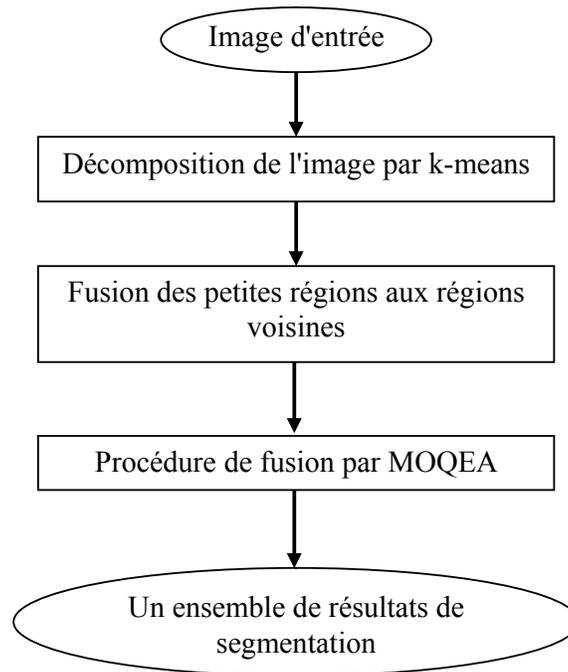


Figure 5.15. Description sommaire de l'algorithme de segmentation

Dans la phase de décomposition, on a adopté l'algorithme *k*-moyennes (*k-means*) qui est assez efficace et plus rapide par rapport d'autres algorithmes comme celui des *c*-moyennes floues (*fuzzy c-means*) [Tou et Gonzales 1974]. *K-means* produit *k* classes (*cluster*) composées chacune d'un ensemble de pixels ayant des valeurs colorimétriques assez proches les unes des autres. Pappas a indiqué dans [Pappas 1992] que la valeur $k=4$ donne des résultats très acceptables pour la plupart des images.

Comme les pixels formant une classe ne sont pas nécessairement adjacents, on utilise le résultat de l'algorithme *k-means* pour rechercher les différentes régions (une région est un ensemble de pixels adjacents appartenant à la même classe). Le nombre de régions obtenues est très souvent supérieur à *k*.

A cause du bruit et pour réduire l'espace de recherche, les petites régions sont filtrées et fusionnées aux régions voisines ayant des les plus proches propriétés colorimétriques. Une région est éliminée si elle comporte moins de 10 pixels [Chun et Yang 1996]. A la fin de cette étape, on a R régions et N frontières.

Le résultat de cette décomposition est considéré comme entrée pour notre algorithme évolutionnaire quantique de segmentation multiobjectif d'images (MOQEA). Le but de cet algorithme est d'établir un ensemble de segmentations non dominées. Chaque résultat de segmentation est obtenu via la fusion de certaines régions adjacentes.

L'algorithme évolutionnaire quantique utilise NC chromosomes quantiques. Chaque chromosome est constitué de N bits quantiques représentant chacun une frontière entre deux régions issues de la segmentation initiale. On maintient également un ensemble des solutions non-dominées.

L'état de base $|0\rangle$ signifie que la frontière que représente le qubit est écartée, c'est-à-dire que les deux régions qu'elle sépare initialement vont être fusionnées. De l'autre coté, l'état de base $|1\rangle$ signifie que la frontière est maintenue. La figure 5.16 illustre un exemple de ce codage.

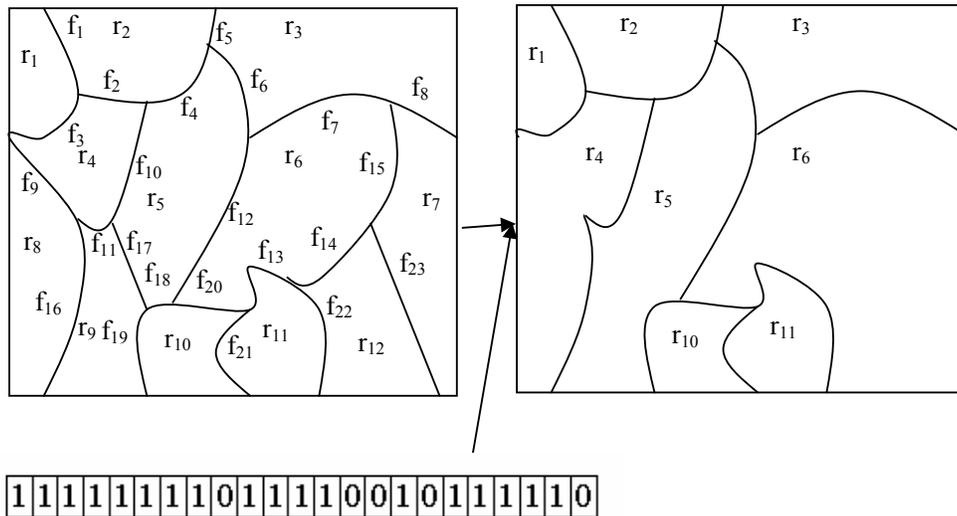


Figure 5.16. Codage d'une solution

Initialement, les chromosomes sont générés aléatoirement et l'ensemble des solutions non-dominées contient seulement une solution, une chaîne de N bits ayant tous la valeur "1". Autrement dit, la seule solution non dominée au début est celle générée par la phase de segmentation préliminaire.

Le schéma général de l'algorithme évolutionnaire quantique destiné à la segmentation multiobjectif d'images est présenté dans la figure 5.17.

