

Rapport d'habilitation à diriger des recherches

Simon de Givry

INRA UBIA Toulouse

[http ://www.inra.fr/mia/T/degivry/](http://www.inra.fr/mia/T/degivry/)

15 février 2011

Table des matières

1	Traitement des réseaux de fonctions de coût	2
1.1	Définition du cadre des réseaux de fonctions de coût	2
1.2	Cohérences locales	3
1.2.1	Schéma chaotique de reformulations locales	3
1.2.2	Schéma planifié de reformulations locales	8
1.2.3	Cohérence locale pour des domaines de grande taille	9
1.2.4	Perspectives en reformulation de fonctions de coût globales	11
1.3	Exploitation de la structure d'un réseau	12
1.3.1	Hybridation de la recherche avec l'élimination de variables	15
1.3.2	Hybridation de la recherche avec l'élimination de clusters	17
1.3.3	Choix d'une bonne décomposition arborescente	19
2	Perspectives d'application dans les réseaux bayésiens	20
2.1	Définition du cadre des réseaux bayésiens	21
2.2	Lien avec l'optimisation dans les réseaux de fonctions de coût	21
2.3	Inférence approchée	23
2.4	Apprentissage de la structure par l'optimisation d'un score	25
A	Applications	35
B	ToulBar2	35
C	Glossaire	36

Introduction

Dans ce rapport, je présente mes travaux autour des réseaux de fonction de coût. Le formalisme *Weighted Constraint Satisfaction Problem* (WCSP) associé à ces réseaux est générique, permettant la modélisation de nombreux problèmes d'optimisation combinatoire sous contraintes, y compris en bioinformatique. La contrepartie à cette grande généricité fut une relative faiblesse des méthodes exactes de résolution initialement proposées. Mon objectif a été d'étendre ces méthodes pour résoudre des problèmes concrets en bioinformatique, en exploitant des spécificités sémantiques ou structurelles observées dans ces problèmes. Ces deux aspects font l'objet de travaux méthodologiques présentés dans les sections 1.2 et 1.3 où je résume mes principales contributions.

L'incertitude sur les données inhérente en bioinformatique conduit souvent à définir un modèle probabiliste. Mon projet de recherche est de pouvoir appliquer avec succès

les méthodes exactes des WCSPs dans le cadre des modèles graphiques probabilistes discrets, en particulier les réseaux bayésiens dont le lien avec les WCSP est présenté en section 2.2. Au delà de la question de l’optimisation, je m’intéresse à l’inférence approchée et à l’apprentissage de la structure d’un réseau bayésien qui constituent les deux derniers axes de mon projet de recherche (le premier étant l’optimisation), présentés dans les sections 2.3 et 2.4. Ces deux axes sont chacun associés à une thèse que je co-encadre. Ces thèses contiennent en outre une partie appliquée en bioinformatique (reconstruction d’haplotypes et de réseaux de régulations de gènes). Une étoile dans la marge souligne ma participation dans une publication. Je ne présente pas mes autres activités faites à THALES (1998-2002) sur l’optimisation

- * en-ligne [de Givry et al., 2002] et la définition d’un langage d’expression de stratégies
- * de recherche hybrides facilitant l’écriture d’algorithmes à contrat de temps [Givry et al., 2001, de Givry and Jeannin, 2006], ainsi que mes travaux à l’INRA depuis
- * 2002 en cartographie de marqueurs [de Givry et al., 2005a, Faraut et al., 2007, Servin et al., 2010].

1 Traitement des réseaux de fonctions de coût

Dans cette partie, je présente le cadre fondateur des problèmes de satisfaction de contraintes et son extension à l’optimisation, les problèmes de satisfaction de contraintes pondérées. Je décris ensuite les principales techniques que j’ai développées pour traiter ces problèmes d’optimisation.

1.1 Définition du cadre des réseaux de fonctions de coût

Definition 1.1 (CSP) *Un réseau de contraintes discret est défini par un triplet (X, D, C) :*

- X , un ensemble de variables ;
- D , un ensemble de domaines finis de symboles (communément appelés valeurs), chaque variable $x \in X$ pouvant prendre une valeur de son domaine $d_x \in D$;
- C , un ensemble de contraintes, chacune $c_S \in C$ portant sur un sous-ensemble des variables $S \subseteq X$, appelé portée de la contrainte, et autorisant seulement une partie des combinaisons de valeurs possibles pour les variables de S .

Le problème de satisfaction de contraintes (Constraint Satisfaction Problem (CSP)) consiste à donner une valeur à chaque variable de telle sorte que toutes les contraintes soient satisfaites. Il s’agit d’un problème NP-complet.

L’arité d’une contrainte c_S est égale à la taille de S . On parle de CSP binaire lorsque toutes les arités sont inférieures ou égales à 2.

Une méthode exacte naïve appelée algorithme *BackTrack* (BT) pour résoudre ce problème consiste à énumérer les valeurs possibles des variables récursivement, en remontant à la variable précédemment affectée dès qu’une contrainte est violée par l’affectation courante. La complexité temporelle de cet algorithme est exponentielle en le nombre de variables (en $O(d^n)$ avec n le nombre de variables et d la taille du plus grand domaine). La complexité spatiale est linéaire. Cette méthode peut être améliorée en pratique par l’ajout d’un filtrage des domaines (en prétraitement ou au cours de la recherche après chaque affectation de variable) [Sabin and Freuder, 1994, Bessière and Régin, 1996, Debruyne and Bessière, 2001]. Ce filtrage consiste à supprimer les valeurs qui ne peuvent pas participer à une solution du problème. Le filtrage le plus souvent utilisé est la cohérence d’arc. Il consiste à considérer chaque contrainte isolément et à supprimer les valeurs qui ne peuvent pas satisfaire la contrainte (*i.e.* qui n’appartiennent à aucune combinaison autorisée par la contrainte).

Definition 1.2 (AC) Une contrainte binaire est arc-cohérente s'il existe pour chaque affectation d'une de ses variables une affectation de l'autre variable qui satisfait la contrainte. Un CSP binaire est arc-cohérent (Arc Consistency (AC)) si toutes ses contraintes sont arc-cohérentes.

Un algorithme de renforcement d'arc-cohérence tel que *AC2001* [Bessière and Régin, 2001] dans le cas d'un CSP binaire a une complexité temporelle et spatiale polynomiale (respectivement en $O(ed^2)$ et $O(ed)$ avec e le nombre de contraintes).

Lorsque le problème n'a pas de solution, il est souvent nécessaire de le relaxer pour trouver une affectation des variables qui satisfait au mieux les contraintes. Un critère simple consiste à minimiser le nombre de contraintes violées (ce qui revient à trouver une relaxation optimale du problème où ces contraintes seraient enlevées). Le problème est alors appelé Max-CSP. Un critère plus fin consiste à associer un coût de violation à chaque affectation des variables d'une contrainte. Ce coût peut être un entier¹ positif quelconque, y compris infini. Il vaut zéro si l'affectation satisfait la contrainte et il vaut l'infini si l'affectation est totalement interdite.

Definition 1.3 (WCSP) Un réseau de fonctions de coût discret est défini par un triplet (X, D, F) :

- X , un ensemble de variables ;
- D , un ensemble de domaines finis de valeurs, chaque variable $x \in X$ pouvant prendre une valeur de son domaine $d_x \in D$;
- F , un ensemble de fonctions de coût, chacune $f_S \in F$ portant sur un sous-ensemble des variables $S \subseteq X$ et associe un coût entier dans $[0, +\infty]$ pour chaque combinaison de valeurs possibles des variables de S .

Le problème de satisfaction de contraintes pondérées (Weighted Constraint Satisfaction Problem (WCSP)) consiste à donner une valeur à chaque variable de telle sorte que la somme des coûts des fonctions soit minimum. Il s'agit d'un problème NP-difficile.

Un cadre unificateur appelé *Valued Constraint Satisfaction Problem* (VCSP) permettant d'exprimer d'autres critères d'aggrégation des fonctions que la somme a été proposé dans [Schiex et al., 1995]. Il introduit une structure algébrique pour représenter les valeurs des fonctions et les agréger. Il a été montré dans [Cooper, 2005] (Appendix A) que pour la plupart des structures algébriques connues, il est possible de se ramener à un nombre fini de problèmes WCSP à résoudre d'où notre intérêt pour ce cadre.

Pour résoudre un WCSP, on utilise classiquement comme méthode exacte l'algorithme de séparation et évaluation (*Depth-First Branch and Bound* (DFBB)). A l'instar de BT, il consiste à énumérer les valeurs possibles des variables récursivement, en remontant à la variable précédemment affectée dès qu'une sous-estimation du coût de n'importe quelle extension complète de l'affectation courante (*i.e.* le minorant du sous-problème courant) dépasse le coût de la meilleure solution déjà visitée noté k (*i.e.* le majorant du problème d'origine, initialement supposé de coût infini). La complexité de DFBB est la même que pour BT. Un minorant trivial consiste à sommer le coût des fonctions dont les variables sont affectées. Nous allons voir dans la section suivante comment calculer de bons minorants.

1.2 Cohérences locales

1.2.1 Schéma chaotique de reformulations locales

Suite aux travaux de Thomas Schiex à l'INRA Toulouse [Schiex, 2000], l'idée principale a été de trouver des reformulations en temps polynomial d'un WCSP de manière

¹Le choix de coûts entiers plutôt que réels est fait pour des raisons d'efficacité sur les ordinateurs mais aussi est nécessaire pour prouver la convergence de certains algorithmes présentés par la suite.

à faire apparaître explicitement un bon minorant de l'optimum du problème traité. Ces reformulations (appelées aussi *Equivalence-Preserving Transformation*) transforment le réseau en un réseau équivalent, c'est à dire qu'elles ne changent pas le coût des affectations complètes. Elles agissent en déplaçant des coûts entre les fonctions, avec pour objectif de transférer les coûts vers des fonctions d'arité un (aussi appelées *coûts unaires*) ou zéro (notée f_\emptyset). f_\emptyset correspond à un coût constant que toute affectation doit payer et sert de minorant utilisé par DFBB [Larrosa, 2002]. L'avantage de cette approche de calcul de minorant est son incrémentalité : les reformulations sont effectuées une fois pour toute dans un même chemin de l'arbre de recherche. A noter que cette idée de reformulation a été étudiée dans d'autres cadres, en particulier en analyse d'images [Koval and Schlesinger, 1976, Werner, 2007], en optimisation de fonctions quadratiques pseudo-bouliennes [Hammer et al., 1984, Boros and Hammer, 2002] et dans les champs de Markov [Wainwright et al., 2005, Kolmogorov, 2006].

Les coûts unaires $f_x(a)$ peuvent être utilisés pour effectuer un filtrage des domaines en supprimant les valeurs dont le coût unaire plus le coût f_\emptyset dépassent le majorant de manière à ce que le réseau vérifie à la fin la propriété de noeud-cohérence (NC) :

$$\forall x \in X, \forall a \in d_x, f_x(a) + f_\emptyset < k$$

De plus, si tous les coûts d'une variable sont non nuls, il est possible de projeter le minimum de ces coûts dans f_\emptyset et de soustraire ce minimum des coûts unaires de façon à avoir un réseau équivalent et au final toujours au moins un coût unaire nul par variable :

$$\forall x \in X, \exists a \in d_x, f_x(a) = 0$$

ce qui combiné à NC donne la propriété NC* qui peut être établie en temps $O(nd)$ [Larrosa, 2002].

Le principal opérateur de reformulation défini dans [Schiex, 2000] est la projection d'une fonction de coût binaire², portant sur les variables x et y , vers un coût unaire lié à une valeur $a \in d_x$:

$$\begin{array}{l} \text{Opérateur AC}(f_{x,y}, x, a) \\ \left[\begin{array}{l} \alpha := \min_{b \in d_y} f_{x,y}(a, b) ; \\ f_x(a) := f_x(a) + \alpha ; \\ \text{foreach } b \in d_y \text{ do } f_{x,y}(a, b) := f_{x,y}(a, b) - \alpha \end{array} \right. \end{array}$$

Si l'on applique cet opérateur de façon chaotique jusqu'à atteindre un point fixe (dont l'existence est garantie), on obtient un réseau vérifiant la propriété d'arc-cohérence souple, *Soft Arc Consistency* (SAC) [Schiex, 2000]. Ce point fixe n'est cependant pas unique. Par exemple, dans le réseau à deux variables à domaine $\{a, b\}$ de la Figure 1.a) où les coûts unaires non nuls sont représentés dans des cercles et les coûts binaires non-nuls sur des arêtes, suivant l'application de $\text{AC}(f_{x,y}, x, a)$ ou de $\text{AC}(f_{x,y}, y, a)$ en premier, le réseau obtenu est respectivement celui de la Figure 1.b) pour lequel le minorant peut être augmenté de 1 (en déplaçant les deux coûts unaires $f_x(a)$ et $f_x(b)$ dans f_\emptyset pour vérifier la propriété NC*), ou celui de la Figure 1.c) dont le minorant reste inchangé.

Ainsi comme pour les cohérences locales dans les CSPs, il est possible de définir des propriétés de cohérence locale dans les WCSPs et de fournir un algorithme de renforcement de la propriété proposée.

Definition 1.4 (AC*) Une fonction de coût binaire est arc-cohérente s'il existe pour chaque affectation d'une de ses variables une affectation de l'autre variable telle

²A noter qu'il est inutile de soustraire α pour des coûts binaires supérieurs à k .

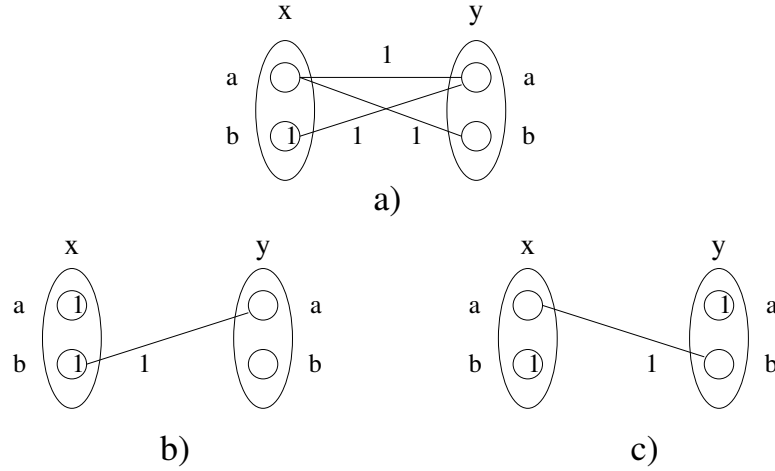


FIG. 1 – Trois réseaux équivalents. L'application chaotique de l'opérateur AC sur le réseau a) conduit à des reformulations différentes.

que le coût soit nul. Un WCSP binaire est arc-cohérent (Arc Consistency (AC^*)) si toutes ses fonctions de coût binaires sont arc-cohérentes et si toutes ses variables sont NC^* .

Un algorithme de renforcement de AC^* présenté dans [Larrosa, 2002, Larrosa and Schiex, 2004] a une complexité temporelle en $O(n^2d^2 + ed^3)$ et spatiale en $O(ed)$. Dans le cas d'un CSP ($f_\emptyset + 1 = k$), AC^* est équivalent à l'arc-cohérence classique. Une amélioration de AC^* consiste à combiner les coûts binaires et les coûts unaires dans le calcul du minimum à projeter :

Opérateur DAC($f_{x,y}, x, a$)

$$\left[\begin{array}{l} \alpha := \min_{b \in d_y} f_{x,y}(a, b) + f_y(b) ; \\ f_x(a) := f_x(a) + \alpha ; \\ \textbf{foreach } b \in d_y, c \in d_x \textbf{ do } f_{x,y}(c, b) := f_{x,y}(c, b) + \max(0, \alpha - f_{x,y}(a, b)) ; \\ \textbf{foreach } b \in d_y \textbf{ do } f_y(b) := f_y(b) - \max(0, \alpha - f_{x,y}(a, b)) ; \\ \textbf{foreach } b \in d_y \textbf{ do } f_{x,y}(a, b) := f_{x,y}(a, b) - \alpha \end{array} \right.$$

Ici, la reformulation peut nécessiter d'étendre les coûts unaires de la variable y vers $f_{x,y}$ pour pouvoir projeter sur la valeur $a \in d_x$ sans faire apparaître de coûts négatifs³. Par exemple, l'application de $DAC(f_{x,y}, x, a)$ sur le réseau de la Figure 1.c) donne la reformulation en Figure 1.b). Malheureusement, l'application répétée de cet opérateur dans le réseau peut ne jamais s'arrêter. C'est par exemple le cas pour un autre réseau à 2 variables à deux valeurs $\{a, b\}$ avec $f_x(a) = 1, f_{x,y}(b, b) = 1$ et les autres coûts étant nuls. La première opération appliquée de x vers y produit le réseau $f_y(b) = 1, f_{x,y}(a, a) = 1$, la seconde en sens inverse produit à nouveau le réseau d'origine.

