

RAPPORT D'HABILITATION

Université de Paris XI Orsay

Spécialité Informatique

Genetic Algorithms and Fractals



Algorithmes Génétiques et Fractales

Evelyne LUTTON

INRIA Rocquencourt, B.P. 105, 78153 LE CHESNAY Cedex, France

Tel : +33 1 39 63 55 23 Fax : +33 1 39 63 59 95

e-mail: Evelyne.Lutton@inria.fr <http://www-rocq.inria.fr/fractales/>

11 Février 1999

<i>Rapporteurs</i>	Wolfgang BANZHAF Pierre LIARDET Edward VRSCAY	<i>Fachbereich Informatik, Universität Dortmund</i> <i>CMI, Université de Marseille</i> <i>Dpt of Applied Mathematics, University of Waterloo</i>
<i>Examineurs</i>	Jean-Paul ALLOUCHE Michel COSNARD Jacques LÉVY VÉHEL Jean-Arcady MEYER Marc SCHOENAUER	<i>LRI, CNRS, Université d'Orsay</i> <i>INRIA Lorraine</i> <i>INRIA Rocquencourt & IRCYN</i> <i>Université Pierre et Marie Curie</i> <i>CMAP, Ecole Polytechnique</i>

à Emma-petit-chat, qui me suit à petits pas,

à Jean-Pierre, qui m'accompagne,

*à Marie-Thérèse, Jean, Madeleine, Emile,
et les autres, qui m'ont montré le chemin ...*

Ainsi dès l'antiquité, se manifestent plusieurs conceptions d'importance : hiérarchie des êtres vivants et gradation naturelle (Aristote) ; production de l'harmonie organique par le hasard et la mort (Empédocle, Démocrite, etc ...) ; lutte des vivants pour la vie (Lucrèce)."

L'évolution des espèces. Histoire des idées transformistes.

Chapitre I. L'antiquité.

Jean Rostand. Librairie Hachette. 1932.

Remerciements

Il m'a fallu exactement 8 ans, 9 mois et 9 jours depuis la soutenance de ma thèse de doctorat pour me décider à présenter le document que voici. En fait, la soutenance d'une habilitation est le prétexte officiel pour faire le point sur quelques années de travail de recherche, d'encadrement, d'organisation, bref, une pause. Et finalement, ce n'est pas si désagréable - autrement dit c'était moins difficile que ce que j'imaginai (à part peut-être les aspects administratifs ...).

Evidemment, le plus dur à rédiger est curieusement la page des remerciements, celle que l'on fait souvent en dernier, où il ne faut oublier personne, celle qui fait penser presque toujours à un catalogue de synonymes (je remercie, merci à, je suis reconnaissante à, etc ...), facilement lyrique, un brin littéraire. Pas facile, donc.

Merci, ce petit mot magique, que j'enseigne maintenant à ma fille ("qu'est-ce qu'on dit à la dame?"), semble donc être le mot le plus employé et décliné dans les documents de thèse, du moins dans les premières pages. Comment faire pour qu'il ne perde pas sa signification à force d'être répété? Alors, voilà, après mûres réflexions (j'ai eu du temps pour y réfléchir, finalement!), je propose un grand ...



- à Wolfgang Banzhaf, Pierre Liardet et Ed Vrscay, qui ont accepté d'être rapporteurs de mon habilitation, pour toutes les discussions passionnantes que nous avons eues,
- à Jean-Paul Allouche, Michel Cosnard, et Jean-Arcady Meyer, qui ont accepté si volontiers de participer à mon jury,
- à Jacques Lévy Véhel, qui m'a fait découvrir et apprécier les fractales ; mon complice de travail préféré depuis 6 ans, années qui m'ont parues tellement courtes,
- à Marc Schoenauer, qui m'a soutenue par ses conseils, et avec qui il est si facile de travailler, contrairement à ce que pourrait laisser croire son emploi du temps de ministre ...
- à Claude Tricot qui m'a souvent si chaleureusement soutenue et recommandée,
- aux membres et ex-membres du projet FRACTALES de l'INRIA, qui ont contribué à créer et maintenir une ambiance de travail si motivante,
- aux étudiants qui ont eu à me subir, et qui ont su garder leur bonne humeur : Benoît Leblanc et Frédéric Raynal, jeunes thésards depuis Octobre (pour eux cela ne fait que commencer!), et par ordre chronologique: Patrice Martinez, Khadidja Benabdallah, Guillaume Cretin, Philippe Glevarec, Cédric Roll, David Bonnefoy,
- à Pierre Collet, pour son enthousiasme, sa curiosité et sa magistrale maîtrise de C++ et de GALIB,
- à Nathalie Gaudechoux, la secrétaire de notre projet, pour sa compétence, sa disponibilité, son efficacité et sa gentillesse,
- aux compagnons évolutionnaires: Jean-Marc Alliot, Jin-Kao Hao, Edmund Ronald, Michèle Sebag, Dominique Snyers, les gentils organisateurs d'EA,
- à Bertrand Braunchweig, pour toutes les répétitions,
- à Jean-Paul Sansonnet puis Christine Froidevaux, à Mme Lefebvre, qui m'ont guidée dans les méandres administratifs du LRI, pour arriver à soutenir enfin cette habilitation,
- aux anglophiles et -phones qui ont bien voulu corriger la partie anglaise de mon manuscrit : Richard James, Catherine Azoulay,
- à Feryel, Martin et Véro Krüger, Françoise Lévy, Jean Louchet pour leur amitié et leur soutien à la cause évolutionnaire (!),
- à Marthe et Jean, Madeleine et Roby, Isabelle et Youssef, Sonia, Yassin et Samy, mes supporters alsaciens.
- à Marie-Thérèse et Jean, mes supporters auvergnats,
- à Jean-Pierre pour nos 10 années de cheminement commun, et à Emma pour tous ses petits pas depuis 2 ans et demi.

Structure of the document

This report represents the synthesis of research about Genetic Algorithms achieved under my supervision from the creation of the FRACTALES group of INRIA in 1993. This document submitted to the Orsay University for the “Habilitation à diriger des recherches” degree, first contains a short french abstract (“the abstract of the abstract”); then a synthetic English presentation of the main results and applications of my publications; these are gathered in the third part of this document. Due to this structure, some redundancies that may seem unaesthetic can be found between the French and English parts: I hope that the bilingual reader will forgive me this offence to scientific concision ...

Contents

I	Algorithmes Génétiques et Fractales	1
II	Genetic Algorithms and Fractals	11
1	Introduction	13
2	Theoretical Analysis	15
2.1	Hölder functions	16
2.2	Sampling rate influence : a general result	17
2.3	GA-Deception analysis of Hölder functions	18
2.3.1	Schema theory	19
2.3.2	Walsh polynomials and Deception characterization	20
2.3.3	Haar polynomials for the deception analysis of Hölder functions	20
2.3.4	Non-deception criterion?	22
2.4	Bitwise regularity analysis	24
2.4.1	Bitwise regularity coefficients.	24
2.4.2	Bitwise regularity coefficients compared to Hölder exponent.	25
2.5	Deception analysis of a GA with uniform crossover	25
2.6	Conclusions and Further work	27
3	Applications	29
3.1	Inverse Problems for IFS	31
3.1.1	Introduction	31
3.1.2	IFS Theory	31
3.1.3	The case of affine IFS : use of a GA	34
3.1.4	Mixed IFS	36
3.1.5	Inverse problem for Mixed IFS : a solution based on GP	37
3.2	Interactive GP for the generation of Mixed IFS attractors	38
3.3	Finite Automata	39
3.4	Other Applications	41
3.4.1	Image Analysis	41
3.4.2	Physics signals analysis	43
3.5	Software	46
3.5.1	ALGON : A GA software	46
3.5.2	PROGON : A GP software	46
4	Concluding remarks and future directions ...	47

III Publications	57
A Hölder functions and Deception of Genetic Algorithms,	
Evelyne LUTTON, Jacques LEVY VEHEL, IEEE transactions on Evolutionary Computation, Vol 2 Number 2, July 1998. This paper is available on : http://www-rocq.inria.fr/fractales/Publications/	59
B Bitwise regularity coefficients as a tool for deception analysis.	
Benoît LEBLANC, Evelyne LUTTON, INRIA Research Report No 3274, October 1997 This paper is available on : http://www-rocq.inria.fr/fractales/Publications/	77
C Mixed IFS: resolution of the inverse problem using Genetic Programming,	
Evelyne LUTTON, Jacques LEVY VEHEL, Guillaume CRETIN, Philippe GLEVAREC, Cédric ROLL Complex Systems, Vol 9, No 5, pp 375-398, October 1995. This paper is available on : http://www-rocq.inria.fr/fractales/Publications/	129
D Fractal Modeling of Speech Signals,	
Jacques LEVY VEHEL, Khalid DAOUDI, Evelyne LUTTON, Fractals, Vol 2, No 6, September 1994. This paper is available on : http://www-rocq.inria.fr/fractales/Publications/	159
E Inverse Problems for Finite Automata: A Solution Based on Genetic Algorithms	
Benoît LEBLANC, Evelyne LUTTON, Jean-Paul ALLOUCHE, Evolution Artificielle, Nîmes, 22-24 Octobre 1997. This paper is available on : http://www-rocq.inria.fr/fractales/Publications/	167
F Genetic Algorithms as a tool in the study of aperiodic order : the case of X-Ray diffraction spectra of GaAs-AlAs multilayer heterostructures,	
Benoît LEBLANC, Evelyne LUTTON, Françoise AXEL, Submitted to The European Physical Journal B., July 1997. This paper is available on demand.	179
G A Genetic Algorithm for the Detection of 2D Geometric Primitives in Images	
Evelyne LUTTON, Patrice MARTINEZ, 12-ICPR, Jerusalem, Israel 9-13 Octobre 1994. This paper is available on : http://www-rocq.inria.fr/fractales/Publications/	201

Part I

**Algorithmes Génétiques et
Fractales**

Les algorithmes génétiques (AG) et plus généralement les algorithmes évolutionnaires (AE) sont des méthodes de résolution de problèmes qui sont fondées sur une métaphore biologique. Schématiquement, ils copient de façon extrêmement simplifiée certains comportements des populations naturelles. Ainsi, ces techniques reposent toutes sur l'évolution d'une population de solutions (c'est-à-dire des points de l'espace de recherche), qui sous l'action de règles précises optimisent un comportement donné, exprimé sous forme d'une fonction, dite fonction d'évaluation, "adaptation à l'environnement" ou encore "fitness".

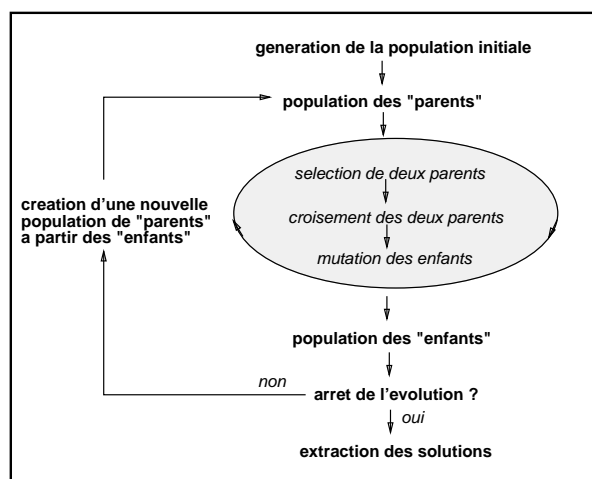


Figure 0.1: *Structure générale d'un algorithme évolutionnaire*

Des populations successives de solutions sont contruites, qui s'adaptent de mieux en mieux à leur "environnement" ; les valeurs de la fonction d'évaluation de ces points augmentent. Les règles d'évolution sont des opérateurs stochastiques: la sélection (où la fonction d'évaluation intervient en tant que biais d'un tirage aléatoire), et les opérateurs "génétiques", qui sont principalement le croisement (combinaison de deux solutions) et la mutation (perturbation d'une solution).

Les AE sont donc tout d'abord des outils d'optimisation très robustes, efficaces lorsque les fonctions à optimiser sont fortement irrégulières, et dépendantes d'un grand nombre de paramètres. Ces stratégies d'optimisation sont fondées sur l'hypothèse que de bonnes solutions (on dit aussi des "individus bien adaptés à leur environnement") peuvent produire des solutions encore plus adaptées par application des opérateurs génétiques. Cette hypothèse peut être reliée à une notion d'AG-difficulté de la fonction à optimiser : c'est dans ce cadre qu'a été élaborée la notion de "déceptivité" (on parle aussi de fonctions "trompeuses").

Les analyses théoriques de ces techniques concernent en général l'analyse de leur convergence (et de leur vitesse de convergence sur des optima localement convexes pour les Stratégies d'Évolution), l'influence des paramètres, l'analyse de l'AG-difficulté. Pour ce qui est des Algorithmes Génétiques, auxquels nous nous intéressons plus particulièrement dans ce rapport, ces analyses sont fondées sur différentes approches :

- des preuves de convergence fondées sur l'analyse par chaînes de Markov: par exemple Davis [19] montre qu'un schéma de décroissance très lent de la probabilité de mutation assure une convergence de l'AG, des approches plus récentes comme [13, 1, 65] fournissent des résultats plus précis en ce qui concerne les conditions de convergence.

- l’analyse de fonctions déceptrives, fondée sur la notion de Schémas et la théorie développée par Holland [33, 26, 27, 29], qui fournit une approche de l’analyse d’efficacité d’un AG, et permet de cerner en quoi certaines fonctions sont “AG-difficiles”.

Certains auteurs relient intuitivement la déceptivité à la notion biologique d’épistasie [18], qui peut être traduite en termes de degrés de non-linéarité. La déceptivité peut aussi être liée à ce que l’on appelle l’analyse des paysages de fitness (“fitness landscape”, voir par exemple [62]). Dans tous les cas, à la base, la déceptivité dépend :

- du jeu de paramètre de l’AG,
 - de la forme de la fonction à optimiser,
 - de la fonction de codage des solutions, c’est-à-dire en quelque sorte de la façon de parcourir l’espace au cours de la recherche.
- enfin, certaines approches très récentes sont fondées sur la modélisation des AG comme systèmes dynamiques [38, 46, 82].

Dans le cadre de la modélisation d’AG comme systèmes dynamiques, des comportements “fractals” ont été mis en évidence, [38]. Cette approche a principalement permis de générer des images fractales (figures de stabilité) et l’aspect fractal des AG pourrait de ce fait éventuellement être considéré comme anecdotique. Le but de ce document est de fournir des justifications plus approfondies du rapport entre algorithmes génétiques et fractales. Notre démarche est la suivante : s’il est vrai que les AG sont adaptés à l’optimisation de fonctions irrégulières en général, il est difficile d’être plus précis sans spécifier le type d’irrégularité auquel on s’intéresse. Or la géométrie fractale se donne justement pour tâche de quantifier cette notion, et à l’intérieur d’une classe de fonctions définie par son irrégularité, il devrait être possible d’obtenir des résultats plus fins sur le comportement d’un AG.

D’un point de vue général la géométrie fractale doit sa célébrité actuelle principalement à ses aspects “graphiques” : images fractales (ensembles de Julia ou de Mandelbrot, montagnes ou paysages fractals), qui possèdent une infinité de détails et des propriétés dites d’auto-similarité. Il existe cependant d’autres aspects de la géométrie fractale, concernant l’analyse de signaux complexes (qui ne sont pas nécessairement fractals) où la théorie fractale et multifractale¹ connaît actuellement un large succès pour des domaines d’application très variés comme l’analyse d’images, la finance ou l’analyse de trafic dans les réseaux, voir par exemple [81].

Dans ce domaine, les algorithmes évolutionnaires se sont révélés être des outils efficaces pour plusieurs applications “fractales” :

- la résolution du problème inverse pour les IFS [84, 83, 60, 56, 77, 49], avec une application en modélisation de signaux de parole [78],
- la compression d’images [79, 25],
- l’optimisation d’antennes fractales [15]

Ce succès est naturellement lié au fait que lorsque que l’on fait une analyse d’un signal irrégulier, on est souvent confronté à des problèmes d’optimisation complexes, où les fonctions à optimiser sont extrêmement irrégulières (pour lesquelles il est très souvent difficile de définir des dérivées) et possèdent de très nombreux maxima locaux. Les méthodes stochastiques et tout particulièrement les algorithmes évolutionnaires sont bien adaptés à ce genre de problèmes d’optimisation.

Plus précisément, le fait de se placer dans un cadre fractal, c’est-à-dire de supposer que les signaux considérés possèdent certaines propriétés précises de régularité locale, permet d’analyser le comportement des AG (analyse pour l’instant fondée sur la théorie des schémas). Comme nous allons le voir

¹Brièvement, l’approche multifractale fournit à la fois des outils de quantification des *irrégularités locales* des signaux considérés, et des outils d’estimation de la *fréquence d’apparition* de ces irrégularités locales.

dans ce rapport, l’analyse de la façon dont un AG optimise certaines fonctions “fractales” (voir chapitre 2) permet de modéliser l’influence d’une partie des paramètres de l’AG sur une mesure de déceptivité. La généralisation de cette analyse fournit des indications sur la façon d’ajuster ces paramètres. Ces résultats peuvent être ensuite exploités pour fournir un outil d’analyse de l’influence du codage des chromosomes dans un AG².

Analyse théorique : la déceptivité dans le cadre fractal

L’analyse résumée ici et développée dans le chapitre 2 est fondée sur la théorie des schémas et l’analyse de déceptivité. Cette théorie offre un modèle extrêmement simplifié du comportement d’un AG qui permet, comme nous le verrons dans ce qui suit, de mener certains calculs jusqu’au bout, chose impossible à faire avec des modèles plus sophistiqués. Il convient cependant d’être prudents en ce qui concerne l’interprétation pratique des résultats tirés de ce modèle simplifié, qui ne peut fournir que des informations sur l’influence relative des paramètres ou du codage (voir annexe B pour des simulations détaillées et une analyse critique de ce modèle).

L’AG modélisé dans le cadre de la théorie des Schémas est le plus simple des AG : l’algorithme canonique, qui utilise la sélection proportionnelle, le croisement à un point et la mutation avec probabilités d’application fixes p_c et p_m .

Supposons que la fonction de fitness f définie sur $\{0, 1\}^l$ est issue par échantillonnage à la précision $\epsilon = \frac{1}{2^l}$ d’une fonction Höldérienne³ F sur l’intervalle $[0, 1]$ (cette hypothèse est toujours valable, même si la fonction sous-jacente unidimensionnelle F ne reflète pas toujours de façon simple le comportement de la fonction de fitness) :

$$\forall x \in \{0, 1\}^l, \quad f(x) = F\left(\frac{I(x)}{2^l}\right)$$

$I(x) \in [0, 2^l - 1]$ est l’entier dont la décomposition binaire est x , $I(x) = \sum_{t=0}^{l-1} x_t 2^t$.