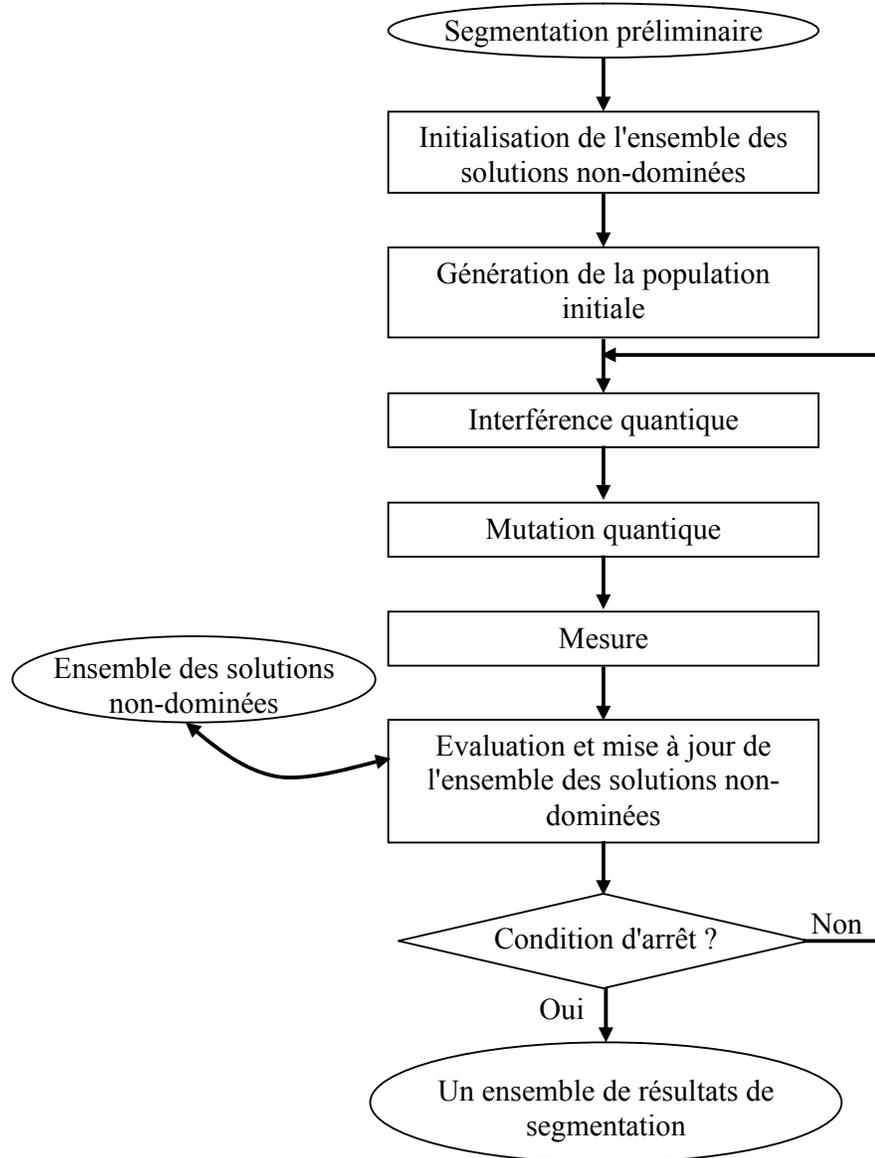


Figure 5.17. L'algorithme évolutionnaire quantique pour la segmentation multiobjectif d'images

Dans cette démarche, l'opération d'interférence doit s'adapter à la nature multicritère de l'optimisation car à la différence des problèmes mono objectif où on mémorise seulement une meilleure solution, dans l'optimisation multiobjectif on a très souvent en mémoire plusieurs solutions non dominées. On doit donc choisir par rapport à quoi appliquer la porte de rotation. On a étudié trois possibilités:

1. *Choisir au hasard, pour chaque chromosome quantique et à chaque itération, une parmi les solutions non dominées qui va interférer avec le chromosome.*

Avantages

- facile à implémenter
- entraîne un coût de calcul additionnel assez négligeable

Inconvénient

- vu que les solutions non dominées sont assez différentes les unes des autres et qu'elles sont éparpillées sur une large portion de l'espace de recherche, cette alternative va certainement compromettre le rôle d'intensification que joue l'opération d'interférence. Il y aura plutôt des fluctuations de nature assez aléatoire.

2. *Effectuer l'interférence par rapport à une solution artificielle issue des solutions non dominées via une certaine formule de synthèse.*

Avantage

- tient en compte toutes les solutions non dominées

Inconvénients

- entraîne un surcoût de calcul considérable
- une solution issue des meilleures solutions n'est pas toujours une bonne solution. Elle peut même pour certains problèmes ne pas représenter une solution faisable. L'opération peut donc ne rien apporter

3. *Effectuer l'interférence par rapport à la solution la plus proche de la solution extraite du chromosome en question en terme de distance euclidienne.*

Avantage

- a plus de sens, car l'interférence dans ce cas déplace le chromosome dans la direction d'une solution non dominée avec prise en considération de son état actuel. On est donc loin des fluctuations totalement aléatoires puisque un chromosome peut être dirigé au cours de plusieurs itérations vers une même solution non dominée

Inconvénient

- entraîne un surcoût de calcul: calculer des distances euclidiennes et trouver leur minimum

Parmi les trois options ci-dessus, on a choisi la troisième qui outre son bien fondé théorique, s'est montrée meilleure lors de l'expérimentation.

A l'issue de l'opération de mesure, on obtient à partir des NC chromosomes quantiques NC solutions binaires. Chaque solution représente un résultat de segmentation qui est par la suite évalué et comparé à tous les résultats représentés par les éléments de l'ensemble des solutions non dominées. Toute solution de cet ensemble dominée par une nouvelle solution est aussitôt retirée. Par ailleurs, si une solution n'est dominée par aucune autre solution, elle est intégrée à l'ensemble des solutions non dominées.

L'évaluation est basée sur deux critères: l'homogénéité intra région et l'hétérogénéité inter région.

Soient R le nombre de régions, m_i la valeur moyenne des pixels appartenant à la région i , var_i la variance à l'intérieur de la région i , ne le nombre des frontières maintenues et nr le nombre de régions résultantes.

L'homogénéité intra région est donnée par:

$$Hom = -\frac{1}{nr} \sum_{i=1}^{nr} var_i \tag{5.7}$$

Et l'hétérogénéité inter région est décrite par:

$$Het = \frac{1}{ne} \sum_{i,j}^{nr} (m_i - m_j)^2 \quad i, j \text{ adjacents} \tag{5.8}$$

Ces deux mesures sont fortement liées l'une à l'autre. L'augmentation de la valeur d'un critère donne souvent une baisse de la valeur que prendra l'autre.

5.3.3. Résultats expérimentaux

L'algorithme proposé a été testé sur plusieurs images naturelles. Pour valider les résultats on les a comparé à ceux obtenus en utilisant un algorithme génétique classique pour l'optimisation multiobjectif (MOGA). Pour notre algorithme, on a choisi expérimentalement les paramètres suivants:

Nombre de chromosomes $NC=3$

Angle de permutation $\delta\theta = \pi/8$

Taux de mutation $pm=0.02$

Pour l'algorithme MOGA, qui a été intégré dans la même approche de décomposition / fusion, n a opté pour les paramètres suivants:

Nombre de chromosomes $NC=25$

Taux de mutation $pm=0.1$, avec l'utilisation d'une opération de croisement.

Pour un premier test on a appliqué les deux approches sur une image de taille 256x256 pixels (figure 5.18).



Figure 5.18. L'image "cameraman" de taille 256x256

Les deux algorithmes ont été exécutés sur 2500 itérations. La figure 5.19 montre l'évolution de la taille de l'ensemble des solutions non dominées.