Pour remédier à ce problème [Cooper, 2003] proposa d'utiliser un ordre total sur les variables de façon à n'appliquer ce second opérateur que dans le sens des variables décroissantes⁴. On parle alors d'arc-cohérence directionnelle (*Directional Arc Consistency* (DAC)) qui peut être établie avec la même complexité que AC dans

³La présence de coûts négatifs rendrait incorrect l'utilisation du minorant dans DFBB.

⁴Cet ordre peut correspondre à mettre au début les variables affectées en premier par DFBB ou encore choisir un ordre topologique dans le cas où le réseau est acyclique.

les CSP. DAC produit un minorant optimal sur les réseaux de fonctions de coût binaires ayant une structure en arbre, en choisissant un ordre topologique sur les variables. Une amélioration consiste à établir DAC dans le sens décroissant et SAC dans l'autre sens, ainsi que NC*, donnant la cohérence locale *Full Directional Arc Consistency* (FDAC*) [Larrosa and Schiex, 2003]. Un algorithme de renforcement de FDAC* a une complexité en temps $O(ed^3)$ et en espace $O(ed)$.

J'ai pu évaluer les performances de FDAC* dans le cadre des problèmes Max-SAT, un cas particulier des WCSPs où les domaines sont de taille deux et où il s'agit de satisfaire le plus grand nombre de clauses d'une formule booléenne. Mon travail a consisté à comparer plusieurs solveurs dédiés à Max-SAT ou à des formalismes voisins comme l'optimisation de formules pseudo-booléennes (*Pseudo-Boolean Optimization* (PBO))⁵ [Boros and Hammer, 2002] ou la *Programmation Linéaire en Nombre Entier* (PLNE)⁶ [Billionnet, 2007]. Les résultats ont montré une nette supériorité de DFBB avec maintien de FDAC* durant la recherche⁷ lorsque le ratio entre le nombre de clauses et le nombre de variables augmente (c-à-d. le cas

- * des problèmes très sur-contraints) [de Givry et al., 2003]. Le solveur *ToolBar*⁸
- * [Bouveret et al., 2004], fruit d'une collaboration entre l'INRA Toulouse et deux équipes à Barcelone, a été mis en domaine public. J'ai par la suite réécrit en C++ une nouvelle version appelée *ToulBar2* en m'inspirant de l'architecture modulaire des outils de programmation par contraintes. Le travail initié autour de *ToolBar* a été suivi d'un important effort de recherche autour de Max-SAT comme l'attestent les compétitions internationales de solveurs Max-SAT depuis 2006. Les minorants fondés sur la reformulation de problèmes comme dans les WCSPs ont été étendus
- * au cas de clauses pondérées d'arité quelconque [Larrosa and Heras, 2005] et repris dans la plupart des solveurs Max-SAT modernes [Heras et al., 2007, Argelich et al., 2007]⁹. Avec l'aide des Barcelonais, j'ai continué à collecter plusieurs benchmarks Max-SAT dont le problème de la *recherche d'une clique de cardinalité maximum dans un graphe* (MAXCLIQUE)¹⁰, incluant un problème de *calcul d'un score de similarité entre deux protéines* (PROTEIN)¹¹ [Larrosa et al., 2008].

L'efficacité de l'algorithme DFBB dépendant principalement de la qualité de son minorant, je me suis intéressé à étudier des cohérences locales de niveau supérieur à l'arc-cohérence et en particulier la *cohérence singleton* (voir [Debruyne and Bessière, 2001] dans le cas CSP) consistant à tester séparément toutes les affectations possibles d'une seule variable *singleton* à la fois et appliquer sur les sous-problèmes résultants un renforcement d'arc-cohérence FDAC*, méthode appelée *Singleton FDAC** dans [de Givry, 2004]. Cette approche permet un filtrage des valeurs plus

⁵Il s'agit d'un formalisme composé de variables booléennes et d'un ensemble de fonctions à valeurs réelles sur ces (sous-ensembles de) variables. L'objectif est de trouver une affectation complète dont la somme des valeurs des fonctions est minimum.

⁶Il s'agit de problèmes exprimés par des variables entières sur lesquelles portent un ensemble de contraintes linéaires et une fonction objectif linéaire également.

⁷C'est la première fois que l'approche WCSP était évaluée sur des problèmes à plusieurs centaines de variables. Il a fallu pour les résoudre plus efficacement mettre au point une heuristique de choix de variable inspirée de la littérature SAT (*Two-sided Jeroslow*), effectuant une estimation du coût moyen des coûts unaires, binaires et ternaires associés à chaque variable.

⁸<http://carlit.toulouse.inra.fr/cgi-bin/awki.cgi/SoftCSP>

⁹L'autre approche étant une reformulation de Max-SAT en SAT avec ajout de contraintes de cardinalité [Marques-Silva and Manquinho, 2008, Ansotegui et al., 2010].

¹⁰Ce problème NP-dur de la théorie des graphes non-orientés consiste à trouver le plus grand ensemble de sommets deux-à-deux adjacents (au sens de son cardinal) dans un graphe.

¹¹Le calcul du score de similarité entre deux protéines s'effectue à partir d'une connaissance de leur structure tridimensionnelle. Chaque protéine est décrite par une séquence d'acides aminés et un ensemble de relations binaires entre paires d'acides aminés qui sont proches l'un de l'autre dans la structure tridimensionnelle. Le problème est de trouver un couplage d'une sous-séquence d'acides aminés de la première protéine à une autre sous-séquence d'acides aminés de la seconde protéine qui maximise le nombre de relations binaires présentes à la fois dans les deux sous-séquences et telles que les acides aminés respectifs soient associés dans le couplage.

fort au prix d'une complexité accrue ($O(en^3d^5)$ en temps et $O(ed)$ en espace). Par contre la reformulation de problème ne peut pas s'appliquer simplement sans ajouter des fonctions de coût d'arité supérieure. Il est cependant possible de produire une reformulation sans ajout de fonction dans le cas de *Singleton NC** qui consiste à appliquer l'opérateur DAC avec un ordre des variables qui oriente toutes les fonctions binaires vers la variable singleton testée. Federico Heras, sous la direction de Javier Larrosa (UPC Barcelone), exprima cette idée sous la forme d'une propriété de cohérence locale, l'arc-cohérence existentielle (*Existential Arc Consistency* (EAC)), permettant la mise au point d'un algorithme efficace de renforcement de EDAC* (*Existential and Full Directional Arc Consistency* (EDAC*)) ayant une complexité en temps $O(ed^2 \max(nd, k))$ et en espace $O(ed)$ [de Givry et al., 2005c, Zytnicki et al., 2005]. *

Definition 1.5 (EAC) Une variable est arc-cohérente existentielle s'il existe une valeur dite existentielle de son domaine telle que son coût unaire soit nul et telle que pour toute fonction de coût binaire portant sur cette variable, il existe une valeur dans le domaine de l'autre variable avec un coût unaire nul et un coût binaire avec cette valeur existentielle nul aussi. Un WCSP binaire est arc-cohérent existentiel (EAC) si toutes ses variables sont arc-cohérentes existentielles.

DFBB avec maintien de EDAC* a obtenu des résultats comparables à ceux de la PLNE sur des instances difficiles d'un problème d'allocation d'entrepôts (*Uncapacitated Warehouse Location Problem* (WAREHOUSE))¹² [de Givry et al., 2005c] et est actuellement l'un des algorithmes les plus performants pour résoudre les WCSPs, comme l'atteste les bons résultats obtenus aux compétitions de solveurs Max-CSP en 2007 et 2008. *

Pour mieux traiter un problème soumis à l'INRA sur la *correction d'erreurs Mendéliennes de génotypage* (MENDEL)¹³ dans des pedigrees d'animaux d'élevage, problème qui s'exprime par des contraintes ternaires entre le génotype d'un enfant et celui de ses deux parents, je me suis intéressé à l'extension de EDAC* au cas de fonctions d'arité ternaire [Sánchez et al., 2008]. L'originalité est d'appliquer un opérateur DAC spécial sur la combinaison d'une fonction ternaire avec l'ensemble des fonctions binaires et unaires incluses dans sa portée sans avoir à restaurer l'arc-cohérence dans l'ordre inverse des variables, comme le fait FDAC* dans le cas binaire. De plus, une fonction de coût binaire peut être partagée par plusieurs fonctions ternaires, améliorant ainsi le transfert d'information entre les fonctions. Un algorithme de renforcement de EDAC* ternaire a pour complexité en temps $O(ed^3 \max nd, k)$ ¹⁴ et en espace $O(ed^2)$. Les fonctions binaires devant être gérées par des tables modifiables, cela limite les possibilités d'étendre cette approche à des arités plus grandes¹⁵. Ce travail a permis de résoudre efficacement des instances MENDEL réelles de grande taille à plus de 10000 variables (correspondant à un pedigree ovin complexe ayant 129516 individus sur 17 générations) avec un critère de parcimonie [de Givry et al., *

¹²Il s'agit d'un problème d'allocation de ressources : la sélection et l'attribution optimale d'entrepôts pour fournir un ensemble de magasins. La fonction objectif combine des coûts d'ouverture et de maintenance d'un entrepôt et des coûts de transports pour aller d'un entrepôt vers un magasin.

¹³Le problème consiste à supprimer le minimum d'observations (fonction objectif linéaire) parmi les génotypes observés d'un ensemble d'individus apparentés à un marqueur donné de manière à satisfaire les lois de l'hérédité énoncées par G. Mendel en 1865. Un génotype est une paire d'allèles (*i.e.* de valeurs) non orientée observable à l'aide d'un marqueur génétique. Le nombre d'allèles pour un marqueur varie entre deux (SNP) à une dizaine.

¹⁴Une autre définition de EDAC* pour des arités quelconques a été proposée dans [Lee and Leung, 2010] pour laquelle un problème de non-convergence de l'algorithme de renforcement a été observé et a nécessité une définition affaiblie.

¹⁵Une forme restreinte de EDAC* a été étendue au cas d'arités quelconques dans [Lee and Leung, 2010].

2005b, 2006b, Sánchez et al., 2008] et a été intégré dans la plate-forme *Quantitative Genetics* du département Génétique Animale de l'INRA (<https://qgp.jouy.inra.fr/>). Par contre avec un critère probabiliste, le problème reste difficile à résoudre (et hélas peu décomposable).

L'inconvénient de toutes ces approches est que la qualité du minorant dépend de l'ordre d'application des opérations AC et DAC (ainsi que l'ordre DAC des variables). Ces opérations sont appliquées de manière chaotique jusqu'à obtenir un point fixe. Il a été démontré dans [Cooper and Schiex, 2004] que trouver une séquence d'opérations AC/DAC qui maximise le minorant f_\emptyset est un problème NP-dur dans le cas de coûts entiers.

1.2.2 Schéma planifié de reformulations locales

- La difficulté de trouver une reformulation optimale (qui maximise f_\emptyset) peut être contournée par utilisation de coûts rationnels. Martin Cooper (IRIT Toulouse) a proposé une formulation en programme linéaire qui maximise f_\emptyset et dont la solution est un ensemble de reformulations locales simultanées effectuant des mouvements de coûts (positifs ou négatifs) entre des fonctions d'arité arbitraire et des coûts unaires (*Optimal Soft Arc Consistency* (OSAC) [Cooper et al., 2007]). Au final, les contraintes du système linéaire assurent que le réseau résultant a des coûts positifs. L'inconvénient de cette approche est la taille du système : $O(ed + n)$ variables et
- * $O(ed' + nd)$ contraintes linéaires avec r l'arité maximum des fonctions [Cooper et al., 2007]. En pratique, cette reformulation optimale peut s'avérer intéressante dans une phase de préprocessing mais il reste à régler la question de la gestion des coûts rationnels. Concrètement, deux approches sont possibles pour se ramener à des
 - * coûts entiers quitte à perdre en optimalité : dans [Cooper et al., 2007], nous avons utilisé la PLNE et donc une méthode de complexité temporelle exponentielle au pire cas et dans [Khemmoudj and Bennaceur, 2007] une méthode heuristique d'arrondi est proposée (*a priori* davantage sous-optimale mais beaucoup plus rapide).

Ce travail sur OSAC nous a permis de découvrir les liens existants avec la Recherche Opérationnelle, une version équivalente à OSAC (sans exploitation des contraintes dures) ayant été proposée en 1976 dans le contexte du traitement d'images [Schlesinger, 1976, Werner, 2007] et dans le cas de formules pseudo-bouliennes quadratiques (équivalent à Max-2SAT), un algorithme spécialisé de maximisation de flot produit également une reformulation optimale [Boros and Hammer, 2002]. La comparaison entre OSAC et cet algorithme reste à faire. Cependant il semble ne pas s'étendre au cas de domaines de taille supérieure à 2.

- Dans le but de trouver un algorithme de reformulation moins coûteux en temps, une version affaiblie de OSAC consiste à détecter si dans un WCSP il existe une séquence d'opérations de reformulations locales qui puisse augmenter le minorant f_\emptyset (sans garantie qu'au final le minorant soit optimum). Pour cela, il suffit d'appliquer l'arc-cohérence classique dans un CSP dérivé du WCSP d'origine, noté $Bool(P)$ (équivalent à remplacer chaque fonction de coût à l'exception de f_\emptyset par une contrainte de même portée avec comme affectations autorisées celles qui ont un
- * coût nul) [Cooper et al., 2008a,b, 2010].

Definition 1.6 (VAC) *Un WCSP P est arc-cohérent virtuel (Virtual Arc Consistency (VAC)) si le renforcement d'arc-cohérence dans le CSP $Bool(P)$ ne produit pas un domaine vide.*

Theorem 1.7 ([Cooper et al., 2008a]) *Il existe une séquence d'opérations locales (opérateurs AC/DAC) dans les coûts rationnels qui augmente f_\emptyset si et seulement si le problème n'est pas VAC.*

- * Un exemple décrit dans [Cooper et al., 2010] montre qu'il peut exister une infinité

de séquences augmentantes faisant croître de manière infinitésimale le minorant. Une collaboration avec Tomas Werner (Université de Prague) nous a permis de débloquent la situation et de mettre au point pour les WCSP binaires un algorithme approché, noté VAC_ϵ (dont la terminaison est assurée) relativement efficace en temps, $O(ed^2k/\epsilon)$ avec ϵ le coût rationnel minimum non nul, et en espace $O(ed)$. J’ai commencé à comparer nos travaux avec l’existant en Recherche Opérationnelle en prenant plusieurs benchmarks dont les problèmes WAREHOUSE et de couverture minimum d’ensembles (TAGSNP et *Steiner Triple Covering Problem* (STEINER))¹⁶ [Östergård and Vaskelainen, 2010]). Il s’avère que la qualité des minorants produits par VAC et OSAC est souvent proche de celle obtenue par la PLNE appliquée sur une formulation directe du problème, mais en un temps plus long¹⁷.

L’intérêt de VAC est de s’appuyer sur l’arc-cohérence classique dont on connaît de bonnes propriétés, en particulier son indépendance à l’ordre des opérations et ses classes traitables dont celle où la structure du réseau est un arbre (sans avoir à spécifier un ordre topologique sur les variables à l’instar de DAC, FDAC* et EDAC*) et celle des contraintes max-closed correspondant à des fonctions sous-modulaires en optimisation. Des exemples de fonctions sous-modulaires sont toutes les fonctions de coût unaires, les fonctions linéaires $\sum_i a_i x_i$, la fonction de distance Euclidienne $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ et la fonction de précédence pondérée $(x \geq y) ? (x - y)^r : +\infty$ avec $r \geq 1$. L’encadrement d’un stage à l’automne 2006 pour étudier cette classe m’a montré les limites de méthodes génériques exactes (fondées sur la recherche d’un flot maximum ou bien VAC) lorsque les domaines des variables sont grands comme c’est le cas pour la recherche de motifs structurés d’ARN (incluant des contraintes de précédence pondérées sous-modulaires) ou pour certains problèmes CSP temporels avec préférences (*Feedback Arc Set Problem* (FEEDBACK))¹⁸ que j’ai étudié dans [Cooper et al., 2010]. De plus, le fait que VAC préserve la structure du problème (pas d’introduction de nouvelles fonctions) peut conduire à des coûts rationnels réduisant la qualité des minorants presque de moitié sur certains problèmes tel que MAXCLIQUE. J’envisage de remédier à ce problème en autorisant l’introduction parcimonieuse et la prise en compte de fonctions d’arité supérieure (au moins ternaire) dans VAC. L’idée serait de choisir entre une reformulation qui préserve la structure mais introduit des coûts fractionnaires comme le fait VAC ou bien une reformulation de type élimination de variable avec compensation à l’instar de la résolution étendue à Max-SAT [Larrosa and Heras, 2005]. Le problème MAXCLIQUE continue d’être l’objet de travaux récents autour de Max-SAT [Heras and Larrosa, 2008, Li and Quan, 2010]. Il peut servir de référence pour orienter et comparer avec le travail d’extension de VAC.

