

# Microstructures pour CSP d'arité quelconque <sup>\*</sup> †

Achref El Mouelhi   Philippe Jégou   Cyril Terrioux

Aix-Marseille Université, LSIS UMR 7296

Avenue Escadrille Normandie-Niemen

13397 Marseille Cedex 20 (France)

{achref.elmouelhi, philippe.jegou, cyril.terrioux}@lsis.org

## Résumé

Il est admis depuis longtemps que la notion de microstructure de CSP offre un cadre graphique utile pour l'étude de propriétés théoriques. Malheureusement, cette représentation graphique est restreinte aux seuls CSP binaires. Aussi, dans cette contribution, nous proposons d'étendre cette notion fondée sur les graphes, aux CSP d'arité quelconque. Cette approche qui évite le recours aux hypergraphes permet ainsi de disposer d'une littérature bien plus riche comme l'est celle de la Théorie (Algorithmique) des Graphes en comparaison à celle offerte par la Théorie des Hypergraphes. Nous introduisons trois définitions possibles de microstructures basées sur les codages binaires d'un CSP non binaire. Nous montrons également que ces représentations peuvent former un nouvel outil théorique pour généraliser certains résultats introduits au niveau des classes polynomiales de CSP binaires. Nous pensons que ces représentations pourraient être utiles à la communauté pour des développements de nature théorique, d'une part pour l'extension de résultats existants, mais aussi pour produire des résultats originaux pour les CSP d'arité quelconque.

## 1 Préliminaires

Les problèmes de satisfaction des contraintes (CSP pour Constraint Satisfaction Problems en anglais [29]) présente un moyen efficace pour la modélisation et la résolution de certains problèmes en Recherche Opérationnelle et en Intelligence Artificielle.

Formellement, une instance CSP est un triplet  $P = (X, D, C)$ , où  $X = \{x_1, \dots, x_n\}$  est un ensemble fini de  $n$  variables,  $D = \{d_1, \dots, d_n\}$  est un ensemble de

domaines finis de valeurs, un pour chaque variable et  $C = \{c_1, \dots, c_e\}$  est un ensemble fini de contraintes. Chaque contrainte  $c_i$  est un couple  $(S(c_i), R(c_i))$ , où  $S(c_i) = \{x_{i_1}, \dots, x_{i_k}\} \subseteq X$  est la portée (aussi appelée « scope ») de la contrainte.  $|S(c_i)|$  est l'arité de la contrainte  $c_i$ , c'est-à-dire, le nombre de variables sur lesquelles la contrainte  $c_i$  porte.  $R(c_i) \subseteq d_{i_1} \times \dots \times d_{i_k}$  est sa relation de compatibilité, chaque combinaison de valeurs d'une relation  $c_i$  est appelée tuple et sera notée  $t$ .

Nous supposons que toute variable apparaît dans la portée d'au moins une contrainte et que toutes les relations sont en extension, c'est-à-dire que les relations contiennent la liste des tuples autorisés. Si la contrainte est d'arité deux, alors elle est dite binaire et elle sera notée  $c_{ij}$  avec  $S(c_{ij}) = \{x_i, x_j\}$ . Si toutes les contraintes sont binaires, le CSP est dit *binnaire*. La structure d'un réseau de contraintes est représentée par un hypergraphe (un graphe pour le cas binaire) noté  $H(P) = (X, C)$ . Les sommets de  $H(P)$  correspondent aux variables et les hyperarêtes aux portées des contraintes. Une affectation d'un sous-ensemble de  $X$  est dite cohérente (ou consistante) si elle ne viole aucune contrainte. Une solution est une affectation cohérente portant sur toutes les variables de  $X$ . Le problème du test d'existence d'une solution est NP-complet. De ce fait, de nombreux travaux ont été réalisés pour rendre la résolution des instances plus efficace en pratique, en utilisant des versions de backtrack améliorées, des techniques de filtrage par consistance partielle, et des heuristiques d'ordonnement des variables, des valeurs, voire des contraintes. Une seconde piste intéressante consiste en l'étude des *classes polynomiales* (dites aussi *traitables*, de l'anglais « tractable »). Une classe polynomiale est un ensemble infini d'instances défini par des restrictions soit

<sup>\*</sup>Ce travail est soutenu par l'Agence Nationale de la Recherche dans le cadre du projet TUPLES (ANR-2010-BLAN-0210).

<sup>†</sup>Ce travail a été présenté dans sa version anglaise lors de la conférence SARA 2013 [14]

sur la portée des contraintes ou du graphe de contraintes, soit sur les relations associées aux contraintes. Les instances d'une telle classe peuvent être résolues en temps polynomial, et souvent la communauté s'accorde pour imposer que le test d'appartenance à une telle classe soit réalisable également en temps polynomial. Par exemple, les CSP binaires ayant un graphe de contraintes acyclique sont résolubles en temps linéaire [16]. Ce résultat a été par la suite généralisé aux CSP d'arité quelconque [20]. Des méthodes de résolution (dont notamment Tree Clustering [11]) ont été fondées sur de telles propriétés. Certaines ont montré l'intérêt pratique de ce type d'approches [25].

La Théorie des Graphes et la Théorie des Hypergraphes ont aussi permis de mettre en évidence des classes polynomiales sur la base de restrictions sur les relations de compatibilité pour le cas des CSP binaires. Elles se sont parfois appuyées sur une représentation appelée *graphe de microstructure* :

**Définition 1 (microstructure)** *Étant donné un CSP  $P = (X, D, C)$ , la microstructure de  $P$  est un graphe non orienté  $\mu(P) = (V, E)$  avec :*

- $V = \{(x_i, v_i) : x_i \in X, v_i \in d_i\}$ ,
- $E = \{ \{(x_i, v_i), (x_j, v_j)\} \mid i \neq j, c_{ij} \notin C \text{ ou } c_{ij} \in C, (v_i, v_j) \in R(c_{ij}) \}$ .

La transformation d'un CSP vers sa microstructure est réalisable en temps polynomial, la résolution du CSP se ramenant alors au fameux problème de la *Clique* [17], comme l'indique le théorème 1 [24] :

**Théorème 1** *Une affectation des variables d'un CSP binaire  $P$  est une solution ssi cette affectation est une clique de taille  $n$  (le nombre de variables) dans  $\mu(P)$ .*

Dans la littérature, plusieurs travaux ont mis en évidence l'intérêt que recèle l'étude des microstructures pour mettre en évidence de nouvelles classes polynomiales. Elles sont parfois fondées sur la Théorie (Algorithmique) des Graphes. Ainsi, en exploitant un résultat important de Gavril [18], il est montré que si le graphe de microstructure est *triangulé* [19], le CSP peut être résolu en temps polynomial, car cela revient à résoudre le problème de la *Clique*, qui est NP-complet, mais de complexité linéaire pour le cas des graphes triangulés. Par la suite, Cohen a appliqué la même approche, dans [6], et a montré que les CSP dont le graphe complémentaire de la microstructure est triangulé définissent une classe polynomiale, résultat qui se déduit immédiatement de [18] et de [24].