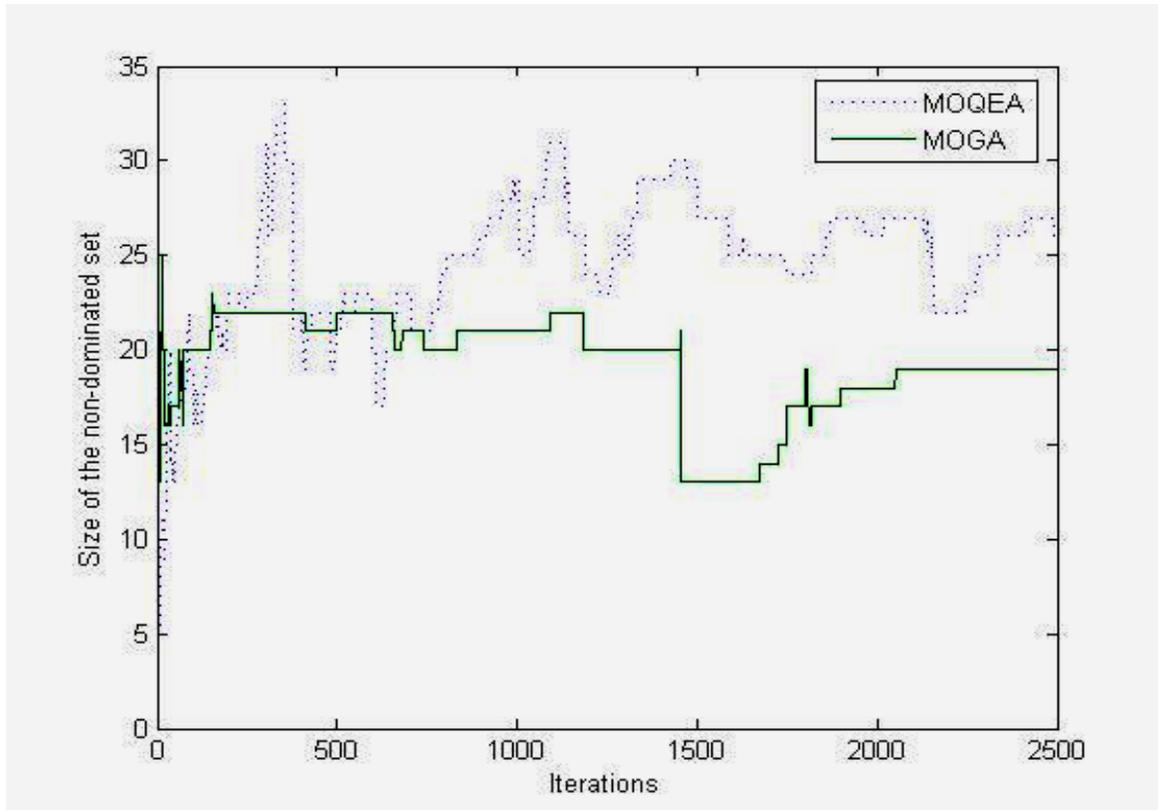


Figure 5.19. L'évolution de la taille de l'ensemble des solutions non dominées

On remarque que notre algorithme évolutionnaire quantique, malgré sa population de petite taille (3 chromosomes quantiques contre 25 chromosomes classiques) donne plus de solutions non dominées et est plus dynamique car il y a le plus souvent un changement de la taille de l'ensemble des solutions non dominées. Cette dynamique est illustrée dans le tableau 5.15 qui donne pour chacun des deux algorithmes le nombre d'itérations parmi les 2500 qui trouvent de nouvelles solutions non dominées, le nombre d'itérations qui ôtent des solutions qui sont dominées par de nouvelles solutions, le nombre total d'insertions et le nombre total de suppressions.

	<i>NI</i>	<i>NI</i>	<i>NI</i>	<i>NI</i>
MOQEA	209	111	235	210
MOGA	67	41	140	121

Tableau 5.15. Résumé de la dynamique de l'ensemble des solutions non dominées

NIA: nombre d'itérations ajoutant des solutions non dominées

NIS: nombre d'itérations supprimant des solutions qui ne sont plus non dominées

NA: nombre total d'insertions

NS: nombre total de suppressions

Il est clair que l'algorithme proposé trouve de nouvelles solutions non dominées environ 3 fois plus fréquemment que le candidat classique.

Dans la figure 5.20, on compare les valeurs prises par les fonctions objectif pour les solutions non dominées résultantes des deux algorithmes.

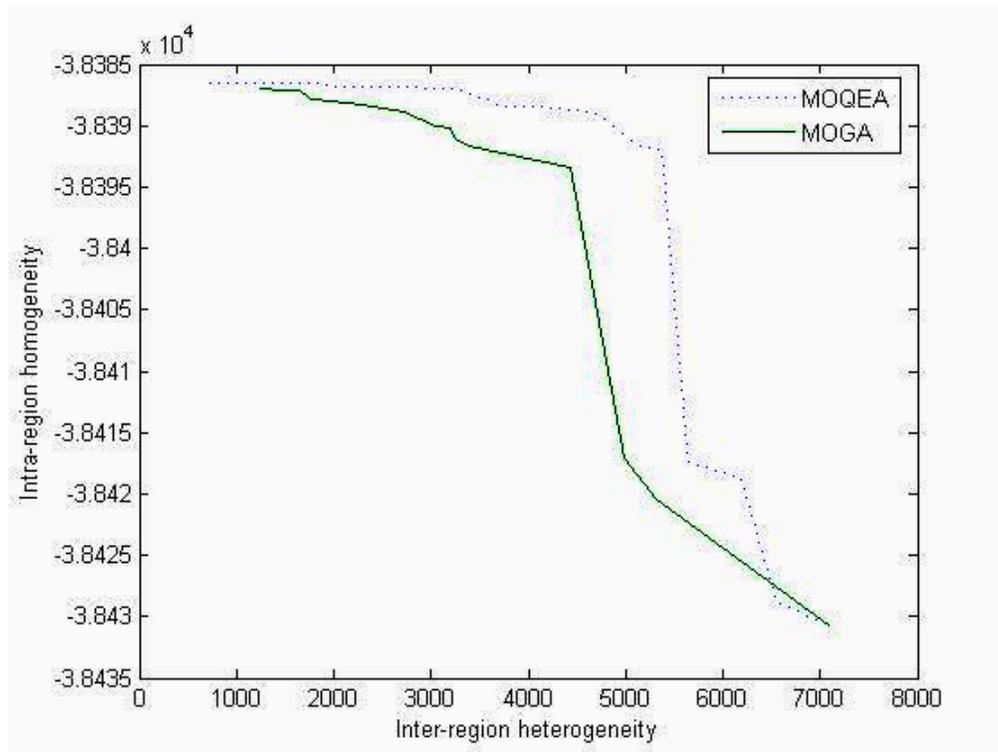


Figure 5.20. Les valeurs des fonctions objectifs pour les solutions non dominées

On remarque que la plupart des solutions non dominées trouvées par l'algorithme génétique classique sont dominées par des solutions découvertes par l'algorithme proposé. De l'autre côté, aucune solution issue de notre algorithme n'est dominée par une solution issue de l'approche classique. Ceci dit, l'algorithme évolutionnaire quantique est nettement plus puissant et efficace que les approches évolutionnaires classiques.

La figure 5.21 présente quelques unes des 26 solutions non dominées découvertes par l'algorithme proposé.