1.2.3 Cohérence locale pour des domaines de grande taille

Un inconvénient des cohérences locales précédentes est leur complexité en espace qui dépend de la taille des domaines, en $O(ed)$. Dans le cas de problèmes ayant des très grands domaines, comme cela a été le cas sur une application de *recherche de motifs structurés de petits ARNs* (RNA)¹⁹ dans une séquence génomique (les

¹⁶Un système de triplets de Steiner consiste en un ensemble V de points et une collection de triplets de V , appelés blocs, tels que toute paire de points est contenue dans un bloc et un seul. Le problème de couverture associé est de sélectionner un nombre minimum de points couvrant tous les triplets.

¹⁷A noter qu’en PLNE le souci d’exactitude des méthodes est plus relatif que pour les WCSPs du fait de l’usage courant des nombres flottants. Il reste qu’un important travail d’optimisation du code de VAC (à l’instar de AC dans les CSP dynamiques) est à faire pour améliorer ses performances en pratique.

¹⁸Le problème consiste à supprimer le minimum d’arcs dans un graphe dirigé pour rendre le graphe acyclique.

¹⁹Un motif structuré est une séquence de nucléotides dans le génome d’une espèce qui contient des éléments structurels (des relations d’appariement entre deux parties de sa séquence) et des

domaines correspondant à plusieurs millions de nucléotides) [Thébault et al., 2006, Zytnicki et al., 2008], il devient nécessaire de travailler sur une représentation plus compacte des domaines à l’instar de ce qui est fait en programmation par contraintes (PPC). Au travers de la thèse de Patricia Thébault, dans le cadre CSP, j’ai aidé à la définition d’opérateurs de renforcement d’arc-cohérence aux bornes pour des contraintes dédiées à la recherche d’ARNs [Thébault et al., 2006]. Puis j’ai contribué lors de la thèse de Matthias Zytnicki à la définition d’une arc-cohérence aux bornes, permettant de filtrer les bornes des domaines dans les WCSPs [Zytnicki et al., 2009].

Definition 1.8 (BAC) *Une variable est arc-cohérente aux bornes si pour chacune des deux bornes de son domaine, la somme du minorant f_0 et du coût minimum de toutes les fonctions de coût portant sur la variable affectée à cette borne est inférieure au majorant k . Un WCSP P est arc-cohérent aux bornes (Bounds Arc Consistency (BAC)) si toutes ses variables sont arc-cohérentes aux bornes.*

BAC procure un filtrage des domaines similaire à ce que ferait Singleton NC* appliqué seulement aux bornes, mais sans augmenter f_0 . Une amélioration de BAC consiste à projeter le minimum de toute fonction de coût dans f_0 , donnant BAC^0 . Un algorithme de renforcement de BAC^0 a une complexité en temps $O(n^2 r^2 d^{r+1})$ et en espace $O(n + er)$ [Zytnicki et al., 2009]. Un gros avantage par rapport à AC*, outre sa meilleure complexité en espace, est sa confluence vers un unique point fixe quelque soit l’ordre d’application des opérations de renforcement. Dans le cas de fonctions binaires fonctionnelles, anti-fonctionnelles, ou semi-convexes, la complexité temporelle est en $O(n^2 d^2)$ et pour des fonctions binaires monotones ou convexes (comme les contraintes de précédence pondérée), cette complexité devient linéaire en d (en $O(n^2 d)$), ce qui s’avère particulièrement intéressant pour la recherche d’ARNs ou des problèmes de raisonnement temporel. Il a été prouvé que BAC^0 domine (au sens de produire un meilleur minorant) l’approche classique en PPC qui consiste à réifier les fonctions de coût (*i.e.* pour chaque fonction, ajout d’une variable reliée à la portée de la fonction et représentant le coût d’une affectation). J’ai pu tester BAC^0 sur un benchmark de *gestion des prises de vue d’un satellite agile* (BEP)²⁰ [de Givry and Jeannin, 2006] fourni par l’ONERA. BAC^0 est presque deux ordres de grandeur plus rapide que EDAC* et surpasse également l’approche réifiée [Zytnicki et al., 2009].

Un résumé des complexités polynomiales des algorithmes de renforcement de cohérence locale est donné en Table 1. Egalement y figurent les classes polynomiales de problèmes WCSPs pour lesquels un algorithme de renforcement est complet. Les cohérences locales sont ordonnées par puissance de filtrage croissante (à l’exception de DAC qui est incomparable à AC* et à BAC^0) en considérant qu’une cohérence A est plus forte qu’une autre cohérence B si une fois le renforcement de A effectué, le réseau vérifie A et aussi B , alors que l’inverse n’est pas vrai. En pratique, EDAC* est souvent le niveau de cohérence locale le plus efficace (à l’exception de BAC^0 pour les grands domaines) et VAC_ϵ peut s’avérer utile pour des problèmes très difficiles. A noter que les coûts unaires ainsi que la valeur existentielle EAC produits par

éléments de contenu (sous-séquence de nucléotides) connus caractérisant une famille d’ARNs. Il peut être décrit par un ensemble de variables indiquant la position des éléments du motif dans le génome et de contraintes exprimant les éléments structurels et de contenu. Le problème consiste à trouver toutes les occurrences d’un motif dans un génome éventuellement en autorisant d’avoir des motifs dégradés. Les petits ARNs sont des séquences de quelques dizaines de nucléotides à rechercher dans des génomes contenant jusqu’à plusieurs milliard de nucléotides.

²⁰Il s’agit d’un problème de maximisation de la sélection (fonction objectif linéaire) et de l’ordonnement des prises de vue d’un satellite sous contraintes de fenêtre temporelle pour chaque prise de vue et d’incompatibilité entre des prises de vue rapprochées. La date d’une prise de vue est discrétisée sur quelques centaines d’unité de temps. Des contraintes quaternaires disjonctives interdisent le chevauchement de deux prises de vue sélectionnées.

EDAC* permettent de guider les heuristiques de choix de variable (à égalité sur la taille des domaines, préférer une variable avec des coûts unaires importants) et de choix de valeur (choisir en premier la valeur EAC).

Cohérence locale	Complexité		Classes polynomiales
	en temps	en espace	
NC*	$O(nd)$	$O(nd)$	-
BAC ⁰	$O(n^2 d^3)$	$O(e)$	-
AC*	$O(n^2 d^2 + ed^3)$	$O(ed)$	-
DAC	$O(ed^2)$	$O(ed)$	Arbre
FDAC*	$O(end^3)$	$O(ed)$	Arbre
EDAC*	$O(ed^2 \max(nd, k))$	$O(ed)$	Arbre
VAC _ε	$O(ed^2 k/\epsilon)$	$O(ed)$	Arbre, fct. sous-modulaires
OSAC	$poly(ed + n)$	$poly(ed^2 + nd)$	Arbre, fct. sous-modulaires

TAB. 1 – Complexités en temps et en espace des algorithmes de renforcement de cohérence locale étudiés pour un réseau de fonctions de coût binaires (ayant n variables, d valeurs au plus par domaine, e fonctions de coût, avec un coût maximum (majorant du problème) k et $\epsilon \in]0, 1]$, le coût rationnel minimum non nul) et classes polynomiales traitées par ces cohérences locales.

1.2.4 Perspectives en reformulation de fonctions de coût globales

Récemment, Jimmy Lee et Mario Leung (Université de Hong-Kong) ont proposé une extension des cohérences locales (NC*,AC*,DAC,FDAC*,EDAC*²¹) à un ensemble de fonctions de coût globales qui peuvent s'exprimer sous la forme d'un problème de flot maximum de coût minimum dans un graphe dirigé particulier [Lee and Leung, 2009]. Un flot dans ce graphe doit correspondre à une unique affectation des variables de la fonction dans le WCSP. De plus chaque couple (variable,valeur) est représenté par un arc distinct dans le graphe permettant de pondérer cet arc par un coût (positif ou négatif) qui puisse être projeté ou étendu à un coût unaire correspondant externe à la fonction globale en temps constant. Les auteurs [Lee and Leung, 2009, 2010] ont intégré leur travail dans notre plate-forme de développement d'algorithmes **ToulBar2** (v0.9.3). Les fonctions de coût globales fondées sur une représentation en flot qui ont été implémentées dans **ToulBar2** sont AllDifferent, GlobalCardinality, Same et Regular. J'ai pu observer sur une instance pondérée du carré latin 6x6 (LATIN), sorte de Sudoku avec préférences (fonctions unaires) sur les cases, une diminution d'un facteur 100 du nombre de noeuds dans l'arbre de recherche par rapport à une représentation en extension binaire de la fonction de coût AllDifferent. La même instance exprimée avec l'approche réifiée en PPC ne termine pas en une demi-heure (résolveur CHOCO 2.1.1). L'utilisation d'une contrainte globale GlobalCardinalityCost [Régim, 2002] (implémentée dans SICS-TUS PROLOGUE 4.1.2) intégrant une partie des fonctions unaires améliore les performances de l'approche PPC mais reste en deçà de l'approche WCSP lorsque la taille du carré latin augmente (supérieure à 7x7). Le travail préliminaire de [Lee and Leung, 2009] ouvre la voie à davantage d'expressivité pour l'approche WCSP et à terme devrait pouvoir remplacer l'approche PPC classique. Une perspective de ce travail est l'intégration dans VAC des fonctions globales. Un domaine d'application privilégié est la fabrication d'emploi du temps comme par exemple pour le cas des infirmières²² [Métivier, 2010].

²¹Pour éviter des problèmes d'oscillation dans les mouvements des coûts unaires lorsque deux fonctions de coût partagent au moins deux variables, une forme simplifiée de EDAC* est proposée dans [Lee and Leung, 2010] qui utilise une partition des variables dans le voisinage d'une variable cible pour déterminer à quelle fonction est associée chaque coût unaire.

²²Cette application a été retenue dans le cadre d'un projet ANR Blanc (Ficolof 2011-2013) dans lequel je serais impliqué.

Lorsque les problèmes sont de grande taille et fortement structurés²³, il est possible de détecter à l'avance l'occurrence de sous-problèmes indépendants au cours de la recherche et éventuellement de mémoriser leur optimum pour éviter d'avoir à les recalculer. Ce travail est relativement orthogonal à celui des cohérences locales et permet aussi d'améliorer la qualité des minorants. Je présente les résultats de mon travail sur ce sujet dans la section suivante.

1.3 Exploitation de la structure d'un réseau

Dans la section 1.2, nous avons montré des mécanismes d'inférence fondés sur la reformulation locale en problème équivalent (via les opérateurs AC et DAC). Ce type d'inférence est incomplet au sens où le minorant produit n'est pas toujours égal à l'optimum du problème, à l'exception des problèmes sous-modulaires ou ayant une structure arborescente dans le cas de certaines cohérences locales vues précédemment. Il existe un mécanisme d'inférence exact appelé algorithme d'élimination de variable (*Variable Elimination* (VE) [Bertelé and Brioshi, 1972, Dechter, 1999]). Ce mécanisme repose sur deux opérateurs : la jointure et la projection de fonctions²⁴. On note \bowtie la **jointure** (ou combinaison) qui permet de combiner l'information de plusieurs fonctions en une seule. La jointure de deux fonctions f_S et f_T donne une nouvelle fonction portant sur $S \cup T$ telle que :

$$f_S \bowtie f_T(l) = f_S(l[S]) + f_T(l[T]) \quad \forall l \in \text{affectations}(S \cup T)$$

avec $l[S]$, la projection de l'affectation l sur les variables S .

La projection d'une fonction f_S sur un sous-ensemble de ses variables $T \subset S$ donne une nouvelle fonction portant sur T définie par :

$$f_S[T](t) = \min_{l \in \text{affectations}(S) \text{ t.q. } l[T]=t} f_S(l) \quad \forall t \in \text{affectations}(T)$$

L'algorithme d'élimination de variable procède par étape en modifiant le réseau d'origine. A chaque étape, une variable du réseau courant est choisie et la jointure des fonctions portant sur cette variable est effectuée. La projection du résultat de la jointure sur l'ensemble des variables voisines de la variable choisie est ajoutée au réseau. Enfin la variable est supprimée du réseau ainsi que toutes les fonctions portant dessus. A la fin, le réseau ne contient aucune variable et seulement une fonction d'arité nulle, f_\emptyset , égale à l'optimum.

Function VE($(X, D, F) : \text{WCSP}$) : entier

```

while  $X \neq \emptyset$  do
    Choisir une variable  $x_i \in X$  ;
     $F_i := \{f_S \in F \mid x_i \in S\}$  ;
     $V_i := \bigcup_{f_S \in F_i} S$  ;
     $f_i := \bowtie F_i[V_i \setminus \{x_i\}]$  ;
     $X := X \setminus \{x_i\}$  ;
     $F := F \setminus F_i \cup \{f_i\}$  ;
return  $f_\emptyset$  ;

```

En conservant les fonctions intermédiaires f_i , il est possible de reconstruire une solution optimale en temps polynomial. La complexité temporelle et spatiale de

²³Ce qui interdit en particulier la présence de fonctions de coût globales portant sur l'ensemble des variables.

²⁴L'opérateur AC (resp. DAC) peut être redéfini par $f_{x,y}[x]$ (resp. $(f_{x,y} \bowtie f_y)[x]$), suivi d'une opération de compensation pour préserver l'équivalence.

VE est exponentielle en la taille du plus grand voisinage V_i rencontré lors des étapes d'élimination. Ce paramètre est appelé *largeur induite* du réseau selon l'ordre d'élimination des variables. Il est inférieur ou égal au nombre de variables du problème et donc fournit une complexité théorique en temps meilleure que DFBB au prix d'une complexité en espace plus grande. Trouver un ordre d'élimination de largeur induite minimum est un problème NP-difficile. Cependant il existe de *bonnes* heuristiques en temps polynomial.

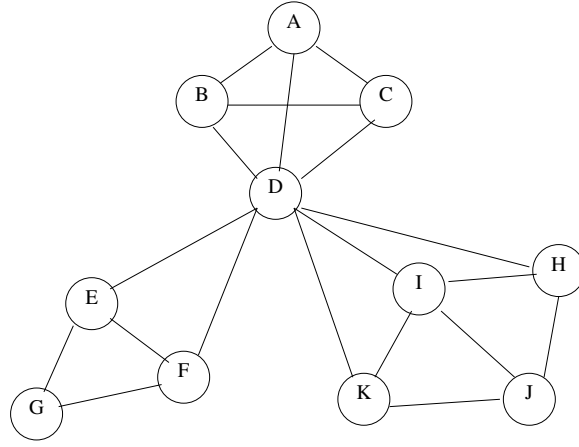
Plutôt que d'éliminer une seule variable à la fois, il peut être plus judicieux pour des questions d'espace mémoire d'en éliminer plusieurs d'un coup. C'est ce qui est fait dans l'algorithme d'élimination d'arbre de clusters (*Cluster Tree Elimination* (CTE) [Dechter and Pearl, 1989]), également appelé arbre de jointure dans les modèles graphiques probabilistes [Lauritzen and Spiegelhalter, 1988, Shafer and Shenoy, 1990]. L'idée repose sur l'exploitation d'une décomposition du problème en arbre de clusters.