Nous avons montré un premier résultat concernant la finesse d’échantillonnage de la fonction sous-jacente F , c’est-à-dire le paramètre l , que l’on peut relier à la précision de localisation de l’optimum de F , voir section 2.2. Ce résultat a une portée générale et est valable pour toute méthode d’optimisation, y compris les AG.

Nous avons développé ensuite une analyse de déceptivité, décrite en section 2.3, qui à l’aide d’une décomposition de la fonction de fitness dans la base de Haar sur $\{0, 1\}^l$ permet d’établir une relation entre une mesure de “déceptivité” sur f , l’exposant de Hölder de F et trois paramètres de l’AG (l , p_m , et p_c). La première conclusion que l’on peut tirer de cette analyse est que l’ajustement des paramètres l , p_m , et p_c tend à rendre le problème moins difficile (au sens de la déceptivité) pour l’AG, et donc potentiellement à améliorer ses performances. On peut aussi imaginer une procédure “a posteriori” de validation des résultats fournis par l’AG. Des expériences simples nous ont montré que même si le comportement de l’AG correspond qualitativement à la modélisation théorique, une analyse plus fine est nécessaire pour être en mesure de proposer des techniques de validation a posteriori robustes (voir [54]). La principale implication pratique de cette analyse concerne l’influence qualitative des paramètres de l’AG sur le comportement de l’algorithme.

L’analyse précédente est fondée sur l’analyse de l’irrégularité d’une fonction sous-jacente unidimensionnelle selon une distance euclidienne sur $[0, 2^l - 1]$. Une distance plus naturelle sur l’espace $\{0, 1\}^l$ est la distance de Hamming, qui peut aussi servir de base à une analyse de l’irrégularité. C’est ce que

²Attention : il n’y a pas de lien direct démontré entre déceptivité et performances de l’AG, un certain nombre de contre-exemples existent, même. L’interprétation que nous donnons ici frôle donc l’abus de langage. Cependant nos résultats de simulation tendent à prouver que l’AG se comporte en général mieux dans le cas où la déceptivité est plus faible.

³une fonction $F : X \rightarrow Y$, (X, d_X) et (Y, d_Y) étant deux espaces métriques, est Höldérienne d’exposant $h > 0$ et de constante $k > 0$ si pour tout $x, y \in X$ tq $d_X(x, y) < 1$ on a $d_Y(F(x), F(y)) \leq k \cdot d_X(x, y)^h$

nous avons développé dans la section 2.4, et qui permet de fournir une relation plus précise entre déceptivité et mesures d'irrégularité sur la fonction. Nous avons ainsi introduit les mesures d'irrégularité "bit à bit" qui reflètent mieux l'irrégularité de la fonction f sur l'espace $\{0, 1\}^l$, et qui sont plus faciles à estimer qu'un exposant de Hölder. Une application directe de cette analyse concerne l'influence (dans une certaine mesure, voir les expérimentations décrites en annexe B) d'un ré-ordonnement des bits du chromosome sur les performances de l'AG.

Cependant pour ce qui concerne les implications pratiques de ce travail, il faut noter que l'AG canonique n'est absolument pas un modèle employé en pratique. Un premier pas vers des modèles plus réalistes a été fait en reprenant l'analyse précédente pour un AG canonique avec croisement uniforme. Cela nous a permis de proposer une application pratique pour l'évaluation du codage chromosomique pour le croisement uniforme. Des simulations numériques (voir annexe B) sur le code de Gray semblent confirmer la validité pratique de cette analyse.

Bien évidemment, la prudence s'impose quant aux implications pratiques de l'analyse que nous venons de décrire brièvement. Cette analyse a le principal intérêt de relier de façon formelle l'irrégularité de la fonction de fitness à une notion de déceptivité: ce fait, qui paraît une évidence intuitive, n'est pas du tout simple à prouver. Cela nous permet aussi de considérer différemment la notion de déceptivité (qui a tendance actuellement à être en disgrâce dans la communauté AG). Cependant, la notion de déceptivité⁴ ne peut pas et ne doit pas être comprise comme une mesure de performance d'un AG: les implications pratiques que nous présentons ici concernant l'influence du codage des chromosomes sur les performances de l'AG peuvent par conséquent paraître quelque peu abusives. Les expériences que nous avons présentées dans l'annexe B tendent pourtant à prouver que la borne déduite des coefficients de régularité bit-à-bit semble être un assez bon indicateur relatif de l'efficacité d'un codage. A partir du moment où les valeurs associées à deux codages diffèrent suffisamment, les performances expérimentales des AG correspondent à la prédiction de l'analyse.

Il est évidemment nécessaire de poursuivre les expérimentations ainsi que l'analyse théorique pour donner plus de précision à ces résultats. Une voie qui semble intéressante concerne l'analyse de l'irrégularité des fonctions de fitness dans le cadre des systèmes dynamiques [3] et de la modélisation Markovienne des AG. Une approche critique de la théorie des schémas se trouve dans [63], où une analyse de la dynamique des schémas au sein d'un modèle Markovien permet d'établir rigoureusement un théorème des schémas pour les AG à population finie (qui possède des caractéristiques proches du théorème des schémas classique).

Ainsi nous pensons qu'il est nécessaire d'analyser plus à fond la signification de la fonction de fitness ajusté f' qui est définie comme l'espérance de la fitness des points que l'on peut atteindre par croisement et mutation à partir du point considéré, et qui sert de base à la mesure de déceptivité sur laquelle nous avons fondée notre analyse (elle est mesurée par $|f - f'|$):

$$f'(x) = E(f(x'))$$

où x' est une v.a. qui représente les individus qui peuvent être issus de x par croisement et mutation.

L'exemple de la figure 0.2 montre une fonction déceptive, selon la définition de Goldberg [26, 27]: la fonction f' représentée en pointillés a pour optimum 0, alors que ce point est un optimum local sur f représentée en traits continus (le véritable optimum de f est en 7). En revanche, on voit sur la figure 0.3 que la fonction $f(x) = x^2$ n'est pas déceptive. Les figures 0.4 et 0.5 montrent les valeurs de f' pour deux fonctions Höldériennes d'exposants différents.

L'analyse de la décomposition dans la base de Walsh de f' et de la notion épistasie définie dans [63] met en évidence les similarités et les différences entre les deux notions: la fonction f' est calculée à partir de f et des paramètres p_c et p_m de l'AG tandis que l'épistasie représente une mesure intrinsèque

⁴Outre le fait qu'il existe plusieurs définitions (non trivialement reliées l'une à l'autre) de cette notion et que la déceptivité n'est pas la seule cause reconnue d'AG-difficulté.

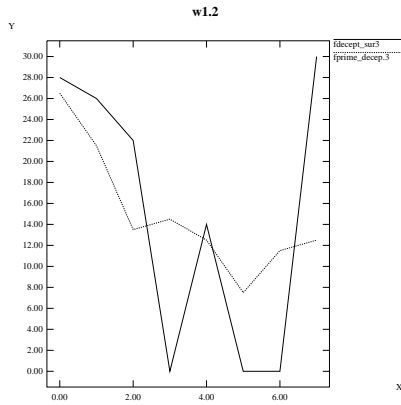


Figure 0.2: Exemple d'une fonction déceptive sur 3 bits

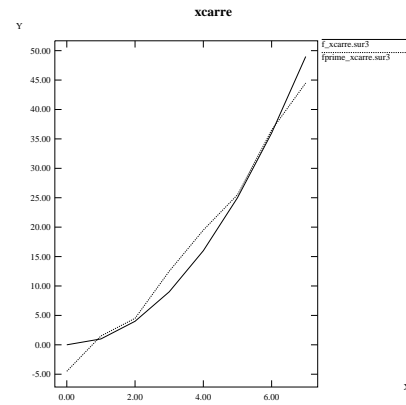


Figure 0.3: la fonction $f(x) = x^2$ sur 3 bits n'est pas déceptive

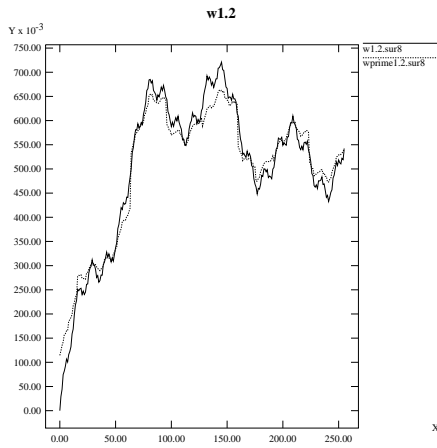


Figure 0.4: La fonction de Weierstrass de dimension fractale 1.2 (exposant de Hölder 0.8) échantillonnée sur 8 bits n'est pas déceptive

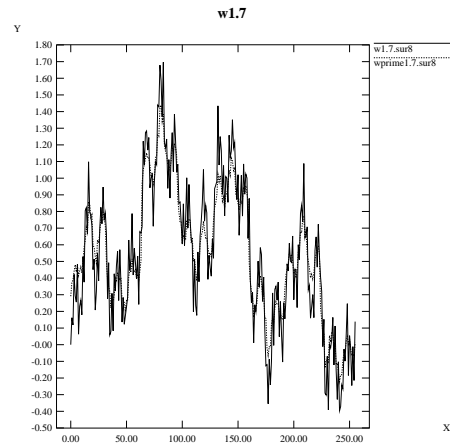


Figure 0.5: La fonction de Weierstrass de dimension fractale 1.7 (exposant de Hölder 0.3) échantillonnée sur 8 bits est déceptive

à f . Les indicateurs dérivés de l'analyse en déceptivité et de l'analyse d'épistasie semblent donc complémentaires pour ce qui est de l'analyse d'AG-difficulté.

Applications

Dans le cadre de l'analyse fractale de données un certain nombre de problèmes délicats peuvent être résolus à l'aide d'AE, le but du chapitre 3 est de décrire les applications que nous avons développées, et qui nous ont permis, parallèlement à l'analyse théorique, d'acquérir une certaine expertise sur le comportement pratique des AE.

D'un point de vue général, ces applications peuvent être formulées comme des problèmes inverses : un problème inverse standard concerne des systèmes complexes pour lesquels à partir d'un ensemble de données d'entrée, on sait calculer (grâce à un modèle) la sortie d'un système, mais pour lesquels il est impossible, à partir des données de sortie de calculer les données d'entrées.

Une stratégie de résolution directe (on pourrait même dire "brutale") utilisable lorsque l'on a très peu de connaissance a priori sur le problème consiste à transformer le problème inverse en un problème d'optimisation : rechercher un ensemble de données d'entrées telles que la sortie correspondante du système soit la plus ressemblante possible à la sortie cible que l'on souhaite inverser.

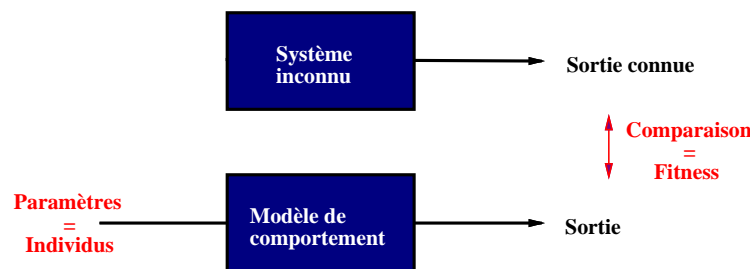


Figure 0.6: Approche de type "boîte noire" d'un problème inverse : un problème d'optimisation.

Usuellement les AE sont bien adaptés à la résolution de problèmes inverses difficiles pour lesquels très peu d'informations a priori sont disponibles (les fonctions à optimiser ne sont connues que de façon implicite et l'on n'a en général aucune connaissances sur leurs dérivées). Dans le cadre de l'analyse fractale de données nous avons pu résoudre un certain nombre de problèmes inverses difficiles de façon satisfaisante à l'aide d'AE :

- Les problèmes inverses pour les IFS, [84, 83, 60, 56]. Des expériences employant des AG dans le cas des IFS affines [77], et de la programmation génétique (PG) pour des IFS non affines (IFS "mixtes") [78] sont décrites dans le chapitre 3.1. Une application à la modélisation de signaux de parole a aussi été développée [78], voir annexe D.
- Le problème inverse pour les automates finis [41], voir chapitre 3.3.

L'emploi d'AE ne se limite pas à l'étude de ces problèmes inverses "académiques", et des applications telles que la compression d'images par IFS [79, 25], ou l'optimisation d'antennes fractales [15] ont pu être abordées avec ces algorithmes.

Les difficultés principales que l'on peut rencontrer dans de telles applications concernent :

- le choix d'un codage adapté et efficace, ce qui permet souvent d'exploiter le peu de connaissances a priori que l'on a sur le système. Dans l'exemple du problème inverse pour les automates finis à point fixe, un codage des longueurs de mot d'un automate a permis de construire un AG beaucoup plus efficace qu'en se fondant sur un codage explicite des mots, voir [41] et l'annexe E.

- la manipulation correcte et économique des contraintes du problème⁵. Le problème inverse pour les IFS en est un bon exemple: les contraintes de contractance peuvent être exploitées avantageusement pour éviter beaucoup de calculs superflus, voir [77, 49] et annexe C.

D'un point de vue général, un AG efficace est construit en exploitant au maximum les connaissances a priori que l'on a sur le problème; ces connaissances a priori peuvent intervenir au niveau du codage, du calcul de la fonction de fitness ou de la gestion des contraintes.

Dans toutes les applications présentées dans le chapitre 3, nous avons choisi d'intégrer les contraintes dans la fonction de fitness sous la forme de pénalités à poids variables, ce qui semble être la meilleure façon de gérer des contraintes qui induisent un espace de recherche très parcellaire. Des poids sont affectés aux diverses contraintes afin de favoriser au début du déroulement de l'AG les termes de pénalité dans la fonction de fitness. Le but est d'attirer les populations dans des régions valides (où les contraintes sont satisfaites) au début de l'évolution de l'AG, puis de favoriser ensuite le terme à optimiser. Cette technique permet de faire très simplement des raccourcis de calculs, principalement lorsque certaines des contraintes ne sont pas satisfaites.

En outre, l'intégration des contraintes par fonctions de pénalités permet de "lisser" la fonction à optimiser, de diminuer de ce fait son irrégularité, et ainsi de tendre à faciliter la tâche de l'AG.

Les applications présentées dans le chapitre 3 ont toutes été développées dans le cadre de l'analyse des signaux :

- liés à l'approche fractale: problème inverse pour les IFS, voir chapitre 3.1 et annexes C et D, pour les automates finis, voir chapitre 3.3 et annexe E,
- en physique: analyse de spectres de diffraction X, voir chapitre 3.4 et annexe F,
- en analyse d'images: détection de primitives géométriques, voir chapitre 3.4 et annexe G.

Ces expériences ont renforcé une fois de plus l'opinion largement répandue parmi les praticiens des AE, qu'un ajustement précis des paramètres, un codage judicieux et une fonction de fitness "économique" d'un point de vue temps de calcul peuvent faire une énorme différence en ce qui concerne l'efficacité et la précision des résultats fournis par un AE.

Bilan

Ce qui est présenté dans ce document résume les travaux que j'ai menés au projet FRACTALES depuis 1992. Il s'en dégage deux aspects: d'une part l'analyse théorique de la déceptivité, et d'autre part un ensemble d'applications touchant des domaines très divers. Ces deux aspects viennent naturellement du mode de travail que j'ai adopté: parallèlement à une réflexion théorique, j'ai mené une politique "d'offre de service" pour des problèmes d'optimisation difficiles. J'ai ainsi pu aborder des problèmes très variés avec mes étudiants (que je tiens à remercier ici pour leur sérieux et leur enthousiasme), en collaboration avec des experts du domaine d'application considéré. Cette stratégie, qui s'est révélée très enrichissante par la variété des problèmes posés et l'expertise pratique que j'ai pu accumuler, se poursuit actuellement, notamment par l'intermédiaire de collaborations qui débutent avec l'équipe du CMAPX, avec le projet SINUS de l'INRIA, avec L'ENSTA, concernant des applications en mécanique, et avec l'IFP pour une étude concernant l'emploi d'algorithmes génétiques en modélisation moléculaire.

⁵La manipulation des contraintes dans les AE n'est pas un problème simple, une bonne revue des méthodes de gestion de contraintes dans se trouve dans [58] ou [67]

Vers une meilleure compréhension de la notion de déceptivité

A la lumière des travaux exposés dans ce rapport, on peut se poser la question de savoir ce que signifie pratiquement la notion de déceptivité. Beaucoup d’encre a coulé à ce propos dans la littérature AE et il est toujours difficile de se faire une opinion. C’est peut-être une des raisons qui a poussé nombre de chercheurs à considérer ce pan de l’analyse théorique des AG comme démodé, et à préférer des approches fondées sur une modélisation des AG sous forme de chaînes de Markov.

Le principal intérêt de l’analyse présentée dans ce rapport est de mettre en évidence le lien entre déceptivité et irrégularité de la fonction de fitness. La notion de déceptivité que nous avons exploitée est fondée sur la notion de fitness ajusté f' , et il semble que l’on pourrait exploiter cette fonction f' dans la construction d’un indicateur de performance d’un AG. Entendons-nous : nous ne prétendons absolument pas relier la performance d’un AG à la mesure $|f - f'|$ (nos simulations numériques nous ont prouvé si besoin était la faiblesse d’une telle prétention), mais nous pensons que la fonction f' , dans le sens où elle fait intervenir certains paramètres de l’AG, permet de mesurer un comportement un peu plus “dynamique” de la fonction de fitness que des mesures de type épistasie, ou corrélation fitness-distance par exemple. Une approche judicieuse serait donc par exemple d’exploiter conjointement différents indices de performance. Nous pensons en outre que la fonction f' peut être analysée dans le cadre d’un modèle Markovien tel celui développé en [63].