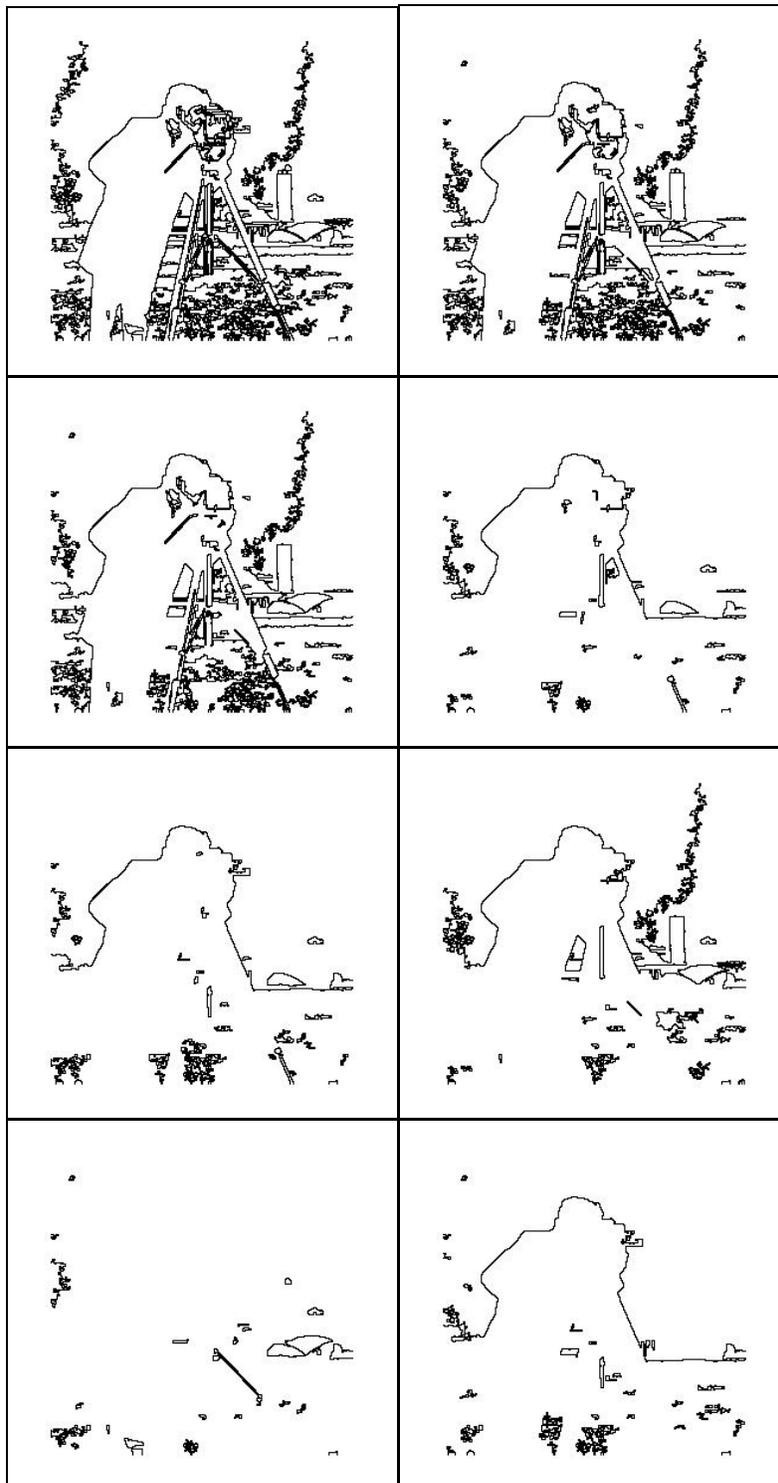


Figure 5.21. Quelques solutions non dominées issue du MOQEA

Conclusion générale

Dans cette thèse a été abordé un sujet qui s'insère dans le cadre général d'optimisation combinatoire et d'optimisation multiobjectif. Il s'agit d'une hybridation entre les algorithmes évolutionnaires et les principes de l'informatique quantique.

L'approche proposée a été étudiée et expérimentée sur des problèmes typiques d'optimisation (problème du sac à dos, problème du voyageur de commerce, problème des N reines) pour démontrer son utilité et prouver son apport et ses qualités supérieures par rapport aux autres approches évolutionnaires. Le rôle des différents paramètres a été étudié et un résultat important qu'on a obtenu est que la *porte D* toute seule ne permet pas une bonne exploration de l'espace de recherche et qu'il nous faudra employer d'autres moyens pour éviter les minima locaux. A cette fin, de nouveaux opérateurs évolutionnaires quantiques ont été définis et leur contribution a été mise en valeur.

Il a été ensuite question d'intégrer les nouveaux algorithmes évolutionnaires quantiques dans un processus de recalage d'images basé sur une approche iconique qui supporte plusieurs types de transformations géométriques (rigide, similitude, affine). Ce processus emploie notamment l'information mutuelle comme mesure de similarité pour pouvoir aligner des images provenant aussi bien d'une même source que de sources différentes. Les résultats obtenus ont été à la hauteur des espérances et des populations de tailles très réduites ont été capables d'aboutir à des solutions optimales en un nombre d'itérations presque négligeable par rapport aux grands nombres d'itérations auxquels nous nous sommes habitués avec les approches évolutionnaires classiques.

Un point intéressant dans cette thèse a été la considération de l'aspect multiobjectif du problème de segmentation d'images. Le but d'une telle démarche est de permettre l'obtention à partir d'une image donnée le maximum d'informations grâce à la génération de plusieurs résultats de segmentation. Si on prend ces résultats deux à deux, aucun résultat n'est moins bon que l'autre sur tous les fronts. Il pourrait donc contribuer à dégager une information utile que les autres résultats n'ont pas la capacité de fournir.

Un algorithme évolutionnaire quantique a été employé pour mener la deuxième phase d'un processus de décomposition/fusion. La différence principale entre cet algorithme et les algorithmes précédents est le maintien d'un ensemble de solutions non dominées et non pas une seule solution. Ceci oblige à conduire différemment l'opération d'interférence. La stratégie qu'on a adoptée a l'avantage de permettre à chaque chromosome quantique d'évoluer dans la direction d'une certaine solution dominée. L'espace de recherche serait donc mieux exploré.

L'expérimentation de notre algorithme de segmentation multiobjectif a mis en évidence son indiscutable supériorité par rapport aux approches évolutionnaires classiques. Une petite population et un petit nombre de générations ont donné naissance à un processus de recherche nettement plus dynamique qui explore mieux l'espace de recherche et donne des solutions de meilleure qualité.