Dans la continuité de ces travaux, Salamon et Jeavons [31] ont généralisé le résultat de [24], aux *graphes parfaits* (les graphes triangulés sont parfaits) en s'appuyant sur des résultats issus de la Conjecture de Berge sur les graphes parfaits [4, 5].

Plus récemment encore, dans [9], Cooper et al. ont introduit une classe polynomiale appelée *BTP* pour *Broken Triangle Property*. BTP permet de capter certaines classes polynomiales comme les CSP binaires acycliques. Et dernièrement, El Mouelhi et al. dans [15] ont présenté des nouveaux résultats fondés sur la microstructure, où ils montrent que des algorithmes usuels de résolution de CSP comme Backtrack (BT), Forward Checking (FC [21]) ou Real Full Look-ahead (RFL [27]) notamment, sont de complexité polynomiale sur la classe d'instances des CSP ayant une microstructure possédant un nombre polynomial de cliques maximales. Au-delà, l'étude des microstructures a également montré son intérêt dans des domaines voisins. Par exemple, pour le problème de comptage de solutions [1], ou encore l'étude des symétries dans les CSP binaires [7, 26].

Il est clair que la microstructure constitue un outil très utile pour l'étude théorique des CSP. Toutefois, cette notion n'a pu être exploitée que dans la limite des CSP binaires. Le complément de la microstructure pour le cas non binaire a été introduite par Cohen dans [6] sans pour autant aboutir à des résultats. Cette généralisation qui s'appuie sur les hypergraphes est définie de la façon suivante :

**Définition 2 (Complément de la microstructure)**

*Étant donné un CSP  $P = (X, D, C)$ , le complément de la microstructure de  $P$  est un hypergraphe  $\overline{\mathcal{M}}(P) = (V, E)$  avec :*

- $V = \{(x_i, v_i) : x_i \in X, v_i \in d_i\}$ ,
- $E = E_1 \cup E_2$  tels que
  - $E_1 = \{ \{(x_i, v_j), (x_i, v_{j'})\} \mid x_i \in X \text{ et } j \neq j'\}$
  - $E_2 = \{ \{(x_{i_1}, v_{i_1}), \dots, (x_{i_k}, v_{i_k})\} \mid c_i \in C, S(c_i) = \{x_{i_1}, \dots, x_{i_k}\} \text{ et } (v_{i_1}, \dots, v_{i_k}) \notin R(c_i) \}$ .

Nous pouvons facilement constater que pour le cas binaire, la définition de Cohen correspond exactement au complément de la microstructure présentée dans [24]. Pour le cas non binaire, malheureusement, la notion de complément d'un hypergraphe ne semble pas avoir été étudié dans la littérature, du moins à notre connaissance. En fait, cette notion pose plusieurs questions dont la principale consiste à savoir s'il faudrait considérer toutes les hyperarêtes qui correspondent aux relations universelles, à l'image de la notion de complémentaire de graphe dans le cas binaire. Mais dans ce cas, la taille de l'hypergraphe serait potentiellement exponentielle en fonction de la taille de l'instance. Pour cette raison, vraisemblablement, cette définition n'a pas été exploitée dans la littérature. Ainsi, exploiter la définition de la microstructure basée sur les hypergraphes semble être plus difficile que l'exploitation des microstructures fondées sur les graphes.

De plus, la littérature de la Théorie des Graphes est clairement plus étendue que celle de la Théorie des Hypergraphes, avec pour conséquence de disposer ainsi d'un plus grand nombre de résultats théoriques ainsi que d'algorithmes.

Dans cette contribution, pour étendre cette notion aux CSP d'arité quelconque, nous proposons une approche différente de celle de [6], puisqu'elle s'appuie tout simplement sur la notion de graphe. A cette fin, nous présentons ici trois types possibles de microstructures basées respectivement sur les différents codages binaires des CSP non binaires : la représentation duale [10], la transformation dite par variable cachée [30] et le codage mixte. Nous étudions les propriétés de base de ces représentations, en suggérant différentes pistes pour leur exploitation théorique, notamment en vue de l'extension de classes polynomiales au cas des CSP d'arité quelconque.

Dans la section suivante, nous introduisons les différentes possibilités de microstructures pour les CSP d'arité quelconque. Ensuite, nous proposons une première exploitation de ces microstructures pour l'étude des classes polynomiales tandis que la dernière section est constituée par la conclusion de cet article.

## 2 Microstructures pour les CSP n-aires

La première évocation de microstructure pour le cas des CSP non binaires a été proposée par Cohen dans [6] et elle est basée sur les hypergraphes. Dans ce qui suit, nous présentons différentes alternatives à cette première approche, toutes basées sur des graphes simples. Chacune est inspirée des différents codages permettant de convertir un CSP non binaire en CSP binaire : il s'agit du codage *dual*, du codage *par variable cachée* et du codage *mixte* (parfois appelé codage *combiné*).

### 2.1 Microstructure basée sur le Codage Dual

Le codage dual, dans le domaine de la Programmation par Contraintes, a été employé pour la première fois dans [10]. Dans la Théorie des (Hyper)Graphes, il est appelé *Line Graph* et il est basé sur la transformation d'un hypergraphe en graphe. Dans la communauté CSP, il est appelé *Graphe Dual* et aussi *Intergraphe* dans [23], mais cette représentation avait précédemment été utilisée dans la Théorie des Bases de Données Relationnelles sous le vocable de *Qual Graphs* [2]. Dans ce codage, les variables correspondent aux contraintes du problème originel et elles sont généralement dites *variables duales*. Le domaine de chaque variable duale est exactement l'ensemble de tuples autorisés par la contrainte. Dans la représentation duale,

des contraintes binaires relient deux variables duales si elles partagent au moins une variable, c'est-à-dire, si l'intersection de leurs portées n'est pas vide. Cette représentation permet de définir une représentation binaire équivalente au problème non binaire de départ, au sens où il existe une bijection entre les ensembles de solutions du CSP de départ et de celui de la représentation duale. La définition de la microstructure associée, notée *DR-Microstructure* correspond ainsi à la microstructure de ce CSP binaire équivalent :

**Définition 3 (DR-Microstructure)** *Étant donné un CSP  $P = (X, D, C)$  d'arité quelconque, la Microstructure basée sur la Représentation Duale de  $P$ , dite DR-Microstructure, est un graphe non orienté  $\mu_{DR}(P) = (V, E)$  avec :*

- $V = \{(c_i, t_i) : c_i \in C, t_i \in R(c_i)\},$
- $E = \{ \{(c_i, t_i), (c_j, t_j)\} \mid i \neq j, t_i[S(c_i) \cap S(c_j)] = t_j[S(c_i) \cap S(c_j)] \}$

où  $t[Y]$  est la restriction de  $t$  aux variables de  $Y$ .

Notons que cette définition a été initialement introduite dans [15]. Comme pour la microstructure des CSP binaires, il existe un lien direct entre les cliques et les solutions.

