



HAL
open science

Algorithmes autonomes et modélisations de problèmes

Frédéric Lardeux

► **To cite this version:**

Frédéric Lardeux. Algorithmes autonomes et modélisations de problèmes. Intelligence artificielle [cs.AI]. Université d'Angers, 2012. tel-03379628

HAL Id: tel-03379628

<https://univ-angers.hal.science/tel-03379628v1>

Submitted on 15 Oct 2021

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

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

Algorithmes autonomes et modélisations de problèmes

Habilitation à Diriger des Recherches

(Spécialité Informatique)

Université d'Angers

présentée et soutenue publiquement le 30 Novembre 2012

par

Frédéric Lardeux

Composition du jury

<i>Rapporteurs :</i>	Laetitia JOURDAN Bertrand NEVEU Lakhdar SAIS	Professeur à l'Université de Lille Ingénieur en chef des Ponts et Chaussées Professeur à l'Université de Lens
<i>Examineurs :</i>	Xavier GANDIBLEUX Marc SCHOENAUER	Professeur à l'Université de Nantes Directeur de recherche INRIA Saclay
<i>Invité :</i>	Éric MONFROY	Professeur à l'Université de Nantes
<i>Directeur :</i>	Frédéric SAUBION	Professeur à l'Université d'Angers

Table des matières

Avant-propos et organisation du mémoire	1
<hr/>	
Partie I Travaux de Recherche	3
Différents axes de recherche	5
1 Modélisation de problèmes	9
1.1 Règles de réduction de contraintes globales codées en SAT	10
1.1.1 Codage CSP et SAT	12
1.1.2 Règles de réduction et de transformation pour la contrainte globale Alldiff .	14
1.1.3 Chevauchements multiples de contraintes Alldiff	16
1.1.4 Évaluation	19
1.1.5 Conclusion	21
1.2 Problème de caractérisation multiple	22
1.2.1 Définition du problème	23
1.2.2 Efficacité des techniques d'apprentissage	24
1.2.3 Formules booléennes partiellement définies	25
1.2.4 Complexité	26
1.2.5 Expérimentations	27
1.2.6 Conclusion	28
2 Recherche Autonome	29
2.1 Introduction	30

2.2	Contrôleur	32
2.2.1	Sélection automatique d'opérateurs	32
2.2.2	Le contrôleur	33
2.2.3	Notre cobaye : GASAT	36
2.2.4	Capacité de contrôle	37
2.2.5	Gestion des opérateurs	40
2.2.6	Algorithmes évolutionnaires	43
2.2.7	Conclusion	43
2.3	Modèle en îles dynamique	47
2.3.1	Principes de base	47
2.3.2	Autonomie	48
2.3.3	Application au problème de SAO	51
2.3.4	Paramétrage du modèle	56
2.3.5	Comparaisons avec des sélections d'opérateurs non dynamiques	57
2.3.6	Conclusion	59
	Perspectives de recherche	61
<hr/>		
	Partie II Curriculum vitae	65
	Curriculum Vitae	67
	État civil	67
	Coordonnées professionnelles	67
	Fonction actuelle	67
	Parcours professionnel	68
	Parcours universitaire	68
	Administration de la recherche	68
	Projets de Recherche	69
	Liste des travaux	71
	Chapitre de livre	71
	Revue	71
	Brevet	71
	Conférences internationales avec actes	72
	Workshops	73
	Conférences nationales avec actes	73

Logiciel	73
Vulgarisation	74
Encadrement et animation de la recherche	75
Co-encadrement de thèses	76
Encadrement de stages de Master Recherche	76
Encadrement de post-doctorant	77
Activités Pédagogiques	79
Synthèse des enseignements	79
Informatique et Sciences du Numérique	79
Responsabilités pédagogiques	79
Activités Administratives	81
Fonctions électives	81
Autres responsabilités	81
<hr/>	
Partie III Sélection de publications	83
Choix des articles	85
SAT Encoding and CSP Reduction for Interconnected AllDIFF Constraints	87
Minimum Multiple Characterization of Biological Data using Partially Defined Boolean Formulas	101
Autonomous Operator Management for Evolutionary Algorithms	109
A dynamic island model for adaptive operator selection	139
<hr/>	
Bibliographie	149

Avant-propos et Organisation du mémoire

CE manuscrit est conçu en trois parties. La première présente mes principaux travaux de recherche, la seconde contient mon curriculum vitae et la troisième est une sélection de mes publications.