Les principaux ingrédients qui ont contribué à l'obtention de bons résultats pour la totalité des problèmes abordés peuvent être résumés dans les points suivants:

- Le codage quantique permet la représentation de toutes les solutions possibles du problème traité par un seul chromosome grâce au principe de la superposition d'états. Ce sont les probabilités d'avoir les différentes solutions comme résultats d'opérations de mesure qui changent au cours de l'évolution. Avec un seul chromosome, on aura plus tendance à mesurer les mêmes solutions et rester dans un petit voisinage ce qui nous mène à employer plus de chromosomes pour avoir plus de chance de ne pas s'emprisonner dans des minima locaux. Une population n'excédant pas 5 chromosomes pourrait très bien être suffisante pour cela.

- L'opération d'interférence réalisée à travers *la porte D* est un moyen puissant de renforcer la recherche dans le voisinage de la meilleure solution en cours. Ceci joue un rôle essentiel dans la diminution du nombre d'itérations nécessaires à la convergence. Les chromosomes sont tout le temps guidés par la meilleure solution actuelle.
- L'opération de mutation ou de permutation, qui est aussi une sorte de mutation employée pour quelques problèmes, est un outil d'échapper aux minima locaux. Ces petites fluctuations permettent aux chromosomes de ne pas être entièrement commandés par la meilleure solution en cours mais d'explorer de temps à autres des zones lointaines qui peuvent englober des solutions meilleures.
- L'opération de croisement quantique permet d'assembler des fragments de chromosomes pour synthétiser de nouveaux chromosomes pouvant générer de meilleures solutions que celles générées par les chromosomes de départ.

Au cours de la présente recherche, certaines difficultés ont été rencontrées, parmi lesquelles:

- L'absence de règles explicites qui permettent le bon choix des valeurs des différents paramètres. Ceci nous a contraint à faire de grosses séries d'expériences mettant en œuvre différentes combinaisons des valeurs de paramètres (taille de la population, valeur de l'angle d'interférence, taux de mutation, taux de croisement, etc.) ce qui a consommé énormément de temps. Les résultats de ces tests pour les différentes applications abordées n'ont pas été tous rapportés dans ce document pour ne pas le surcharger avec des informations qui sont assez fréquemment redondantes.
- La difficulté de choisir des approches de référence avec lesquelles on doit comparer les algorithmes développés pour les valider. Une question qui s'est toujours posée est: doit-on comparer le nouvel algorithme au meilleur algorithme qui effectue la même chose ou bien à un algorithme appartenant à la même classe? Si c'est la première voie, on aura la difficulté de désigner le meilleur algorithme. Si c'est la deuxième, il y aura toujours des critiques sur le choix d'un algorithme au dépend de l'autre surtout un tel algorithme n'a pas été utilisé spécialement pour résoudre un tel problème.

A partir du travail réalisé dans le cadre de cette thèse, plusieurs voies pour des recherches futures peuvent être dégagées:

- Traiter plus de problèmes d'optimisation avec les algorithmes évolutionnaires quantiques.
- Concevoir et réaliser des versions distribuées des algorithmes évolutionnaires quantiques pour améliorer les temps d'exécution.
- Réfléchir sur l'intégration des principes de l'informatique quantique à d'autres techniques, notamment les réseaux de neurones, les colonies de fourmis et les essaims de particules.

Bibliographie

- [Aarts et al. 2005] E. Aarts, J. Korst, W. Michiels. "Simulated annealing". In E. K. Burke and G. Kendall, editors, *Search methodologies: Introductory tutorials in optimization and decision support techniques*, pp. 187-210. Springer Verlag, 2005.
- [Aarts et Korst 1988] E. Aarts, J. Korst. *Simulated annealing and boltzmann machines: A stochastic approach to combinatorial optimization and neural computing*. John Wiley & Sons, Chichester, 1988.
- [Adams et Bischof 1994] R. Adams, L. Bischof. "Seeded region growing". *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 16(6), pp. 641-647, 1994.
- [Al-kazemi et Mohan 2002] Al-kazemi, C. K. Mohan. "Multi-phase discrete particle swarm optimization". In *Fourth International Workshop on Frontiers in Evolutionary Algorithms*, Kinsale, Ireland, 2002.
- [Alpert et al. 1996] N. M. Alpert, D. Berdichevsky, Z. Levin, E. D. Morris, A. J. FISCHMAN. "Improved methods for image registration" *NeuroImage*, Vol. 3, pp.10-18, 1996.
- [Ayache 1989] N. Ayache. "Vision stéréoscopique et perception multi-sensorielle". *InterEditions, Paris*, 1989.
- [Bendiab et al. 2003] E. Bendiab, S. Meshoul, M. Batouche. "Une stratégie de recherche pour l'alignement d'images multi-sources". *Proceedings of the 6th International Symposium on Programming and Systems (ISPS'03)*, May 2003.
- [Berro 2001] A. Berro. *Optimisation multiobjectif et strategies d'évolution en environnement dynamique*. PhD thesis, Université des Sciences Sociales Toulouse I, Toulouse, France, December 2001.
- [Bhandarkar et Zhang 1999] S. M. Bhandarkar, H. Zhang "Image segmentation using evolutionary computation". *IEEE Transaction On Evolutionary Computation*, Vol. 3(1), pp. 1-21, 1999.

- [Blahut 1987] R. E. Blahut. *Principles and practice of information theory*. Addison-Wesley Pub. Comp, 1987.
- [Blum et Roli 2003] C. Blum, A. Roli. "Metaheuristics in combinatorial optimization: Overview and conceptual comparison". *ACM Computing Surveys*, Vol. 35(3), pp. 268-308, 2003.
- [Brown 1992] L. G. Brown. "A survey of image registration techniques". *ACM computing surveys*, vol. 24(4), pp. 325-376, 1992.
- [Buzug et Weese 1998] T. M. Buzug, J. Weese. "Voxel-based similarity measures for medical image registration in radiological diagnosis and image guided surgery". *Journal of Computing and Information Technology*, pp. 165-179, 1998.
- [Chang et Li 1994] Y. L. Chang, X. Li. "Adaptive image region-growing". *IEEE Transaction on Image Processing*, Vol. 3(6), pp. 868-872, 1994.
- [Chun et Yang 1996] D. N. Chun, H. S. Yang. "Robust image segmentation using genetic algorithm with a fuzzy measure". *Pattern Recognition*, Vol. 29(7), pp. 1195-1211, 1996.
- [Coello 1999] C. A. Coello Coello. "A comprehensive survey of evolutionary-based multiobjective optimization techniques". *Knowledge and Information Systems*. Vol. 1(3), pp. 269-308, 1999.
- [Coello 2006] C. A. Coello Coello. "20 years of evolutionary multi-objective optimization: what has been done and what remains to be done". In *Computational Intelligence: Principles and Practice*, Chapter 4, pp. 73-88, IEEE Computational Intelligence Society, 2006.
- [Coello et al. 2002] C. A. Coello Coello, D. A. Van Veldhuizen, G. B. Lamont. *Evolutionary algorithms for solving multi-objective problems*. Kluwer Academic Publisher, Springer, 2002.
- [Coello et Toscano 2001] C. A. Coello Coello, G. Toscano. "A micro-genetic algorithm for multiobjective optimization". In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, pp. 126-140. Springer-Verlag. Lecture Notes in Computer Science No. 1993, 2001.