Definition 1.9 (TREEDEC) Une décomposition arborescente d'un WCSP (X, D, F) est définie par un couple (C, T) où C est un ensemble fini de clusters, chaque cluster étant un sous-ensemble de variables ($\forall C_e \in C, C_e \subseteq X$), couvrant toutes les variables ($\bigcup_{C_e \in C} C_e = X$) et toutes les fonctions ($\forall f_S \in F, \exists C_e \text{ t.q. } S \subseteq C_e$), et T est un arbre, les noeuds étant les clusters et les arêtes formant un graphe acyclique connexe tel que pour toute variable $x \in X$, l'ensemble des clusters qui contiennent cette variable forme un sous-graphe acyclique connexe dans T .

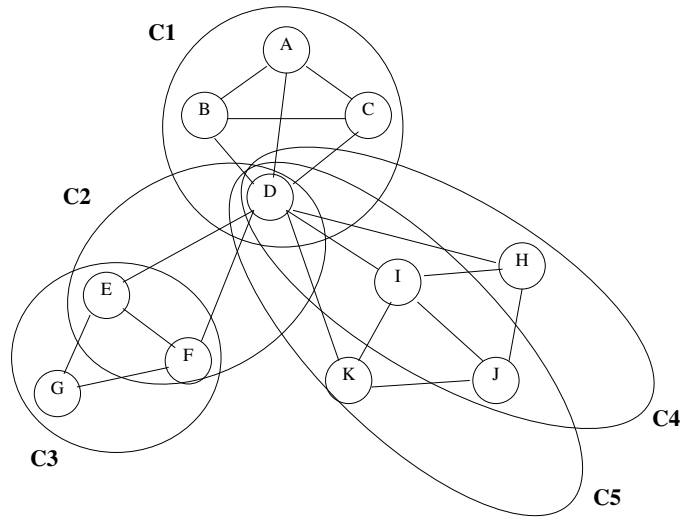
En choisissant d'enraciner T à un cluster racine $C_1 \in C$, on peut définir $Pere(C_e)$ (resp. $Fils(C_e)$), le cluster parent (resp. les clusters fils) de C_e dans T . On appelle *séparateur* de C_e le sous-ensemble de variables $S_e = C_e \cap Pere(C_e)$. L'ensemble des *variables propres* d'un cluster C_e est noté $V_e = C_e \setminus S_e$. Chaque fonction de coût $f_S \in F$ est associée à un unique cluster $C_e \supseteq S$ le plus proche de la racine dans T . Un exemple de décomposition arborescente est donné dans la Figure 2. Le problème original est un WCSP binaire à 11 variables représenté par le graphe de la Figure 2(a). Il est ici recouvert par 5 clusters indiqués en Figure 2(b). L'arbre de clusters est donné en Figure 2(c). La largeur d'arbre est égale à 4. Les variables propres du cluster C_4 sont $V_4 = \{I, J, H\}$ et le séparateur avec son cluster père C_1 est $S_4 = \{D\}$. C_4 contient les fonctions de coût $\{f_{DI}, f_{DH}, f_{IJ}, f_{HI}, f_{HJ}\}$.

L'algorithme CTE consiste à éliminer les clusters en partant des feuilles de l'arbre de décomposition et en remontant au cluster racine. L'élimination d'un cluster consiste à faire la jointure de toutes les fonctions associées à ce cluster et à projeter le résultat sur son séparateur. La complexité temporelle est exponentielle en la taille du plus grand cluster appelée largeur d'arbre. La complexité spatiale est exponentielle en la taille du plus grand séparateur, donc inférieure à la largeur d'arbre. A partir d'un ordre d'élimination de VE, il est possible de construire une décomposition arborescente dont la largeur d'arbre soit égale à la largeur induite (l'inverse étant vrai aussi). Il en résulte que la largeur d'arbre minimum est égale à la largeur induite minimum.

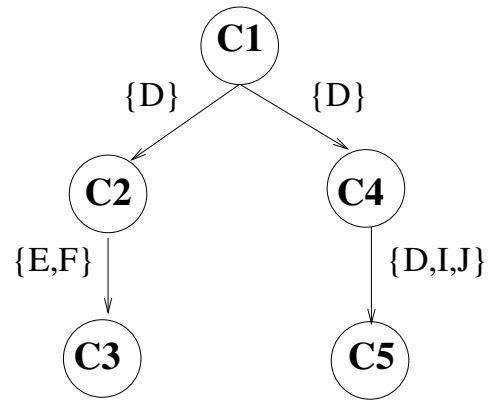
Dans les deux cas, VE et CTE ont une complexité pratique identique à leur complexité théorique ce qui les rend peu exploitables en pratique à l'exception du cas où la largeur d'arbre est très petite. Divers algorithmes hybrides combinant une recherche arborescente et VE ou CTE dans le but d'obtenir des algorithmes plus efficaces en pratique, capables d'exploiter la structure des problèmes, et ayant de bonnes garanties théoriques ont été proposés [Darwiche, 2001, Dechter, 2003, Terrioux and Jegou, 2003, Bacchus et al., 2003]. La Figure 3 replace la famille des méthodes hybrides à la croisée des deux grandes familles de méthodes d'optimisation combinatoire fondées soit sur l'exploration d'un espace de recherche par une recherche arborescente ou une recherche locale, soit par des méthodes d'inférence,



(a) Problème original



(b) Recouvrement en clusters



(c) Arbre des clusters (enraciné en C1) et séparateurs

FIG. 2 – Exemple de décomposition arborescente d'un graphe à 11 sommets.

dans les deux cas, les méthodes pouvant être exactes ou approchées (*i.e.* sans garantie d’optimalité).

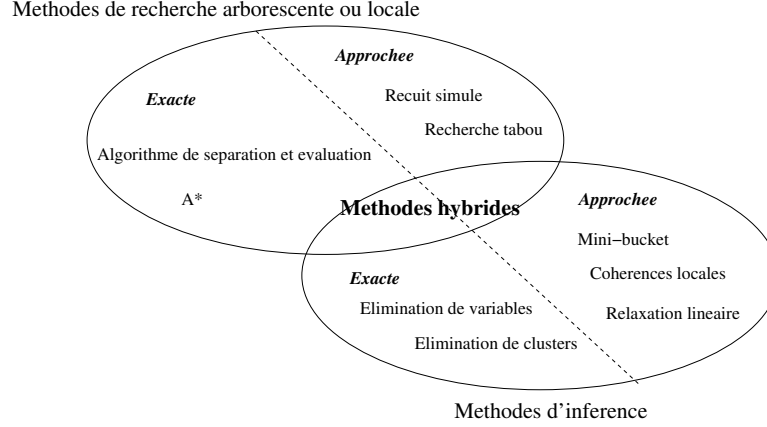


FIG. 3 – Méthodes hybrides.

Je présente mes travaux sur ces hybridations dans les sections suivantes. Ma motivation vient du fait que plusieurs problèmes et benchmarks rencontrés à l’INRA et durant ma thèse ont un graphe primal²⁵ fortement structuré (présence de *modules* faiblement reliés entre eux), comme le montre la Figure 4 pour le problème de correction d’erreurs de génotypage MENDEL [Sánchez et al., 2008], le problème de la * *sélection de marqueurs représentatifs d’une population d’individus* (TAGSNP)²⁶ en * *génétique* [Sanchez et al., 2009], le problème d’*allocation de fréquence à des liens* * *radios* (CELAR)²⁷ [Cabon et al., 1999], ou encore le problème de *sélection des prises* * *de vue d’un satellite d’observation terrestre* (SPOT5)²⁸ [Bensana et al., 1999]. Une telle structure se prête *a priori* bien aux techniques d’élimination de variable et de cluster.

1.3.1 Hybridation de la recherche avec l’élimination de variables

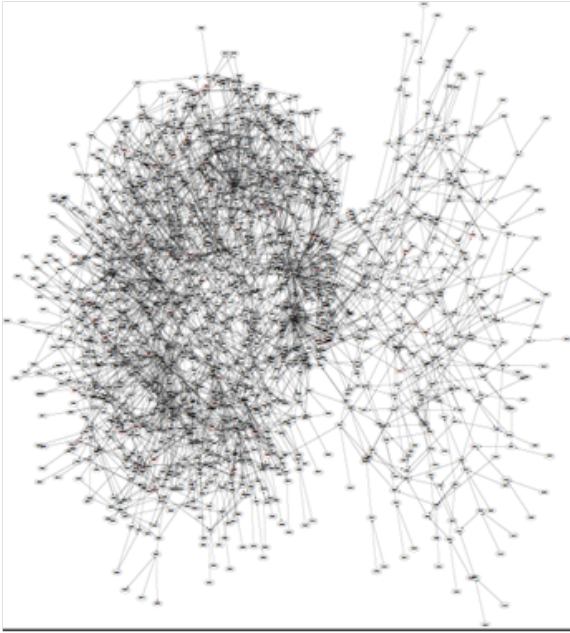
L’idée proposée initialement dans l’algorithme *Boosting Search with Variable Elimination* (DFBB-VE) [Larrosa, 2000] est d’introduire dans un algorithme DFBB une élimination à la volée des variables ayant un petit *degré* (nombre de variables qui sont reliées à la variable à éliminer par une fonction de coût). Typiquement, l’utilisation d’un degré maximum égal à 2 donnant l’algorithme DFBB-VE(2) a pour conséquence de ne créer que des fonctions de coût binaires qui sont bien propagées

²⁵Ce graphe contient un noeud par variable du WCSP et chaque fonction de coût est remplacée par une clique sur les variables de sa portée dans le graphe.

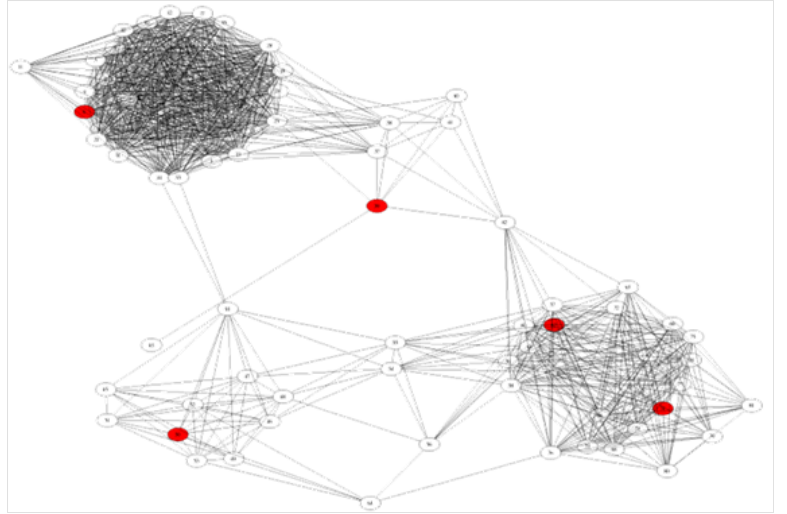
²⁶Le problème consiste à sélectionner un nombre minimum de marqueurs génétiques SNP fortement corrélés au reste des marqueurs pour une petite population d’individus génotypés. Cette compression de l’information permet ensuite de génotyper à moindre coût, car seulement sur un faible nombre de marqueurs sélectionnés (tagSNP), pour un grand échantillon. Ce problème peut se représenter par un problème de *Set Covering* avec des critères secondaires quadratiques de dispersion entre les tagSNPs et de taux de couverture des non-tagSNPs maximaux.

²⁷Le problème consiste à sélectionner une fréquence parmi un ensemble fini de fréquences disponibles à un ensemble de liaisons radios prédéterminées. Des fonctions de coût binaires pénalisent les possibles interférences entre deux liens radios. Un benchmark issu d’une application militaire du Centre d’Electronique de l’Armement (CELAR) a été distribué en 1993 et fut étudié dans un projet européen de comparaison de méthodes d’optimisation combinatoire (CALMA).

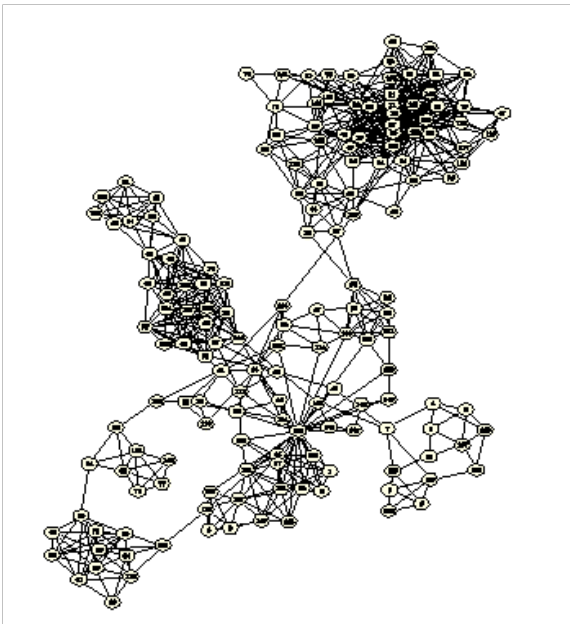
²⁸Il s’agit d’un problème fourni par le CNES et l’ONERA de maximisation de la sélection (fonction objectif linéaire) des prises de vue du satellite SPOT5 sous contraintes d’incompatibilité entre des prises de vue rapprochées utilisant un même instrument optique. Le domaine des variables correspond au choix de l’instrument ou à la non-sélection d’une prise de vue.



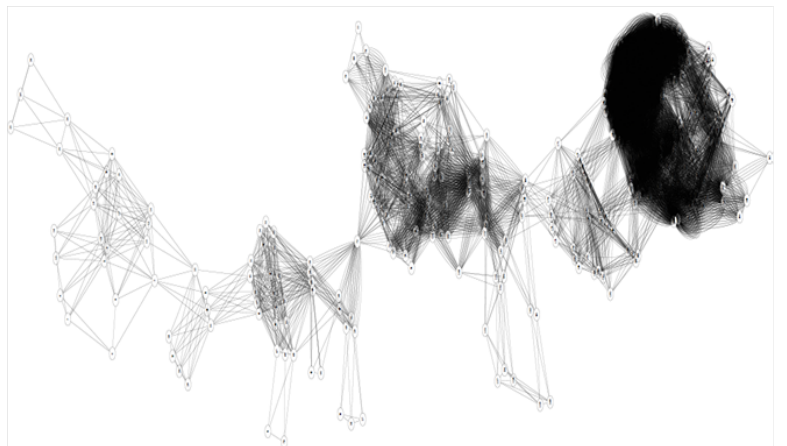
(a) Pedigree langladeM7



(b) HapMap chr01



(c) CELAR scen07r



(d) SPOT5 509

FIG. 4 – Exemples de problèmes fortement structurés.

par les cohérences locales. De plus il a été démontré que l'ordre d'élimination de variables de degré au plus 2 n'avait pas d'importance vis à vis du réseau final résultant [Dechter, 2003]. Cette élimination permet de résoudre les réseaux ayant une largeur d'arbre minimum égale à 2, ce qui inclut les structures en arbre. Par conséquent, DFBB-VE(2) est comparable à une méthode de coupe-cycle d'autant plus que les heuristiques de choix de variables telles que $\frac{dom}{ddeg}$ ou $\frac{dom}{wdeg}$ [Lecoutre et al., 2009] ont naturellement tendance à choisir des variables fortement connectées conduisant à de *bons* ensembles coupes-cycles. Dans l'exemple de la Figure 2(a), l'affectation de la variable D crée 3 sous-problèmes indépendants, $\{A, B, C\}$, $\{E, F, G\}$ et $\{H, I, J, K\}$, ayant chacun une largeur d'arbre égale à 2.

Avec l'aide de Marti Sanchez, nous avons mis en oeuvre cet algorithme en intégrant l'élimination à la volée avec la propagation des fonctions (pouvant par filtrage affecter des variables et donc réduire le degré d'autres variables) jusqu'à obtention d'un point fixe, ceci à chaque noeud de la recherche. Cette élimination à la volée a permis de gagner jusqu'à deux ordres de grandeur sur certains problèmes dont MENDEL [Sánchez et al., 2008]. En pratique, DFBB-VE(2) avec maintien de EDAC* et une heuristique dynamique de choix de variable basée sur les conflits [Lecoutre et al., 2009] s'avère la méthode la plus robuste sur l'ensemble de nos applications et benchmarks. Elle obtient les meilleurs résultats aux compétitions de solveurs Max-CSP 2007 & 2008. *

1.3.2 Hybridation de la recherche avec l'élimination de clusters

Ici nous avons étudié deux stratégies d'hybridation : soit la recherche arborescente est maître et la décomposition arborescente lui sert de guide, soit l'élimination de clusters est maître et la recherche arborescente lui sert à éliminer les clusters.

Recherche arborescente guidée par la décomposition arborescente.