La bibliographie de la partie consacrée aux travaux de recherche contient certaines des publications auxquelles j'ai participé. Afin que le lecteur puisse facilement retrouver les références de mes publications liées à chacune des thématiques de recherche présentées, une liste de ces publications est présente à chaque fin de section. La liste complète est proposée dans la partie contenant mon curriculum vitae.

Lors de la rédaction de la partie sur les travaux de recherche, j'ai privilégié l'utilisation du pronom "nous" plutôt que le pronom "je". En effet, je ne conçois pas d'effectuer ma recherche en m'isolant de mes collègues. Les échanges que l'on peut avoir lors de réunions, de conférences voire même lors de pauses café sont à mon avis une des sources principales des avancées dans le domaine de la recherche. C'est pour cela que je tiens à associer tous ceux avec qui je travaille ou avec qui j'ai travaillé à la partie consacrée aux travaux de recherche en parlant de "nos" travaux plutôt que de "mes" travaux.

Partie I

Travaux de Recherche

Différents axes de recherche

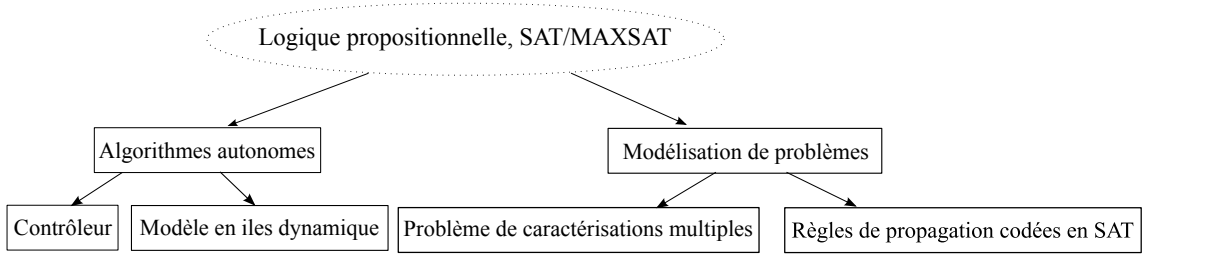
D'une manière générale, mon activité de recherche s'articule autour de la résolution de problèmes combinatoires. Un problème combinatoire est modélisé au moyen de variables pouvant prendre des valeurs discrètes (typiquement des nombres entiers), devant satisfaire un ensemble de contraintes et permettant éventuellement d'optimiser une fonction de coût. De tels problèmes se rencontrent naturellement dans le monde industriel : planification de production, ordonnancement de tâches, emploi du temps, tournées de véhicules... Il est alors possible d'utiliser différents formalismes pour modéliser de tels problèmes. Une fois le modèle fixé, il est nécessaire de fournir des outils, algorithmes spécifiques ou solveurs de problèmes plus généraux, pour les résoudre. De nombreuses approches ont été développées, émanant des communautés informatique et mathématiques appliquées, et touchant notamment les domaines de la recherche opérationnelle, de l'intelligence artificielle ou encore de la programmation mathématique.