- [Collignon et al. 1995] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, G. Marchal. "Automated multi-modality image registration based on information theory". In *Proceeding of XIVth international conference on information processing in medical imaging (IPMI'95)*, pp. 263-274, Kluwer Academic Publishers, 1995.
- [Colorni et al. 1992] A. Colorni, M. Dorigo, V. Maniezzo. "Distributed optimization by ant colonies". In F. J. Varela and P. Bourguine, editors, *Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*, pp 134-142. The MIT Press, Cambridge, MA. 1992.
- [Corne et al. 2001] D. W. Corne, N. R. Jerram, J. D. Knowles, M. J. Oates. "PESA-II: region-based selection in evolutionary multiobjective optimization". In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, pp. 283-290, Morgan Kaufmann Publishers, San Francisco, California, July 2001.
- [Cover et Thomas 1991] T. M. Cover, J. A. Thomas. *Elements of information theory*. John Wiley & Sons, inc. 1991.
- [Deb et al. 2000] K. Deb, S. Agrawal, A. Pratab, T. Meyarivan. "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II". *KanGAL report 200001*, Indian Institute of Technology, Kanpur, India, 2000.
- [Deb et Goel 2001] K. Deb, T. Goel. "A hybrid multi-objective evolutionary approach to engineering shape design". In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, pp. 385-399. Springer-Verlag. Lecture Notes in Computer Science No. 1993, 2001.
- [Deutsch 1985] D. Deutsch. "Quantum theory, the Church-Turing principle and the universal quantum computer". In *Proceedings of Royal Society of London*, A 400, pp. 97-117, 1985.
- [Deutsch et Ekert 1998] D. Deutsch, A. Ekert. "Quantum computation". *Physics World*, Vol. 11, pp. 47-52, march 1998.

- [Deutsch et Jozsa 1992] D. Deutsch, R. Jozsa. "Rapid solution of problems by quantum computation". *Proceedings of Royal Society of London A439*, pp. 553–558. October 1992.
- [Dirac 1958] P. Dirac, *The Principles of quantum mechanics*, Oxford University Press, 1958.□
- [Dorigo et Stützle 2004] M. Dorigo, T. T. Stützle. *Ant colony optimization*. The MIT Press, Cambridge, MA. 2004.
- [Draa et al. 2005] A. Draa, H. Talbi, M. Batouche. "A quantum-inspired genetic algorithm for solving the N-Queens problem". In *Proceedings of the 7th International Symposium on Programming and Systems, (ISPS'2005)*, pp.145-152, Algiers, May 2005.
- [Ducheyne et al. 2001] E. I. Ducheyne, R. R. De Wulf, B. De Baets. "Bi-objective genetic algorithm for forest management: a comparative study". *Proceedings of the 2001 Genetic and Evolutionary Computation Conference. Late-Breaking Papers*, pp. 63-66, San Francisco, California, July 2001.
- [Duda et Hart 1973] R. O. Duda, P. E. Hart. *Pattern classification and scene analysis*. John Wiley & Sons, New-York, 1973.
- [Ekert et al. 2000] A. Ekert, P. Hayden, H. Inamori. "Basic concepts in quantum computation". *Center for Quantum Computation*, University of Oxford, January 2000.
- [Erickson et al. 2001] M. Erickson, A. Mayer, J. Horn. "the niched pareto genetic algorithm 2 applied to the design of groundwater remediation systems". In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, pp. 681-695, Springer-Verlag, Lecture Notes in Computer Science No. 1993, 2001.
- [Faugeras 1993] O. Faugeras. *Three-Dimensional Computer Vision*. The MIT Press, 1993.
- [Feynman 1965] R. Feynman. *The character of physical law*. Cambridge, M.I.T. Press, 1965.
- [Fonseca et Fleming 1993] C.M. Fonseca, P.J. Fleming. "Multiobjective genetic algorithms". In *IEE Colloquium on Genetic Algorithms for Control Systems Engineering*, pp. 6/1-6/5, IEE, UK, 1993.

- [Fourman 1985] M. P. Fourman. "Compaction of symbolic layout using genetic algorithms". In *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*, pp. 141-153, London, Lawrence Erlbaum, 1985.
- [Freville 2000] A. Freville. "Méthodes de recherche locale". *Journée AFPLC*, École des Mines de Nantes, 2000.
- [Fu et Mei 1981] K. S. Fu, J. K. Mei. "A survey on image segmentation". *Pattern Recognition*, Vol. 13, pp. 3-16, 1981.
- [Gendreau et Potvin 2005] M. Gendreau, J.-Y. Potvin. "Metaheuristics in combinatorial optimization". *Annals of Operations Research*, Vol. 140(1), pp. 189-213. 2005.
- [Giraldi et al. 2004] G. A. Giraldi, R. Portugal, R. N. Thess. "Genetic algorithms and quantum computation". *Technical report*, 0403003, National Laboratory for Scientific Computing, Petropolis, RJ, Brazil, 2004.
- [Goldberg 1989] D. E. Goldberg. *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1989.
- [Goldberg et al. 1989] D. E. Goldberg, B. Korb, K. Deb. "Messy genetic algorithms: Motivation, analysis, and first results". *Complex Systems*, Vol. 3(5), pp. 493-530, 1989.
- [Grover 1996] L. K. Grover. "A fast quantum mechanical algorithm for database search". In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, pp. 212-219, 1996.
- [Gutin et Punnen 2006] G. Gutin, A.P. Punnen. *The traveling salesman problem and its variations*, Springer, 2006.
- [Han 2003] K. H. Han. *Quantum-inspired evolutionary algorithm*. Ph.D thesis, Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology (KAIST), June 2003.
- [Han et Kim 2000] K. H. Han, J. H. Kim. "Genetic quantum algorithm and its application to combinatorial optimization problem". In *Proceedings of the 2000 IEEE Congress on Evolutionary Computation*, IEEE Press, pp. 1354-1360, July 2000.