**Théorème 2** *Un CSP  $P$  a une solution ssi  $\mu_{DR}(P)$  a une clique de taille  $e$  (le nombre de contraintes).*

**Preuve :** Par construction,  $\mu_G(P)$  est un graphe  $e$ -parti, et toute clique contient au plus un sommet  $(c_i, t_i)$  pour chaque contrainte  $c_i \in C$ . Donc, une  $e$ -clique de  $\mu_{DR}(P)$  correspond exactement à une clique avec un seul sommet  $(c_i, t_i)$  de chaque contrainte  $c_i \in C$ . De plus, chaque couple de sommets  $(c_i, t_i)$  et  $(c_j, t_j)$  reliés par une arête satisfait  $t_i[S(c_i) \cap S(c_j)] = t_j[S(c_i) \cap S(c_j)]$ . Donc, tous les sommets  $(c_i, t_i)$  d'une clique sont deux à deux adjacents et donc compatibles, et en particulier, une  $e$ -clique de  $\mu_G(P)$  correspond exactement à  $e$  tuples  $t_i$  autorisés par toutes les contraintes, ce qui est équivalent à une solution de  $P$ .  $\square$

Considérons l'exemple suivant qui sera utilisé par la suite.

**Exemple 1**  $P = (X, D, C)$  a cinq variables  $X = \{x_1, \dots, x_5\}$  avec les domaines suivants :  $D = \{d_1, d_2, d_3, d_4, d_5\}$  avec  $d_1 = \{a, a'\}$ ,  $d_2 = \{b\}$ ,  $d_3 = \{c\}$ ,  $d_4 = \{d, d'\}$  et  $d_5 = \{e\}$ .  $C = \{c_1, c_2, c_3, c_4\}$  est un ensemble de quatre contraintes avec  $S(c_1) = \{x_1, x_2\}$ ,  $S(c_2) = \{x_2, x_3, x_5\}$ ,  $S(c_3) = \{x_3, x_4, x_5\}$  et  $S(c_4) = \{x_2, x_5\}$ . Les relations associées aux contraintes sont données par les tables suivantes :

$R(c_1)$		$R(c_2)$			$R(c_3)$			$R(c_4)$	
$x_1$	$x_2$	$x_2$	$x_3$	$x_5$	$x_3$	$x_4$	$x_5$	$x_2$	$x_5$
$a$	$b$	$b$	$c$	$e$	$c$	$d$	$e$	$b$	$e$
$a'$	$b$				$c$	$d'$	$e$		

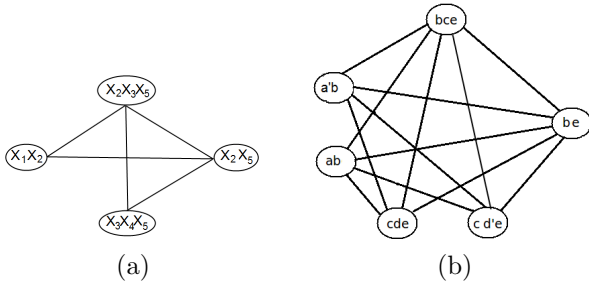


FIGURE 1 – Graphe Dual (a) et DR-Microstructure (b) du CSP de l'exemple 1.

La DR-Microstructure de cet exemple est présentée par la figure 1. Nous avons 4 contraintes, donc  $e = 4$ .

Conformément au théorème 2, une solution de  $P$  est une clique de taille 4, ce qui est le cas de  $\{ab, bce, be, cde\}$ . Dans les exemples, nous noterons directement par  $t_i$  le sommet  $(c_i, t_i)$ .

En supposant que les relations des instances sont données sous forme de tables, la taille de la DR-Microstructure est bornée par un polynôme fonction de la taille du CSP, c'est-à-dire  $|E| \leq |V|^2$  avec  $|V| = \sum_{c_i \in C} |\{t_i \in R(c_i)\}|$ . De plus, il est clair que le calcul de la DR-Microstructure peut se réaliser en temps polynomial.

Il est bien connu que dans le graphe dual certaines arêtes redondantes peuvent être éliminées sans toucher à l'équivalence avec le problème de départ [22, 23]. Sur cette base, nous pouvons ainsi définir un ensemble de microstructures différentes, voisines de la DR-Microstructure. Dans [23], il est montré que pour un CSP d'arité quelconque, il existe un ensemble de CSP binaires équivalents construits sur la base de l'ensemble des intergraphes, le maximal d'entre eux correspondant au codage dual. En considérant cet ensemble de graphes partiels, nous pouvons étendre la définition précédente de DR-Microstructure :

**Définition 4 (DSR-Microstructure)** *Étant donné un CSP  $P = (X, D, C)$  (d'arité quelconque) et un de ses intergraphes  $(C, F)$ , la Microstructure basée sur la représentation des graphes partiels duaux de  $P$  est un graphe non orienté  $\mu_{DSR}(P, (C, F)) = (V, E)$  avec :*

- $V = \{(c_i, t_i) : c_i \in C, t_i \in R(c_i)\}$ ,
- $E = E_1 \cup E_2$  tels que
  - $E_1 = \{ \{(c_i, t_i), (c_j, t_j)\} \mid \{(c_i, c_j) \in F, t_i[S(c_i) \cap S(c_j)] = t_j[S(c_i) \cap S(c_j)]\} \}$
  - $E_2 = \{ \{(c_i, t_i), (c_j, t_j)\} \mid \{(c_i, c_j) \notin F\} \}$ .

Avec cette représentation, nous disposons des mêmes propriétés en termes de taille de la DSR-Microstructure puisqu'elle demeure bornée par le même polynôme que la DR-Microstructure, et le cal-

cul de cette microstructure sera également réalisable en temps polynomial.

On peut constater que la DR-Microstructure est un graphe partiel de la DSR-Microstructure vu que pour chaque arête supprimée dans le graphe dual, une relation universelle la remplacera dans la DSR-Microstructure. Nous pouvons facilement constater que le résultat sur les cliques est toujours vrai :

**Théorème 3** *Un CSP  $P$  possède une solution ssi  $\mu_{DSR}(P, (C, F))$  possède une clique de taille  $e$ .*

La démonstration de ce théorème est presque identique à celle portant sur la DR-Microstructure.

## 2.2 Microstructure basée sur le Codage par Variable Cachée

Le codage basé sur la variable cachée (*Hidden Transformation en anglais*) est inspiré par Peirce [28] (cité dans [30]). Dans cette transformation, l'ensemble de variables contient les variables originelles de  $X$  plus l'ensemble des variables duales issues de  $C$ . Les nouvelles contraintes binaires vont relier une variable duale à une variable originelle si la variable originelle appartient à la portée de la variable duale. La microstructure sera donc basée sur cette représentation binaire.