Au lieu d'éliminer les clusters des feuilles vers la racine de la décomposition arborescente, l'idée ici est d'appliquer une recherche arborescente en partant du cluster racine et en allant vers les clusters feuilles. L'idée principale est que l'affectation des variables d'un cluster sépare les sous-problèmes enracinés aux clusters fils. Chaque sous-problème peut être résolu indépendamment des autres. Par l'exemple, l'affectation des variables du cluster C_1 de la Figure 2(c) donne deux sous-problèmes indépendants ($C_2 \cup C_3$ et $C_4 \cup C_5$). Ainsi la complexité en temps ne dépend plus que de la profondeur de la décomposition arborescente (nombre maximum de variables différentes de la racine à une feuille de la décomposition) qui est bornée par sa largeur d'arbre fois le logarithme du nombre de variables du problème. La complexité en espace reste la même que DFBB. C'est ce qui est fait dans des algorithmes comme Pseudo-Tree Search [Larrosa et al., 2002] ou AND/OR tree search [Marinescu and Dechter, 2009a]. Dans l'exemple de la Figure 2(c), la profondeur de la décomposition est 8 ($|C_1 \cup C_4 \cup C_5|$).

Il est possible d'obtenir les mêmes complexités que CTE si l'on mémorise l'optimum des sous-problèmes résolus dans les séparateurs. En Figure 2(c), pour chaque affectation de D , on mémorise l'optimum des sous-problèmes enracinés en C_2 et en C_4 . A la différence de CTE, cette mémorisation se fait de manière paresseuse en fonction de l'ordre d'exploration de l'arbre de recherche et des coupes de DFBB (par exemple, certaines affectations de D seront rejetées du fait des affectations précédentes sur $\{A, B, C\}$), conduisant à des complexités spatiales en pratique bien meilleures. C'est le cas de l'algorithme *Backtrack with Tree Decomposition* (BTD) étendu à l'optimisation dans [Jégou and Terrioux, 2004] et AND/OR graph search [Marinescu and Dechter, 2006, 2009b]. La différence principale entre ces deux approches étant que BTD exploite un minorant fourni par les cohérences locales lui permettant de choisir dynamiquement l'ordre d'affectation des variables à l'intérieur des clusters, tandis

que AND/OR search utilise un minorant obtenu par une élimination de variable approchée (Mini-Bucket [Dechter and Rish, 2003]) conduisant à un ordre statique d’affectation des variables.

Nos travaux ont permis de réduire encore un peu plus la consommation mémoire de l’algorithme BTM et aussi souvent les temps de calcul en évitant de résoudre à l’optimum des sous-problèmes pour lesquels même une solution optimale ne suffirait pas à trouver une meilleure solution pour le problème complet du fait des décisions effectuées au-dessus dans l’arbre de recherche. Cela veut dire aussi qu’un sous-problème peut être résolu plusieurs fois, à chaque fois un meilleur minorant est enregistré jusqu’à trouver et prouver son optimum. La complexité théorique de ce nouveau BTM est k (le majorant) fois celle de l’ancien [de Givry et al., 2006a].

- * De plus nous avons montré comment exploiter les cohérences locales comme EDAC* dans BTM. Pour cela il est nécessaire d’enregistrer les mouvements de coût possibles entre clusters de manière à corriger l’information stockée dans les séparateurs. L’effacement de valeurs doit également être traité avec prudence pour conserver des minorants valides. Finalement le minorant utilisé par ce nouveau BTM exploite de manière conjointe (on retient récursivement le maximum dans le sous-arbre courant de clusters) le minorant f_\emptyset fourni par les cohérences locales et l’information stockée dans les séparateurs.

Cette approche a permis de résoudre une instance réputée difficile du CELAR (scen06) en 3h sur un PC 2.4 GHz en ne mémorisant que 326 minorants dans les séparateurs comparé à environ 1 million pour CTE (pour un temps de calcul comparable de 8h sur un DEC Alpha 500MP) [Koster, 1999]. La même instance avait été résolue précédemment (seulement sa preuve d’optimalité) en 26 jours sur un SUN UltraSparc 167 MHz (resp. 3 jours sur un SUN Sparc 2) par l’algorithme *Russian Doll Search* (RDS)²⁹ [Verfaillie et al., 1996] (resp. PFC-MRDAC [Larrosa et al., 1998]) sur une décomposition faite à la main [de Givry et al., 1997]. Parallèlement l’amélioration du schéma de branchement (*branchement binaire dichotomique*) et des heuristiques dynamiques de choix de variable (*last conflict* et *weighted degree* [Lecoutre et al., 2009]) intégrés dans BTM a permis de faire des gains supplémentaires. La preuve d’optimalité de l’instance scen06 met moins d’une minute sur un PC Xeon 2.66 GHz avec la méthode BTM de ToulBar2 (v0.9.3) en exploitant une décomposition arborescente fournie par l’heuristique *Maximum Cardinality Search* (MCS).

Pourtant les résultats étaient décevants comparés à ceux de l’algorithme RDS sur un benchmark de sélection des prises de vue d’un satellite d’observation terrestre (SPOT5) et à ceux de AND/OR graph search exploitant un autre minorant que celui fourni par les cohérences locales (Mini-Bucket [Dechter and Rish, 2003]) sur certains benchmarks des modèles graphiques probabilistes discrets.

- * Une généralisation de RDS à BTM, donnant l’algorithme *Russian Doll Search with Tree Decomposition* (RDS-BTM) [Sanchez et al., 2009] améliora le pouvoir de coupe de la méthode au prix d’un preprocessing relativement peu coûteux en pratique³⁰, toujours avec une bonne consommation mémoire et des complexités théoriques équivalentes à BTM. Ce preprocessing consiste à résoudre une relaxation de tous les sous-problèmes de la décomposition arborescente en commençant par les feuilles et en remontant jusqu’à la racine. La relaxation d’un sous-problème enraciné en

²⁹L’algorithme RDS résout une succession de problèmes emboîtés en partant d’un sous-problème contenant une seule variable et en ajoutant une nouvelle variable à chaque itération. Les optima des sous-problèmes résolus précédemment servent de minorants pour la résolution du problème courant. Cette stratégie impose un ordre statique de choix de variables.

³⁰Il reste à tester une amélioration de la gestion du cache de RDS-BTM autorisant des affectations partielles des séparateurs. Cela peut avoir un impact important surtout dans le cas Max-SAT où l’effacement des clauses satisfaites rend inutile l’affectation des variables devenues déconnectées du problème.

un cluster C_e consiste à supprimer les variables de son séparateur S_e et les fonctions qui portent dessus. Ainsi on obtient un minorant quelque soit l'affectation du séparateur. Sur l'exemple de la Figure 2(c), RDS-BTD résout successivement les sous-problèmes relaxés enracinés en C_3, C_2, C_5, C_4 et enfin le problème complet. Le sous-problème relaxé enraciné en C_4 contient les variables $C_4 \cup C_5 \setminus S_4 = \{I, J, H, K\}$ et les fonctions de coût $\{f_{IJ}, f_{HI}, f_{HJ}, f_{IK}, f_{KJ}\}$. Chaque résolution est effectuée par BTD ce qui permet de commencer à mémoriser dans les séparateurs lors de cette phase de préprocessing. Le minorant utilisé combine trois informations : le minorant f_\emptyset des cohérences locales, l'information des séparateurs et l'information de RDS. RDS-BTD généralise les algorithmes DFBB, BTD, Pseudo-Tree RDS [Larrosa et al., 2002] et RDS suivant la décomposition arborescente fournie en entrée et l'utilisation des informations mémorisées.

Pour la première fois, la preuve d'optimalité de l'instance scen07 du CELAR était obtenue. RDS-BTD résout entièrement les instances SPOT5 (bien qu'ici l'interaction entre cohérence locale et RDS soit délicate). RDS-BTD s'avère particulièrement efficace sur des problèmes dont la structure est modulaire comme pour le CELAR ou pour la sélection de marqueurs représentatifs d'une population d'individus en génétique (TAGSNP) [Sanchez et al., 2009]³¹. *

CTE avec élimination de clusters par recherche arborescente

Si l'on dispose d'un grand nombre de processeurs (c'est le cas de la Génomole Toulouse équipée d'une machine de 400 cœurs), il est alors intéressant de résoudre séparément chaque cluster pour chaque affectation de son séparateur. Cette version parallèle de CTE profite d'une résolution efficace des clusters par un algorithme de séparation et évaluation [Allouche et al., 2010b]. L'instance scen08 du CELAR ouverte depuis 17 ans a ainsi pu être résolue à l'optimum. Ma grande surprise a été la variation modérée des temps de calcul entre les différentes affectations de séparateur (par exemple, sur 70000 résolutions d'un même cluster, le temps a varié entre 11' et 300') évitant ainsi des questions d'équilibrage de charge. De plus, il s'avère possible d'appliquer cette approche même à des tailles de séparateur importantes (environ une quinzaine de variables) comme cela a été le cas sur certaines instances d'un benchmark d'analyse de liaison génétique (LINKAGE) donnant parfois de meilleurs résultats dans le cas mono-processeur comparés à RDS-BTD. *

1.3.3 Choix d'une bonne décomposition arborescente

Il reste la question de trouver les modules dans un problème, c'est-à-dire une décomposition arborescente avec des petits séparateurs et un relativement faible nombre de clusters. Les heuristiques de décomposition arborescente habituellement utilisées (MCS, Min-Fill) ont montré leurs limites pour identifier ces séparateurs en particulier sur les instances TAGSNP. Ce problème d'identification reste ouvert³² et fera l'objet de futures recherches. Une récente définition de la modularité [Newman, 2006] pourrait servir de base à la construction de bonnes décompositions arborescentes. Sonia Cafieri a appliqué un algorithme exact de maximisation de la modularité [Aloise et al., 2010] sur l'instance CELAR scen07r de la Figure 4 et trouve 9 clusters (en moins de 2 min) proches de ceux obtenus avec l'heuristique MCS et une limite sur la taille des séparateurs choisie à la main [Sanchez et al., 2009]. Une autre approche capable de traiter des très grands graphes est fondée sur le partitionnement de graphes, à voir les récents travaux [Bichot, 2007]. *

³¹Sur ce problème, il s'avère que la programmation linéaire en nombre entier donne les meilleurs résultats [Allouche et al., 2010a].

³²Il existe bien des méthodes exhaustives d'identification des petits séparateurs [Berry et al., 2000, Ndiaye, 2007] difficilement applicables en pratique car ne prenant pas en compte la taille résultante des clusters qui doit rester suffisamment grande pour limiter le nombre total de clusters.

Nous avons pu observer qu'en jouant sur la taille limite des séparateurs retenus issus de l'heuristique MCS, il existait un réglage optimal dépendant de l'instance considérée. Ce réglage dépend du gain en réduction de la largeur d'arbre par rapport à la perte en augmentation de la taille des séparateurs [Allouche et al., 2009].

En conclusion de cette partie sur l'hybridation de la recherche avec l'élimination de clusters, il s'avère difficile d'exploiter à la BTM la structure d'un problème en général (lors des compétitions de solveurs, **ToulBar2/DFBB-VE(2)** était globalement plus performant que **ToulBar2/BTD**). Il est nécessaire de définir des caractéristiques de problème permettant de sélectionner de manière automatique la méthode la plus judicieuse. Par exemple, si le problème a une faible largeur d'arbre et des petits domaines alors appliquer VE, s'il est modulaire avec des petits séparateurs et des fonctions de faible arité alors appliquer RDS-BTD, s'il dispose d'un faible nombre de variables *hubs* très connectées alors mieux vaut appliquer DFBB-VE(2). Si la structure est une grille comme souvent en image ou pour LINKAGE, alors l'approche AND/OR search avec Mini-Bucket semble meilleure. Il serait intéressant d'étudier d'autres relaxations que Mini-Bucket exploitant de façon dynamique la structure d'un problème relaxé (de petite largeur d'arbre) à l'instar des travaux de thèse de Matthew Kitching [Kitching, 2010].

Pour conclure les travaux sur les méthodes hybrides, la Table 2 rappelle les complexités en temps et en espace des différentes méthodes exactes présentées dans ce rapport (voir [Bacchus et al., 2009] pour une analyse plus détaillée).

Méthode exacte	Complexité	
	en temps	en espace
DFBB	$O(\exp(n))$	$O(n)$
VE	$O(n \exp(w))$	$O(n \exp(w))$
CTE	$O(n \exp(w))$	$O(n \exp(s))$
DFBB-VE(i) [Larrosa, 2000]	$O(\exp(n))$	$O(n \exp(i))$
Pseudo-Tree Search [Freuder and Quinn, 1985, Larrosa et al., 2002]	$O(n \exp(w \log(n)))$	$O(n)$
AND/OR tree search [Marinescu and Dechter, 2005, 2009a]	$O(n \exp(w \log(n)))$	$O(n)$
AND/OR graph search [Marinescu and Dechter, 2006, 2009b]	$O(n \exp(w))$	$O(n \exp(s))$
BTD [Jégou and Terrioux, 2004, de Givry et al., 2006a]	$O(k n \exp(w))$	$O(n \exp(s))$
RDS-BTD [Sanchez et al., 2009]	$O(k n \exp(w))$	$O(n \exp(s))$

TAB. 2 – Complexités en temps et espace des méthodes exactes étudiées pour un réseau de fonctions de coût à n variables ayant une largeur d'arbre w et des séparateurs contenant au plus s variables ($s \leq w \leq n$) et un majorant du problème k .

2 Perspectives d'application dans les réseaux bayésiens

Dans cette seconde partie, j'introduis le cadre des réseaux bayésiens et ouvre des perspectives de recherche qui constituent mon projet de recherche. Mon objectif est d'appliquer et d'étendre les techniques développées pour les WCSPs afin d'essayer de mieux traiter les problèmes d'optimisation, d'inférence et d'apprentissage dans les réseaux bayésiens.

2.1 Définition du cadre des réseaux bayésiens

Définition 2.1 (Réseau bayésien) Un réseau bayésien (Bayesian Network (BN)) discret est défini par un triplet (X, D, P) :

- X , un ensemble de n variables aléatoires discrètes, $X = \{x_1, \dots, x_n\}$;
- D , un ensemble de domaines finis de valeurs, chaque variable $x_i \in X$ pouvant prendre une valeur de son domaine $d_{x_i} \in D$;
- P , un ensemble de n probabilités conditionnelles, $P = \{P_1, \dots, P_n\}$, chacune $P_i = \mathbb{P}(x_i | Pa(x_i))$ étant définie conditionnellement aux variables $Pa(x_i) \subseteq X \setminus \{x_i\}$ telles que le graphe dirigé composé des noeuds X et des arcs $\{(x_j, x_i) | x_i \in X, x_j \in Pa(x_i)\}$ soit acyclique.

Un réseau bayésien représente une distribution de probabilité sur X dont la loi de probabilité jointe se factorise de la manière suivante :

$$\mathbb{P}(X) = \prod_{i=1}^n \mathbb{P}(x_i | Pa(x_i))$$

On distingue principalement trois requêtes sur ce réseau :

- Le problème Most Probable Explanation (MPE) consiste à donner une valeur à chaque variable³³ de telle sorte que la probabilité jointe soit maximum. Il s'agit d'un problème NP-difficile.
- Le problème Probability of Evidence (PR) consiste à donner la probabilité des variables observées (i.e. affectées à une valeur connue) et donc à intégrer sur les autres variables, c'est à dire à calculer la somme des probabilités jointes de toutes les affectations complètes possibles compatibles avec les observations. Il s'agit d'un problème #P-complet.
- Le problème Maximum a Posteriori Hypothesis (MAP) consiste à donner une valeur à un sous-ensemble de variables $Y \subset X$ de telle sorte que la probabilité jointe $\mathbb{P}(Y)$ soit maximum (il faut donc intégrer sur les autres variables). Il s'agit d'un problème NP^{PP} -complet [Park and Darwiche, 2004].

2.2 Lien avec l'optimisation dans les réseaux de fonctions de coût

Le problème MPE de maximisation d'un produit de n fonctions (les probabilités conditionnelles) se traduit directement en un problème WCSP de minimisation de la somme de n fonctions de coût via la transformation suivante³⁴ : $\forall 1 \leq i \leq n, f_{\{x_i\} \cup Pa(x_i)} = -\log \mathbb{P}(x_i | Pa(x_i))$. Ainsi l'optimum du WCSP minimise $-\log \mathbb{P}(X)$ et donc maximise bien $\mathbb{P}(X)$.