Au cours de ma thèse, mes travaux de recherche s'intéressaient aux problèmes de satisfiabilité booléenne SAT et MAX-SAT. Une formule booléenne est construite en utilisant des variables binaires (qui peuvent être vraies ou fausses) et permet de modéliser de nombreux problèmes pratiques tels que la vérification de circuits électroniques. Par exemple, la formule booléenne $(a \text{ ou } b) \text{ et } c$ sera vérifiée si c est vraie et que soit a , soit b , sont vraies. Ce problème, en dépit de la simplicité de sa formulation, révèle une complexité de résolution importante, puisqu'intuitivement le temps nécessaire à vérifier si une telle formule admet une solution augmente exponentiellement en fonction de la taille de la formule. Le problème SAT sert ainsi en quelque sorte de mètre étalon lorsque l'on étudie la complexité de familles de problèmes et il a donc suscité un très grand nombre de travaux, à la fois théoriques et pratiques. A l'heure actuelle, une communauté très active tente d'améliorer sans relâche les performances des solveurs SAT et ce afin de pouvoir traiter des problèmes pratiques dont la taille et la difficulté ne cessent de croître.

Dans ce contexte, l'un des principaux résultats de ma thèse et des années qui ont suivi mon recrutement a été le développement de l'algorithme évolutionnaire GASAT [LSH06b]. Cet algorithme combine un algorithme génétique contenant des opérateurs spécialisés pour SAT et une recherche locale intégrant un voisinage multiple ainsi qu'un mécanisme permettant de sortir des minima locaux. GASAT fut proposé à la compétition SAT en 2004 [BS04] et a fini quatrième pour la catégorie des instances aléatoires. Depuis, nos travaux de recherche se sont diversifiés mais restent toujours de près ou de loin en relation avec le problème SAT. Nous avons alors abordé aussi bien la résolution efficace de problèmes que leur modélisation, ne négligeant pas des applications vers d'autres champs disciplinaires. Le schéma ci-dessous synthétise l'organisation de mon activité depuis mon recrutement comme maître de conférences

à l'université d'Angers en 2006.

FIGURE 1 : Travaux de recherche



Algorithmes Autonomes

La recherche d'algorithmes de résolution toujours plus efficaces et surtout capables de faire face aux nouvelles dimensions des problèmes pratiques à résoudre¹ a conduit les chercheurs à hybrider, comme nous l'avons fait dans nos travaux sur SAT, des techniques très spécialisées et à augmenter la flexibilité de leurs solveurs, souvent par l'introduction de nombreux paramètres.

Compte tenu des fameux théorèmes "No Free Lunch", il n'est pas réaliste de penser définir un paramétrage universel qui amènerait des performances optimales sur toutes les instances de problèmes à résoudre. Lorsque le nombre de paramètres est limité, l'utilisateur peut tenter de les régler manuellement sur des cas de test. Toutefois, lorsque ce nombre augmente et/ou que les stratégies de résolution interagissent fortement dans un même algorithme, cela nécessite une expertise qui met ces outils hors de portée de l'utilisateur final visé, en général l'ingénieur. Ainsi, la conception d'algorithmes de recherche plus autonomes, capables d'automatiser le réglage de leurs paramètres, est clairement un enjeu important pour la résolution de problèmes. Ce nouveau champs de recherche, en pleine expansion, se positionne à l'intersection de plusieurs domaines de l'informatique, faisant notamment appel à des techniques issues de l'intelligence artificielle et de l'apprentissage automatique. Il convient en général de distinguer le réglage préliminaire des paramètres du contrôle en ligne du comportement des algorithmes pendant leur exécution.

En ce qui nous concerne plus directement, lors du travail sur GASAT, la partie expérimentale a été très fastidieuse avant de trouver un réglage efficace. Ceci nous a donc amené naturellement à essayer de trouver une nouvelle façon d'utiliser nos algorithmes tout en limitant le plus possible la phase de paramétrage. Les bons résultats de certains algorithmes utilisant un paramétrage automatique (Reactive Tabu Search [BT94] par exemple) nous ont poussé à approfondir cette approche en proposant deux "méta" algorithmes génériques (appelés *contrôleur* et *modèle en îles dynamique*) capables de gérer automatiquement et dynamiquement le comportement d'algorithmes de résolution quelconques :