Le problème inverse pour les IFS : ce n’est pas terminé

L’emploi des AG et de la PG pour trouver des solutions au problème inverse pour les IFS ouvre la voie à un certain nombre d’études. On peut imaginer trois grandes orientations pour nos travaux futurs sur les IFS :

- Les structures géométriques fractale (du fait de leurs irrégularités liées par certaines homogénéités d’échelles) ont des propriétés physiques intéressantes, voir par exemple [81]. La résolution du problème inverse pour les IFS simplifie la représentation de telles formes. Une application en collaboration avec le CMAPX a été initialisée concernant le calcul de structures ayant certaines propriétés physiques données (amortissement, résistance).
- La résolution du problème inverse pour les IFS peut aussi avoir une influence dans le domaine de la compression fractale de données (signaux ou images). En effet les techniques actuelles de compression fractales sont fondées sur la résolution aux moindres carrés d’une simplification du problème inverse pour les IFS : l’emploi d’AG ou de PG dans un tel cadre permettrait d’utiliser des transformations plus complexes pour le codage des données. Les travaux que nous avons menés au projet FRACTALES [78, 17] sur ce sujet seront poursuivis dans ce sens.
- Enfin, nous envisageons d’aborder le problème inverse pour les IFS par AG et PG par le biais d’approches de type “distribuée” (nous avons adopté le terme “individuelle”) : dans l’approche actuelle un IFS est représenté par un individu de la population ; dans le cas “distribuée” la population entière représente un IFS, un individu étant une fonction de cet IFS (un peu comme dans les approches des systèmes de classeurs pour des application en IA). Cette approche a l’avantage de permettre une exploration de l’espace de recherche plus “économique” en temps de calcul (les tailles de population sont de plus nettement plus petites), au dépens bien sûr d’un calcul plus complexe des fonctions de fitness. Des premières expérimentations nous ont permis de nous rendre compte que cette stratégie permettait un convergence très rapide vers une approximation grossière de la forme cible, mais que la convergence fine était une tâche beaucoup plus complexe (et qui dépend évidemment énormément de la fonction de fitness choisie).

Part II

Genetic Algorithms and Fractals

Chapter 1

Introduction

Genetic Algorithms (GA) and more generally evolutionary algorithms (EA) are currently known as efficient stochastic optimization tools, and are widely used in various application domains. These techniques are based on the evolution of a population of solutions to the problem, the evolution being driven by a “fitness” function that is maximized during the process. Successive populations of solutions are thus built, that fit increasingly well (the values of the fitness function increase). Their evolution is based on stochastic operators : selection (the fitness function is used as a sort of “bias” for a random selection in the population), and the “genetic” operators, mainly crossover (combination of two solutions) and mutation (perturbation of a solution). This technique is based on the assumption that well fitted solutions (also called individuals) can provide better solutions with help of the genetic operators. This assumption can be proven to be connected to some notions of “GA-difficulty” of the function to be optimized : one usually talks about “deception” or sometimes also about “fitness landscape”.

Theoretical investigations on GA and EA generally concern convergence analysis (and convergence speed analysis on a locally convex optimum for Evolution Strategies), influence of the parameters, GA-easy or GA-difficulty analysis. For GA, our main concern here, these analyses are based on different approaches :

- proof of convergence based on Markov chain modeling: for example, Davis [19] has shown that a very low decrease in the mutation probability p_m throughout the generations ensures the theoretical convergence onto a limiting distribution, more recent approaches such as [13, 1, 65] present more precise convergence results.

- deceptive functions analysis, based on Schema analysis and Holland’s original theory [33, 26, 27, 29], which characterizes the efficiency of a GA, and sheds light on “GA-difficult” functions.

Deception has been intuitively related to the biological notion of epistasis [18], which can be understood as a sort of “non-linearity” degree. It can also be related, as we will see later, to the so called “fitness landscape” analyses (see for example [62]). In any case, deception basically depends on:

- the parameter setting of the GA,
 - the shape of the function to be optimized,
 - the coding of the solutions, i.e. the ”way” the search space is scanned.
- some rather new approaches are based on an explicit modeling of a GA as a dynamic system [38, 46, 82].

It has to be noted first that in the modeling of GA as dynamic systems, some fractal features have been exhibited [38]. This approach has mainly led to the generation of fractal images, and may be considered as anecdotal. The aim of this report is to offer evidence that there are stronger justifications, both theoretical and applied, for considering Genetic Algorithms and Fractals together.

Fractals are largely known as a way to generate “nice” images (Julia sets, Mandelbrot sets, Von Koch curves, Sierpinski gasket), that present the characteristic of having an infinity of details and that show a sort of “self-similarity”. We are dealing here with other aspects of fractals, that is the use of fractal tools in order to perform analyses of complex signals. The use of fractal or multifractal theoretical tools in order to perform analyses of signals that are not necessarily “fractal” is now an important trend in this area, and has been proven successful in various domains of applications, such as image analysis, finance, or network traffic analysis, see [81].

In this framework, evolutionary algorithms have been proven to be efficient tools for several applications :

- resolution of the inverse problem for IFS [84, 83, 60, 56, 77, 49] with some applications to speech signal modeling [78],
- image compression [79, 25],
- fractal antennas optimization [15].

This success is mainly due to the fact that when we are dealing with fractal analysis of signals, we often encounter complex optimization problems, with very irregular functions (it is difficult to define local derivatives on such functions), having many local optima, and on large search spaces. Stochastic optimization methods and Evolutionary algorithms in particular are well suited to this type of optimization problems.

From the theoretical viewpoint some tools, developed in the framework of fractal theory, can be used in order to perform a finer analysis of Genetic Algorithms behaviour (mainly based on the schema theory). As we will see in the following, an analysis of how GA optimize some “fractal” functions (see chapter 2) makes it possible to model the influence of some parameters of the GA. Such an analysis can then be generalized and gives some clues about how to tune some of the GA parameters in order to improve its efficiency. Finally, a further analysis on the same theoretical basis allows the influence of the coding in a GA to be analyzed.

This theoretical analysis is presented in chapter 2. In Chapter 3 applications that have been developed in the FRACTALES team are described. We present in section 3.5 our GA software, that is available on <http://www-rocq.inria.fr/fractales/>.

Chapter 2

Theoretical Analysis

The analysis developed in this chapter is based on schema theory and deception analysis. The GA modeled in this framework is the so-called canonical GA, i.e. with proportionate selection (roulette wheel selection), one point crossover and mutation, at fixed rates p_c and p_m throughout the GA run.

If we suppose that the fitness function f , defined on $\{0, 1\}^l$ is the sampling with precision $\epsilon = \frac{1}{2^l}$ of a Hölder function F defined on the interval $[0, 1]$ (this hypothesis is always valid, even if the function F does not reflect in a simple way the behaviour of f):

$$\forall x \in \{0, 1\}^l, \quad f(x) = F\left(\frac{I(x)}{2^l}\right)$$

$I(x) \in [0, 2^l - 1]$ is the integer whose binary expansion is x , $I(x) = \sum_{t=0}^{l-1} x_t 2^t$.

A first result relates l , associated to the sampling precision of the underlying Hölder function F , to the precision of the maximum location that can be obtained, see section 2.2. This result is valid for any optimization method, including Genetic Algorithms.

A deception analysis on Hölder function has then been derived (see section 2.3), that provides a relation between a measure of “deceptiveness” of f , the Hölder exponent of F , and some parameters of the GA (l , p_m , and p_c). This relation suggests that an adjustment of these parameters may tend to improve the performances of a GA. One can also feature an *a posteriori* validation method of the results provided by a GA. Anyway, experiments have proven that even if the behaviour of the GA corresponds to the theoretical relation, one need a finer theoretical analysis in order to be able to build a practical and robust *a posteriori* validation method. The main practical implication of this study is to indicate in which way the parameters must be tuned in order to reduce deceptiveness.

The previous analysis is based on an underlying distance that is the euclidean distance on $[0, 2^l - 1]$. A more “natural” distance on the search space $\{0, 1\}^l$ is the Hamming distance. A similar analysis is described in section 2.4, which provides a more precise relation between deceptiveness and some irregularity measures on the function. This second approach has two main advantages: the irregularity measures (a set of l coefficients, called bitwise regularity coefficients) are easier to estimate than the two coefficients of an underlying Hölder function, and the deceptiveness bound is more precise. A direct practical application is then the estimation of the influence of the bits order in the chromosome coding. Of course this estimation has some limits (mainly due to the theoretical limitations of the static deception model) and does not reflect the behaviour of the simple GA on “very epistatic” functions.

Another important point concerning practical implications of such models is the simple GA model, that is usually not used on real world applications. This is one reason why we have extended the previous model to a GA with uniform crossover in section 2.5. The same relations as for one point crossover can be established in both cases: underlying Hölder functions and general case. Experiments

are then reported on the deceptiveness and convergence behaviour of a GA with and without Gray encoding of the chromosomes.

The theoretical analysis presented in this chapter has of course two main limitations: first it models only the simplest GA, that are not used in real world applications, second the deception analysis has some known weaknesses that do not allow the full complexity of a GA behaviour to be modelled. This analysis however provides important qualitative and, to a certain extent, quantitative information about how a GA converges on a given fitness function. The main result concerns the influence of chromosome length, mutation and crossover probabilities, and provides some tools to evaluate the efficiency of a chromosome encoding.

Further work on this topic is outlined in section 2.6, based on more sophisticated GA models.

2.1 Hölder functions

Definition 1 (Hölder function of exponent h)

Let (X, d_X) and (Y, d_Y) be two metric spaces. A function $F : X \rightarrow Y$ is called a Hölder function of exponent $h > 0$, if for each $x, y \in X$ such that $d_X(x, y) < 1$, we have:

$$d_Y(F(x), F(y)) \leq k \cdot d_X(x, y)^h \quad (x, y \in X) \quad (2.1)$$

for some constant $k > 0$.

Although a Hölder function is always continuous, it need not be differentiable (see the example of Weierstrass functions below).

Intuitively a Hölder function with a low value of h looks much more irregular than a Hölder function with a high value of h (in fact, this statement only makes sense if we consider the highest value of h for which (2.1) holds).

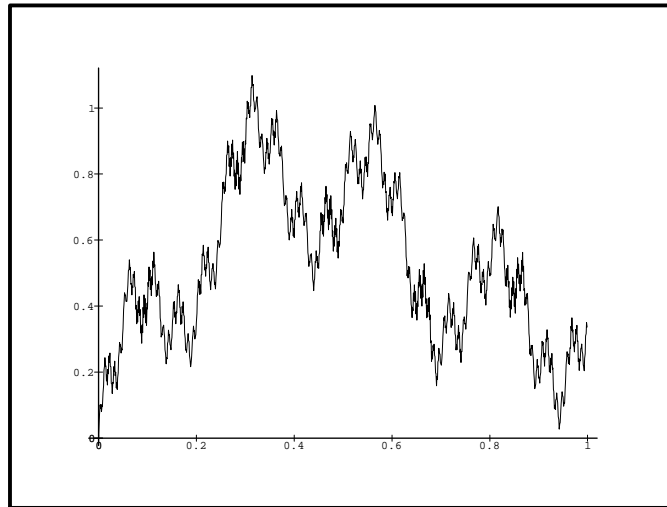


Figure 2.1: Weierstrass function of dimension 1.5.

The frame of Hölder functions, while imposing a condition that will prove useful for tuning the parameters of the GA, makes it possible to consider very irregular functions, such as the Weierstrass

function displayed in figure 2.1 and defined by :

$$W_{b,s}(x) = \sum_{i=1}^{\infty} b^{i(s-2)} \sin(b^i x) \quad \text{with } b > 1 \text{ and } 1 < s < 2 \quad (2.2)$$

This function is nowhere differentiable, possesses infinitely many local optima, and may be shown to satisfy a Hölder condition with $h = s$ [23]. For such “monofractal” functions (i.e. functions having the same irregularity at each point), it is often convenient to talk in terms of box dimension d (sometimes referred to as “fractal” dimension), which, in this simple case, is $2 - h$.

Hölder functions appear naturally in some practical situations where no smoothness can be assumed and/or where a fractal behaviour arises (for example, to solve the inverse problem for IFS [60], in constrained material optimization [74], or in image analysis tasks [51, 4]). It is thus important to obtain even very preliminary clues that allow the parameters of a stochastic optimization algorithm like GA to be tuned, in order to perform an efficient optimization on such functions.

2.2 Sampling rate influence : a general result

We first have derived a relation between the Hölder exponent of the function, the sampling rate, and the accuracy of the optimum localization, both in the domain and the range of the function [48]. This relation holds for any optimization method which works on sampled search spaces.

We first address the problem of the tuning of the resolution (or sampling precision) in the general case, i.e. without assumptions on the discrete optimization method used. This is indeed a crucial problem since if the sampling precision is inadequate, any optimization technique (even exhaustive search) may grossly fail to estimate the right position of the global optimum (see figure 2.2).

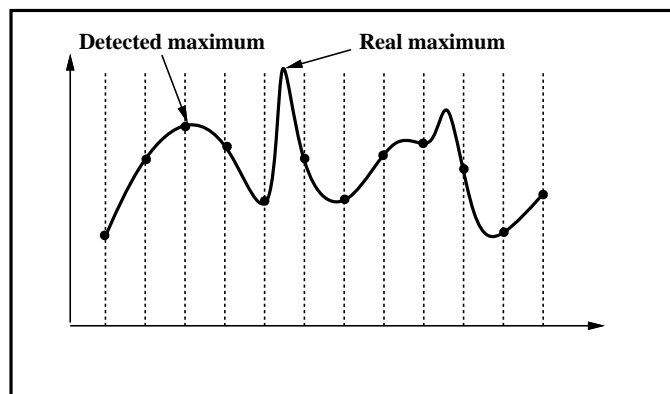


Figure 2.2: An inadequate sampling precision may mislead the optimization process.

In the case of a Hölder function, a very simple remark allows one to verify *a posteriori* that the chosen resolution ϵ has led to a correct estimate. The hypotheses are the following ones :

- i) the function $F : \mathbf{R} \rightarrow \mathbf{R}$ is Hölder with exponent $h > 0$ and constant k (all results in this section remain true if F goes from \mathbf{R}^n to \mathbf{R} , $n \in \mathbf{N}^*$),
- ii) the discrete optimization method has a sampling precision of $\epsilon < 1$ (for instance, $\epsilon = \frac{1}{2^l}$ for a GA where l is the chromosome length). More precisely, the underlying continuous search space is sampled at regularly spaced points (x_n) , with $|x_i - x_{i+1}| = \epsilon$ for all i ,
- iii) the discrete optimization method always gives the right answer on the discrete data: if x_m is found by the algorithm, then it is true that: $\forall i, F(x_m) \geq F(x_i)$ (when we are looking for a maximum).

This last hypothesis implies that the method is also able to locate the “true” second maximum in the discrete space, i.e. the point x'_m such that: $\forall i, i \neq m \Rightarrow F(x'_m) \geq F(x_i)$
 (For a GA, it is possible to locate x_m and x'_m using a sharing technique [30]).

Proposition 1 *Under conditions i), ii) and iii) above, we have :*

$$F(x_m) - F(x'_m) \geq k\epsilon^h \Rightarrow \begin{cases} x^* \in]x_{m-1}, x_{m+1}[\\ F(x^*) \in]F(x_m) - k\epsilon^h, F(x_m) + k\epsilon^h[\end{cases} \quad (2.3)$$

where x^* is the position of the maximum in the continuous space.

This relation quantifies the intuitive guess that if h is low (i.e. if the function is very irregular), then $F(x_m)$ and $F(x'_m)$ should clearly differ in order to yield reliable information. Otherwise, because F has large oscillations, the absolute maximum of F could be in the neighborhood of x'_m instead of in that of x_m .

Numerical simulations presented in [54] prove that this condition can be easily fulfilled, and sketch an *a posteriori* validation procedure of the results accuracy.

2.3 GA-Deception analysis of Hölder functions

This analysis is based on Goldberg’s deception analysis [26, 27], which uses a decomposition of the function to be optimized, f , on Walsh polynomials. This decomposition allows the definition of a new function f' , which reflects the behaviour of the GA, and which represents the expected fitness value that can be reached from the point x :

$$f'(x) = E(f(x'))$$

where x' is a random variable that represents the individuals that can be reached by mutation and crossover from x .

The GA is said to be deceptive when the global maxima of f and f' do not correspond to the same points of the search space.

2.3.1 Schema theory

A schema represents an affine variety of the search space: for example the schema $01\star\star 11\star 0$ is a sub-space of the space of codes of 8 bits length (\star represents a “wild-card”, which can be 0 or 1).

The GA modelled in schema theory is a canonical GA which acts on binary strings, and for which the creation of a new generation is based on three operators:

- a proportionate *selection*, where the fitness function steps in: the probability that a solution of the current population is selected is proportional to its fitness,
- the *genetic operators*: one point crossover and bit-flip mutation, randomly applied, with probabilities p_c and p_m .

Schemata represent global information about the fitness function, but it has to be understood that schemata are just tools which help to understand the codes structure: A GA works on a population of N codes, and implicitly uses information on a certain number of schemata.

We recall below the so called “schema theorem” which is based on the observation that the evaluation of a single code makes it possible to deduce some (partial) knowledge about the schemata to which that code belongs.

Theorem 1 (Schema theorem) (Holland)

For a given schema H , let :

- $m(H, t)$ be the relative frequency of the schema H in the population of the t^{th} generation,
- $f(H)$ be the mean fitness of the elements of H ,
- $\mathcal{O}(H)$ be the number of fixed bits in the schema H , called the order of the schema,
- $\delta(H)$ be the distance between the first and the last fixed bit of the schema, called the definition length of the schema.
- p_c be the crossover probability,
- p_m be the mutation probability of a gene of the code,
- \bar{f} be the mean fitness of the current population.

Then :

$$E[m(H, t + 1)] \geq m(H, t) \frac{f(H)}{\bar{f}} \left[1 - p_c \frac{\delta(H)}{l-1} - \mathcal{O}(H)p_m \right]$$

From a qualitative view point, this formula means that the “good” schemata, having a short definition length and a low order, tend to grow very rapidly in the population. These particular schemata are called *building blocks*.

The usefulness of the schema theory is twofold: first, it supplies some tools to check whether a given representation is well-suited to a GA. Second, the analysis of the nature of the “good” schemata, using for instance Walsh functions [28, 34], can give some ideas on the GA efficiency [18], via the notion of deception that we describe below.

2.3.2 Walsh polynomials and Deception characterization

Goldberg has suggested using a method based on a decomposition of f on the orthogonal basis of Walsh functions on $[0..2^l - 1]$, where $[0..2^l - 1]$ denotes the set of integers of the interval $[0, 2^l - 1]$.

On the search space $[0..2^l - 1]$, we can define 2^l Walsh polynomials as:

$$\Psi_j(x) = \prod_{t=0}^{l-1} (-1)^{x_t j_t} = (-1)^{\sum_{t=0}^{l-1} x_t j_t} \quad \forall x, j \in [0..2^l - 1]$$

x_t and j_t are the values of the t^{th} bit of the binary decomposition of x and j .

They form an orthogonal basis of the set of functions defined on $[0..2^l - 1]$, and we let $f(x) = \sum_{j=0}^{2^l-1} w_j \Psi_j(x)$ be the decomposition of the function f .

The function f' [26, 27] can thus be written as follows:

$$f'(x) = \sum_{j=0}^{2^l-1} w'_j \Psi_j(x) \quad \text{with} \quad w'_j = w_j \left(1 - p_c \frac{\delta(j)}{l-1} - 2p_m \mathcal{O}(j)\right) \quad (2.4)$$

The quantities δ and \mathcal{O} are defined for every j in a similar way as for the schemata: $\delta(j)$ is the distance between the first and the last non-zero bits of the binary decomposition of j , and $\mathcal{O}(j)$ is the number of non-zero bits of j .