- [Han et Kim 2002] K. Han, J. Kim. "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization". *IEEE Transaction On Evolutionary Computation*, Vol. 6(6), Dec. 2002.
- [Helterbrand 1996] J. D. Helterbrand. "One-pixel-wide closed boundary identification". *IEEE Transaction on Image Processing*, Vol. 5(5), pp.780-783, 1996.
- [Herreros et al. 1999] A. Herreros, E. Baeyens, J. R. Perán. "Design of multiobjective robust controllers using genetic algorithms". In *Proceedings of the 1999 Genetic and Evolutionary Computation Conference*, pp. 131-132, Orlando, Florida, July 1999.
- [Ho et Lee 2001] S. Y. Ho, K. Z. Lee. "An efficient evolutionary image segmentation algorithm". In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1327-1334, 2001.
- [Hojjatoleslami et Kittler 1998] S. A. Hojjatoleslami, J. Kittler. "Region growing: a new approach". *IEEE Transaction on Image Processing*, Vol. 7(7), pp. 1079-1084, 1998.
- [Holland 1975] J. H. Holland. *Adaptation in natural and artificial systems*. MIT Press, Cambridge, MA, 1975.
- [Horaud et Monga 1995] R. Horaud, O. Monga. *Vision par ordinateurs : outils fondamentaux*. Edition Hermes, 1995.
- [Horn et al. 1994] J. Horn, N. Nafpliotis, D. E. Goldberg. "A niched pareto genetic algorithm for multiobjective optimization". In *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, Vol.1, pp. 82-87, Piscataway, New Jersey, June 1994.
- [Horn et Nafpliotis 1993] J. Horn, N. Nafpliotis. "Multiobjective optimization using the niched pareto genetic algorithm". *Technical Report IlliGAL Report 93005*, University of Illinois at Urbana-Champaign, Urbana, Illinois, USA, 1993.
- [Horowitz et Pavlidis 1974] J.L. Horowitz, T. Pavlidis. "Picture segmentation by a direct split-and-merge procedure". In *Proc. of the 2nd ICPR*, pp. 424-433, 1974.

- [Kane 1998] B. E. Kane. "A silicon-based nuclear spin quantum computer". *Nature*, Vol. 393, pp. 133-137, 1998.
- [Karp 1975] R. M. Karp. On the computational complexity of combinatorial problems. *Networks*, Vol. 5, pp. 45-68. 1975.
- [Kim et al. 2006] Y. Kim, J. Kim, K. Han. "Quantum-inspired multiobjective evolutionary algorithm for multiobjective 0/1 knapsack problems". In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 9151-9156, Jul. 2006.
- [Knowles et al. 2000] J. D. Knowles, D. W. Corne, M. J. Oates. "On the assessment of multiobjective approaches to the adaptive distributed database management problem". In *Proceedings of the Sixth International Conference on Parallel Problem Solving from Nature (PPSN VI)*, pp. 869-878, Berlin, September 2000. Springer.
- [Knowles et al. 2000] J. D. Knowles, D. W. Corne, M. J. Oates. "The Pareto-envelope based selection algorithm for multiobjective optimization". In *Proceedings of the Sixth International Conference on Parallel Problem Solving from Nature (PPSN VI)*, pp. 839-848, Berlin, September 2000. Springer.
- [Knowles et Corne 1999] J. D. Knowles, D. W. Corne. "The Pareto archived evolution strategy: a new baseline algorithm for multiobjective optimisation". In *1999 Congress on Evolutionary Computation*, pp. 98-105, Washington, D.C., July 1999.
- [Knowles et Corne 2000] J. D. Knowles, D. W. Corne. "Approximating the nondominated front using the Pareto archived evolution strategy". *Evolutionary Computation*, Vol.8(2), pp.149-172, 2000.
- [Laburthe 1998] F. Laburthe. *Contraintes et algorithmes en optimisation combinatoire*. Thèse de Doctorat, PARIS VII, 1998.
- [Lahanas et al. 2001] M. Lahanas, N. Milickovic, D. Baltas, N. Zamboglou. "Application of multiobjective evolutionary algorithms for dose optimization problems in brachytherapy". In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, pp. 574-587, Springer-Verlag, Lecture Notes in Computer Science No. 1993, 2001.

- [Lawler et al. 1985] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, D.B. Shmoys. *The traveling salesman problem: a guided tour of combinatorial optimization*. John Wiley & Sons, 1985.
- [Magniez 2002] F. Magniez. "Introduction au calcul quantique". *Centre national de la recherche scientifique (CNRS)*, Décembre 2002.
- [Meshoul et al. 2001] S. Meshoul, M. Batouche, K. Belhadj-Moustefa. "An evolutionary framework for image data fusion based on the maximization of mutual information." *Proceedings of the International Symposium on Software and Systems (ISS'2001)*, February 2001.
- [Meshoul 2004] S. Meshoul. *Optimisation par les systèmes complexes pour le recalage et la mise en correspondance en analyse d'images*. Thèse de Doctorat, Université Mentouri, Constantine, 2004.
- [Narayanan et Moore 1996] A. Narayanan, M. Moore. "Quantum-inspired genetic algorithm". In *Proceedings of the IEE International Conference on Evolutionary Computation*, pp. 41-46, 1996.
- [Nedjah et al. 2008] N. Nedjah, L.S. Coelho, L.M. Mourelle. *Quantum inspired intelligent systems*. Springer-Verlag Berlin Heidelberg. 2008.
- [Osman 1993] I. H. Osman. "Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem". *Annals of Operations Research*, Vol. 41, pp. 421-451. 1993.
- [Osyczka 1985] A. Osyczka. Multicriteria optimization for engineering design. In *Design Optimization*, J. S. Gero, Ed. Academic Press, Inc., New York, NY, 193-227, 1985.
- [Pal et Pal 1993] R. Pal, S. K. Pal. "A review in image segmentation techniques". *Pattern Recognition*, Vol. 26, pp. 1277-1294, 1993.
- [Pappas 1992] T. N. Pappas. "An adaptive clustering algorithm for image segmentation". *IEEE Transaction on Signal Processing*, Vol. 40(4), pp. 901-914, 1992.
- [Pareto 1886] V. Pareto. *Cours d'économie politique*, Ed. Rouge, Lausanne, Suisse, 1886.

- [Pennec 1996] X. Pennec. *L'incertitude dans les problèmes de reconnaissance et de recalage. Application en imagerie médicale et biologie moléculaire*. Thèse de Doctorat, Ecole Polytechnique, 1996.
- [Petrovski et McCall 2001] A. Petrovski, J. McCall. "Multi-objective optimisation of cancer chemotherapy using evolutionary algorithms". In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, pp. 531-545, Springer-Verlag, Lecture Notes in Computer Science No. 1993, 2001.
- [Reeves et Rowe 2003] C. R. Reeves, J. E. Rowe. *Genetic algorithms: principles and perspectives- a guide to GA theory*. Operations Research/Computer Science Interfaces Series. Kluwer Academic Publishers, Boston, MA, USA, 2003.
- [Rezaee 2000] M. R. Rezaee, P. M. J. van der Zwet, B. P. E Lelieveldt, R. J. van der Geest, J. H. C. Reiber. "A multiresolution image segmentation technique based on pyramidal segmentation and fuzzy clustering". *IEEE Transaction on Image Processing*, Vol. 9(7), pp. 1238-1248, 2000.
- [Rieffel et Polak 2000] E. Rieffel, W. Polak. "An introduction to quantum computing for non-physicists". *arxiv.org, quant-ph/9809016 v2*, Jan. 2000.
- [Roche 2001] A. Roche. *Recalage d'images médicales par inférence statistique*. Thèse de doctorat, Université de NICE-SOPHIA ANTIPOLIS, Février 2001.
- [Sahoo et al. 1988] P. K. Sahoo, S. Soltani, A. K. C. Wong "A survey of thresholding technique ". *CVGIP 41*, pp. 233-260, 1988.
- [Sarrut 2000] D. Sarrut. *Recalage multimodal et plate-forme d'imagerie médicale à accès distant*. Thèse de Doctorat. Université lumière lion 2, Janvier 2000.
- [Schaffer 1985] J. D. Schaffer. "Multiple objective optimization with vector evaluated genetic algorithms". In *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*, pp. 93-100, London, Lawrence Erlbaum, 1985.
- [Schmidt-Kaler et Grangier 2006] F. Schmidt-Kaler, P. Grangier. "Les constructeurs de qubits". *La recherche*. N°398, pp. 38-41. Juin 2006.