- Le contrôleur est un module se connectant à un algorithme existant. Il analyse l'impact de chaque étape de la résolution afin de récompenser ou de pénaliser les actions qui ont été réalisées. L'objectif est d'apprendre au cours de la recherche une politique optimale de résolution en adaptant dynamiquement les paramètres de l'algorithme à l'état de calcul courant.
- Le modèle en îles dynamique permet d'utiliser de manière parallèle différentes configurations d'un même algorithme en les répartissant sur des îles. Sur chaque île, chaque algorithme tente alors de

1. Les défis organisés dans le cadre de la société française de recherche opérationnelle illustrent bien ces nouvelles problématiques notamment liées à la gestion de l'énergie ou à l'allocation de ressources, avec des volumes de données et un nombre de variables de décisions très importants.

résoudre le même problème. Les solutions potentielles qui sont explorées sont alors échangées selon des probabilités de migration entre les îles. Il suffit ensuite de laisser évoluer une population de solutions sur ce modèle qui détermine le paramétrage ou les séquences de paramétrages optimaux. L'aspect dynamique de notre modèle en îles vient du fait que les probabilités de migration évoluent en fonction de l'amélioration des solutions visitées.

A l'aide de ces méthodes, il n'est plus nécessaire de tester de manière exhaustive tous les paramétrages afin de trouver le meilleur. Il est même possible de détecter des combinaisons de paramétrages au cours de la recherche permettant d'être plus efficace. Pour cela, le contrôleur et le modèle en îles dynamique travaillent simultanément avec tous les paramétrages potentiels et apprennent dynamiquement, et de manière autonome, ceux qui semblent les plus efficaces à un instant donné de la recherche. De manière naturelle, la notion d'efficacité d'un paramétrage était initialement calculée en fonction de l'évolution de la fonction d'évaluation. Nos méthodes abordent maintenant l'efficacité en fonction de plusieurs critères (qualité, entropie, temps d'exécution, ...). En effet, il peut être intéressant, à certains moments de la recherche, de vouloir intensifier alors qu'à d'autres moments il serait plus judicieux de diversifier. Cette alternance de comportements de recherche est bien connue mais il est toujours difficile de trouver un bon compromis entre eux. Nous rendons maintenant cela possible en évaluant chacun des paramétrages étudiés selon tous ces critères ce qui permet de mettre plus en évidence les différentes phases de la recherche. La création d'un algorithme est donc ramenée à une définition de stratégie de recherche de plus haut niveau pouvant faire varier les critères à améliorer (par exemple une alternance d'intensification et de diversification) et non à une définition d'un paramétrage fixe plus ou moins évolué pour un problème donné.

L'autre aspect intéressant de ces méthodes est la possibilité de faire un bilan critère par critère pour chacun des paramétrages. Nous arrivons à montrer que certains d'entre eux ne modifient qu'un seul critère de manière constante alors que d'autres voient leur impact beaucoup plus sensible à l'état d'avancement de la recherche. Une meilleure compréhension des caractéristiques des paramétrages est donc possible.

Modélisation

De manière concomitante à ces travaux sur les aspects pratiques de la résolution de problèmes, nous nous intéressons aussi à leur modélisation.

Un paradigme de modélisation bien connu pour l'optimisation combinatoire est le problème de satisfaction de contraintes (CSP), dans lequel un problème est modélisé par un ensemble de variables de décision, un ensemble de valeurs possibles pour ces variables (en général un sous-ensemble de valeurs entières) et un ensemble de contraintes devant être satisfaites. La résolution d'un CSP consiste donc à trouver une combinaison (affectation) de valeurs pour chaque variable de telle sorte que toutes les contraintes soient vérifiées. La résolution de ces problèmes peut être abordée en utilisant un algorithme qui explore l'arbre des affectations possibles de valeurs aux variables afin de trouver une solution. Cet arbre étant évidemment de taille trop importante pour une exploration exhaustive, des techniques, dites de propagation, permettent d'en élaguer des parties afin de rendre cette recherche de solution possible.