Nous avons pu appliquer directement les méthodes décrites dans la Section 1 pour résoudre le problème MPE. DFBB-VE(2) avec maintien de EDAC* et comme heuristique dynamique de choix de variable *last conflict* et *weighted degree* [Lecoutre et al., 2009] a obtenu de très bons résultats dans la catégorie MPE pour les modèles graphiques probabilistes discrets (incluant les réseaux bayésiens et les champs de Markov aléatoires) aux challenges proposés lors des conférences *Uncertainty in AI* 2008³⁵ et 2010³⁶.

La difficulté principale de l'optimisation dans les réseaux bayésiens vient de la taille parfois importante des tables de probabilités conditionnelles se traduisant par des fonctions de coût de grande arité ($r = \max_{x \in X} 1 + |Pa(x)|$) pour lesquelles les

³³Certaines variables pouvant être pré-affectées à leur valeur observée.

³⁴A noter qu'il faut également transformer les coûts réels en entiers (multiplication par une grande constante).

³⁵<http://graphmod.ics.uci.edu/uai08/Evaluation>

³⁶<http://www.cs.huji.ac.il/project/UAI10/>

cohérences locales sont coûteuses à propager durant la recherche³⁷. Un important travail de mise au point de cohérences locales efficaces pour des arités supérieures à trois reste à faire dans le cadre des WCSPs. Ce travail est motivé par l’observation de gains importants sur la qualité du minorant produit par l’algorithme *Min-Sum Diffusion* [Kovalevsky and Koval, 1975, Cooper et al., 2010], une alternative à VAC_ϵ implémentée pour des arités quelconques, sur plusieurs réseaux bayésiens, dont LIN-KAGE et MENDEL³⁸. La convergence lente de *Min-Sum Diffusion* justifie la mise au point de méthodes plus rapides.

La connaissance des probabilités conditionnelles au travers de tables exhaustives comme c’est le cas dans le format des challenges UAI peut masquer un modèle générateur de ces probabilités (par exemple, *noisy-or* [Kim and Pearl, 1983] ou plus simplement une expression logique dans le cas d’une fonction déterministe reliant une variable à ses parents) ou encore des indépendances conditionnelles ne pouvant être capturées par la structure acyclique d’un réseau bayésien. Il est alors intéressant de rechercher une décomposition des probabilités conditionnelles en produits de fonctions d’arité inférieure. Cette approche est utile pour réduire la complexité du maintien des cohérences locales et aussi des méthodes exploitant la structure du réseau (comme VE, CTE ou BTD) en réduisant potentiellement la taille des cliques maximales (elle peut également servir à apprendre des réseaux bayésiens avec moins de paramètres). [Heckerman and Breese, 1996] ont étudié le cas de la décomposition d’une généralisation de *noisy-or*³⁹ incluant les modèles linéaires gaussiens par l’ajout de variables intermédiaires (passant d’une probabilité conditionnelle d’arité $n + 1$ à $2n$ fonctions ternaires et $3n$ variables supplémentaires). Ce travail de modélisation proposant d’autres modèles générateurs décomposables a été poursuivi dans [Diez and Galan, 2002, Zagorecki et al., 2006]. Diverses approches pour automatiser la décomposition des probabilités conditionnelles, en s’autorisant éventuellement d’approximer ces probabilités, ont été proposées [Vomlel, 2002, Poole and Zhang, 2003, Chavira and Darwiche, 2005, Zagorecki and Druzdzel, 2006, Chavira and Darwiche, 2007, Savicky and Vomlel, 2007]. Nous souhaitons poursuivre cette direction de recherche en s’intéressant à la décomposition exacte ou approchée de probabilités conditionnelles avec ou sans ajout de variables supplémentaires et en prenant en compte la partie déterministe (*i.e.* probabilité nulle ou contrainte dure) au sein des probabilités conditionnelles, en s’inspirant des travaux de [Dechter, 1987, Dechter and Pearl, 1992, Meiri et al., 1996].

Ce travail est en cours d’application sur un *problème de reconstruction d’haplotypes dans des pedigrees complexes* (HAPLOTYPE)⁴⁰ [Fichelson et al., 2005]. La décomposition de contraintes dures (en l’occurrence des quaternaires⁴¹ décomposées en deux contraintes ternaires) et l’élimination de variables de faible degré ou reliées fonctionnellement à d’autres variables (via une contrainte binaire définissant une bijection entre les domaines de valeurs) a permis de retrouver une formulation compacte du problème telle qu’initialement proposée par Jean-Michel Elsen dans le cas

³⁷En particulier, une itération de VAC a une complexité en $O(ed^r)$ [Cooper et al., 2010], nécessitant l’usage d’un algorithme d’arc-cohérence généralisée optimal, en pratique délicat à implémenter [Gent et al., 2007, 2010] et à rendre incrémental suite à des mouvements de coûts à la fin de chaque itération de VAC.

³⁸Sur l’instance langladeM7 de la Figure 4, le minorant produit par *Min-Sum Diffusion* est égal à l’optimum comparé à celui de EDAC* qui est à 9% de l’optimum.

³⁹L’idée générale de la décomposition est que la fonction $or(x_1, \dots, x_n) = or(x_1, or(x_2, or(\dots or(x_{n-1}, x_n))))$.

⁴⁰A partir d’un pedigree connu et d’un ensemble d’observations partielles des génotypes sur plusieurs marqueurs positionnés le long d’une paire de chromosomes homologues (*i.e.* possédant les mêmes gènes mais pas obligatoirement les mêmes allèles), le problème est de reconstruire la séquence d’allèles, appelé haplotype, sur chacun des deux chromosomes.

⁴¹Ces contraintes relient la variable d’allèle paternel du fils aux deux allèles de son père et à une variable de ségrégation ($\mathbb{P}(GP_{Ci}|GP_{Ai}, GM_{Ai}, SP_{Ci}) = (SP_{Ci} \wedge (GP_{Ci} = GP_{Ai})) \vee (SP_{Ci} \wedge (GP_{Ci} = GM_{Ai}))$) dans la Figure 5(b)).

de pedigrees de demi-frères (situation courante pour les animaux d'élevage) [Favier et al., 2010b,a]⁴². La Figure 5 montre le cas pour un pedigree avec deux demi-frères. Le réseau bayésien initial (modèle avec variables de ségrégation [Lauritzen and Sheehan, 2003]) contient de l'ordre de $5 * (2 * N + 1) * M$ variables pour N demi-frères et M marqueurs (SNP). La réduction a pour effet de ne conserver plus qu'une seule variable par marqueur *hétérozygote*⁴³ du père, éliminant ainsi toutes les variables des mères (qui sont toutes indépendantes et non observées) et des fils. Le résultat est un problème Max-2SAT ayant en pratique une faible largeur d'arbre et donc très bien résolu par VE. Aurélie Favier a pu ainsi résoudre à l'optimum des pedigrees de demi-frères contenant un grand nombre de marqueurs et de demi-frères (incluant des données réelles HAPMAP comportant 36000 marqueurs SNP et plus de 20 demi-frères) [Favier et al., 2010b]. Avec $N = 4$ et $M = 1500$, le réseau initial * contient 67500 variables et après réduction il ne contient plus que 360 variables booléennes et 661 fonctions de coût binaires avec une largeur d'arbre $w = 3$. Ce travail se poursuit sur l'application à des pedigrees complexes (comportant des boucles dues à des accouplements entre individus apparentés) et sur la prise en compte d'un *a priori* sur les haplotypes des fondateurs via le *déséquilibre de liaison*⁴⁴. Les travaux de Radu Marinescu [Marinescu and Dechter, 2006, 2009b] montraient l'efficacité d'un algorithme de séparation et évaluation exploitant la structure des problèmes (AND/OR graph search) par rapport à l'état de l'art pour résoudre les réseaux bayésiens et en particulier le problème de reconstruction d'haplotypes. Notre expérience sur BTD (et dans une moindre mesure RDS-BTD) ne semble pas confirmer ces bons résultats. En effet, l'approche DFBB-VE avec une ordre de choix de variable libre semble plus efficace que BTD sur la plupart des réseaux bayésiens testés, dont les benchmarks du *Bayesian Network Repository* (ERGO)⁴⁵ et ceux du challenge UAI 2008, la largeur d'arbre étant souvent trop importante (et à l'inverse une faible largeur d'arbre est traitée plus efficacement avec l'algorithme VE) et les problèmes peu modulaires. Le choix d'une bonne décomposition pour ces problèmes reste une question à traiter (voir la Section 1.3.3).

2.3 Inférence approchée

Philippe Jégou et Aurélie Favier ont proposé une extension de BTD au comptage (#BTD) ainsi qu'une version approchée (Approx#BTD) fondée sur un partitionnement des contraintes en sous-graphes triangulés⁴⁶. Nous avons ensuite montré que la complexité de cette méthode approchée est au pire exponentielle en la taille de la plus grande clique du problème. Or sur de nombreux benchmarks, cette taille est relativement faible (et bien inférieure à la largeur d'arbre), conduisant à une méthode très performante mais sans garantie de qualité. La plupart des outils existants étant dans #SAT, il a été difficile de montrer expérimentalement les avantages de #BTD à quelques exceptions près (problème des tours d'Hanoi). Par contre de très bons résultats ont été obtenus sur le problème de *coloriage de graphe* (COLORING)⁴⁷ [Favier et al., 2009b,a]. *

⁴²Cette formulation fait également une hypothèse simplificatrice d'équi-probabilité sur les allèles des fondateurs qui rend possible une telle réduction.

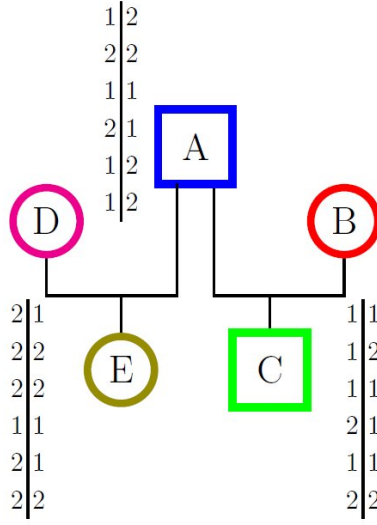
⁴³Dont les deux valeurs observées de ses deux allèles sont différentes.

⁴⁴Le déséquilibre de liaison mesure la corrélation entre les allèles sur un même haplotype à deux positions différentes. Celui-ci peut être obtenu soit à partir des génotypes observés, soit à partir des haplotypes partiellement reconstruits de façon certaine qui ont été transmis par les mères. Il doit permettre d'améliorer la qualité de la reconstruction pour un faible nombre d'individus (cas le plus intéressant en pratique).

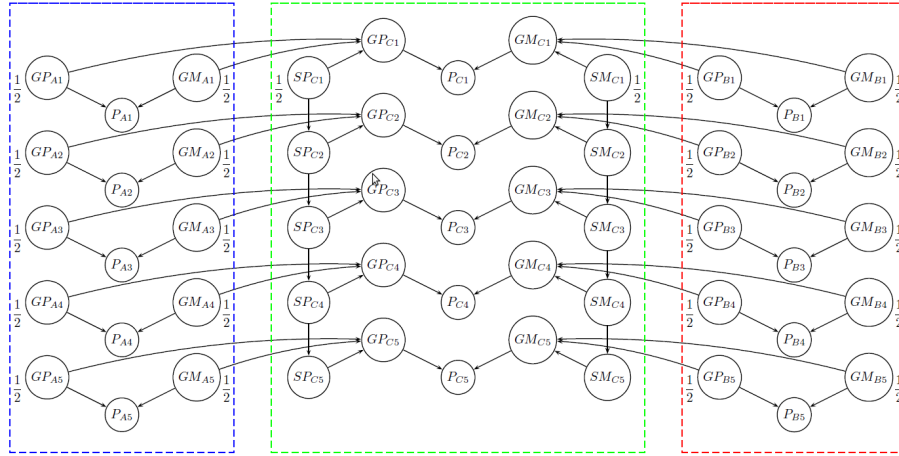
⁴⁵<http://www.cs.huji.ac.il/~galel/Repository/>

⁴⁶Nous avons en vain cherché une méthode de décomposition avec garantie de maximalité dans le cas de contraintes non binaires. Finalement une heuristique simple sans garantie a été retenue.

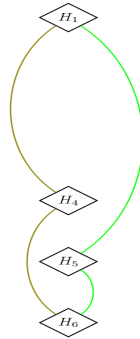
⁴⁷Le problème est d'attribuer une couleur à chacun des sommets d'un graphe de manière à ce que deux sommets reliés par une arête soient de couleur différente.



(a) Pedigree avec le père A et les enfants C, E génotypés sur 6 marqueurs (mères B, D non génotypées)



(b) Vue partielle du réseau bayésien



(c) Réseau complet après réduction

FIG. 5 – Exemple de réduction du réseau bayésien correspondant à un problème de reconstruction d'haplotypes pour un pedigree de demi-frères.

Une extension de #BTD à la question de l'intégration discrète (calcul d'inférence dans les réseaux bayésiens) a été initiée. Outre l'inférence exacte, du fait de sa complexité, il est important de s'intéresser aussi à des méthodes approchées. Un travail d'extension d'Approx#BTD à l'inférence est en cours qui sera suivi d'une évaluation par rapport à l'état de l'art. Il sera également intéressant d'étudier l'apport des cohérences locales pour fournir une sur- et une sous-approximation de la masse de probabilité et voir comment cela peut être exploité dans un algorithme de recherche arborescente. Ce sujet sera abordé au travers d'un projet ANR Blanc (Ficololo 2011-2013) et fait partie d'un nouveau thème de recherche du laboratoire : *Algorithmique approchée pour les modèles graphiques stochastiques*.

2.4 Apprentissage de la structure par l'optimisation d'un score

Lorsque l'on ne connaît pas la structure du réseau bayésien ni les probabilités conditionnelles, il est possible de les apprendre à partir d'un ensemble d'observations sur les variables aléatoires. Il s'agit alors de trouver un graphe dirigé acyclique avec les probabilités conditionnelles associées qui représente au mieux les observations tout en ayant une complexité raisonnable⁴⁸. Cette question de l'apprentissage de la structure d'un réseau bayésien fait l'objet d'une seconde thèse également appliquée à un problème de bioinformatique sur la *reconstruction de réseaux de régulations de gènes à partir de données d'expression et de données de polymorphisme* (GEN-GEN)⁴⁹ [Jansen and Nap, 2001]. Jimmy Vandel a évalué des méthodes d'apprentissage existantes au travers d'une librairie d'algorithmes d'apprentissage en Matlab⁵⁰. Des premiers résultats ont été obtenus et publiés [Vandel and de Givry, 2009, Vandel et al., 2010a]. Le passage à une librairie en Java⁵¹ a permis d'introduire des connaissances biologiques en interdisant ou forçant certains arcs (en fonction de régulations locales *en cis* ou distantes *en trans*) et de tester d'autres critères de score amenant à un nouvel article pour un workshop [Vandel et al., 2010b]. Ce travail a permis d'avoir une comparaison de l'approche bayésienne par rapport à l'état de l'art des méthodes de reconstruction de réseaux de gènes. Des résultats préliminaires montrent que cette approche semble plus robuste que les autres approches sur nos données simulées. Ces bons résultats ont été en partie confirmés lors d'une compétition internationale DREAM5⁵² sur des données de grande taille ($p = 1000$ gènes) pour un nombre d'observations suffisant (de l'ordre de p) et un graphe peu dense (notre approche bayésienne obtenant le meilleur compromis précision/sensibilité). Les perspectives de ce travail sont une plus grande prise en compte des aspects biologiques (contraindre la nature des régulations apprises, prise en compte d'un grand nombre de SNP, etc).

Au niveau algorithmique, le problème de trouver un réseau bayésien qui optimise un score (AIC, BIC, BDe, etc) est un sujet rarement traité par des méthodes exactes (à l'exception de [Silander and Myllymäki, 2006]). Il y a là une opportunité pour tester les méthodes WCSPs en s'inspirant de la réduction de l'espace de recherche

⁴⁸La complexité d'un réseau bayésien est mesuré par le nombre de paramètres indépendants définissant les probabilités conditionnelles.

⁴⁹Le problème consiste à exploiter un grand nombre d'individus génétiquement différents sur lesquels on mesure les données d'expression de l'ensemble des gènes ainsi que des données de polymorphisme (observation des génotypes d'un grand nombre de marqueurs). La variabilité génétique influant directement sur la régulation des gènes, cela permet de retrouver un plus grand nombre de régulations. Nous étudions l'exploitation conjointe des données d'expression et des données génétiques dans le formalisme des réseaux bayésiens discrets.