- [Shannon 1951] C. E. Shannon. "Prediction and entropy of printed English". *The Bell System Technical Journal*, Vol. 30, pp. 50-64, January 1951.
- [Shor 1994] P. Shor. "Algorithms for quantum computation: discrete logarithms and factoring". In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science, IEEE Press*, Nov. 1994.
- [Sofge 2008] D. A. Sofge. "Prospective algorithms for quantum evolutionary computation". In *Proceedings of the Second Quantum Interaction Symposium (QI-2008)*, College Publications, UK, 2008.
- [Srinivas et Deb 1994] N. Srinivas, K. Deb. "Multiobjective optimization using nondominated sorting in genetic algorithms". *Evolutionary Computation*, Vol.2(3), pp.221-248, Fall 1994.
- [Stolze et Suter 2008] J. Stolze, D. Suter. *Quantum computing: a short course from theory to experiment*. Wiley-VCH, 2008.
- [Studholme et al. 1995] C. Studholme, D. L. G. Hill, D. J. Hawkes. "Multiresolution voxel similarity measures for MR-PET registration". In *Proceeding of XIVth international conference on information processing in medical imaging (IPMI'95)*, pp. 287-298. Kluwer Academic Publishers, 1995.
- [Studholme et al. 1998] C. Studholme, D. L. G. Hill, D.J. Hawkes. "An overlap invariant entropy measure of 3D medical image alignment". *Pattern Recognition*, Vol. 1(32), pp. 71-86. 1998.
- [Talbi et Batouche 2004] Ha. Talbi, M. Batouche. "L'optimisation par essaims de particules : application au recalage d'images". *Proceeding du 4^{ieme} Séminaire National sur l'Informatique, (SNIB'04), Biskra*, pp. 195-204, Mai 2004.
- [Talbi et Layeb 2000] H. Talbi, A. Layeb. *Un réseau de neurone artificiel pour la mise en correspondance des primitives en recalage d'image*. Mémoire d'Ingénieur, Université Mentouri, Constantine, Juin 2000.

- [Talbi et al. 2004a] H. Talbi, A. Draa, M. Batouche. "A quantum genetic algorithm for image registration". In *Proceedings of the International Conference on Information & Communication Technologies: from Theory to Applications (ICTTA'2004)*, Damascus, IEEE Press, April 2004.
- [Talbi et al. 2004b] H. Talbi, M. Batouche, A. Draa. "A quantum-inspired genetic algorithm for multi-source affine image registration". *International Conference on Image Analysis and Recognition (ICIAR'2004)*, Porto, September 2004, Lecture Notes in Computer Science: Image Analysis and Recognition Springer-Verlag Press, LNCS 3211 pp. 147-154.
- [Talbi et al. 2004c] H. Talbi, A. Draa, M. Batouche. "A new quantum-inspired genetic algorithm for solving the travelling salesman problem". In *Proceedings of the IEEE International Conference on Industrial Technology. (ICIT'2004)*, Hammamet, Volume 3, pp. 1192 - 1197, IEEE Press, December 2004.
- [Talbi et al. 2006] H. Talbi, A. Draa, M. Batouche. "A novel quantum-inspired evolutionary algorithm for multi-sensor image registration". *International Arabic Journal on Information Technology*, Vol. 3(1), pp. 9-15, Jan. 2006.
- [Tou et Gonzalez 1974] J. T. Tou, R. C. Gonzalez. *Pattern Recognition Principles*. Reading, MA, Addison-Wesley, 1974.
- [TSPLIB95] <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/tsp/>
- [Van et Aarts 1987] P.J.M. Van, E.H.L. Aarts. *Simulated annealing theory and applications*. Reidel, Dordrecht, Holland, 1987.
- [Van Veldhuizen et Lamont 2000] D. A. Van Veldhuizen, G. B. Lamont. "Multiobjective optimization with messy genetic algorithms". In *Proceedings of the 2000 ACM Symposium on Applied Computing*, pages 470-476, Villa Olmo, Como, Italy, 2000. ACM.
- [Viola 1995] P. Viola. *Alignment by maximization of mutual information*. Thèse de Doctorat, Massachusetts Institute of Technology. 1995.

- [Waldner 2006] J. B. Waldner. *Nano-informatique et intelligence ambiante: Inventer l'ordinateur du XXIème siècle*, Hermes Science, London, 2006.
- [Wells et al.1996] W. M. Wells, P. Viola, H. Atsumi, S. Nakajima "Multi_modal volume registration by maximization of mutual information". *Medical Image Analysis*, Vol. 1(1), pp. 35-51, 1996.
- [Woods et al. 1992] R. P. Woods, S. R. Cherry, J. C. Mazziotta. "Rapid automated algorithm for aligning and reslicing PET images" *Journal of Computer Assisted Tomography*, Vol. 16(4), pp. 620-633, 1992.
- [Zadeh 1963] L. A. Zadeh. "Optimality and non-scalar valued performance criteria". *IEEE Transactions on Automatic Control*, Vol. 8(1), pp. 59-60, 1963.
- [Zhang 1993] Z. Zhang. "Le problème de mise en correspondance : l'état de l'art". *INRIA RR n°2146*, 1993.
- [Zitzler et al. 1999] E. Zitzler, J. Teich, S. S. Bhattacharyya. "Multidimensional exploration of software implementations for DSP algorithms". *Journal of VLSI Signal Processing*, Vol. 24(1), pp. 83-98, February 1999.
- [Zitzler et al. 2001] E. Zitzler, M. Laumanns, L. Thiele. "SPEA2: Improving the strength pareto evolutionary algorithm". *TIK Report, ETH Zurich*. May 2001.
- [Zitzler et Thiele 1999] E. Zitzler, L. Thiele. "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach". *IEEE Transactions on Evolutionary Computation*, Vol.3(4), pp. 257-271, November 1999.
- [Zydallis et al. 2001] J. B. Zydallis, D. A. Van Veldhuizen, G. B. Lamont. "A statistical comparison of multiobjective evolutionary algorithms including the MOMGA-II". *In Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, pp. 226-240. Springer-Verlag. Lecture Notes in Computer Science No. 1993, 2001.