2.3.3 Haar polynomials for the deception analysis of Hölder functions

If we consider the fitness function f as the sampling of some Hölder function defined on the interval $[0, 1]$, it is intuitively obvious that the more irregular the function is (i.e. the lower the Hölder exponent is), the more deceptive it is likely to be. The intuition can be reinforced by theoretical arguments, as we will see in the following.

Another decomposition than the previous one is then more suited to a deception analysis of Hölder functions. There exist simple bases which permit, in a certain sense, the irregularity of a function to be characterized in terms of its decomposition coefficients. Wavelet bases possess such a property. The simplest wavelets, i.e. Haar wavelets, are defined on the discrete space $[0..2^l - 1]$ as:

$$H_0(x) = 1 \quad \text{for all } x \text{ in } [0..2^l - 1]$$

$$H_{2^q+m}(x) = \begin{cases} 1 & \text{for } (2m)2^{l-q-1} \leq x < (2m+1)2^{l-q-1} \\ -1 & \text{for } (2m+1)2^{l-q-1} \leq x < (2m+2)2^{l-q-1} \\ 0 & \text{otherwise in } [0..2^l - 1] \end{cases}$$

with $q = 0, 1, \dots, l-1$ and $m = 0, 1, \dots, 2^q - 1$: q is the degree of the Haar function.

These functions form an orthogonal basis of the set of functions defined on $[0..2^l - 1]$. Any function f of $[0..2^l - 1]$ can be decomposed as:

$$f(x) = \sum_{j=0}^{2^l-1} h_j H_j(x) \quad h_j = \frac{1}{2^{l-q}} \sum_{x=0}^{2^l-1} f(x) H_j(x)$$

Haar coefficients can be bounded

The following result can be easily proven (see [54]):

$$\forall j, \quad |h_j| \leq \frac{k}{2} 2^{-h(q+1)}$$

This inequality is shown in figure 2.3. The following remark is relevant for practical implementation: the optimal value of k (i.e. the lowest one) depends on the sampling precision. The curves of figure 2.3 are drawn with $k = 2.5$ for a Weierstrass function sampled on 12 bits, and with $k = 3$ for an FBM¹ sampled on 10 bits.

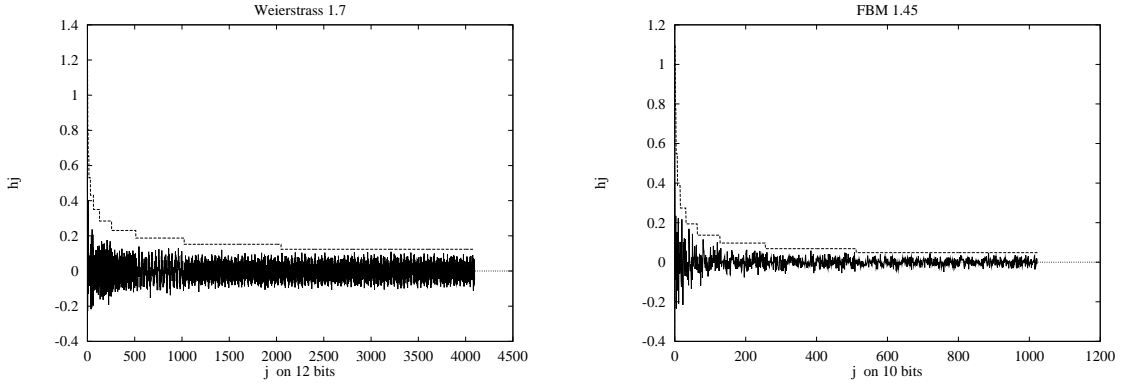


Figure 2.3: Haar coefficients (continuous) and bound (dotted) for a Weierstrass function of dimension 1.7 sampled on 12 bits (left) and an FBM of dimension 1.45 sampled on 10 bits (right).

Deception for Hölder functions

An upper bound for the quantity $|f(x) - f'(x)|$ can be computed (see [54]):

Theorem 2 *Let f be the sampling on l bits of a Hölder function of exponent h and constant k , defined on $[0, 1]$, and let f' be defined as in (2.4). Then :*

$$\forall x \in [0..2^l - 1], \quad |f'(x) - f(x)| \leq k * B(p_m, p_c, l, h) \quad (2.5)$$

with

$$B(p_m, p_c, l, h) = \frac{p_c}{l-1} 2^{-h} \left[\frac{2^{-l(h+1)} - 1}{2^{-(h+1)} - 1} + \frac{(1 - 2^{1-h})(2^{-hl} - 1) - l2^{-hl}(1 - 2^{-h})}{(2^{-h} - 1)^2} \right] + p_m \frac{2^{-h}}{(2^{-h} - 1)^2} [1 + 2^{-hl}(l2^{-h} - l - 1)]$$

¹FBM stands for Fractional Brownian Motion. For definition and properties of the Fractional Brownian Motion (FBM) see for instance [55]. As Weierstrass functions, paths of FBM (almost certainly) verify a Hölder property, the irregularity being the same at each point. Thus an FBM with Hölder exponent h has box dimension equal to $2 - h$.

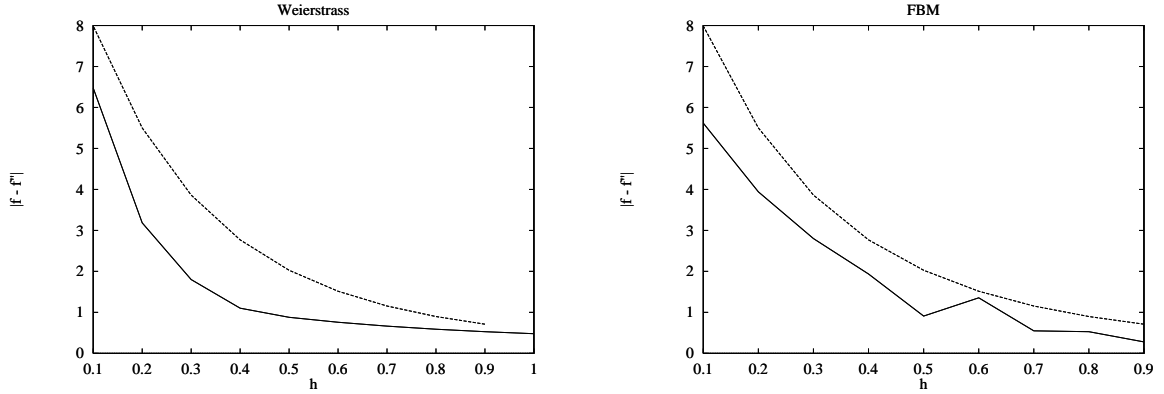


Figure 2.4: $B(p_m, p_c, l, h)$ (dotted) and computed maximum differences between f and f' (continuous) as a function of h , for Weierstrass functions (left), and FBM's (right), $l = 8$ bits, $p_c = 0.9$, $p_m = 0.25$.

Since for all admissible values of l, p_m, p_c , B is an increasing function of h , this relation implies that the smaller h is (i.e. the more irregular the function is), the greater the difference between functions f and f' may be, thus the more deceptive f is likely to be. This first fact bears some analogy with the results stated in section 2.2, and is confirmed by numerical simulations displayed in figure 2.4.

A fine analysis of the function $B(p_m, p_c, l, h)$ is rather difficult, because B defines a hyper-surface of \mathbf{R}^5 , but the following results may be stated (see [54]).

- $B(p_m, p_c, l, h)$ increases with l for small values of l , reaches a maximum l_{max} and then decreases for larger values of l . It has the following asymptotic behaviour when $l \rightarrow \infty$:

$$\lim_{l \rightarrow \infty} B(p_m, p_c, l, h) = p_m \frac{2^{-h}}{(2^{-h} - 1)^2}$$

- $B(p_m, p_c, l, h)$ decreases with respect to p_c and p_m , and deception is less influenced by p_c than by p_m .

2.3.4 Non-deception criterion ?

A sufficient condition for non-deception is $B(p_m, p_c, l, h) = 0$, which in general is not possible. A qualitative approach is then to keep B as small as possible, in order to keep f and f' as close as possible to each other. Formula (2.5) provides a relation involving some of the GA parameters, which may help to set them in order to make the convergence of the GA easier. Notice however that this relation only gives a bound, which may not be optimal.

A simple sufficient non-deception condition in the same spirit as in section 2.2 may be sketched: Formula (2.5) points out that the function f' is located inside a strip of extent $2kB$ around f . If we suppose that we have detected the two first optima of the functions (using for example a GA with sharing), i.e. x_1^* and x_2^* corresponding to the values f_1^* and f_2^* (see [54]), then if the following relation holds:

$$|f_1^* - f_2^*| > 2 * k * B(p_m, p_c, l, h)$$

we are ensured that the maximum of the function f' will be near to the one of f (i.e. near f_1^*). This situation is depicted in figure 2.5. In this case, the function is not deceptive. If this relation does not hold (figure 2.6), we cannot say anything about the deception of the function.

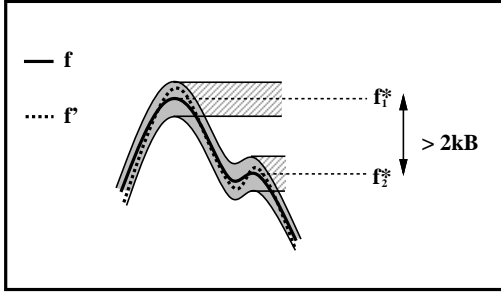


Figure 2.5: Favorable case: no deception.

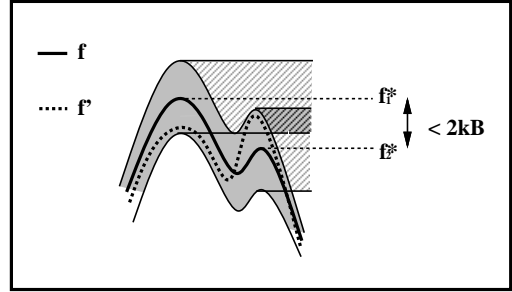


Figure 2.6: Unfavorable case: possible deception.

The extent of the strip around f may be tuned by changing the values of the parameters p_c , p_m and l , with the constraint that l must be larger than a fixed threshold, which ensures a correct sampling: the relation $|f_1^* - f_2^*| > k \frac{1}{2^{lh}}$ is used to find a lower bound for l , while the relation $|f_1^* - f_2^*| > 2 * k * B(p_m, p_c, l, h)$ provides conditions on p_m , p_c and l .

To summarize, an *a posteriori* validation test may be written as :

$$|f_1^* - f_2^*| > k * \max(2 * B(p_m, p_c, l, h), \frac{1}{2^{lh}})$$

It must however be pointed out that while the accuracy test seems to give precise indications, the deception test has to be used more carefully, only as a “confidence” measure on the results of the GA. This is confirmed by numerical simulations presented in [54].

2.4 Bitwise regularity analysis

The previous analysis is based on an irregularity characterization with respect to an underlying distance that is the Euclidian distance on $[0, 1]$. This approach is straightforward for fitness functions defined on \mathbb{R} , and in the general case it is always possible to consider the fitness function as the sampling of an underlying one-dimensional Hölder function. It is however less obvious in this latter case that the Hölder exponent reflects the irregularity of the fitness function in a simple way (it may appear for example more irregular than it is in a multidimensional space). This is why we have performed a similar irregularity analysis but with respect to the Hamming distance on the set of binary strings. Another justification is that the use of Hamming distance is more closely related to the action of genetic operators.

2.4.1 Bitwise regularity coefficients.

Due to the above considerations, we introduce the following coefficients, that are derived from Hölder grained exponents with respect to a distance proportional to the Hamming distance (see [40] and appendix B for a detailed justification):

Definition 2 (Bitwise regularity coefficients) *Let f be a function defined on $\{0, 1\}^l$:*

$$\forall q \in \{0, \dots, l-1\}, C_q = \sup_{x \in \{0,1\}^l} \{|f(x) - f(x'_{l-q-1})|\}$$

*with x'_{l-q-1} and x differing only with respect to one bit at the position $(l-q-1)$.*²

In other terms, the C_q coefficient represents the maximum fitness variation due to a bit flip at the position $(l-q-1)$. Therefore, we can show that:

$$\forall j = 2^q + m, \quad |h_j| \leq \frac{C_q}{2}$$

In the same way as in [54], with the help of the Haar basis, the following theorem has been established (see [40] and appendix B for a proof):

Theorem 3 *Let f be a function defined on $\{0, 1\}^l$ with bitwise regularity coefficients $(C_q)_{q \in \{0, \dots, l-1\}}$, and let f' be defined as in (2.4). Then $\forall x \in \{0, 1\}^l$:*

$$\begin{aligned} |f(x) - f'(x)| \leq & \frac{p_c}{l-1} * \sum_{q=0}^{l-1} C_q * \left(\frac{1 + 2^q(q-1)}{2^q} \right) \\ & + p_m * \sum_{q=0}^{l-1} C_q * (q+1) \end{aligned}$$

Furthermore, this result still holds when the order of the C_q is reversed, so the final bound is the one minimizing the preceding expression.

We also have to note that the bits do not have the same role in this bound expression. In fact their relative weight strictly increases with respect to the index q . Sorting (either in increasing or decreasing order) would then minimize this bound suggesting that the simple change of coding consisting of a permutation on the bits would make the function easier. This feature can be explained by the fact that the one point crossover more easily disrupts a combination of a few genes spread at each extremity of the chromosome than if these genes were grouped at one extremity. Reordering the bits in order to

²The less significant bit being at position 0.

sort the bitwise regularity coefficients is then equivalent to grouping the most “sensitive” genes at one extremity of the chromosome. Some experiments presented in [40] and appendix B partially support this hypothesis, but also reveal that other phenomena (such as epistasis [18]) have to be taken into account in order to predict the sensitivity of GA to such encoding changes.

2.4.2 Bitwise regularity coefficients compared to Hölder exponent.

If we suppose that the fitness function f is the sampling on l bits of a Hölder function of exponent h and constant k , defined on $[0, 1]$, the bound of *theorem 3* is lower than the bound of *theorem 2*.

One can easily show, (see [40] and appendix B), that :

$$C_q \leq k * 2^{-(q+1)h} \quad (2.6)$$

as we have :

$$|h_j| \leq \frac{C_q}{2} \quad \text{and} \quad |h_j| \leq \frac{k}{2} * 2^{-(q+1)h}$$

and as the bound on $|f - f'|$ is a linear function of the bounds on the $|h_j|$, it immediately follows that the bound of *theorem 3* is the lowest. Moreover, the estimation of the bitwise regularity coefficients is computationally cheaper than the estimation of the Hölder exponent and its associated constant k .

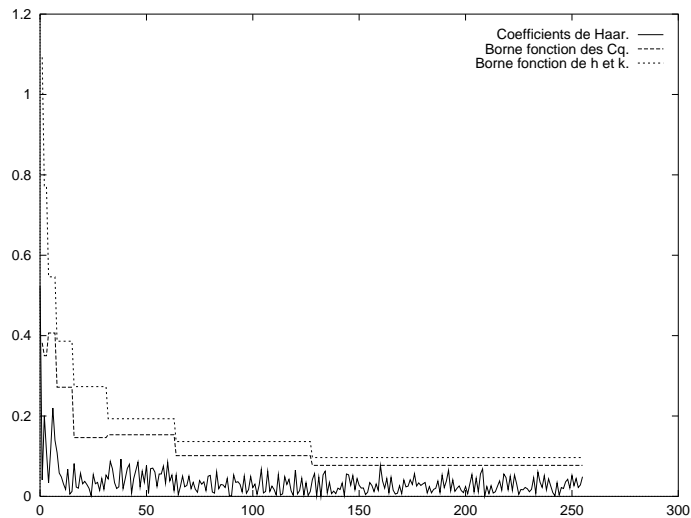


Figure 2.7: Haar coefficients, majoration with h and k , and majoration with C_q , for a Weierstrass function of dimension 1.5 ($h = 0.5$)

2.5 Deception analysis of a GA with uniform crossover

As we have seen, the bound on $|f - f'|$ derived from the bitwise regularity coefficients C_q depends on their relative order, due to the use of one point crossover. The aim of this section is to present analogous results that have been established for the uniform crossover ([73], for which the positional bias no longer exists).

The only change is to replace the schema disruption probability p_d for this version of crossover :

$$p_d \leq \left(1 - \left(\frac{1}{2} \right)^{O(H)-1} \right) \quad (2.7)$$

This upper bound is obtained by observing that once the first fixed bit of the schema is allocated to one of the offsprings, it will always survive if all other fixed bits are allocated to the same offspring.

Theorem 4 (Schema theorem with uniform crossover)

For a given schema H , let :

- $m(H, t)$ be the relative frequency of the schema H in the population of the t^{th} generation,
- $f(H)$ be the mean fitness of the elements of H ,
- $O(H)$ be the number of fixed bits in the schema H , called the order of the schema,
- p_c be the crossover probability,
- p_m be the mutation probability of a gene of the code,
- \bar{f} be the mean fitness of the current population.

Then :

$$m(H, t+1) \geq m(H, t) \frac{f(H)}{\bar{f}} \left[1 - p_c \left(1 - \left(\frac{1}{2} \right)^{O(j)-1} \right) - O(H)p_m \right]$$

Then the new adjusted Walsh coefficients are:

$$w'_j = w_j \left[1 - p_c \left(1 - \left(\frac{1}{2} \right)^{O(j)-1} \right) - 2p_m O(j) \right]$$

Notice that $O(j)$ no longer depends on the defining length of the schema. Furthermore as the order of a schema is invariant with respect to a permutation on the bits, the following theorem has been proven (see [40] and appendix B for a demonstration) :

Theorem 5 Let f be a function defined on $\{0, 1\}^l$ with bitwise regularity coefficients $(C_q)_{q \in \{0, \dots, l-1\}}$, and let f' be defined as in (2.4). Then for all permutation σ defined on the set $\{0, \dots, l-1\}$, $\forall x \in \{0, 1\}^l$:

$$|f(x) - f'(x)| \leq p_c * \sum_{q=0}^{l-1} C_{\sigma^{-1}(q)} + p_m * \sum_{q=0}^{l-1} C_{\sigma^{-1}(q)} * (q+1) \quad (2.8)$$

We immediately see that this upper bound is minimal when the $C_{\sigma^{-1}(q)}$ are ordered in decreasing order.