⁵⁰<http://bnt.insa-rouen.fr/>, librairie *BNT Structure Learning* en partie écrite par Kevin Murphy (Vancouver, Canada) et Philippe Leray (Polytech, Nantes).

⁵¹<http://www.cs.duke.edu/~amink/software/banjo/>

⁵²<http://wiki.c2b2.columbia.edu/dream>

proposée dans [de Campos and Ji, 2010] et les modélisations déjà étudiées pour le problème FEEDBACK (de manière à garantir l’acyclicité du graphe) [Lesaint et al., 2010]. Ce problème d’optimisation est également abordé en recherche locale en utilisant la librairie Comet⁵³. L’objectif est de tester d’autres voisinages (par exemple, en remettant en cause tous les parents d’une ou deux variables) et diverses méta-heuristiques (*Tabu Search* [Glover and Laguna, 1993], *Variable Neighborhood Search* [Mladenovic and Hansen, 1997], *Guided Local Search* [Voudouris, 1997]). A terme, cela doit permettre de trouver efficacement une structure optimale pour ensuite pouvoir parcourir la distribution des structures sous-optimales de manière à en déduire une structure consensus robuste, à l’instar des travaux effectués sur

* l’ordonnancement de marqueurs en génétique [Servin et al., 2010].

Les questions concernant l’impact de la discrétisation des données et la détermination des causalités seront également importantes à poursuivre.

Conclusion

En 2003 à Kinsale lors de la conférence *Principle and Practice of Constraint Programming*, Jean-François Puget, alors vice-président de la R&D optimisation à ILOG, déclarait que les travaux autour des WCSPs se limitaient à résoudre des problèmes (aléatoires) de petite taille. J’espère avoir montré dans ce rapport que ce n’est plus le cas. Le problème d’allocation de fréquence (CELAR) en est un bon exemple. Face à la PLNE, l’approche WCSP est une alternative intéressante pour traiter des problèmes d’optimisation complexes. Les WCSPs sont une extension pragmatique du cadre CSP à l’optimisation autorisant des gains d’efficacité importants par rapport à l’approche usuelle en programmation par contraintes consistant à encapsuler la fonction objectif dans une variable.

Les évolutions à mener dans les prochaines années sont d’une part la définition de nouvelles cohérences locales adaptées aux fonctions de coût globales ou de grande arité dont la convergence soit plus rapide que VAC et d’autre part une prise en compte automatique (sans réglage) de la structure des problèmes.

Outre la mise en évidence de nouvelles applications des WCSPs en bioinformatique, je souhaite poursuivre mon étude des techniques WCSPs appliquées aux problèmes de l’inférence approchée et de l’apprentissage de structure dans les modèles graphiques probabilistes discrets.

Enfin je tiens à remercier l’ensemble de mes co-auteurs à l’INRA (Thomas, Christine, David,...) et aussi à l’IRIT (Hélène et Martin), à l’ONERA (Gérard, Fred et Martin), à THALES (Pierre, Juliette et Jean) et encore à Barcelone (Javier, Marti et Pedro) et Marseille (Philippe), sans lesquels toute cette aventure n’aurait pu voir le jour.

⁵³<http://www.comet-online.org>

Références

- D Allouche, S de Givry, M Sanchez, and T Schiex. TagSNP selection using Weighted CSP and Russian Doll Search with Tree Decomposition. In *CP-09 workshop on Constraint Based Methods for Bioinformatics*, Lisbon, Portugal, 2009.
- D. Allouche, S. de Givry, and T. Schiex. Sélection de tagSNP : une approche PLNE. In *Proc. of ROADEF-10*, Toulouse, France, 2010a.
- D. Allouche, S. de Givry, and T. Schiex. Towards parallel non serial dynamic programming for solving hard weighted csp. In *Proc. of CP-10*, St Andrews, Scotland, 2010b.
- D. Aloise, S. Caferri, G. Caporossi, P. Hansen, S. Perron, and L. Liberti. Column generation algorithms for exact modularity maximization in networks. *Phys. Rev. E*, 82(4), 2010.
- C. Ansotegui, M. Bonet, and J. Levy. A new algorithm for weighted partial maxsat. In *Proc. of AAAI-10*, Atlanta, Georgia, USA, 2010.
- J. Argelich, C.M. Li, F. Manya, and J. Planes. New inference rules for max-sat. *Journal of Artificial Intelligence Research*, 30 :321–359, 2007.
- F. Bacchus, S. Dalmao, and T. Pitassi. Value elimination : Bayesian inference via backtracking search. In *Proc. of UAI-03*, pages 20–28, 2003.
- F. Bacchus, S. Dalmao, and T. Pitassi. Solving #sat and bayesian inference with backtracking search. *Journal of Artificial Intelligence Research*, 34 :391–442, 2009.
- E. Bensana, M. Lemaître, and G. Verfaillie. Earth observation satellite management. *Constraints*, 4(3) :293–299, 1999.
- A. Berry, J.-P. Bordat, and O.Cogis. Generating all the minimal separators of a graph. *International Journal of Foundations of Computer Science*, 11 :397–404, 2000.
- U. Bertelé and F. Brioschi. *Nonserial Dynamic Programming*. Academic Press, 1972.
- C. Bessière and J-C. Régin. Mac and combined heuristics : Two reasons to forsake fc (and cbj ?) on hard problems. In *Proc. of the Second International Conference on Principles and Practice of Constraint Programming*, pages 61–75, Cambridge (MA), 1996.
- C. Bessière and J-C. Régin. Refining the basic constraint propagation algorithm. In *Proc. of IJCAI’2001*, pages 309–315, 2001.
- C-E. Bichot. *Elaboration d’une nouvelle métaheuristique pour le partitionnement de graphe : la méthode de fusion-fission. Application au découpage de l’espace aérien*. PhD thesis, ENAC, Toulouse, 2007.
- A. Billionnet. *Optimisation discrète*. Dunod, 2007.
- E. Boros and P. Hammer. Pseudo-Boolean Optimization. *Discrete Appl. Math.*, 123 :155–225, 2002.
- S. Bouveret, F. Heras, S.de Givry, J. Larrosa, M. Sanchez, and T. Schiex. Toolbar : a state-of-the-art platform for wcsp. <http://www.inra.fr/mia/T/degivry/ToolBar.pdf>, 2004.

- B. Cabon, S. de Givry, L. Lobjois, T. Schiex, and J.P. Warners. Radio Link Frequency Assignment. *Constraints*, 4(1) :79–89, 1999.
- M. Chavira and A. Darwiche. Compiling bayesian networks with local structure. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1306–1312, 2005.
- M. Chavira and A. Darwiche. Compiling Bayesian networks using variable elimination. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2443–2449, 2007.
- M. Cooper. Reduction operations in fuzzy or valued constraint satisfaction. *Fuz. Sets and Sys.*, 134(3) :311 – 342, 2003.
- M. Cooper. High-order consistency in Valued Constraint Satisfaction. *Constraints*, 10 :283–305, 2005.
- M. Cooper and T. Schiex. Arc consistency for soft constraints. *Artificial Intelligence*, 154 :199–227, 2004.
- M. Cooper, S. de Givry, and T. Schiex. Optimal soft arc consistency. In *Proc. of IJCAI-07*, pages 68–73, Hyderabad, India, 2007.
- M. Cooper, S. de Givry, M. Sanchez, T. Schiex, and M. Zytnicki. Virtual arc consistency for weighted csp. In *Proc. of AAAI-08*, Chicago, IL, 2008a.
- M. Cooper, S. de Givry, M. Sanchez, T. Schiex, and M. Zytnicki. Cohérence d’arc virtuelle pour les csp pondérés. In *Proc. of JFPC-08*, Nantes, France, 2008b.
- M. Cooper, S. de Givry, M. Sanchez, T. Schiex, M. Zytnicki, and T. Werner. Soft arc consistency revisited. *Artificial Intelligence*, 174(7–8) :449–478, 2010.
- A. Darwiche. Recursive Conditioning. *Artificial Intelligence*, 126(1-2) :5–41, 2001.
- C. de Campos and Q. Ji. Properties of bayesian dirichlet scores to learn bayesian network structures. In *Proc. of AAAI-94*, pages 49–54, Washington, DC, 2010.
- S. de Givry. Singleton consistency and dominance for weighted csp. In *Proc. of 6th International CP-2004 Workshop on Preferences and Soft Constraints*, page 15p., Toronto, Canada, 2004.
- S. de Givry and L. Jeannin. A unified framework for partial and hybrid search methods in constraint programming. *Computer & Operations Research*, 33(10) : 2805–2833, 2006.
- S. de Givry, G. Verfaillie, and T. Schiex. Bounding the Optimum of Constraint Optimization Problems. In *Proc. of CP-97*, pages 405–419, Schloss Hagenberg, Austria, October 29 - November 1 1997.
- S. de Givry, L. Jeannin, F. Josset, J. Mattioli, N. Museux, and P. Savéant. The THALES constraint programming framework for hard and soft real-time applications. The PLANET Newsletter, Issue 5 ISSN 1610-0212, pages 5-7, December 2002. <http://planet.dfki.de/service/Resourses/Rome/degivry.pdf> (slides).
- S. de Givry, J. Larrosa, P. Meseguer, and T. Schiex. Solving max-sat as weighted csp. In *Proc. of CP-03*, pages 363–376, Kinsale, County Cork, Ireland, 2003.
- S. de Givry, M. Bouchez, P. Chabrier, D. Milan, and T. Schiex. CARTHAGENE : multipopulation integrated genetic and radiated hybrid mapping. *Bioinformatics*, 21(8) :1703–1704, 2005a.

- S. de Givry, I. Palhiere, Z. Vitezica, and T. Schiex. Mendelian error detection in complex pedigree using weighted constraint satisfaction techniques. In *ICLP-05 workshop on Constraint Based Methods for Bioinformatics*, page 9p., Sitges, Spain, 2005b.
- S. de Givry, M. Zytnicki, F. Heras, and J. Larrosa. Existential arc consistency : Getting closer to full arc consistency in weighted CSPs. In *Proc. of IJCAI-05*, pages 84–89, Edinburgh, Scotland, 2005c.
- S. de Givry, T. Schiex, and G. Verfaillie. Exploiting Tree Decomposition and Soft Local Consistency in Weighted CSP. In *Proc. of AAAI-06*, Boston, MA, 2006a.
- S. de Givry, Z. Vitezica, I. Palhiere, and T. Schiex. Mendelsoft : Mendelian error detection in complex pedigree using weighted constraint satisfaction techniques. In *8th World Congress on Genetics Applied to Livestock Production*, page 2p., Belo Horizonte, Brazil, 2006b.
- R. Debruyne and C. Bessière. Domain filtering consistencies. *Journal of Artificial Intelligence Research*, 14 :205–230, May 2001.
- R. Dechter. *Constraint Processing*. Morgan Kaufmann Publishers, 2003.
- R. Dechter. Decomposing an n-ary relation into a tree of binary relations. In *In Proc. of PODS-87*, pages 185–189, 1987.
- R. Dechter. Bucket elimination : A unifying framework for reasoning. *Artificial Intelligence*, 113(1–2) :41–85, 1999.
- R. Dechter and J. Pearl. Tree clustering for constraint networks. *Artificial Intelligence*, 38 :353–366, 1989.
- R. Dechter and J. Pearl. Structure identification in relational data. *Artificial Intelligence*, 58 :237–270, December 1992.
- R. Dechter and I. Rish. Mini-buckets : A general scheme for bounded inference. *J. ACM*, 50(2) :107–153, 2003.
- F. Diez and S. Galan. Efficient computation for the noisy max. *International Journal of Intelligent Systems*, 18 :2003, 2002.
- T. Faraut, S. de Givry, P. Chabrier, T. Derrien, F. Galibert, C. Hitte, and T. Schiex. A comparative genome approach to marker ordering. In *Proc. of ECCB-06*, page 7p., Eilat, Israel, 2007.
- A. Favier, S. de Givry, and P. Jégou. Exploiting problem structure for solution counting. In *Proc. of CP-09*, pages 335–343, Lisbon, Portugal, 2009a.
- A. Favier, S. de Givry, and P. Jégou. Comptage de solutions en exploitant la structure du graphe de contraintes. In Yves Deville, editor, *Actes des Cinquièmes Journées Francophones de Programmation par Contraintes*, Orléans, 2009b.
- A. Favier, J-M. Elsen, S. de Givry, and A. Legarra. Optimal haplotype reconstruction in half-sib families. In *ICLP-10 workshop on Constraint Based Methods for Bioinformatics*, Edinburgh, UK, 2010a.
- A. Favier, J-M. Elsen, S. de Givry, and A. Legarra. Exact Haplotype Reconstruction in Half-Sib Families with Dense Marker Maps. In *9th World Congress on Genetics Applied to Livestock Production (WCGALP-10)*, Leipzig, Germany, 2010b.

- M. Fichelson, N. Dovgolevsky, and D. Geiger. Maximum Likelihood Haplotyping for General Pedigrees. *Human Heredity*, 59(1) :41–60, 2005.
- E. Freuder and M. Quinn. Taking advantage of stable sets of variables in constraint satisfaction problems. In *Proc. of IJCAI-85*, pages 1076–1078, Los Angeles, CA, 1985.
- I. Gent, C. Jefferson, I. Miguel, and P. Nightingale. Data structures for generalised arc consistency for extensional constraints. In *Proc. of AAAI-07*, Boston, MA, 2007.
- I. Gent, C. Jefferson, I. Miguel, and P. Nightingale. Generating special-purpose stateless propagators for arbitrary constraints. In *Proc. of CP-10*, pages 206–220, St Andrews, Scotland, 2010.
- S. de Givry, Y. Hamadi, J. Mattioli, P. Gérard, M. Lemaître, G. Verfaillie, A. Aggoun, I. Gouachi, T. Benoist, E. Bourreau, F. Laburthe, P. David, S. Loudni, and S. Bourgault. Towards an on-line optimisation framework. In *CP-2001 Workshop on On-Line combinatorial problem solving and Constraint Programming (OLCP’01)*, pages 45–61, Paphos, Cyprus, December 1 2001.
- F. Glover and M. Laguna. Tabu Search. In *Modern Heuristic Techniques for Combinatorial Problems*, pages 70–141. Blackwell Scientific Publishing, Oxford, 1993.
- C. Gomes, B. Selman, and H. Kautz. Boosting combinatorial search through randomization. In *Proc. of AAAI-98*, Madison, WI, 1998.
- P. Hammer, P. Hansen, and B. Simeone. Roof duality, complementation and persistency in quadratic 0-1 optimization. *Math. Programming*, 28 :121–155, 1984.
- W. Harvey and M. Ginsberg. Limited discrepancy search. In *Proc. of the 14th IJCAI*, Montréal, Canada, 1995.
- D. Heckerman and J. Breese. Causal independence for probability assessment and inference using bayesian networks. *IEEE Systems, Man, and Cyber.*, 26(6) :826–831, 1996.
- F. Heras and J. Larrosa. A max-sat inference-based pre-processing for max-clique. In *Proc. of SAT-08*, pages 139–152, Guangzhou, China, 2008.
- F. Heras, J. Larrosa, and A. Oliveras. MiniMaxSat : A New Weighted Max-SAT Solver. In *Proc. of SAT’2007*, number 4501 in LNCS, pages 41–55, Lisbon, Portugal, May 2007.
- R. Jansen and J. Nap. Genetical genomics : the added value from segregation. *Trends in genetics*, 17(7) :388–391, July 2001.
- P. Jégou and C. Terrioux. Decomposition and good recording. In *Proc. of ECAI-2004*, pages 196–200, Valencia, Spain, 2004.
- M. Khemmoudj and H. Bennaceur. Valid inequality based lower bounds for wcsp. cp 2007 : 394-408. In *Proc. of CP-07*, pages 394–408, Providence, RI, USA, 2007.
- J. Kim and J. Pearl. A computational model for causal and diagnostic reasoning in inference systems. In *In Proc. of IJCAI-83*, pages 190–193, Karlsruhe, Germany, 1983.
- M. Kitching. *Decomposition and Symmetry in Constraint Optimization Problems*. PhD thesis, University of Toronto, 2010.