**Définition 5 (HT-Microstructure)** *Étant donné un CSP  $P = (X, D, C)$  (d'arité quelconque), la Microstructure basée sur la Transformation par Variable Cachée de  $P$  est un graphe non orienté  $\mu_{HT}(P) = (V, E)$  avec :*

- $V = S_1 \cup S_2$  tel que :
  - $S_1 = \{(x_i, v_i) : x_i \in X, v_i \in d_i\}$ ,
  - $S_2 = \{(c_i, t_i) : c_i \in C, t_i \in R(c_i)\}$ ,
- $E = \{ \{(c_i, t_i), (x_j, v_j)\} \mid \text{soit } x_j \in S(c_i) \text{ et } v_j = t_i[x_j] \text{ soit } x_j \notin S(c_i) \}$ .

La figure 2 représente la HT-Microstructure basée sur cette transformation pour le CSP de l'exemple 1. Nous pouvons constater que la HT-Microstructure est un graphe biparti car nous avons d'un côté les valeurs des domaines, et de l'autre, les tuples des relations. Pour rappel, une *biclique* est un sous graphe biparti complet, c'est-à-dire que chaque sommet du premier ensemble est connecté à tous les sommets du second, et réciproquement. Une biclique entre deux ensembles de sommets de taille respective  $i$  et  $j$  est notée  $K_{i,j}$ . Dans la HT-Microstructure, une solution correspond à une biclique particulière comme le montrera le théorème 4 qui se déduit directement des deux lemmes suivants.

**Lemme 1** *Dans une HT-Microstructure, une biclique  $K_{n,e}$  avec  $e$  tuples appartenant à des relations deux*

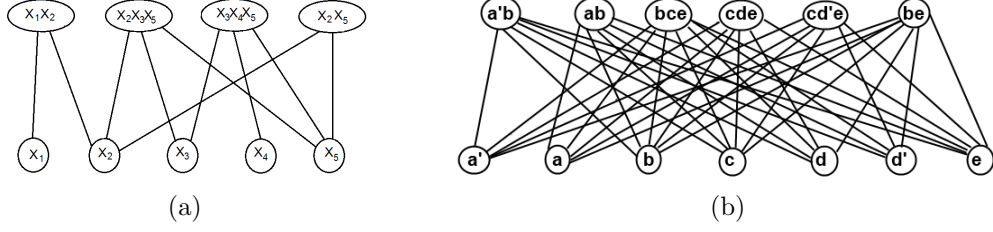


FIGURE 2 – Graphe de la Hidden Transformation (a) et HT-Microstructure (b) du CSP de l'exemple 1.

à deux différentes ne peut pas contenir deux valeurs différentes d'un même domaine.

**Preuve :** Supposons qu'une biclique  $K_{n,e}$  avec  $e$  tuples appartenant à des relations deux à deux différentes contienne deux valeurs différentes  $v_j$  et  $v'_j \in d_j$ . Il existe alors au moins une contrainte  $c_i$  telle que  $x_j \in S(c_i)$  et par conséquent  $t_i[x_j] = v_j, v'_j$  ou une autre valeur  $v'_j$ . Dans les trois cas, ceci contredit que  $v_j$  et  $v'_j$  sont dans une même biclique car il n'est pas possible d'avoir deux tuples d'une même relation.  $\square$

**Lemme 2** Dans une HT-Microstructure, une biclique  $K_{n,e}$  avec  $n$  valeurs issues de domaines différents ne peut pas contenir deux tuples différents d'une même relation.

**Preuve :** Supposons qu'une biclique  $K_{n,e}$  avec  $n$  valeurs appartenant à des variables deux à deux différentes contienne deux tuples  $t_i$  et  $t'_i$  d'une même contrainte  $c_i$ . Donc, il existe au moins une variable  $x_j$  telle que  $t_i[x_j] \neq t'_i[x_j]$ . Si  $v_j = t_i[x_j]$  et  $v'_j = t'_i[x_j]$  appartiennent toutes les deux à la biclique  $K_{n,e}$ , on a une contradiction car nous ne pouvons pas avoir deux valeurs d'une même variable.  $\square$

En utilisant ces deux lemmes, nous pouvons déduire que toute biclique  $K_{n,e}$  avec  $n$  valeurs et  $e$  tuples tels que chaque couple de valeurs appartient à un couple de variables différentes et chaque couple de tuples appartient à un couple de relations différentes, correspond à une affectation de toutes les variables qui satisfait toutes les contraintes. Nous pouvons dans ce cas énoncer le théorème suivant :

**Théorème 4** Étant donné un CSP  $P = (X, D, C)$  et sa HT-Microstructure  $\mu_{HT}(P)$ ,  $P$  possède une solution ssi  $\mu_{HT}(P)$  possède une biclique  $K_{n,e}$  avec  $n$  valeurs et  $e$  tuples tels que tous les tuples appartiennent à des relations deux à deux différentes et toutes les valeurs appartiennent à des domaines deux à deux différents.

En revenant à l'exemple précédent, nous pouvons facilement observer qu'une biclique ne correspond pas forcément à une solution. Bien que

$\{a, a', b, c, e, ab, ab', bce, be\}$  soit une biclique  $K_{5,4}$ , elle ne constitue pas pour autant une solution. Par contre,  $\{a, b, c, d, e, ab, bce, be, cde\}$  est une biclique  $K_{5,4}$  et est aussi une solution de  $P$ . Donc, l'ensemble de solutions n'est pas équivalent à l'ensemble des bicliques  $K_{n,e}$ . Cet ensemble est équivalent seulement à l'ensemble des bicliques  $K_{n,e}$  avec  $n$  valeurs et  $e$  tuples telles qu'aucun couple de valeurs (resp. de tuples) n'appartient à un même domaine (resp. à une même relation).

Comme pour la DR-Microstructure, la taille de la HT-Microstructure est bornée polynomialement par la taille du CSP :

- $|V| = \sum_{x_i \in X} |d_i| + \sum_{c_i \in C} |\{t_i \in R(c_i)\}|$  et
- $|E| \leq \sum_{x_i \in X} |d_i| \times \sum_{c_i \in C} |\{t_i \in R(c_i)\}|$ .

De plus, étant donnée une instance CSP, calculer sa HT-Microstructure peut aussi se réaliser en temps polynomial.

Il existe une autre façon de représenter la microstructure à partir du codage caché et qui est liée à une autre manière pour compléter la microstructure. Nous développons ce point ci-dessous.

### 2.3 Microstructure basée sur le Codage Mixte

Le codage mixte [32] d'un CSP non binaire, combine à la fois le codage dual et le codage par variable caché. C'est pour cette raison que dans la littérature nous le trouvons aussi sous le nom de codage *combiné*. Cette approche consiste à connecter les valeurs des variables duales aux valeurs des variables originelles, les valeurs des variables originelles étant connectées entre elles si elles n'appartiennent pas à un même domaine et les tuples entre eux s'ils sont compatibles :

**Définition 6 (ME-Microstructure)** Étant donné un CSP  $P = (X, D, C)$  d'arité quelconque, la Microstructure basée sur le codage Mixte de  $P$  est un graphe non orienté  $\mu_{ME}(P) = (V, E)$  avec :

- $V = S_1 \cup S_2$  tel que
  - $S_1 = \{(c_i, t_i) : c_i \in C, t_i \in R(c_i)\}$ ,
  - $S_2 = \{(x_j, v_j) : x_j \in X, v_j \in d_j\}$ ,
- $E = E_1 \cup E_2 \cup E_3$  tel que
  - $E_1 = \{ \{(c_i, t_i), (c_j, t_j)\} \mid i \neq j, t_i[S(c_i) \cap S(c_j)] = t_j[S(c_i) \cap S(c_j)] \}$

- $E_2 = \{ \{(c_i, t_i), (x_j, v_j)\} \mid \text{soit } x_j \in S(c_i) \text{ et } v_j = t_i[x_j] \text{ soit } x_j \notin S(c_i)\}$
- $E_3 = \{ \{(x_i, v_i), (x_j, v_j)\} \mid x_i \neq x_j\}$ .