Practically, if it is possible to get the C_q values (or good estimations), it is hard to draw conclusions from the value of the bound (2.8). But if we consider the effect of an encoding change on it, it is interesting to see if its variation is experimentally correlated to the performances of the GA. Intuitively, the hypothesis is formulated as follows : if an encoding change (such as Gray code) induces a decrease of the bound (2.8), the GA should perform better with this new encoding, and conversely. We present experiments with the Gray code in [40] and appendix B.

2.6 Conclusions and Further work

Besides the intuitive fact that it relates the irregularity of the fitness function to its “difficulty”, and provides information about how to modify some of the GA parameters in order to improve its performances, one important application of this theoretical analysis is that it provides a means of measuring (of course to a certain extent, due to the intrinsic limitations of deception theory) the influence of the chromosome encoding. We present in [40] and appendix B some experimentations with the Gray encoding that prove the interest of such an approach: these tests show that the bound derived from the bitwise regularity coefficients are relatively reliable as long as the variations in the bound between two different encodings is large enough.

It has to be noted finally that Schema theory has some known limitations, and thus that the practical implications of the previous analysis must be considered with care. See [40]³ and appendix B for a more detailed criticism. In spite of these limitations, Schema theory has the great advantage of providing a simple model of the GA behaviour, and allows to make some computations, that are much more complicated or even infeasible for other models (other theoretical analyses of GA provide different models, that are also simplified in a different way). Conclusions built from this model, as it is a rough simplification of the behaviour of a real GA, must thus be mainly considered as qualitative analyses.

Further work can concern the analysis of Hölder exponents and bitwise regularity coefficients in the framework of dynamical systems [3] and Markov-based modeling [12]. A critical analysis of epistasis measures can be found in [63], that leads to the conclusion that epistasis is not a reliable performance indicator, but can be considered as a performance measure of the crossover operator for the choice of different encodings. This result is similar to those presented before. Moreover, a look at the expression of the epistasis in the Walsh basis compared to that of f' shows the resemblance and differences between the two measures: the function f' takes into account the influence of the genetic operators while epistasis represents an intrinsic non-linearity measure of the function. These two quantities computed on the fitness function represent two different viewpoints of the fitness difficulty. A first idea would be to analyse, in the same way as for the epistasis, a correlation between f and f' . This may provide a more reliable indicator that the estimation of the bound $|f - f'|$. A second idea would be to use a combination of these two measures in order to better capture the notion of GA-difficulty.

³An interesting viewpoint on schema theory can also be found in [63] which proves results similar to the schema theorem with the help of a Markov chain model. This work clearly presents the limitation of schema theory model.

Chapter 3

Applications

Most of the applications described in this chapter are related to the inverse problems that arise in signal analysis applications. A standard inverse problem can be stated as follows: from a given set of data, it is possible to compute the output of a system, but from a given output (the “target”) it is impossible to calculate the input data set.

A classical strategy, a “black box” technique, is to turn this inverse problem into an optimization problem: search for the input data set such that the output of the system resemble the target. Usually, evolutionary algorithms are adapted to the resolution of difficult inverse problems for which there is little *a priori* information (the functions to be optimized are not explicitly known, and *a fortiori* their derivatives). In the framework of fractal data analysis, some difficult inverse problems have been successfully solved using EA, for example:

- inverse problems for IFS [84, 83, 60, 56]. We present in this chapter some experiments with GA for affine IFS [77], and with GP in the general case of mixed IFS, that is more complex [49]. An application to speech signals modeling has also been developed [78].
- inverse problems for finite automata [41].

The resolution of these “academic” inverse problems has led to experimenting GA on applied problems such as image compression [79, 25], or fractal antennas optimization [15].

The main difficulties of such applications are:

- How to find an adequate coding of the problem, which often makes it possible to efficiently exploit some *a priori* knowledge about the system. In the example of the inverse problem for fixed point finite automata, a coding of the lengths of the words of an automaton yields a much more efficient GA than a direct encoding of the words, see [41] and appendix E.
- How to conveniently handle the constraints of the problem, that also allows some shortcuts in the computation of the fitness function. A good example is the inverse problem for IFS, where contractivity constraints have been exploited in order to prune a lot of extra computations, see [77, 49] and appendix C.

In a general manner, an efficient GA for a difficult problem can be built using as much *a priori* knowledge as possible. This is particularly true when constraints can be identified and handled with the function to be optimized.

Constraints handling is a crucial problem for EA in general and several methods have been proposed for handling nonlinear constraints in a EA [68, 69, 67, 66], they can be grouped into four categories, see [58] for a complete review of these methods:

- methods based on preserving feasibility of solutions, for example using specialized operators which transform valid individuals into other valid ones, or repair operators (projection on the feasible region),
- methods based on penalty functions (weighted sum of the constraints violations included in the fitness function),
- methods which separate feasible and infeasible solutions,
- hybrid methods.

For the inverse problems presented hereafter, all the constraints of the problems were integrated as penalty functions with variable weights within the fitness function (this was the simplest way to properly handle the constraints for feasible spaces that can be very sparse). These weights were designed in order to favour the penalty term at the beginning of the evolution of the GA, in order to first draw the populations near feasible regions, and then to progressively favour the term to be optimized. This structure allows us to easily make computation cut-off when some constraints are not fulfilled. Another justification for using such a strategy is that weighted penalty functions allow “smoother” fitness functions, which clearly tend to make the job easier for the GA : this intuitive fact is also enforced by the conclusions of the theoretical analysis presented above.

The applications presented in the following have been developed in the framework of signal analysis :

- related to fractal theory: inverse problems for IFS, see chapter 3.1, appendices C and D, for finite automata, see chapter 3.3 and appendix E,
- related to physics: X-Ray diffraction spectra analysis, see chapter 3.4 and appendix F
- related to image analysis: geometric primitives detection, see chapter 3.4 and appendix G

These experiments have enforced once more, if needed, the commonly-held opinion known among EA-people, that careful parameter setting, efficient encoding and an “economic” fitness function computation can make a huge difference to the efficiency and accuracy of the GA, and even turn an “infeasible” problem into a “feasible” one.

We finally present in chapter 3.5 the two software products that have been developed in order to support the various applications we have presented here: ALGON for GA and PROGON for GP. These are available on the FRACTALES GROUP WEB pages: <http://www-rocq.inria.fr/fractales/>

3.1 Inverse Problems for IFS

3.1.1 Introduction

Iterated functions system (IFS) theory is an important topic in fractals, introduced by J. Hutchinson [35]. These studies have provided powerful tools for the investigation of fractal sets, and the action of systems of contractive maps to produce fractal sets has been considered by numerous authors (see, for example, [6, 7, 22, 31]).

A major challenge of both theoretical and practical interest is the resolution of the so-called inverse problem [8, 56, 83, 84]. Except for some particular cases, no exact solution is known.

A lot of work has been done in this framework, and some solutions exist based on deterministic or stochastic optimization methods. As the function to be optimized is extremely complex (see figure 3.1), most of them make some *a priori* restrictive hypotheses: use of an affine IFS, with a fixed number of functions [9, 24, 37, 80, 36]. Solutions based on GA or EA have been presented for affine IFS [25, 60, 79, 83].

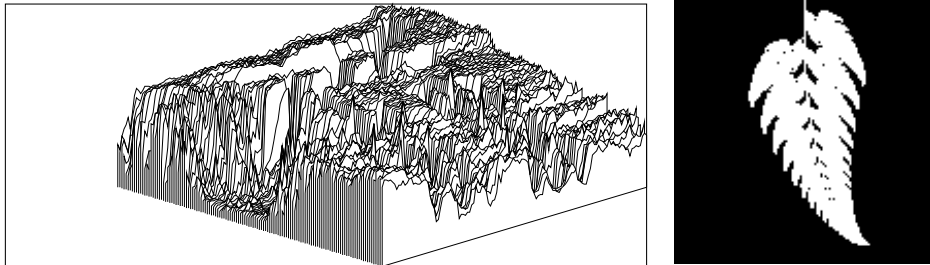


Figure 3.1: Left: A 2D slice of the error function for the Barnsley fern (right), for affine IFS (the dimension of the search space is 24).

We present below a simple implementation of a GA for the resolution of the inverse problem for affine IFS (section 3.1.3). If we let the contractions be non affine (as in numerous applications the IFS are implicitly supposed to be affine, we use the term “mixed” IFS, in order to emphasize the fact that the functions are not constrained to be affine, see section 3.1.4), the inverse problem cannot be addressed using “classical” techniques. A solution based on GP is presented in section 3.1.5. In appendix D an application related to speech signal modeling is presented.

3.1.2 IFS Theory

Attractor

An IFS $\mathcal{U} = \{F, (w_n)_{n=1, \dots, N}\}$ is a collection of N functions defined on a complete metric space (F, d) .

Let W be the Hutchinson operator, defined on H , the set of non-empty compact subsets of F :

$$\forall K \subset F, W(K) = \bigcup_{n \in [0, N]} w_n(K)$$

Then, if the w_n functions are contractive (the IFS is then called a *hyperbolic* IFS), there is a unique set A , the **attractor** of the IFS, such that:

$$W(A) = A$$

Recall: A mapping $w : F \rightarrow F$, from a metric space (F, d) onto itself, is called **contractive** if there exists a positive real number $\lambda < 1$ such that $d(w(x), w(y)) \leq \lambda \cdot d(x, y) \quad \forall x, y \in F$.

The uniqueness of a hyperbolic attractor is a result of the contractive mapping fixed-point theorem for W , which is contractive according to the Hausdorff distance:

$$d_H(A, B) = \max[\max_{x \in A} (\min_{y \in B} d(x, y)), \max_{y \in B} (\min_{x \in A} d(x, y))]$$

From a computational viewpoint, an attractor can be generated according to two techniques:

- **Stochastic method (toss-coin)**

Let x_0 be the fixed point of one of the w_i functions. We build the point sequence x_n as follows:
 $x_{n+1} = w_i(x_n)$, i being randomly chosen in $\{1..N\}$.

Then $\bigcup_n x_n$ is an approximation of the real attractor of \mathcal{U} . The larger n is, the more precise the approximation is.

- **Deterministic method:**

From any kernel S_0 , we build the set sequence $\{S_n\}$,

$$S_{n+1} = W(S_n) = \bigcup_n w_n(S_n)$$

When n is large, S_n is an approximation of the real attractor of \mathcal{U} .

Invariant measure

Let us attach to each w_n a probability p_n , such that :

$$\forall n, p_n > 0 \quad \text{and} \quad \sum_{n=1}^N p_n = 1$$

Let M be the Markovian operator on the set of measures defined on F :

$$M(\nu) = \sum_{i=1}^N p_i \nu \circ w_i^{-1}$$

There exists a unique measure μ , the invariant measure, such that :

$$M\mu = \mu$$

Note that (for hyperbolic IFS) the support of μ is A , independent of the p_n 's, as long as they all are strictly positive.

Furthermore, μ is attractive in the following sense : let ν_0 be any probability measure on F , and define a sequence of probability measures by :

$$\nu_m(B) = \sum_n p_n \nu_{m-1}(w_n^{-1}(B))$$

then $(\nu_m)_m$ converges weakly to μ .

Invariant measures can be generated with the same techniques as for attractors :

- **Stochastic method (toss-coin)**

The point sequence x_n is built in the same way as for attractors: $x_{n+1} = w_i(x_n)$, w_i being randomly chosen in $\{1..N\}$ with probability p_i

Let δ_x be the point measure in $x \in F$. Then $\sum \delta_{x_n}$ approximates μ for large n .

- **Deterministic method :**

From any initial measure, for example δ_{x_0} , we build the measures sequence ν_n :

$$\nu_{n+1} = M(\nu_n) = \sum_{i_1, i_2, \dots, i_n=1}^N p_{i_1} p_{i_2} \dots p_{i_n} \delta_{w_{i_1} \circ w_{i_2} \circ \dots \circ w_{i_n}(x_0)}$$

Then ν_n is an approximation of μ for large n .

Inverse problem

The computation for any hyperbolic IFS of its attractor and its invariant measure is an easy task, but the inverse problem is a complex one :

Given a subset A of H , and a probability measure μ on A , find an IFS and some $(p_n)_n$ whose attractor is A and stationary measure is μ .

This problem has been considered by a number of authors. An exact solution can be found in the particular case where each $w_n(A) \cap w_m(A)$ is of zero measure (see [8]). In the general case, no exact solution is known. Usually, the problem is divided into two parts :

1. finding the IFS that generate A ,
2. finding the (p_n) that generate μ .

An essential tool for solving 1. is the **collage theorem** :

Theorem 6 (Collage theorem) *Let $E \subset F$ be such that :*

$$d_H(E, \bigcup_n w_n(E)) < \epsilon \quad \text{then} \quad d_H(A, E) < \frac{\epsilon}{1 - \lambda}$$

where $\lambda = \sup_n \{\lambda_n\}$ is the largest contractive factor, and A is the attractor of $(F, (w_n))$.

Based on this, several optimization methods have been proposed, see [8, 44, 70, 20, 56].

The resolution of 2. requires the definition of a distance between measures. It has been considered in [70, 35, 6, 84]. An exact solution for a special case, using an extension of the theory presented in [7], is given in [42] : the authors in [7] allow the probabilities to be functions instead of numbers (“place-dependent probabilities”), thus permitting a larger variety of measures to be generated. Conversely, this generalization makes the inverse problem easier to solve in some special cases.

In the following, we will restrict ourselves to the resolution of 1.¹

In [70], the authors propose a GA for solving the 1D inverse problem. Their algorithm proves to be efficient in the examples presented. Note that the 2D problem is much harder than the 1D one,

¹The image compression method using IFS does not solve this inverse problem, but a simplified one, based on the separation of the image into small blocks and the finding of optimal mappings between blocks of different sizes. A simple attempt at using a GA for fractal image compression can be found in [79].

since it involves many more unknowns. Typically, 6 unknowns are present in [70], and for example 24 in the 2D problem for affine IFS with 4 functions.

The 2D problem can be precisely stated as follows :

Given a compact subset G of the plane, find n contractions (w_i) such that the attractor of (w_i) approximates at best G , in the sense of the Hausdorff metric

Note that we only need the attractor to approximate G , since, in general, there might exist no exact solution. The first thing to compute is the number n of contractions. If the fractal dimension of G is D and if $w_i(G) \cap w_j(G)$ is of zero measure for each (i,j) , then we know that :

$$\sum_{i=1}^n \lambda_i^D = 1$$

Thus if we ask that all λ_i be lower than λ , we have a lower bound n^* for n , in the special case of non overlapping $w_i(G)$. n^* will also be a lower bound for the general case. In practice, we may compute the fractal dimension of G , and start our investigations with n^* functions. If we are not satisfied with the result we increase the number of functions.

An unsolved problem is the choice of the basis of functions. Most people use affine transforms for simplicity, but the use of other types of functions, such as sine polynomials (see [44] and figure 3.2) can lead to a much richer class of shapes.

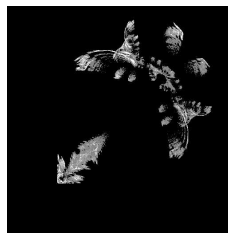


Figure 3.2: The logo of the FRACTALES group : a sine polynomials IFS

3.1.3 The case of affine IFS : use of a GA

In the case of an affine IFS, each contractive map w_i of \mathcal{U} is represented as

$$w_i(x, y) = \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \end{bmatrix}$$

The inverse problem corresponds to the optimization of the values $(a_i, b_i, c_i, d_i, e_i, f_i)$ to get the attractor that most resembles the target.

Figure 3.1 displays a 2D slice of an error function based on the collage theorem, i.e. the Hausdorff distance between the attractor of (w_n) and the attractor of $(w_n + \epsilon_n)$ where ϵ_n is a small perturbation. We can see that it is extremely irregular.

Figures 3.3 and 3.4 present the results obtained with a simple implementation of a GA on two attractors : the well-known fern [9], and a perturbed Sierpinski triangle. See [47] for details of the experiment.



Figure 3.3: (left) Original 128x128 image : fern - (right) Reconstruction

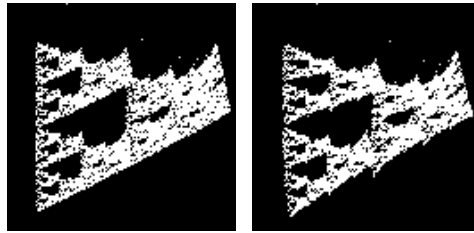


Figure 3.4: (left) Original 128x128 image : perturbed Sierpinski triangle - (right) Reconstruction

Improvements of this technique are based on:

- the use of the sharing method developed in [52], that improves the exploration capability of the GA, and allows the simultaneous detection of several optima (if any),
- the use of a multi-resolution and iterative scheme: a first very rough and then increasingly precise approximation of the fitness function is used in successive runs of the GA (the result of the previous run being used as an initial solution in the next one, at a finer resolution). A reduced number of generations has been necessary in order to obtain an approximation, see figure 3.5,
- a more precise computation of distances between target images and attractors, with the help of distance images (a well-known tool of mathematical morphology [11]), which seems to be more stable than the Hausdorff distance, and less irregular than L^1 or L^2 distances.

An experimental comparison of two fitness functions: the one based on the classical toss coin algorithm, the other using the collage theorem, yields as a first conclusion that the toss coin fitness



Target → successive approximations

Figure 3.5: Successive approximations of a square with help of a fitness function based on the toss coin algorithm (from left to right)

provides more reliable results. This conclusion has been confirmed in the case of the inverse problem for mixed IFS.

$$\begin{aligned}
 w_1(x, y) &= \left(\begin{array}{c} \sqrt{|\sin(\cos 0.90856 - \log(1 + |x|))|} \\ \sin y \end{array} \right) \\
 w_2(x, y) &= \left(\begin{array}{c} \cos(\cos(\sqrt{|x|})) \\ \cos(\log(1 + |y|)) \end{array} \right) \\
 w_3(x, y) &= \left(\begin{array}{c} \log(1 + |\cos(\log(1 + |y + x|))|) \\ \sqrt{|\sin 0.084698|} \end{array} \right) \\
 w_4(x, y) &= \left(\begin{array}{c} \log(1 + |\sin(\sqrt{|0.565372|})|) \\ \sqrt{|0.81366 - ((\log(1 + |0.814259|)) * \cos y)|} \end{array} \right) \\
 w_5(x, y) &= \left(\begin{array}{c} \log(1 + |\sqrt{|0.747399 + \cos y|}|) \\ \sin\left(\frac{0.73624}{0.0001 + |0.264553 * y + 0.581647 + x|}\right) \end{array} \right)
 \end{aligned}$$

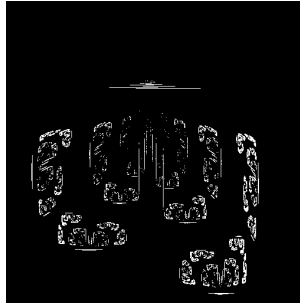


Figure 3.6: A mixed IFS and its attractor.