- V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(10) : 1568–1583, 2006.
- A. Koster. *Frequency assignment : Models and Algorithms*. PhD thesis, Maastricht, The Netherlands, 1999.
- V. Koval and M. Schlesinger. Dvumernoe programmirovaniye v zadachakh analiza izobrazheniy (Two-dimensional programming in image analysis problems). *USSR Academy of Science, Automatics and Telemechanics*, 8 :149–168, 1976. In Russian.
- V. Kovalevsky and V. Koval. A diffusion algorithm for decreasing energy of max-sum labeling problem. Technical report, Glushkov Institute of Cybernetics, Kiev, USSR, 1975.
- J. Larrosa. Boosting search with variable elimination. In *CP*, pages 291–305, Singapore, 2000.
- J. Larrosa. On arc and node consistency in weighted CSP. In *Proc. AAAI-02*, pages 48–53, Edmondton, (CA), 2002.
- J. Larrosa and F. Heras. Resolution in max-sat and its relation to local consistency in weighted csp. In *Proc. of the 19th IJCAI*, pages 193–198, Edinburgh, Scotland, 2005.
- J. Larrosa and T. Schiex. Solving Weighted CSP by Maintaining Arc-consistency. *Artificial Intelligence*, 159(1-2) :1–26, 2004.
- J. Larrosa and T. Schiex. In the quest of the best form of local consistency for weighted CSP. In *Proc. of IJCAI-03*, pages 239–244, Acapulco, Mexico, 2003.
- J. Larrosa, P. Meseguer, T. Schiex, and G. Verfaillie. Reversible DAC and other improvements for solving max-CSP. In *Proc. of AAAI’98*, Madison, WI, July 1998.
- J. Larrosa, P. Meseguer, and M. Sanchez. Pseudo-tree search with soft constraints. In *Proc. of ECAI-02*, pages 131–135, Lyon, France, 2002.
- J. Larrosa, F. Heras, and S. de Givry. A logical approach to efficient max-sat solving. *Artificial Intelligence*, 172(2–3) :204–233, 2008.
- S. Lauritzen and N. Sheehan. Graphical Models for Genetic Analyses. *Statistical Science*, 18(4) :489–514, 2003.
- S. Lauritzen and D. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society – Series B*, 50 :157–224, 1988.
- C. Lecoutre, L. Saïs, S. Tabary, and V. Vidal. Reasoning from last conflict(s) in constraint programming. *Artificial Intelligence*, 173 :1592,1614, 2009.
- J. Lee and K. Leung. Towards efficient consistency enforcement for global constraints in weighted constraint satisfaction. In *Proc. of IJCAI’09*, pages 559–565, 2009.
- J. Lee and K. Leung. A stronger consistency for soft global constraints in weighted constraint satisfaction. In *Proc. of AAAI’10*, Atlanta, Georgia, USA, 2010.

- D. Lesaint, D. Mehta, B. O’Sullivan, L. Quesada, and N. Wilson. Developing approaches for solving a telecommunications feature subscription problem. *J. Artif. Intell. Res. (JAIR)*, 38 :271–305, 2010.
- C.M. Li and Z. Quan. An efficient branch-and-bound algorithm based on maxsat for the maximum clique problem. In *Proc. of AAAI-10*, Atlanta, Georgia, USA, 2010.
- R. Marinescu and R. Dechter. And/or branch-and-bound search for combinatorial optimization in graphical models. *Artificial Intelligence*, 173(16-17) :1457–1491, 2009a.
- R. Marinescu and R. Dechter. Memory intensive and/or search for combinatorial optimization in graphical models. *Artificial Intelligence*, 173(16-17) :1492–1524, 2009b.
- R. Marinescu and R. Dechter. And/or branch-and-bound for graphical models. In *Proc. of IJCAI-05*, pages 224–229, Edinburgh, Scotland, 2005.
- R. Marinescu and R. Dechter. Memory intensive branch-and-bound search for graphical models. In *Proc. of AAAI-06*, Boston, MA, 2006.
- J. Marques-Silva and V. Manquinho. Towards more effective unsatisfiability-based maximum satisfiability algorithms. In *Proc. of SAT-08*, pages 225–230, Guangzhou, China, 2008.
- I. Meiri, R. Dechter, and J. Pearl. Uncovering trees in constraint networks. *Artif. Intell.*, 86(2) :245–267, 1996.
- J-P. Métevier. *Relaxation de contraintes globales*. PhD thesis, Université de Caen, 2010.
- N. Mladenovic and P. Hansen. Variable neighbourhood search. *Computer and Operations Research*, 24 :1097–1100, 1997.
- S. Ndiaye. *Calcul et exploitation de recouvrements acycliques pour la résolution de (V)CSP*. PhD thesis, LSIS Marseille, 2007.
- M. Newman. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev.*, 74, 2006.
- P. Östergård and V. Vaskelainen. Russian doll search for the steiner triple covering problem. *Optimization Letters*, pages 1–8, 2010.
- J. Park and A. Darwiche. Complexity results and approximation strategies for map explanations. *Journal of Artificial Intelligence Research*, 21 :101–133, 2004.
- D. Poole and N. Zhang. Exploiting contextual independence in probabilistic inference. *Journal of Artificial Intelligence Research*, 18 :263–313, 2003.
- J-C. Régim. Cost based arc consistency for global cardinality constraints. *Constraints*, 7(3-4) :387–405, 2002.
- D. Sabin and E. Freuder. Contradicting conventional wisdom in constraint satisfaction. In *Proceedings of the 2nd International Workshop on Principles and Practice of Constraint Programming*, number 874 in LNCS, Rosario, Orcas Island (WA), May 1994. Springer-Verlag.

- M. Sánchez, S. de Givry, and T. Schiex. Mendelian error detection in complex pedigrees using weighted constraint satisfaction techniques. *Constraints*, 13(1) : 130–154, 2008.
- M Sanchez, D Allouche, S de Givry, and T Schiex. Russian doll search with tree decomposition. In *Proc. of IJCAI’09*, Pasadena (CA), USA, 2009. http://www.inra.fr/mia/T/degivry/rdsbtd_ijcai09_sdg.ppt.
- P. Savicky and J. Vomlel. Exploiting tensor rank-one decomposition in probabilistic inference. *Kybernetika*, 43(5) :747–764, 2007.
- T. Schiex. Arc consistency for soft constraints. In *Principles and Practice of Constraint Programming - CP 2000*, volume 1894 of *LNCS*, pages 411–424, Singapore, September 2000.
- T. Schiex, H. Fargier, and G. Verfaillie. Valued constraint satisfaction problems : Hard and easy problems. In *Proc. of IJCAI-95*, pages 631–637, Montréal, Canada, 1995.
- M. Schlesinger. Sintaksicheskiy analiz dvumernykh zritelnykh signalov v usloviyakh pomekh (Syntactic analysis of two-dimensional visual signals in noisy conditions). *Kibernetika*, 4 :113–130, 1976. In Russian.
- B. Servin, S. de Givry, and T. Faraut. Statistical confidence measures for genome maps : application to the validation of genome assemblies. *Bioinformatics*, 2010.
- G. Shafer and P. Shenoy. Probability propagation. *Annals of Mathematics and Artificial Intelligence*, 2(1–4) :327–352, 1990.
- T. Silander and P. Myllymäki. A simple approach for finding the globally optimal bayesian network structure. In *Proc. of UAI-06*, pages 445–452, Cambridge, MA, 2006.
- C. Terrioux and P. Jegou. Hybrid backtracking bounded by tree-decomposition of constraint networks. *Artificial Intelligence*, 146(1) :43–75, 2003.
- P. Thébault, S. de Givry, T. Schiex, and C. Gaspin. Searching RNA motifs and their intermolecular contacts with constraint networks. *Bioinformatics*, 22(17) : 2074–2080, 2006.
- J. Vandel and S. de Givry. Inférence de réseaux de régulation de gènes à partir de données d’expression et de polymorphisme. In *JOBIM-09 workshop sur Modèles graphiques probabilistes pour l’intégration de données hétérogènes et la découverte de modèles causaux en biologie*, Nantes, France, 2009.
- J. Vandel, S. de Givry, B. Mangin, and M. Vignes. Reconstruction de réseau de régulation de gène à l’aide de données génomiques et de données génétiques. In *Proc. of CAP-10*, 2010a.
- J Vandel, B Mangin, M Vignes, and S de Givry. Extended bayesian scores for reconstructing gene regulatory networks. In *ECCS-10 workshop on graphical models for reasoning on biological systems : computational challenges*, Lisbon, Portugal, 2010b.
- G. Verfaillie, M. Lemaître, and T. Schiex. Russian doll search. In *Proc. of AAAI’96*, pages 181–187, Portland, OR, 1996.
- J. Vomlel. Exploiting functional dependence in bayesian network inference. In *Proc. of UAI-06*, pages 528–535, Alberta, Canada, 2002.

- C. Voudouris. *Guided local search for combinatorial optimisation problems*. PhD thesis, Department of Computer Science, University of Essex, Colchester, UK, July 1997.
- M. Wainwright, T. Jaakkola, and A. Willsky. MAP estimation via agreement on (hyper)trees : message passing and linear programming approaches. *IEEE Trans. on Information Theory*, 51(11) :3697–3717, 2005.
- T. Werner. A Linear Programming Approach to Max-sum Problem : A Review. *IEEE Trans. on Pattern Recognition and Machine Intelligence*, 29(7) :1165–1179, July 2007.
- A. Zagorecki and M. Druzdzal. Knowledge engineering for bayesian networks : How common are noisy-max distributions in practice? In *Proc. of ECAI-06*, pages 482–487, Riva del Garda, Italy, 2006.
- A. Zagorecki, M. Voortman, and M. Druzdzal. Decomposing local probability distributions in bayesian networks for improved inference and parameter learning. In *FLAIRS Conference*, pages 860–865, Florida, USA, 2006.
- M. Zytnicki, F. Heras, S. de Givry, and J. Larrosa. Cohérence d’arc existentielle : un pas de plus vers la cohérence d’arc complète. In *Proc. of JFPC-05*, pages 139–148, Lens, France, 2005.
- M. Zytnicki, C. Gaspin, and T. Schiex. Darn ! a soft constraint solver for rna motif localization. *Constraints*, 13(1), 2008.
- M. Zytnicki, C. Gaspin, S. de Givry, and T. Schiex. Bounds Arc Consistency for Weighted CSPs. *Journal of Artificial Intelligence Research*, 35 :593–621, 2009.

A Applications

La Table 3 résume les principales applications mentionnées dans ce rapport en donnant la dimension des plus grandes instances résolues (n , nombre de variables, d , taille du plus grand domaine, r , arité maximum des fonctions de coût, e , nombre de fonctions de coût) en précisant quelle(s) méthode(s) se sont avérées les plus efficaces pour résoudre ces instances à l’optimum.

Problème	Référence	n	d	r	e	Méthodes
<i>Allocation de ressources</i>						
CELAR	[Cabon et al., 1999]	458	44	2	5000	CTE, RDS-BTD, VAC
SPOT5	[Bensana et al., 1999]	364	4	3	10108	RDS, RDS-BTD
WAREHOUSE	[de Givry et al., 2005c]	1100	300	2	100000	PLNE, EDAC, VAC
<i>Bioinformatique</i>						
HAPLOTYPE	[Fichelson et al., 2005]	1200	7	5	2000	AND/OR graph search
(demi-frères)	[Favier et al., 2010b]	36000	2	2	20000	VE
MENDEL	[Sánchez et al., 2008]	20000	66	3	30000	EDAC3, DFBB-VE
RNA	[Zytnicki et al., 2008]	20	4.10^9	4	10	BAC
TAGSNP	[Sanchez et al., 2009]	1500	266	2	150000	PLNE, RDS-BTD

TAB. 3 – Aperçu des principales applications traitées comme des réseaux de fonctions de coût.

L’ensemble de ces applications (excepté RNA) ainsi que de nombreux benchmarks sont disponibles sur le site <http://mulcyber.toulouse.inra.fr/projects/costfunctionlib/> au format `wcsp` et à terme sur le site <http://costfunction.org/>.

B ToulBar2

ToulBar2 est un résolveur WCSP en domaine public disponible sur la Forge logiciel de l’INRA Toulouse à l’adresse <http://mulcyber.toulouse.inra.fr/projects/toulbar2/>. Il intègre l’ensemble des méthodes présentées dans ce rapport : les cohérences locales (NC,AC,FD(G)AC,ED(G)AC,VAC,BAC) incluant des algorithmes de renforcement pour des fonctions de coût globales (AllDiff,GCC,Same,Regular) et les algorithmes hybrides combinant une recherche arborescente et l’élimination de variable ou de cluster (DFBB-VE, BTD, RDS-BTD). Il dispose également de stratégies de diversification de la recherche (*Limited Discrepancy Search* [Harvey and Ginsberg, 1995] et *Random Restart* [Gomes et al., 1998]). Son architecture modulaire permet d’ajouter de nouvelles fonctions de coût à l’instar des outils de programmation par contraintes ainsi que de nouveaux algorithmes de recherche, disposant d’une séparation claire entre le maintien des cohérences locales et l’exploration de l’espace des solutions. Il peut aussi être utilisé pour résoudre les réseaux bayésiens (problème MPE) et les champs de Markov, les deux au format UAI-2008, les problèmes Max-SAT au format `(w)cnf`, ainsi que le problème du comptage du nombre de solutions d’un CSP (au format `wcsp` ou XCSP 2.1). ToulBar2 a remporté de nombreux succès dans des compétitions de solveurs, notamment Max-CSP 2006 & 2008, UAI 2008 & 2010.

C Glossaire

algorithme	(BT) : BackTrack , 2
algorithme	(BTD) : Backtrack with Tree Decomposition , 17
algorithme	(CTE) : Cluster Tree Elimination , 13
algorithme	(DFBB) : Depth-First Branch and Bound , 3
algorithme	(DFBB-VE) : Boosting Search with Variable Elimination , 15
algorithme	(RDS) : Russian Doll Search , 18
algorithme	(RDS-BTD) : Russian Doll Search with Tree Decomposition , 18
algorithme	(VE) : Variable Elimination , 12
cohérence	(AC) : Arc Consistency , 3
cohérence	(AC*) : Arc Consistency , 5
cohérence	(BAC) : Bounds Arc Consistency , 10
cohérence	(DAC) : Directional Arc Consistency , 5
cohérence	(EAC) : Existential Arc Consistency , 7
cohérence	(EDAC*) : Existential and Full Directional Arc Consistency , 7
cohérence	(FDAC*) : Full Directional Arc Consistency , 6
cohérence	(OSAC) : Optimal Soft Arc Consistency , 8
cohérence	(SAC) : Soft Arc Consistency , 4
cohérence	(VAC) : Virtual Arc Consistency , 8
formalisme	(BN) : Bayesian Network , 21
formalisme	(CSP) : Constraint Satisfaction Problem , 2
formalisme	(PBO) : Pseudo-Boolean Optimization , 6
formalisme	(PLNE) : Programmation Linéaire en Nombre Entier , 6
formalisme	(VCSP) : Valued Constraint Satisfaction Problem , 3
formalisme	(WCSP) : Weighted Constraint Satisfaction Problem , 3
problème	(BEP) : gestion des prises de vue d'un satellite agile , 10
problème	(CELAR) : allocation de fréquence à des liens radios , 15
problème	(COLORING) : coloriage de graphe , 23
problème	(ERGO) : Bayesian Network Repository , 23
problème	(FEEDBACK) : Feedback Arc Set Problem , 9
problème	(GENGEN) : reconstruction de réseaux de régulations de gènes à partir de données d'expression et de données de polymorphisme , 25
problème	(HAPLOTYPE) : problème de reconstruction d'haplotypes dans des pedigrees complexes , 22
problème	(MAP) : Maximum a Posteriori Hypothesis , 21
problème	(MAXCLIQUE) : recherche d'une clique de cardinalité maximum dans un graphe , 6
problème	(MENDEL) : correction d'erreurs Mendéliennes de génotypage , 7
problème	(MPE) : Most Probable Explanation , 21
problème	(PR) : Probability of Evidence , 21
problème	(PROTEIN) : calcul d'un score de similarité entre deux protéines , 6
problème	(RNA) : recherche de motifs structurés de petits ARNs , 9
problème	(SPOT5) : sélection des prises de vue d'un satellite d'observation terrestre , 15
problème	(STEINER) : Steiner Triple Covering Problem , 9
problème	(TAGSNP) : sélection de marqueurs représentatifs d'une population d'individus , 15
problème	(WAREHOUSE) : Uncapacited Warehouse Location Problem , 7