La microstructure basée sur le codage mixte du CSP de l'exemple 1 est donnée dans la figure 3. Nous pouvons constater que dans ce codage, nous gardons le même ensemble de sommets que celui de la HT-Microstructure. Pour les arêtes, nous aurons les arêtes de la DR-Microstructure et de la HT-Microstructure, plus toutes les arêtes reliant les valeurs de domaines différents, et correspondent à un sous-graphe  $n$ -parti complet, une partie étant alors constituée des sommets d'un même domaine. Ceci aura une influence sur les liens entre définition des solutions et les propriétés en termes de graphes, comme le précise le lemme suivant :

**Lemme 3** *Dans une ME-Microstructure, une clique de  $n + e$  sommets ne peut pas contenir deux valeurs d'une même variable, ni deux tuples d'une même relation.*

**Preuve :** Soient  $v_i$  et  $v'_i$  deux valeurs du domaine d'une variable  $x_i$ . Par définition, les sommets correspondant à  $v_i$  et  $v'_i$  ne peuvent être adjacents et donc, ne peuvent figurer ensemble dans une même clique.

De même, deux tuples d'une même relation ne sont pas compatibles entre eux et ils ne peuvent figurer ensemble dans une même clique.  $\square$

En s'appuyant sur ce lemme, nous pouvons illustrer la relation entre cliques et solutions de CSP :

**Théorème 5** *Un CSP  $P$  possède une solution ssi  $\mu_{ME}(P)$  possède une clique de taille  $n + e$ .*

**Preuve :** Dans la ME-Microstructure, en se référant au lemme 3, une clique de  $n + e$  sommets contient exactement un sommet par variable et par contrainte. Donc, elle correspond à une affectation de  $n$  variables qui satisfait les  $e$  contraintes, et il s'agit donc d'une solution.  $\square$

Comme pour les autres microstructures, la taille de la ME-Microstructure est bornée par un polynôme fonction de la taille du CSP.

$$\begin{aligned} - |V| &= \sum_{x_i \in X} |d_i| + \sum_{c_i \in C} |\{t_i \in R(c_i)\}| \text{ et} \\ - |E| &\leq \sum_{x_i \in X} |d_i| \times \sum_{c_i \in C} |\{t_i \in R(c_i)\}| + \\ &\quad (\sum_{x_i \in X} |d_i|)^2 + (\sum_{c_i \in C} |\{t_i \in R(c_i)\}|)^2. \end{aligned}$$

De plus, calculer la ME-Microstructure d'un CSP est réalisable en temps polynomial.

## 2.4 Comparaisons entre microstructures

Tout d'abord, nous pouvons constater qu'aucune de ces microstructures ne peut être considérée comme une généralisation de la microstructure présentée pour

le cas binaire. En effet, étant donné un CSP binaire  $P$ , nous avons  $\mu(P) \neq \mu_{DR}(P), \mu(P) \neq \mu_{HT}(P)$  et  $\mu(P) \neq \mu_{ME}(P)$ . De plus, si la DR-Microstructure correspond exactement à la microstructure du dual, ni la HT-Microstructure, ni la ME-Microstructure ne correspondent à la microstructure du codage binaire associé, ceci étant dû à la façon dont les graphes ont été complétés. Par ailleurs, si finalement toutes ces microstructures peuvent être calculées en temps polynomial, d'un point de vue pratique, il semble difficile en général de les construire et de les manipuler efficacement, en particulier quand les contraintes ne sont pas données en extension et ceci même pour la microstructure de CSP binaires. Par contre, ce dernier point ne présente pas un obstacle pour l'étude théorique que nous désirons proposer. Il faut rappeler que notre but consiste à introduire un outil théorique qui nous permettra d'étudier les CSP d'arité quelconque. La section suivante présente quelques résultats préliminaires portant sur l'exploitation de ces représentations afin d'étendre aux CSP d'arité quelconque, des classes polynomiales définies initialement pour les seuls CSP binaires.

## 3 Quelques premiers résultats

Nous allons présenter maintenant quelques résultats qui peuvent être déduits de l'étude des microstructures. Pour cela, nous allons étudier trois classes polynomiales introduites pour les CSP binaires. Il s'agit de la classe des CSP ayant un nombre polynomial de cliques maximales [15], de la classe BTP [9] et enfin de la classe ZUT [8].

### 3.1 Microstructures et cliques maximales

Dans [15], il est montré que si le nombre de cliques maximales dans la microstructure d'un CSP binaire - noté  $\omega_{\#}(\mu(P))$  - est borné par un polynôme, les algorithmes comme BT, FC ou RFL, peuvent résoudre ce CSP en temps polynomial. Plus précisément, ce coût est borné par  $O(n^2 d \cdot \omega_{\#}(\mu(P)))$  pour BT et FC, et par  $O(ned^2 \cdot \omega_{\#}(\mu(P)))$  pour RFL. Donc, nous montrons ici comment étendre ce résultat aux CSP d'arité quelconque en exploitant ces microstructures. Ainsi, dans [15], ces résultats ont été généralisés aux CSP non binaires en exploitant la représentation duale et en utilisant les algorithmes nBT, nFC et nRFL, soit les versions non binaires de BT, FC et RFL. Plus précisément, en exploitant un ordre particulier pour l'affectation des variables, il a été montré que la complexité est bornée par  $O(nea \cdot d^a \cdot \omega_{\#}(\mu_{DR}(P)))$  pour nBT, et par  $O(nea \cdot r^2 \cdot \omega_{\#}(\mu_{DR}(P)))$  pour nFC et nRFL, où  $a$  est l'arité maximale des contraintes et  $r$  le nombre