La modélisation SAT implique des sacrifices par rapport à une modélisation CSP. En effet, les techniques de propagations utilisées par les algorithmes de résolution pour les problèmes SAT sont beaucoup moins puissantes que celles utilisées par les problèmes de satisfaction de contraintes. Notre objectif est de passer d'une modélisation CSP à une modélisation SAT en conservant la puissance des règles de propagation des CSP dans la résolution en SAT. La propagation unitaire est la principale propagation appliquée pour SAT alors que pour les CSP de nombreuses règles de propagation existent. De plus, il

est très difficile d'ajouter de nouvelles règles de propagation pour la résolution en SAT en s'appuyant sur la structure du problème codé tout comme cela est fait du côté CSP. En effet, le codage souvent automatique en SAT "casse" la structure originelle du problème traité et seule la propagation unitaire peut s'appliquer. Afin de garder la puissance des règles de propagation des CSP, il est donc nécessaire de les intégrer dans le codage même du problème sous forme de clauses. Des codages de CSP vers SAT de type variable/valeur ont déjà été étudiés (par exemple dans [Wal00b]) mais ils ne cherchent pas à coder des règles de réduction propres aux combinaisons de contraintes.

Afin de mettre en avant les possibilités qu'un tel codage apporte, nous avons décidé de manipuler des règles de propagation assez puissantes propres aux contraintes globales AllDiff [Rég94], qui permettent d'indiquer que des variables doivent avoir des valeurs toutes distinctes, ce qui est très commode pour un bon nombre de problèmes. Nous définissons de nouvelles règles dans le cas où plusieurs contraintes AllDiff ont des variables en commun. En fonction du nombre de contraintes AllDiff se chevauchant, nous proposons différentes règles de propagation. Le problème concret sur lequel nous testons nos transformations est le problème du Sudoku car il permet de manipuler un grand nombre de contraintes de type AllDiff fortement imbriquées.

Application à la biologie

Les travaux que nous avons réalisés sur la logique propositionnelle en nous intéressant au problème SAT nous ont conduits à collaborer avec des collègues statisticiens et biologistes. Le problème qui a initié ce projet est celui de la caractérisation des collections de souches bactériennes responsables d'importantes pathologies végétales. Cette caractérisation précise est un enjeu scientifique majeur, car les bactéries sont responsables d'importantes maladies sur les plantes, et donc soumises à des procédures de contrôle officielles (par exemple, en Europe, la directive 2000/29/CE). Le développement de tests de diagnostic est donc un enjeu important pour systématiquement identifier les souches de ces espèces. Ces souches sont très nombreuses et il est impossible de toutes les tester. Caractériser un groupe de souches bactériennes vis à vis des autres, revient à trouver une combinaison logique de certains caractères n'apparaissant que dans celui-ci. Ce problème serait simple à traiter s'il ne fallait pas trouver la plus petite caractérisation possible afin de limiter le coût des analyses généralement faites à l'aide de puces à ADN.

On peut facilement s'abstraire de l'application biologique et reformuler ce problème comme la caractérisation minimale multiple de groupes de vecteurs booléens selon un ensemble de variables. Au premier abord, pour résoudre ce problème, il pourrait paraître logique d'utiliser des techniques d'apprentissage classiques telles que le classifieur Support Vecteur Machine [MLH03]. Le soucis est que ces techniques ne garantissent pas que tous les vecteurs booléens soient caractérisés de manière exacte. D'autres techniques de type FOIL [Qui90] semblent plus adaptées mais elles ne permettent pas la minimisation des caractérisations.

Pour résoudre ce problème, nous proposons des algorithmes travaillant avec des formules en forme normale conjonctive mais aussi avec des fonctions booléennes partiellement définies. Les résultats obtenus sur des instances réelles fournies par des biologistes ont permis de caractériser différents types de souches et de mettre au point un test de diagnostic efficace. Un brevet a ainsi été déposé à l'INPI.

Nous présenterons tout d'abord notre travail sur la modélisation de problèmes avec une partie sur le codage en problème SAT des règles de propagation utilisées pour les CSP et une autre partie sur le problème de caractérisation minimale. Ensuite, nous aborderons nos contributions en terme de recherche autonome avec la description du contrôleur et du modèle en files dynamique.