3.1.4 Mixed IFS

When the w_i are no longer restricted to being affine functions, the corresponding inverse problem cannot be addressed in a simple way, unless some *a priori* hypotheses on the structure of the IFS (number and type of functions) are made. In the following, we have chosen to call these IFS **mixed IFS** in order to emphasize the fact that they are no longer restricted to being affine. When dealing with mixed IFS, the first point to be addressed is to find an adequate representation. A natural one is to code the mixed IFS as trees.

The attractors of figures 3.6 and 3.7 are random mixed IFSs: the w_i functions have been recursively built with the help of random shots in a set of basic functions, a set of terminals (x and y), and a set of constants. In our examples, the constants belong to $[0, 1]$, and the basic functions set is $\{+, -, \times, \text{div}(x, y) = \frac{x}{0.0001 + |y|}, \cos, \sin, \text{root}(x) = \sqrt{|x|}, \text{loga}(x) = \log(1 + |x|)\}$

Each w_i can thus be represented as a tree (see for example figure 3.8). The trees of the w_i are then gathered to build the main tree representing the IFS \cup (figure 3.9). This very simple structure allows an IFS to be coded with different numbers and different types of functions.

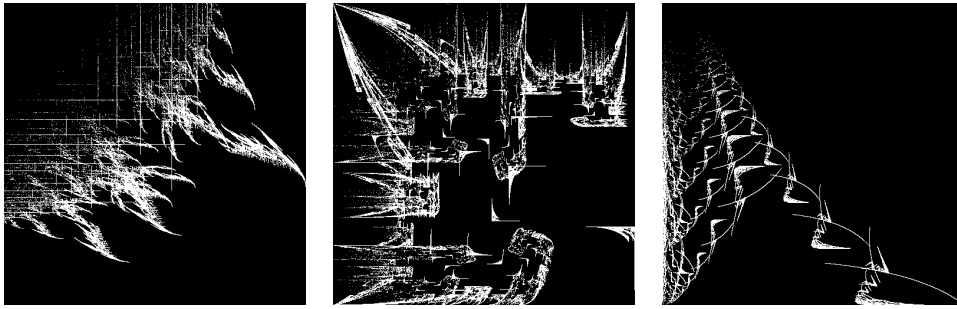


Figure 3.7: Other examples of attractors generated with mixed IFSs.

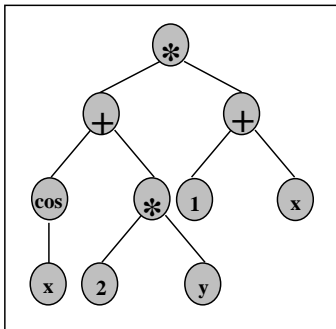
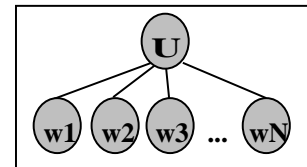
Figure 3.8: The function $((\cos(x) + 2 * y) * (1 + x))$.

Figure 3.9: Representation of a mixed IFS.

The set of possible IFSs depends on the choice of the basic function set and constant set. A difficult problem for a mixed IFS is to verify that the w_i are contractive, in order to select a *hyperbolic* IFS. Unlike the case of an affine IFS, this verification is not straightforward for a mixed IFS and is in fact computationally intractable. We have proposed using some heuristics that reject strongly noncontractive functions (see [49] for details).

3.1.5 Inverse problem for Mixed IFS : a solution based on GP

Genetic programming can be efficiently used in order to solve a more “general” inverse problem for IFS : a simultaneous numeric and a symbolic optimization can be efficiently performed with GP. The interest for the inverse problem with “mixed IFS” is important and may enlarge the scope of applications, for example image compression, because it allows a wider range of shapes to be coded.

A GP algorithm has been tested on shapes that were actual attractors of IFSs, some of which being constituted with randomly chosen contractive maps. The choice of basic functions for the GP is the one presented in section 3.1.4. Initial populations were randomly chosen. See appendix C for a detailed description of the algorithm and experiments.

The inverse problem for mixed IFS has been solved within a reasonable computation time (a few hours on Sparc 10 and Dec 5000 stations). This computation time is similar to computation times of GA applied to the inverse problem for the affine IFS [47], although in the case of the mixed IFS the size of the search space is much larger. This fact may be explained by the use of variable-sized

structures in the GP algorithm, which seems to perform a more efficient search in a large space. Parameter adjustment remains a challenging task, but we empirically noticed the following facts :

- The distance images are very efficient, and especially their associated multiresolution scheme : increasingly precise evaluation of the difference between the current IFS and the target are used throughout the evolution of the GP algorithm.
- The mutation operator is important : a specialised mutation on constants has been used in order to make the constants evolve.

Finally, the target images that yield good results are rather compact ; the convergence to line-shaped targets is more difficult.

3.2 Interactive GP for the generation of Mixed IFS attractors

A genetic Programming algorithm may also be used in an interactive way in order to investigate the space of mixed IFS attractors and invariant measures. We present below such an interactive implementation for the generation of attractors images. The GP algorithm is used as an exploration tool in an image space: the implicitly optimised function is the “users satisfaction”. This interactive approach is not new : Karl Sims [71] has extensively shown the power of the method in the framework of computer graphics.

An application has been developed (in JAVA language and based on the software PROGON) in order to provide an artistic exploration tool. A beta version of this algorithm is available on the FRACTALES WEB pages: <http://www-rocq.inria.fr/fractales/>.

The originality of our implementation is based on the use of mixed IFS, which are fractal structures represented as variable size trees, and on the use of genetic programming.

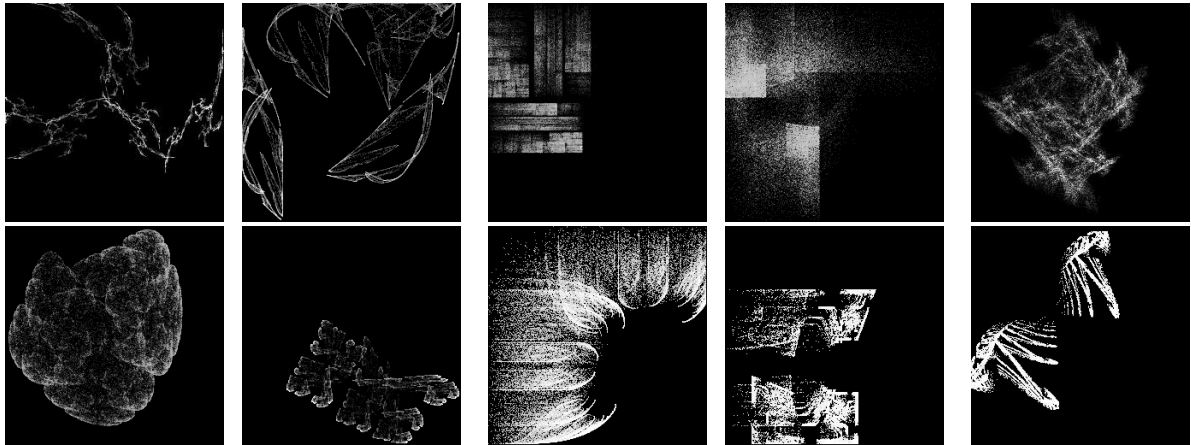


Figure 3.10: Some Mixed IFS invariant measures and attractors generated using interactive GP

3.3 Finite Automata

The problem presented here deals with the mathematical analysis of sequences generated by finite automata. These sequences have some fractal characteristics (infinite sequence of words issued from a fixed point finite automaton are self similar) and the corresponding inverse problem can also be related to some data compression problems in the same way as the inverse problem for IFS. There is no known general exact method for solving the associated inverse problem and GA optimization techniques can provide useful results. Appendix E presents some results obtained on the inverse problem for fixed point automata.

A *finite automaton* is defined as a symbolic substitution σ acting on strings of symbols. More precisely σ is a map from a finite set of symbols S to S^* , the set of strings of symbols in S . The elements of S^* are called *words* and the images by σ of elements of S are called *words of the automaton*.

A sequence of words can be produced by successive applications of σ to an initial word s_0 . If we denote by $s_n = s_{n_1}s_{n_2}\dots s_{n_p}$ the word at step n , the word obtained at step $n + 1$ is then:

$$s_{n+1} = \sigma(s_n) = \sigma(s_{n_1})\sigma(s_{n_2})\dots\sigma(s_{n_p}).$$

Example 1 $S = \{1, 2, 3\}$

$$\sigma \begin{cases} 1 \rightarrow 211 \\ 2 \rightarrow 13 \\ 3 \rightarrow 123 \end{cases}$$

Iteration	Word
0	1
1	211
2	13211211
3	2111231321121113...

If the word s_0 happens to be a prefix of the word $s_1 = \sigma(s_0)$, it is not hard to see that the sequence of words $(s_n)_{n \in \mathbb{N}}$ converges to an infinite word. And this infinite word is clearly a *fixed point* of the substitution σ (extended to infinite words by concatenation).

Example 2 $S = \{1, 2, 3\}$

$$\sigma \begin{cases} 1 \rightarrow 21 \\ 2 \rightarrow \mathbf{231} \\ 3 \rightarrow 13 \end{cases}$$

Iteration	Word
0	2
1	231
2	2311321
3	23113212111323121...

Now suppose an infinite word is given, is this word the fixed point of a substitution ? (i.e. given a finite word s , find an automaton σ and an initial word s_0 such that $\sigma^n(s_0) = s$ for some n .) No general answer to this question is known : a theoretical answer is known if the substitution has constant length [14] (i.e., if all words of the automaton have the same length), and also if the substitution is *primitive* (i.e., is such that there exists an m with the property that σ^m of any symbol contains at least one occurrence of each symbol of the set S) as proved recently by Durand [21].

This inverse problem can be reformulated as an optimization problem on the search space of all possible σ , n and s_0 that minimizes a distance between $\sigma^n(s_0)$ and s . In order to reduce the complexity of this problem² one also has to give some restrictions to the search space. The length of the words

²The search space is massively multimodal : for a given solution couple (σ, s_0) and for any divisor p of n , the couple

of σ is limited to l_{max} and s_0 is a single symbol.

The inverse problem for a fixed point automaton is much easier to solve than in the general case. Indeed, the information contained in the target word can be efficiently exploited, taking advantage of the fact that a fixed point is a succession of words of the automaton as well as the succession of symbols which generated them. Of course, it is necessary to know the lengths of the words in order to identify the connection between the two successions. A simple assumption on the lengths of words of the automaton then makes it possible to identify it with a mechanism of simultaneous identification and reconstruction.

Interesting results were obtained on fixed points automata, see appendix E. These experiments suggest that a coding of the general problem based on a set of possible words observed in the target sequence to be analysed, by considerably reducing the search space of possible automata, would make the resolution of the general case easier.

(σ^P, s_0) is also a solution.

3.4 Other Applications

Below are presented applications that are not related to fractal signal analysis.

3.4.1 Image Analysis

Geometric Primitives extraction is an important task in image analysis. It is especially important in the case of indoor vision, where most of the objects to be analysed are manufactured. Their description with the help of bidimensional or tridimensional geometric primitives is well adapted. We present here an alternative to the Hough Transform (HT). HT is a very efficient method for lines or simple primitives detection, but reaches its limits for complex primitives. HT consists in the searching of maxima in the space of parameters which describe the primitive. It explicitly constructs the function to be optimized, represented by an “accumulator”, i.e. a sampling of the parameter space. The effective detection of primitives is thus done by a rough sequential search on the accumulator. It becomes rapidly untractable to store an accumulator and detect optima on it when the number of parameters to estimate (and thus the dimension of the search space) increases. This is why we have to think about efficient optimization techniques to solve the problem for complex geometric primitives.

This optimization problem can be written as follows : optimize the position and size of a geometric primitive (or equivalently the values of parameters), knowing the edges detected on an image. The function implicitly optimized in the HT is the total number of contour points which coincide with the trace of the primitive defined by the parameters. When the dimension of the space to search is large, this function can be very irregular.

Roth and Levine [64] proposed a GA-based method for 2D and 3D primitives detection in 1992. For the detection of 2D primitives, we have improved their method (detailed in [50]) :

- by using distance images instead of directly using contour images, which tends to smoothen the function to optimize (see figure 3.11)
- by using a GA-sharing technique, to detect several image primitives in the same step (see figure 3.12),

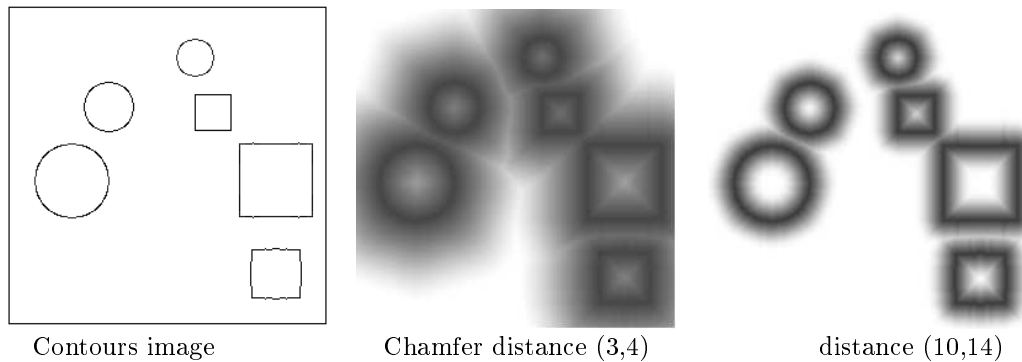


Figure 3.11: Example of distance images on a synthetic image

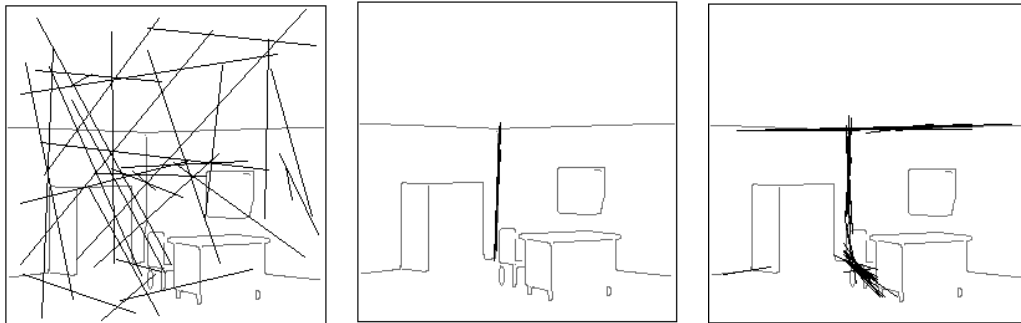


Figure 3.12: *Left, initial random population of segments (black) on a contour image (grey). Middle, final classical convergence of the population (after 80 generations). Right, convergence with sharing*



Figure 3.13: *Left, real image. Right, segments detected (black) and contours (grey)*

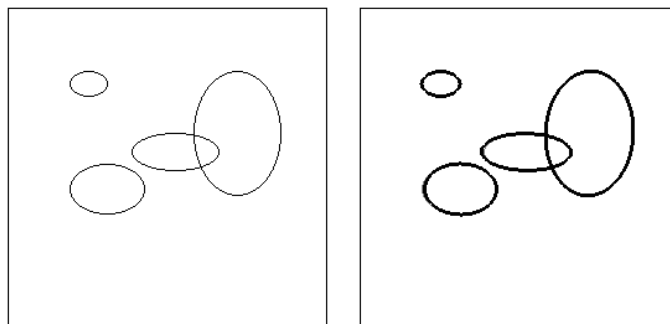


Figure 3.14: *Left, synthetic contours. Right, ellipses detected*

3.4.2 Physics signals analysis

We present below the use of a GA for the analysis of data from an aperiodically ordered system, high resolution X-Ray diffraction spectra from multilayer heterostructures arranged according to a deterministic or random scheme. This is again an “inverse problem”: the retrieval of the generating disorder from the investigation of the spectra of an unknown sample having non crystallographic, non quasi-crystallographic long range order.

The systems under study are multilayer heterostructures composed of planar layers of two kinds (the two letter alphabet) arranged according either to deterministic algorithms, aperiodic substitutional or automatic sequences, or analogous systems where the disorder generating sequence is unknown.

X-Ray diffraction spectra analysis: the model

In [61] is presented a theoretical and numerical study for the analysis of XRay diffraction spectra of Prouhet-Thue-Morse GaAs-AlAs multilayer heterostructures. Their calculation is based on kinematic diffraction theory.

The Prouhet-Thue-Morse sequence $\{\epsilon_n\}$ can be defined in several equivalent ways as follows :

- Let σ be a substitution acting on a two letter alphabet, for example $(0, 1)$:

$$\sigma \begin{cases} 0 \rightarrow 01 \\ 1 \rightarrow 10 \end{cases} \quad (3.1)$$

The sequence is then characterized by its initial conditions ϵ_0 and the number n of iterations of σ . Its length is $N = 2^n$.

With $\epsilon_0 = 0$, the sequence is :

$$0110100110010110\dots$$

- A recursive definition. With ϵ_i the i^{th} element in the sequence, one has :

$$\begin{cases} \epsilon_{2n} = \epsilon_n \\ \epsilon_{2n+1} = 1 - \epsilon_n \end{cases} \quad \text{with } \epsilon_0 \in \{0; 1\} \quad (3.2)$$

- A definition using an algorithmic machine known as a 2-automaton (see [57], [61]).

Let us notice that for a given length there are two possible sequences called *mirror* sequences corresponding to the two initial conditions $\epsilon_0 = 0$ and $\epsilon_0 = 1$.

Taking advantage of the specific properties of the Prouhet-Thue-Morse sequence and using the atomic structure factors of the GaAs and AlAs layers, the authors calculate a general formula for the diffraction amplitude $\hat{S}_n(q)$ with q the wave vector [61].

The intensity of the high resolution X-Ray diffraction spectrum is then :

$$I_n(q) = \left| \hat{S}_n(q) \right|^2 = \hat{S}_n^*(q) \hat{S}_n(q) \quad (3.3)$$

The authors have thus been able to successfully reproduce experimental high resolution X-Ray diffraction spectra from 2^7 and 2^{10} Prouhet-Thue-Morse multilayer heterostructures originally published in reference [5].

Generalization of the model

This model can be generalized for multilayer heterostructures having any kind of generating binary sequence $(s_N(k))_{k \in [0;N-1]}$ with N the total length of the sequence.

With the symbolic association and using the notations of [61]:

0 : codes a GaAs layer, with thickness d_0 and diffraction amplitude $\hat{\mu}_0(q)$ as a function of wave vector q .

1 : codes an AlAs layer, with thickness d_1 and diffraction amplitude $\hat{\mu}_1(q)$ as a function of wave vector q .