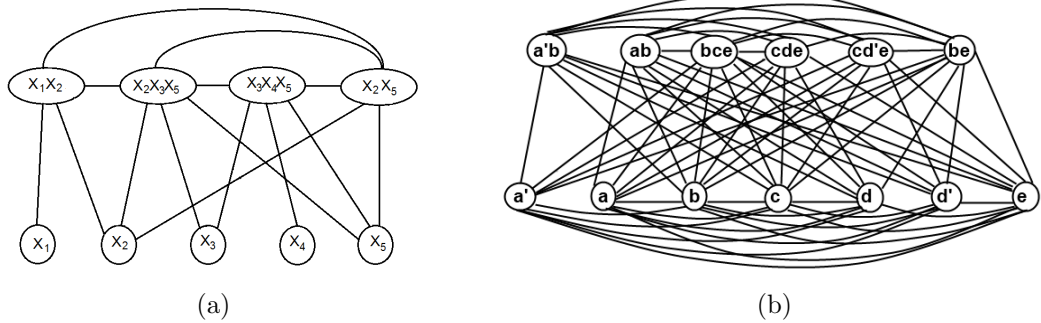


FIGURE 3 – Graphe de la Transformation Mixte (a) et ME-Microstructure (b) du CSP de l'exemple 1.

maximum de tuples dans une relation. En se basant sur la complexité en temps de ces algorithmes, et en se focalisant sur les classes de graphes ayant un nombre polynomial de cliques maximales, les auteurs ont pu facilement définir de nouvelles classes traitables. Parmi ces classes de graphes, nous pouvons citer les graphes *planaires*, les graphes *Toroïdaux*, les graphes *plongeables sur des surfaces* [12] et les graphes  $CSG^k$  [3]. Le résultat peut se résumer par le théorème suivant :

**Théorème 6** *Les CSP d'arité quelconque dont la DR-Microstructure est soit un graphe planaire, soit un graphe toroïdal, soit un graphe plongeable dans une surface ou soit un graphe CSG sont résolubles en temps polynomial.*

Pour la HT-Microstructure, un tel résultat ne peut pas exister. En effet, la HT-Microstructure est un graphe biparti et donc ses cliques maximales sont toutes de taille 2 puisque les cliques maximales sont limitées aux arêtes. Donc, le nombre de cliques maximales est toujours polynomial.

Pour la ME-Microstructure, un tel résultat ne peut pas plus exister, mais pour des raisons différentes. Par construction, les arêtes de l'ensemble  $E_3 = \{ \{(x_i, v_i), (x_j, v_j)\} \mid x_i \neq x_j \}$  de la définition 6 permettent toutes les combinaisons de valeurs possibles. Ceci rendra le nombre de cliques maximales exponentiel à l'exception des CSP monovalent (CSP dont la taille de tous les domaines est 1).

### 3.2 Microstructures et BTP

La propriété BTP (Broken Triangle Property) [9] définit une classe traitable pour les CSP binaires en exploitant des caractéristiques de la microstructure. La classe BTP recèle un certain intérêt car elle capte plusieurs classes polynomiales bien connues, comme les CSP binaires arborescents ainsi que des classes polynomiales relationnelles telles que RRM. La question

est donc : cette propriété peut-elle être étendue aux CSP d'arité quelconque en exploitant les caractéristiques de leurs microstructures ? Notons qu'une première évocation de cela apparaît dans [9]. Ici, nous essayons d'étendre ces travaux en utilisant les trois types de microstructures, mais auparavant, nous rappelons la propriété BTP :

#### Définition 7 (Broken-Triangle Property [9])

Un CSP binaire  $P$  satisfait la Broken Triangle Property (BTP) par rapport à un ordre sur les variables  $< si, pour tout triplet de variables  $(x_i, x_j, x_k)$  tel que  $x_i < x_j < x_k$ , si  $(v_i, v_j) \in R(c_{ij})$ ,  $(v_i, v_k) \in R(c_{ik})$  et  $(v_j, v'_k) \in R(c_{jk})$ , alors soit  $(v_i, v'_k) \in R(c_{ik})$ , soit  $(v_j, v_k) \in R(c_{jk})$ . Si  $(v_i, v'_k) \notin R(c_{ik})$  et  $(v_j, v_k) \notin R(c_{jk})$ , alors on a un triangle cassé sur  $x_k$ .$

Il est prouvé dans [9] que si un CSP binaire  $P$  satisfait BTP, trouver un ordre et résoudre le CSP peut s'effectuer en  $O(n^3 d^4 + ed^2)$ .

#### 3.2.1 DR-Microstructure

Pour étendre BTP aux CSP non binaires, une extension de BTP, appelée DBTP (pour Dual BTP), est proposée dans [13]. La définition, qui s'appuie sur la DR-Microstructure est rappelée ci-dessous :

#### Définition 8 (Dual Broken-Triangle Property)

Un CSP  $P = (X, D, C)$  vérifie la Dual Broken Triangle Property (DBTP) par rapport à un ordre  $<$  sur les contraintes si pour tout triplet de contraintes  $(c_i, c_j, c_k)$  tel que  $c_i < c_j < c_k$ , pour tout  $t_i \in R(c_i)$ ,  $t_j \in R(c_j)$  et  $t_k, t'_k \in R(c_k)$  tels que

- $t_i[S(c_i) \cap S(c_j)] = t_j[S(c_i) \cap S(c_j)]$
- $t_i[S(c_i) \cap S(c_k)] = t_k[S(c_i) \cap S(c_k)]$
- $t'_k[S(c_j) \cap S(c_k)] = t_j[S(c_j) \cap S(c_k)]$

alors

- soit  $t'_k[S(c_i) \cap S(c_k)] = t_i[S(c_i) \cap S(c_k)]$
- soit  $t_j[S(c_j) \cap S(c_k)] = t_k[S(c_j) \cap S(c_k)]$

La figure 4 illustre graphiquement cette définition, l'aspect formel de la définition 8 étant manifestement plus difficile à appréhender. En fait, il s'agit juste de remplacer les valeurs par les tuples et les variables par les contraintes pour avoir cette extension. Dans la figure 4 (a), nous pouvons observer la présence d'un triangle cassé sur la contrainte  $c_3$ . Donc, si on considère l'ordre suivant sur les contraintes  $c_1 \prec c_2 \prec c_3$ , le CSP en question ne satisfait pas DBTP par rapport à  $\prec$ . Au contraire, dans la figure 4(b), si  $t_1$  et  $t'_3$  (arête bleue) ou  $t_2$  et  $t_3$  (arête rouge) sont compatibles, alors le CSP satisfait DBTP par rapport à  $\prec$ .

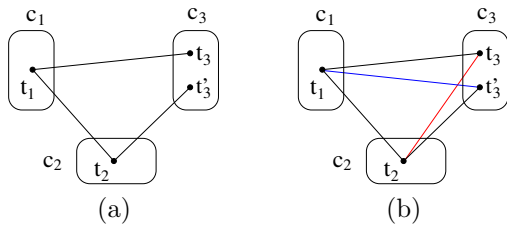


FIGURE 4 – DR-Microstructure d'un CSP n-aire satisfaisant BTP sur sa représentation duale.

Au même titre que pour BTP dont on sait que cette classe inclut les CSP binaires arborescents, dans [13] il est montré que dans le cas non binaire, les CSP dont l'hypergraphe de contraintes est  $\beta$ -acyclique sont aussi inclus dans DBTP. Certains autres résultats sur BTP restent vrais aussi pour DBTP. Plus de détails peuvent être trouvés dans [13].

### 3.2.2 HT et ME-Microstructures

Pour la HT-Microstructure, nous pouvons immédiatement constater que les triangles cassés ne peuvent pas exister car le graphe est biparti. Pour analyser BTP sur cette microstructure, on doit considérer les contraintes universelles (i.e. avec des relations universelles) entre sommets du graphe de contraintes résultant de la transformation par variable cachée. Aussi, nous étudions directement la ME-Microstructure car elle possède les mêmes sommets et par définition, elle est déjà complétée avec des arête entre ces sommets.

Considérons maintenant la HT-Microstructure. Étendre BTP est bien plus compliqué ici car nous devons considérer au moins quatre cas différents de triangles, car contrairement à BTP et DBTP sur la DR-Microstructure, nous trouvons deux types de sommets : des valeurs ou des tuples. De plus, puisque pour BTP, nous devons considérer des ordres tels que  $i < j < k$ , en fait, nous devons considérer six sortes de triangles sachant que les deux premiers termes de chaque triangles peuvent être permutés : (1)  $x_i < x_j < x_k$ , (2)  $x_i < x_j < c_k$ , (3)  $x_i < c_j < x_k$  (ou

$c_i < x_j < x_k$ ), (4)  $x_i < c_j < c_k$  (ou  $c_i < x_j < c_k$ ), (5)  $c_i < c_j < x_k$ , (6)  $c_i < c_j < c_k$ . Remarquons que le premier cas revient à vérifier la condition classique de BTP et que le dernier correspond à DBTP.

On peut noter l'existence d'un lien entre DR-Microstructure et ME-Microstructure pour BTP. En effet, étant donné un CSP, la présence d'un triangle cassé dans sa DR-Microstructure aura une influence sur l'ordre des contraintes. Elle aura aussi une influence sur l'ordre dans le cadre de sa ME-Microstructure, car celui-ci dépend à la fois des variables et des contraintes. Ceci nous conduit au théorème suivant :

**Théorème 7** *Si un CSP  $P$  satisfait BTP en considérant sa ME-Microstructure par rapport à un ordre sur les contraintes, alors il existe un ordre sur les contraintes pour lequel  $P$  satisfait BTP en considérant sa DR-Microstructure.*

### 3.3 Microstructures et Classe ZUT

Dans ce qui précède, il apparaît que la DR-Microstructure semble être la microstructure la plus intéressante. Ceci restera-t-il vrai pour les autres classes polynomiales ? Pour répondre à cette question, nous allons procéder de la même façon en étudiant cette fois-ci la classe polynomiale dite *Zéro-Un-Tous* que nous noterons *ZUT* (*Zero/One/All* en anglais) introduite par Cooper et al. dans [8], et dont nous rappelons la définition :

**Définition 9 (ZUT [8])** *Un CSP binaire  $P$  est dit ZUT si pour chaque contrainte  $c_{ij}$  de  $C$ , pour chaque valeur  $v_i \in d_i$ ,  $c_{ij}$  vérifie l'une des conditions suivantes :*

- (ZERO)  $\forall v_j \in d_j, (v_i, v_j) \notin R(c_{ij})$ ,
- (UN) *il existe une seule valeur  $v_j \in d_j$  telle que  $(v_i, v_j) \in R(c_{ij})$ ,*
- (TOUS)  $\forall v_j \in d_j, (v_i, v_j) \in R(c_{ij})$ .

Cette propriété peut être représentée graphiquement en utilisant la microstructure. Dans le cas de la DR-Microstructure, il semble facile d'appliquer le même principe que pour la microstructure classique des CSP binaires. Pour cela, par rapport à la définition de ZUT pour le cas binaire, il suffit seulement de remplacer les valeurs par des tuples. Donc, satisfaire ZUT en considérant la DR-Microstructure dépendra des relations de l'instance.

Pour la HT-Microstructure, les arêtes relient les tuples (sommets associés aux variables duales) aux valeurs (sommets associés aux variables originelles de l'instance). Nous analysons la situation selon deux directions : des tuples vers les valeurs et des valeurs vers les tuples.



- *Arêtes des tuples vers les valeurs.* Il existe deux façons pour connecter un tuple  $t_i$  à une valeur  $v_j$ . Si la variable  $x_j$  de cette valeur appartient à la portée de la contrainte  $c_i$  de ce tuple, alors  $t_i$  n'est connecté qu'à une seule valeur de  $x_j$ , à savoir  $v_j$ . Si la variable  $x_j$  n'appartient pas à la portée de la contrainte  $c_i$  de tuple  $t_i$ , alors  $t_i$  est connecté à toutes les valeurs de  $x_j$ . Donc, nous avons seulement des connexions de type « Un » ou « Tous ».
- *Arêtes des valeurs vers les tuples.* Étant donnée une contrainte associée à la HT-Microstructure, une valeur est connectée aux tuples dans lesquels elle apparaît. Nous discutons ci-dessous les différentes possibilités :
  - connexion « Zéro ». Une valeur n'est pas supportée par des tuples. C'est l'équivalent de la connexion « Zéro » de la définition de ZUT pour les CSP binaires.
  - connexion « Un ». Une valeur est supportée par un seul tuple. C'est l'équivalent de la connexion « Un » de la définition de ZUT pour les CSP binaires.
  - connexion « Tous ». Une valeur est supportée par tous les tuples de la contrainte. C'est aussi l'équivalent de la connexion « Tous » de la définition de ZUT pour les CSP binaires.

Il est donc possible qu'une instance satisfasse la propriété ZUT dans la HT-Microstructure.

Enfin, pour la ME-Microstructure, nous devons vérifier les conditions définies seulement pour la DR et HT-Microstructures car les relations entre les variables sont universelles (donc, la connexion « Tous » est toujours vraie).

Pour conclure, par construction, rien ne s'oppose à ce que les conditions de la propriété ZUT soient satisfaites, mais elles s'avèrent cependant très restrictives, comme d'ailleurs pour le cas binaire.

## 4 Conclusion

Dans cette contribution, nous avons introduit le concept de microstructure pour le cas des CSP d'arité quelconque. Si ce concept pour le cas binaire est désormais bien établi et utilisé comme un outil théorique, notamment pour la définition de nouvelles classes polynomiales pour les CSP, pour le cas des CSP non binaires, la notion de microstructure n'était pas clairement établie auparavant. Nous avons travaillé à la définition explicite de cette notion pour les CSP d'arité quelconque en nous basant sur différents codages binaires de CSP étudiés précédemment dans la littérature, de sorte à disposer de microstructures graphiques, plutôt qu'en nous appuyant sur la

notion d'hypergraphe comme proposé dans [6]. Nous avons présenté trois types de microstructures : la DR-Microstructure (raffinée avec la DSR-Microstructure), la HT-Microstructure et la ME-microstructure, qui sont inspirées respectivement de la représentation duale, de la transformation par variable cachée et de l'approche mixte. Pour le cas binaire, aucune de ces trois microstructures ne correspond à la microstructure classique, de sorte qu'aucune d'entre elles ne peut être considérée comme une généralisation de la notion classique. Pour montrer l'intérêt de ce travail, nous avons exploité ces microstructures afin d'étendre certaines classes polynomiales définies au niveau des CSP binaires sur la base de leurs microstructures. Un premier résultat porte sur le cas des microstructures de CSP binaires dont le nombre de cliques maximales est borné par un polynôme, cas connus pour être traitables en temps polynomial par les algorithmes usuels de résolution de CSP binaires, comme BT, FC ou RFL. Ces classes s'étendent naturellement aux CSP non binaires dont les microstructures satisfont les mêmes propriétés. Nous avons également montré comment la classe BTP peut naturellement être étendue aux CSP non binaires, l'étude de cette classe étant particulièrement intéressante car elle inclut plusieurs classes polynomiales bien connus de CSP binaires, et qui par cette approche sont désormais définies en termes de contraintes d'arité quelconque.