One has :

$$\hat{S}_N(q) = \mu_{s_N(0)}(q) + \sum_{j=1}^{N-1} e^{-2i\pi q \left[\sum_{k=0}^{j-1} d_{s_N(k)} \right]} \mu_{s_N(j)}(q) \quad (3.4)$$

We note that for the calculation of $\hat{S}_N(q)$ one has to use a summation of $N = 2^n$ terms, whereas the calculation of $\hat{S}_N(q)$ in the Prouhet-Thue-Morse case requires only n factors. The generalization in the present context is at the cost of going from $O(n)$ to $O(2^n)$ in computational time.

Nature and interest of the “inverse problem”

The goal is to retrieve from the experimental X-Ray diffraction spectrum of an unknown multilayer heterostructure sample the binary sequence after which the layers are arranged.

Such a problem is well resolved in “classical” crystallography where the possible lattices of crystalline samples have symmetries belonging to one of the 230 crystallographic groups. The analysis of the XRay diffraction spectrum then allows the complete retrieval of the structure, as the chemical composition is known. This is also true for quasicrystals [16, 72], but not for materials where long range order is “less regular”, such as aperiodic deterministic order generated by a substitutive sequence or for glassy materials where the disorder is in general thought of as being of random origin. The interest of finding or approaching a solution to this problem is obvious.

A GA has been used in order to find solutions to these difficult “inverse problems”, where no analytical solution is known. That means here solving the optimization problem that is the minimization of a distance between the experimental spectrum and the computed spectrum according to equation 3.4, with respect to the generating binary sequence $(s_N(k))_{k \in [0;N-1]}$. Preliminary results are presented in appendix F for synthetic data (see figures 3.15 and 3.16). We are currently working on experimental data (see figures 3.17 and 3.18), where a precise substrate model is needed in order to obtain reliable results.

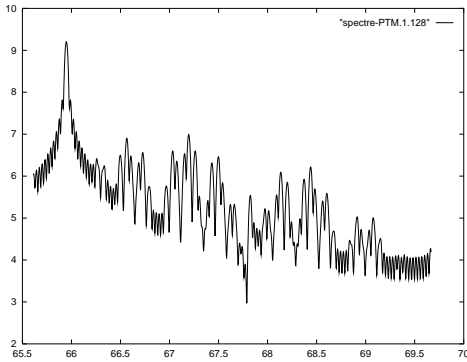


Figure 3.15: Spectrum generated with a Prouhet-Thue-Morse sequence of size 128

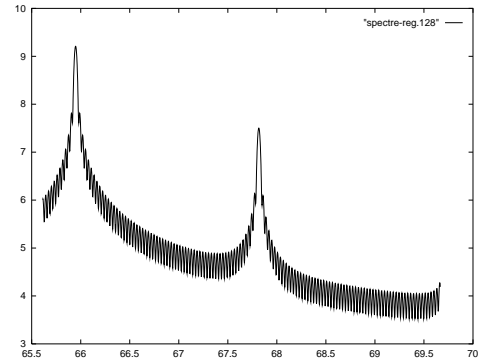


Figure 3.16: Spectrum generated with a periodic sequence of size 128

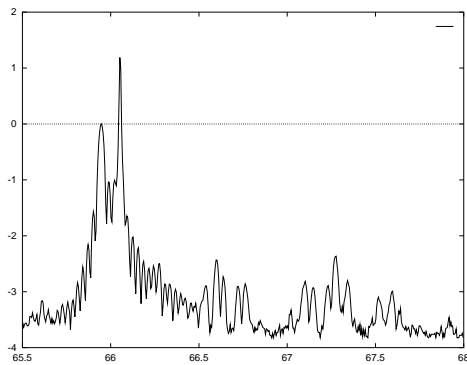


Figure 3.17: Experimental spectrum for a Prouhet-Thue-Morse GaAs-AlAs multilayer with $N=128$ (courtesy F. Laruelle, L. Leprince and J. Schneck, CNET-Bagneux)

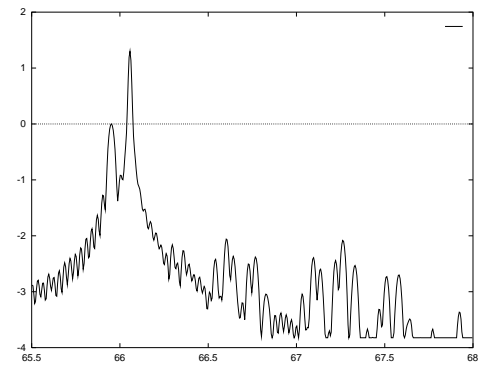


Figure 3.18: Computed spectrum from the model of [18] including a simple substrate model

3.5 Software

ALGON and PROGON are our EA toolboxes, ALGON for GA and PROGON for GP. These softwares have been built in order to simplify the implementation of various applications, and provide, besides a variety of classical operators and strategies, original tools (for non binary strings, chromosomes vectors and variable length chromosomes for ALGON, and new mutation operators for PROGON). ALGON and PROGON have been developed in C and a graphic interface is available that allow to visualize the progress of the algorithm and parameters setting.

3.5.1 ALGON : A GA software

ALGON is a set of functions and procedures, written in C language. It is designed for building Genetic Algorithm applications.

The main characteristics of ALGON are:

- 1 an encoding alphabet of arbitrary size.
- 2 variable size chromosomes.
- 3 an arbitrary number of chromosomes per individual.
- 4 various choices for the selection method, the crossover and mutation operators, the *sharing* method and the population replacement scheme.
- 5 a graphic interface (using TixWISH) for GA interactive parameterization and performance visualization.

The user has to supply some application specific functions, notably:

- the coding of individuals.
- the *fitness* function (to be maximized).

With these specifications, an executable file is compiled which supplies information about the GA performances among other things, via a dedicated file used by the graphic interface.

This software is available on the WEB pages of the FRACTALES Group.

3.5.2 PROGON : A GP software

PROGON is being built in the same way as ALGON, but for the evolution of tree-structured data. The main characteristics of this GP toolbox are:

- various choices for the definition sets of the trees (set of nodes, leaves and variables),
- specialized crossover and mutation operators for trees,
- various choices for selection, sharing and population replacement scheme,
- a graphic interface (using TixWISH) for GA interactive parameterization and performance visualization.

A C++ version of this software will be soon available on the WEB pages of the FRACTALES Group.

Chapter 4

Concluding remarks and future directions ...

I have presented in this document an overview of the work I have done on Genetic Algorithms (with my students I would like to warmly thank again) since 1993 in the FRACTALES Group. The main theoretical contribution of this report concerns a new look at deception theory: the analysis of the adjusted fitness function f' has helped to shed new light on the notion of deception, that can be related – to a certain extent – to some irregularity measures on the function. This adjusted fitness function can be considered as an interesting performance indicator of the GA, as it involves some of the GA parameters¹. However, a lot of work remains to be done, especially concerning the exact signification of this function f' and its expression in the framework of other models than deception theory.

Another contribution of this work I would like to emphasize concerns the application of GA and GP to the inverse problems for IFS. The resolution of the inverse problem for IFS is an important topic for fractal analysis of data. We intend to continue on this topic in the following way :

- It is well known that fractal geometrical structures (as they are extremely irregular, but with some scaling homogeneity) have interesting physical properties, see [81]. An application related to structural mechanics has been initialized in collaboration with the CMAPX laboratory, based on the inverse problem for IFS.
- We are currently starting the experimentation of a different approach with GA/GP to the inverse problem for IFS, using a sort of “distributed” approach. An IFS is represented in the “classical” approach as an individual of a population: in the “distributed” approach, an IFS is represented with a whole population, an individual being a function. This tends to explore the search space in a more economical way (population sizes are smaller, too), at the expense of a more complex fitness computation, of course. This approach has been first implemented with GA for affine IFS, and preliminary experiments tend to prove that the GA converge very rapidly onto a rough approximation of the target shape, precise converge being a more difficult task (where a careful design of the fitness function is extremely important).
- Finally, an efficient resolution of the inverse problem for IFS may influence the classical techniques in the field of fractal compression of data (signal or images). Some studies in the FRAC-

¹Of course, deceptivity cannot be related to GA performance in a simple way, and we by no means claim that the bound on $|f - f'|$ is **the** performance indicator. The experimentations presented in appendix B tend to prove that this bound is reliable in a relative way to estimate encoding efficiency, but of course extensive experiments must be carried out before claiming such a fact.

TALES group concerning data compression [78, 17] will be pursued based on more complex transformation and on the use of GA/GP optimization techniques.

Bibliography

- [1] A. Agapie. Genetic algorithms : Minimal conditions for convergence. In *Artificial Evolution, European Conference, AE 97, Nimes, France, October 1997.*, Springer Verlag, 1997.
- [2] J. Albert, F. Ferri, J. Domingo, and M. Vincens. An Approach to Natural Scene Segmentation by Means of Genetic Algorithms with Fuzzy Data . In *Pattern Recognition and Image Analysis*, pages 97–113, 1992. Selected papers of the 4th Spanish Symposium (Sept 90), Perez de la Blanca Ed.
- [3] Lee Altenberg. The Schema Theorem and Price’s Theorem . In *Foundation of Genetic Algorithms 3*, 1994.
- [4] P. Andrey and P Tarroux. Unsupervised image segmentation using a distributed genetic algorithm. *Pattern Recognition*, 1993.
- [5] F. Axel and H. Terauchi. High resolution x-ray diffraction spectra of thue-morse gaas-alas heterostructures. *Phys. Rev. Lett.*, 66:2223, 1991.
- [6] M. Barnsley and S. Demko. Iterated function system and the global construction of fractals. *Proceedings of the Royal Society, A* 399:243–245, 1985.
- [7] M. Barnsley, S. Demko, J. Elton, and Geronimo J. Invariant measures for Markov processes arising from iterated function systems with place-dependent probabilities. Georgia Tech. preprint.
- [8] M. Barnsley, V. Ervin, D. Hardin, and J. Lancaster. Solution of an inverse problem for fractals and other sets. *Proc. Natl. Acad. Sci. USA*, 83, 1986.
- [9] M. F. Barnsley. *Fractals Everywhere*. Academic Press, N Y, 1988.
- [10] J.-P. Berroir and J. Lévy Véhel. Multifractal tools for image processing. In *SCIA, Tromso*, 1993.
- [11] G. Borgefors. Distance transformation in arbitrary dimension. *Computer Vision, Graphics, and Image Processing*, 27, 1984.
- [12] R. Cerf. *Une théorie asymptotique des Algorithmes Génétiques*. PhD thesis, Université Montpellier II, 1994.
- [13] R. Cerf. *Artificial Evolution, European Conference, AE 95, Brest, France, September 1995, Selected papers*, volume Lecture Notes in Computer Science 1063, chapter Asymptotic convergence of genetic algorithms, pages 37–54. Springer Verlag, 1995.
- [14] A. Cobham. Uniform tag sequences. *Math. Systems Theory*, 6:164–192, 1972.
- [15] Nathan Cohen. Antennas in chaos : Fractal-element antennas. In *Fractals in Engineering 97*. INRIA, 1997. Hot Topic Session, Arcachon, France, June 25-27.

- [16] M. Cornier-Quiquandon, M. Quivy, S. Lefebvre, E. Elkaim, G. Heger, A. Katz, and D. Gratias. Neutron diffraction study of icosahedral al-cu-fe single quasicrystals. *Phys. Rev. B* *44*, page 2071, 1991.
- [17] Guillaume Cretin and Evelyne Lutton. Fractal image compression : Experiments on hv partitioning and linear combination of domains. In *Fractals in Engineering 97*. INRIA, 1997. Poster Session, Arcachon, France, June 25-27.
- [18] Y. Davidor. *Genetic Algorithms and Robotics. A heuristic Strategy for Optimization*. World Scientific Series in Robotics and Automated Systems - vol 1. World Scientific, Teaneck, NJ, 1991.
- [19] Thomas E. Davis and Jose C. Principe. A Simulated Annealing Like Convergence Theory for the Simple Genetic Algorithm . In *Proceedings of the Fourth International Conference on Genetic Algorithm*, pages 174–182, 1991. 13-16 July.
- [20] S. Demko, L. Hodges, and B. Naylor. Construction of fractal objects with ifs. *Siggraph*, 19(D 3), 1985.
- [21] F. Durand. A characterization of substitutive sequences using return words. *Disc. Math.*, 1997. to appear.
- [22] J. H. Elton. An ergodic theorem for iterated maps. In *Georgia Tech. preprint*, 1986.
- [23] K. J. Falconer. *Fractal Geometry : Mathematical Foundation and Applications*. John Wiley & Sons, 1990.
- [24] Y. Fisher. Fractal image compression. In *Siggraph 92 course notes*, 1992.
- [25] B. Goertzel. Fractal image compression with the genetic algorithm. *Complexity International*, 1, 1994.
- [26] D. E. Goldberg. Genetic Algorithms and Walsh functions : Part I, a gentle introduction. *TCGA Report No. 88006*, University of Alabama, Tuscaloosa, US, 1988.
- [27] D. E. Goldberg. Genetic Algorithms and Walsh functions : Part II, deception and its analysis. *TCGA Report No. 89001*, University of Alabama, Tuscaloosa, US, 1989.
- [28] David A. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Publishing Company, inc., Reading, MA, January 1989.
- [29] David E. Goldberg. Construction of high-order deceptive functions using low-order walsh coefficients. IlliGAL Report 90002, University of Illinois at Urbana-Champaign, Urbana, IL 61801, December 1990.
- [30] David E. Goldberg and J. Richardson. Genetic algorithms with sharing for multimodal function optimization. In J. J. Grefenstette, editor, *Genetic Algorithms and their Applications*, pages 41–49, Hillsdale, New Jersey, 1987. Lawrence Erlbaum Associates.
- [31] D. P. Hardin. *Hyperbolic Iterated Function Systems and Applications*. PhD thesis, Georgia Institute of Technology, 1985.
- [32] A. Hill and C. J. Taylor. Model-Based Image Interpretation using Genetic Algorithms . *Image and Vision Computing*, 10(5):295–301, June 1992.
- [33] J. H. Holland. *Adaptation in Natural and Artificial System* . Ann Arbor, University of Michigan Press, 1975.

- [34] A. Homaifar and S. Guan. Training weights of neural networks by genetic algorithms and messy genetic algorithms. In M. H. Hamza, editor, *Proceedings of the Second IASTED International Symposium. Expert Systems and Neural Networks*, pages 74–77, Hawaii, 17-17 Aug 1990. Acta Press, Anaheim, CA.
- [35] J. Hutchinson. Fractals and self-similarity. *Indiana University Journal of Mathematics*, 30:713–747, 1981.
- [36] E. W. Jacobs, Y. Fisher, and R. D. Boss. Fractal image compression using iterated transforms. In *Data Compression*, 1992.
- [37] A.E. Jacquin. Fractal image coding: A review. *Proc. of the IEEE*, 81(10), 1993.
- [38] J. Juliany and M. D. Vose. The genetic algorithm fractal. *Evolutionary Computation*, 2(2):165–180, 1994.
- [39] J. R. Koza. *Genetic Programming*. MIT Press, 1992.
- [40] Benoit Leblanc and Evelyne Lutton. Bitwise regularity coefficients as a tool for deception analysis of a genetic algorithm. Technical Report RR-3274, INRIA research report, 1997. October.
- [41] Benoit Leblanc, Evelyne Lutton, and Jean-Paul Allouche. *Artificial Evolution, European Conference, AE 97, Nimes, France, October 1997, Selected papers*, volume Lecture Notes in Computer Science, chapter Inverse problems for finite automata: a solution based on Genetic Algorithms. Springer Verlag, 1997.
- [42] J. Lévy Véhel. Fractal probability functions : an application to image analysis. In *CVPR, Maui*, 1991.
- [43] J. Lévy Véhel. *Bases et Applications de la Géométrie Fractale*. INRIA, 1993. à paraître.
- [44] J. Lévy Véhel and A. Gagalowicz. Fractal approximation of 2D objects. In *EUROGRAPHICS*, 1988. Nice.
- [45] J. Lévy Véhel, P. Mignot, and J.-P. Berroir. Multifractals, texture, and image analysis. In *CVPR, Urbana Champaign*, 1992.
- [46] P. Liardet. Algorithmes genetiques d’un point de vue ergodique. In *Mini-colloque “Algorithmes Evolutifs : de la theorie aux applications”*, 1997. Centre de Mathematiques et Informatique, Marseille, 19-20 juin 1997.
- [47] E. Lutton and J. Levy-Vehel. Optimization of fractal functions using genetic algorithms. Technical Report 1941, INRIA research report, June 1993.
- [48] E. Lutton and J. Lévy Véhel. Some remarks on the optimization of hölder functions with genetic algorithms. *Submitted to Evolutionary Computation*, 1995. (see also Inria Research Report No 2627).
- [49] E. Lutton, J. Lévy Véhel, G. Cretin, P. Glevarec, and C. Roll. Mixed ifs: resolution of the inverse problem using genetic programming. *Complex Systems*, 9:375–398, 1995. (see also Inria Research Report No 2631).
- [50] Evelyne Lutton and Patrice Martinez. A genetic algorithm for the detection of 2d geometric primitives in images. Research Report 2110, INRIA, November 1993.

- [51] Evelyne Lutton and Patrice Martinez. A Genetic Algorithm for the Detection of 2D Geometric Primitives in Images. In *12-ICPR*, 1994. Jerusalem, Israel, 9-13 October.
- [52] Evelyne Lutton and Patrice Martinez. Détection de primitives géométriques bidimensionnelles dans les images à l'aide d'un algorithme génétique. In *Evolution Artificielle 94*, 1994. Toulouse, France, 19-23 Septembre.
- [53] Evelyne Lutton and Jacques Lévy Véhel. Action fractales : Approches fractales pour l'analyse et la modélisation des signaux : thèmes de recherche autour des algorithmes génétiques. In *Evolution Artificielle 94*, 1994. Toulouse, France, 19-23 Septembre.
- [54] Evelyne Lutton and Jacques Lévy Véhel. Some remarks on the optimization of holder functions with genetic algorithms. Research Report 2627, INRIA, July 1995.
- [55] B.B. Mandelbrot and J.W. Van Ness. Fractional Brownian Motion, fractional Gaussian noises and applications. *SIAM Review* 10, 4:422–437, 1968.
- [56] G. Mantica and A. Sloan. Chaotic optimization and the construction of fractals : solution of an inverse problem. *Complex Systems*, 3:37–62, 1989.
- [57] M. Mendès France and J-P Allouche. In *Beyond Quasicrystals*. Les Editions de Physique/Springer Verlag, 1995. Axel, F. and Gratias, D., eds.
- [58] Zbigniew Michalewicz and Marc Schoenauer. Evolutionary algorithms for constrained parameter optimization problems. *Evolutionary Computation*, 4(1):1–32, 1997.
- [59] P. Mignot, J. Lévy Véhel, and Y. Lechevallier. Arthur : un système d'analyse de textures. *TS*, 1993.
- [60] D. J. Nettleton and R. Garigliano. Evolutionary algorithms and a fractal inverse problem. *Biosystems*, 33:221–231, 1994. Technical note.
- [61] Jacques Peyrière, Eric Cockayne, and Françoise Axel. Line-shape analysis of high resolution x-ray diffraction spectra of finite thue-morse gaas-alas multilayer heterostructures. *J. Phys. I France*, 5:111–127, 1995.
- [62] S. Rochet, G. Venturini, M. Slimane, and E.M. El Kharoubi. *Artificial Evolution, European Conference, AE 97, Nimes, France, October 1997, Selected papers*, volume Lecture Notes in Computer Science, chapter A critical and empirical study of epistasis measures for predicting GA performances : a summary. Springer Verlag, 1997.
- [63] Sophie Rochet. *Convergence des Algorithmes Génétiques : modèles stochastiques et epistasie*. PhD thesis, Université de Provence, Janvier 1998. Spécialité Mathématiques et Informatique.
- [64] G. Roth and M. D. Levine. Geometric Primitive Extraction Using a Genetic Algorithm . In *IEEE Computer Society Conference on CV and PR*, pages 640–644, 1992.
- [65] G. Rudolph. Asymptotical convergence rates of simple evolutionary algorithms under factorizing mutation distributions. In *Artificial Evolution, European Conference, AE 97, Nimes, France, October 1997*,. Springer Verlag, 1997.
- [66] Marc Schoenauer. *Evolutionary Computation and Numerical Optimization*. Dossier d'habilitation à diriger des recherches, Université Paris XI Orsay, January 1997. Spécialité Informatique.
- [67] Marc Schoenauer and Zbigniew Michalewicz. Boundary operators for constrained parameter optimization problems. In *ICGA97*.

- [68] Marc Schoenauer and Zbigniew Michalewicz. Evolutionary computation at the edge of feasibility. In *PPSN 96, Parallel Problem Solving from Nature*.
- [69] Marc Schoenauer and Spyros Xanthakis. Constrained ga optimization. In *Proceedings of the first International Conference on Genetic Algorithms, Urbana Champaign, 1993*.
- [70] R. Shonkwiler, F. Mendivil, and A. Deliu. Genetic algorithms for the 1-d fractal inverse problem. In *Proceedings of the Fourth International Conference on Genetic Algorithms*, pages 495–502, 1991. 13-16 July.
- [71] K. Sims. Interactive evolution of dynamical systems. In *First European Conference on Artificial Life*, pages 171–178, 1991. Paris, December.
- [72] W. Steurer. Five-dimensional patterson analysis of the decagonal phase of the system al-mn. *Acta Cryst. B45*, page 534, 1989.
- [73] Gilbert Syswerda. Uniform crossover in genetic algorithms. In J.D. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms*. Morgan Kaufmann publishers, San Mateo, 1989.
- [74] P. Trompette, J. L. Marcelin, and C. Schmeling. Optimal damping of viscoelastic constrained beams or plates by use of a genetic algorithm. In *IUTAM, 1993*. Zakopane Pologne.
- [75] S. Truvé. Using a Genetic Algorithm to solve Constraint Satisfaction Problems Generated by an Image Interpreter. In *Theory and Applications of Image Analysis : 7th Scandinavian Conference on Image Analysis*, pages 378–386, August 1991. Aalborg (DK).
- [76] J. Lévy Véhel and J.-P. Berroir. Image segmentation via multifractal analysis. 1993. *Fractal 93*, Londres.
- [77] J. Lévy Véhel and E. Lutton. Optimization of fractal functions using genetic algorithms. In *Fractal 93*, 1993. London.
- [78] Jacques Lévy Véhel, Khalid Daoudi, and Evelyne Lutton. Fractal modeling of speech signals. *Fractals*, 2(3):379–382, September 1994.
- [79] L. Vences and I. Rudomin. Fractal compression of single images and image sequences using genetic algorithms., 1994. The Eurographics Association.
- [80] J. Lévy Véhel. *Analyse et synthèse d'objets bi-dimensionnels par des méthodes stochastiques*. PhD thesis, Université de Paris Sud, Décembre 1988.
- [81] Jacques Lévy Véhel, Evelyne Lutton, and Claude Tricot (Eds). *Fractals in Engineering : From Theory to Industrial Applications*. Springer Verlag, 1997. ISBN 3-540-76182-9.
- [82] M.D. Vose. Formalizing genetic algorithms. In *Genetic Algorithms, Neural Networks and Simulated Annealing Applied to Problems in Signal and Image processing*. The institute of Electrical and Electronics Engineers Inc., 8th-9th May 1990. Kelvin Conference Centre, University of Glasgow.
- [83] E. R. Vrscay. *Fractal Geometry and Analysis*, chapter Iterated function Systems: theory, applications and the inverse problem, pages 405–468. 1991.
- [84] R. Vrscay. Moment and collage methods for the inverse problem of fractal construction with iterated function systems. In *Fractal 90 Conference*, 1990. Lisbonne, June 6-8.

Bibliographie Personnelle

Congrès avec comité de lecture

- [1] Evelyne LUTTON, Mathilde DELMAS, Henri MAITRE, "Une base de données pour la représentation de scènes industrielles complexes d'extérieur", PIXIM, 25-29 Septembre 1989, Paris, pp. 17-31.
- [2] Evelyne LUTTON, Henri MAITRE, "Etude des symétries du problème de perspective à trois lignes", 7e congrès AFCET, RFIA, 29 Novembre - 1er Décembre 1989, Paris, pp. 537-546.
- [3] Evelyne LUTTON, Henri MAITRE, Jaime LOPEZ-KRAHE, "Determining Vanishing Points with Hough Transform", Visual Communications and Image Processing '90, SPIE, Lausanne, Switzerland, 2-4 October 1990.
- [4] Evelyne LUTTON, Jean-Marc VEZIEN, André GAGALOWICZ, "About Criteria for 3D reconstruction of Planar Facets from a stereo pair", International Symposium on Computer and Information Sciences, ISCIS VII, 2-4 November 1992, Antalya, Turkey.
- [5] Véronique SERFATY, Andrew ACKAH-MIEZAN, Evelyne LUTTON, André GAGALOWICZ, "Photometric Analysis as an aid to 3D reconstruction of indoor scenes", IS & T / SPIE Symposium on Electronic Imaging : Science and Technology, January 31 - February 5, 1993, San Jose, California, USA.
- [6] Véronique SERFATY, Andrew ACKAH-MIEZAN, Evelyne LUTTON, André GAGALOWICZ, "Towards a "Visual" Model for Robot Vision : From Wireframe to Photometric Representation", IMAGE'COM 93, Bordeaux, France.
- [7] Evelyne LUTTON, Jean-Marc VEZIEN, André GAGALOWICZ, "Model-based stereo reconstruction by energy minimization", IMAGE'COM 93, Bordeaux, France.
- [8] Evelyne LUTTON, "3D Model-Based Stereo Reconstruction using Coupled Markov Random Fields", 5th International Conference CAIP'93, Computer Analysis of Images and Pattern (IAPR), September 13-15, 1993, Budapest, Hungary.
- [9] Jacques LEVY VEHEL, Evelyne LUTTON, "Optimization of Fractal Functions using Genetic Algorithms", Fractal'93, Londres, Sept 1993.
- [10] Evelyne LUTTON, Patrice MARTINEZ, "Détection de primitives géométriques bidimensionnelles dans les images à l'aide d'un algorithme génétique", Evolution Artificielle, Toulouse, 19-23 Septembre 1994.
- [11] Evelyne LUTTON, Patrice MARTINEZ, "A Genetic Algorithm for the Detection of 2D Geometric Primitives in Images", 12-ICPR, Jerusalem, Israel 9-13 Octobre 1994.
- [12] Evelyne LUTTON, "Genetic Algorithms and the Optimization of Fractal Functions: Deceptivity analysis", Fractal Geometry and Self-Similar Phenomena, Symposium celebrating the 70th birthday of Benoît Mandelbrot, February 2-4 1995, Curacao.

- [13] Evelyne LUTTON, Guillaume CRETIN, Jacques LEVY VEHEL, Philippe GLEVAREC, Cedric ROLL, "Mixed IFS : resolution of the inverse problem using Genetic Programming", Evolution Artificielle, Brest, 4-6 Septembre 1995.
- [14] Guillaume CRETIN, Evelyne LUTTON, "Fractal Image Compression: Experiments on HV Partitioning and Linear Combination of Domains", Fractals in Engineering, Arcachon, June 25-27, 1997.
- [15] Benoît LEBLANC, Evelyne LUTTON, Jean-Paul ALLOUCHE, "Inverse Problems for Finite Automata: A Solution Based on Genetic Algorithms", Evolution Artificielle, Nîmes, 22-24 Octobre 1997.
- [16] Benoît LEBLANC, Evelyne LUTTON, "Bitwise regularity and GA-hardness", 1998 IEEE World Congress on Computational Intelligence, Anchorage, Alaska, May 4-9, 1998.

Journaux

- [1] Evelyne LUTTON, Henri MAITRE, Jaime LOPEZ-KRAHE, "Determination of Vanishing Points using Hough Transform", PAMI, IEEE transactions on Pattern Analysis and Machine Intelligence, No 16, Vol 4, 1994.
- [2] Jacques LEVY VEHEL, Khalid DAOUDI, Evelyne LUTTON, "Fractal Modeling of Speech Signals", Fractals, No 3, Vol 2, September 1994.
- [3] Evelyne LUTTON, Guillaume CRETIN, Jacques LEVY VEHEL, Philippe GLEVAREC, Cedric ROLL "Mixed IFS : resolution of the inverse problem using Genetic Programming", Complex Systems, Vol 9, No 5, pp 375-398, October 1995.
- [4] Evelyne LUTTON, Jacques LEVY VEHEL, "Hölder functions and deception of Genetic Algorithms", IEEE transactions on Evolutionary Computation, Volume 2, Number 2, pp 56-72, July 1998.

Articles soumis

- [1] Benoît LEBLANC, Evelyne LUTTON, Françoise AXEL, "Genetic Algorithms as a tool in the study of aperiodic order : the case of X-Ray diffraction spectra of GaAs-AlAs multilayer heterostructures", presented to The European Physical Journal B., July 1997.

Livres édités

- [1] Jean-Marc ALLIOT, Evelyne LUTTON, Edmund RONALD, Marc SCHOENAUER (Eds), *Evolution Artificielle 94*, CEPADUES, 1994
- [2] Jean-Marc ALLIOT, Evelyne LUTTON, Edmund RONALD, Marc SCHOENAUER, Dominique SNYERS (Eds), *Artificial Evolution, European Conference, AE 95, Brest, France, September 1995, Selected papers*, volume Lecture Notes in Computer Science 1063, Springer Verlag, 1995.
- [3] Jacques LÉVY VÉHEL, Evelyne LUTTON, Claude TRICOT (Eds), *Fractals in Engineering : From Theory to Industrial Applications*, Springer Verlag, 1997.
- [4] Jin-Kao HAO, Evelyne LUTTON, Edmund RONALD, Marc SCHOENAUER, Dominique SNYERS (Eds), *Artificial Evolution, European Conference, AE 97, Nîmes, France, October 1997, Selected papers*, volume Lecture Notes in Computer Science 1363, Springer Verlag, 1997.

Conférences invitées

- [1] Evelyne LUTTON, "Decision Making Problems : Genetic Algorithms Principles and Applications", CODATA conference, Paris, 4-6 Novembre 1993.
- [2] Evelyne LUTTON, "Analyse de déceptivité des algorithmes génétiques et exposant de Hölder", Mini-colloque "Algorithmes Evolutifs : de la theorie aux applications", Centre de Mathématiques et Informatique, Marseille, 19-20 juin 1997.

Divers

- [1] Evelyne LUTTON, Jacques LEVY VEHEL, "ACTION FRACTALES : Approches Fractales pour l'Analyse et la Modélisation des Signaux : thèmes de recherche autour des algorithmes génétiques", Présentation d'équipe, Evolution Artificielle, Toulouse, 19-23 Septembre 1994.
- [2] Evelyne LUTTON, Jacques LEVY VEHEL, Guillaume CRETIN, "Mixed IFS", 2mn, film Vidéo VHS, 1995.
- [3] Benoît LEBLANC, Evelyne LUTTON, "ALGON: A genetic Algorithm Software Package", June 1997, <http://www-rocq.inria.fr/fractales/>

Thèse

- [1] Evelyne LUTTON, "Reconnaissance du point de prise de vue d'une photographie à partir d'un modèle de scène", Thèse Télécom Paris, 90E005, Mai 1990.

Rapports de recherche

- [1] Evelyne LUTTON, Henri MAITRE, Jaime LOPEZ-KRAHE, "Determination of Vanishing Points using Hough Transform", Rapport Télécom Paris 91 D 008, Département Images, Septembre 1991.
- [2] Evelyne LUTTON, Henri MAITRE, "Positionning a Camera from a Photograph and a Scene Model", Rapport Télécom Paris 92 D 002, Département Images, Janvier 1992.
- [3] Evelyne LUTTON, "3D Model-Based Stereo Reconstruction using Coupled Markov Random Fields", Rapport de Recherche INRIA No 1951, Juin 1993.
- [4] Jacques LEVY VEHEL, Evelyne LUTTON, "Optimization of Fractal Functions using Genetic Algorithms", Rapport de Recherche INRIA No 1941, Juin 1993.
- [5] Evelyne LUTTON, Patrice MARTINEZ "A Genetic Algorithm for the Detection of 2D Geometric Primitives in Images", Rapport de Recherche INRIA No 2210, Novembre 1993.
- [6] Evelyne LUTTON, Jacques LEVY VEHEL, "Some remarks on the optimization of Hölder functions with Genetic Algorithms", Rapport de Recherche INRIA No 2627, Juillet 1995.
- [7] Evelyne LUTTON, Guillaume CRETIN, Jacques LEVY VEHEL, Philippe GLEVAREC, Cedric ROLL "Mixed IFS : resolution of the inverse problem using Genetic Programming", Rapport de Recherche INRIA No 2631, Août 1995.
- [8] Benoît LEBLANC, Evelyne LUTTON, "Bitwise regularity coefficients as a tool for deception analysis", Rapport de Recherche INRIA No 3274, Octobre 1997.

Rapports d'expertise

- [1] Evelyne LUTTON, Henri MAITRE, "Systèmes de gestion de bases de données et images", Narcisse Project, T104 Workpackage Report, Juin 1990.
- [2] Evelyne LUTTON, "Etat de l'art des Algorithmes Génétiques", Rapport d'expertise pour le SGDN, Décembre 1993.

Part III
Publications

Appendix A

Hölder functions and Deception of Genetic Algorithms,

Evelyne LUTTON, Jacques LEVY VEHEL,
IEEE transactions on Evolutionary Computation, Vol 2 Number 2, July 1998.

This paper is available on : <http://www-rocq.inria.fr/fractales/Publications/>

Appendix B

Bitwise regularity coefficients as a tool for deception analysis.

Benoît LEBLANC, Evelyne LUTTON,
INRIA Research Report No 3274, October 1997

This paper is available on : <http://www-rocq.inria.fr/fractales/Publications/>

Appendix C

Mixed IFS : resolution of the inverse problem using Genetic Programming,

Evelyne LUTTON, Jacques LEVY VEHEL, Guillaume CRETIN,
Philippe GLEVAREC, Cédric ROLL
Complex Systems, Vol 9, No 5, pp 375-398, October 1995.

This paper is available on : <http://www-rocq.inria.fr/fractales/Publications/>

Appendix D

Fractal Modeling of Speech Signals,

Jacques LEVY VEHEL, Khalid DAOUDI, Evelyne LUTTON,
Fractals, Vol 2, No 6, September 1994.

This paper is available on : <http://www-rocq.inria.fr/fractales/Publications/>

Appendix E

Inverse Problems for Finite Automata: A Solution Based on Genetic Algorithms

Benoît LEBLANC, Evelyne LUTTON, Jean-Paul ALLOUCHE,
Evolution Artificielle, Nîmes, 22-24 Octobre 1997.

This paper is available on : <http://www-rocq.inria.fr/fractales/Publications/>

Appendix F

Genetic Algorithms as a tool in the study of aperiodic order : the case of X-Ray diffraction spectra of GaAs-AlAs multilayer heterostructures,

Benoît LEBLANC, Evelyne LUTTON, Françoise AXEL,
Submitted to The European Physical Journal B., July 1997.

This paper is available on demand.

Appendix G

A Genetic Algorithm for the Detection of 2D Geometric Primitives in Images

Evelyne LUTTON, Patrice MARTINEZ,
12-ICPR, Jerusalem, Israel 9-13 Octobre 1994.

This paper is available on : <http://www-rocq.inria.fr/fractales/Publications/>

Abstract

This report presents the synthesis of research that has been done on Genetic Algorithms in the FRACTALES group of INRIA since its creation in 1993. The use of Genetic Algorithms for the resolution of optimization problems which arise when dealing with fractal signal analysis was an important step towards the efficient resolution of basic inverse problems such as inverse problems for Iterated Functions Systems. In this report we present, among other applications, how Genetic Algorithms and Genetic Programming techniques have been implemented in order to find solutions in a tractable computation time.

Furthermore, fractal tools have been used to carry out a theoretical investigation of deception properties of Genetic Algorithms. A link has thus been established between some irregularity measures on the function to be optimized and its deception. A qualitative analysis of the influence of some of the GA parameters has been made and a chromosomal encoding evaluation tool has been designed.

Résumé

Nous présentons dans ce document la synthèse des travaux concernant les Algorithmes Génétiques menés au projet FRACTALES de l'INRIA depuis sa création en 1993. L'emploi d'algorithmes génétiques pour la résolution de problèmes d'optimization qui se posent lorsque l'on fait de l'analyse de données avec des méthodes issues de la géométrie fractale a permis de proposer des solutions efficaces à des problèmes de base comme le problème inverse pour les IFS. Nous exposons notamment dans ce document comment des méthodes à base d'algorithmes génétique et de programmation génétique ont été employées pour fournir des solutions en un temps de calcul raisonnable.

Parallèlement une étude théorique de la notion de déceptivité des algorithmes génétiques a été menée à l'aide d'outils "fractals" permettant de quantifier l'irrégularité des fonctions. Cette étude a notamment permis de prouver un lien entre déceptivité et mesures d'irrégularités sur la fonction à optimiser. Une analyse qualitative de l'influence de certains paramètres de l'algorithme génétique sur la déceptivité a pu être dérivée, ainsi qu'un outil d'évaluation du codage chromosomique.