Nous espérons que ces outils seront utilisés au niveau non binaire comme cela a pu être le cas au niveau binaire pour la microstructure classique. Par contre, il est clair qu'une utilisation pratique de ces notions semble plus que difficile, en particulier pour le cas des contraintes pour lesquelles les relations ne sont pas définies en extension. Toutefois, ces microstructures pourraient être utilisées virtuellement comme dans le cas notamment des filtrages par cohérence partielle.

## Références

- [1] Ola Angelsmark and Peter Jonsson. Improved algorithms for counting solutions in constraint satisfaction problems. In *CP*, pages 81–95, 2003.
- [2] P.A. Bernstein and N. Goodman. The power of natural semijoins. *SIAM J. Comput.*, 10-4 :751–771, 1981.
- [3] A. Chmeiss and P. Jégou. A generalization of chordal graphs and the maximum clique problem. *Information Processing Letters*, 62 :111–120, 1997.
- [4] M. Chudnovsky, X. Liu G. Cornuejols, P.D. Seymour, and K. Vuskovic. Recognizing berge graphs. *Combinatorica*, 25 :143–186, 2005.

- [5] M. Chudnovsky, N. Robertson, P.D. Seymour, and R.Thomas. The strong perfect graph theorem. *Ann. Math.*, 164 :51–229, 2006.
- [6] David A. Cohen. A New Classs of Binary CSPs for which Arc-Consistency Is a Decision Procedure. In *CP*, volume 2833 of *LNCS*, pages 807–811. Springer, 2003.
- [7] David A. Cohen, Peter Jeavons, Christopher Jefferson, Karen E. Petrie, and Barbara M. Smith. Symmetry definitions for constraint satisfaction problems. *Constraints*, 11(2-3) :115–137, 2006.
- [8] M. Cooper, D. Cohen, and P. Jeavons. Characterising Tractable Constraints. *Artificial Intelligence*, 65(2) :347–361, 1994.
- [9] M. Cooper, Peter Jeavons, and Andras Salamon. Generalizing constraint satisfaction on trees : hybrid tractability and variable elimination. *Artificial Intelligence*, 174 :570–584, 2010.
- [10] R. Dechter and J. Pearl. Network-based heuristics for constraint satisfaction problems. *Artificial Intelligence*, 34 :1–38, 1987.
- [11] R. Dechter and J. Pearl. Tree-Clustering for Constraint Networks. *Artificial Intelligence*, 38 :353–366, 1989.
- [12] Vida Dujmovic, Gasper Fijavz, Gwenaël Joret, Thom Sulanke, and David R. Wood. On the maximum number of cliques in a graph embedded in a surface. *European J. Combinatorics*, 32(8) :1244–1252, 2011.
- [13] Achref El Mouelhi, Philippe Jégou, and Cyril Terrioux. A hybrid tractable class for non-binary cps. In *ICTAI*, pages 947–954, 2013.
- [14] Achref El Mouelhi, Philippe Jégou, and Cyril Terrioux. Microstructures for cps with constraints of arbitrary arity. In *SARA*, 2013.
- [15] Achref El Mouelhi, Philippe Jégou, Cyril Terrioux, and Bruno Zanuttini. Some new tractable classes of cps and their relations with backtracking algorithms. In *CPAIOR*, pages 61–76, 2013.
- [16] E. Freuder. A Sufficient Condition for Backtrack-Free Search. *JACM*, 29 (1) :24–32, 1982.
- [17] M.R. Garey and D.S. Johnson. *Computer and Intractability*. Freeman, 1979.
- [18] F. Gavril. Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph. *SIAM Journal on Computing*, 1 (2) :180–187, 1972.
- [19] M. Golumbic. *Algorithmic Graph Theory and Perfect Graphs*. Academic Press, New York, 1980.
- [20] G. Gottlob, N. Leone, and F. Scarcello. A Comparison of Structural CSP Decomposition Methods. *Artificial Intelligence*, 124 :343–282, 2000.
- [21] R. Haralick and G. Elliot. Increasing tree search efficiency for constraint satisfaction problems. *Artificial Intelligence*, 14 :263–313, 1980.
- [22] P. Janssen, P. Jégou, B. Nougouier, and M.C. Vilarem. A filtering process for general constraint satisfaction problems : achieving pairwise-consistency using an associated binary representation. In *Proceedings of IEEE Workshop on Tools for Artificial Intelligence*, pages 420–427, 1989.
- [23] P. Jégou. *Contribution à l'étude des problèmes de satisfaction de contraintes : Algorithmes de propagation et de résolution – Propagation de contraintes dans les réseau dynamiques*. PhD thesis, Université des Sciences et Techniques du Languedoc, January 1991.
- [24] P. Jégou. Decomposition of Domains Based on the Micro-Structure of Finite Constraint Satisfaction Problems. In *AAAI*, pages 731–736, 1993.
- [25] P. Jégou and C. Terrioux. Hybrid backtracking bounded by tree-decomposition of constraint networks. *Artificial Intelligence*, 146 :43–75, 2003.
- [26] Christopher Mears, Maria Garcia de la Banda, and Mark Wallace. On implementing symmetry detection. *Constraints*, 14(4) :443–477, 2009.
- [27] B. Nadel. *Tree Search and Arc Consistency in Constraint-Satisfaction Algorithms*, pages 287–342. In *Search in Artificial Intelligence*. Springer-Verlag, 1988.
- [28] C.S. Peirce, C. Hartshorne, and P. Weiss. *Collected Papers of Charles Sanders Peirce*, volume vol. 3. Harvard University Press, 1933.
- [29] F. Rossi, P. van Beek, and T. Walsh. *Handbook of Constraint Programming*. Elsevier, 2006.
- [30] Francesca Rossi, Charles J. Petrie, and Vasant Dhar. On the equivalence of constraint satisfaction problems. In *ECAI*, pages 550–556, 1990.
- [31] András Salamon and Peter Jeavons. Perfect Constraints Are Tractable. In *CP*, pages 524–528, 2008.
- [32] K. Stergiou and T. Walsh. Encodings of Non-Binary Constraint Satisfaction Problems. In *AAAI*, pages 163–168, 1999.