



HAL
open science

Caractérisation de la complexité des classes de CSP définies par des motifs interdits à deux contraintes.

Martin Cooper, Guillaume Escamocher

► **To cite this version:**

Martin Cooper, Guillaume Escamocher. Caractérisation de la complexité des classes de CSP définies par des motifs interdits à deux contraintes.. JFPC - Journées Francophones de Programmation par Contraintes - 2012, May 2012, Toulouse, France. hal-00811871

HAL Id: hal-00811871

<https://inria.hal.science/hal-00811871>

Submitted on 11 Apr 2013

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

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

Caractérisation de la complexité des classes de CSP définies par des motifs interdits à deux contraintes*

Martin C. Cooper Guillaume Escamocher

IRIT, Université de Toulouse, France

{cooper, Guillaume.Escamocher}@irit.fr

Résumé

De nouvelles classes traitables du CSP (Problème de Satisfaction de Contraintes) ont été récemment découvertes via l'étude de classes d'instances définies par l'exclusion de sous-problèmes décrits par des motifs. La caractérisation complète de toutes les classes traitables définies par des motifs interdits est un problème ambitieux. Nous démontrons une dichotomie dans le cas de motifs interdits constitués de deux contraintes. Ce travail nous a permis d'identifier de nouvelles classes traitables.

Abstract

Novel tractable classes of the binary CSP (constraint satisfaction problem) have recently been discovered by studying classes of instances defined by excluding subproblems described by patterns. The complete characterisation of all tractable classes defined by forbidden patterns is a challenging problem. We demonstrate a dichotomy in the case of forbidden patterns consisting of two constraints. This has allowed us to discover new tractable classes.

1 Introduction

Dans une instance CSP le but est de déterminer l'existence d'une affectation de valeurs à des variables telle qu'un ensemble de contraintes soient satisfaites simultanément. Une question fondamentale est l'identification de sous-problèmes traitables du CSP. Les approches classiques consistent à identifier des

restrictions sur soit les types de contraintes soit l'(hyper)graphe des portées des contraintes qui impliquent l'existence d'un algorithme de complexité polynomiale. Dans certains cas, des dichotomies ont même été démontrées [2, 3, 14, 15, 16].

Récemment, une nouvelle voie de recherche s'est ouverte : l'identification de classes traitables d'instances CSP définies par l'interdiction d'un (ensemble de) sous-problème(s) spécifique(s). De nouvelles classes traitables ont été découvertes en interdisant des sous-problèmes simples sur 3 variables [9, 10]. Nous présentons dans ce papier un premier pas vers l'identification de toutes ces classes traitables, une dichotomie pour le cas particulier de sous-problèmes interdits à deux contraintes. Nous avons omis les détails de certaines preuves très longues ; les preuves complètes se trouvent dans la version arXiv de ce papier [8].

Nous définissons tout d'abord la notion de motif CSP que nous utiliserons plus tard pour décrire des classes d'instances CSP dans lesquelles le motif n'apparaît pas. Un motif peut représenter un ensemble de sous-problèmes (interdits) en laissant la compatibilité de certaines affectations indéfinie. Nous utilisons le terme *point* pour dénoter l'affectation d'une valeur à une variable. Un motif est un graphe dans lequel les sommets correspondent aux points et à la fois les sommets et les arêtes sont étiquetés. L'étiquette d'un sommet correspondant à une affectation variable-valeur $\langle v, d \rangle$ est simplement la variable v et l'étiquette d'une

* financé par le projet ANR-10-BLAN-0210.

arête entre deux sommets décrit la compatibilité de la paire d'affectations correspondante.

Définition 1. Un motif est un quintuplet $\langle V, A, var, E, cpt \rangle$ comprenant : un ensemble V de variables, un ensemble A de points (affectations), une fonction de variable $var : A \rightarrow V$, un ensemble $E \subseteq \binom{A}{2}$ d'arêtes (paires non ordonnées d'éléments de A) tel que $\{a, b\} \in E \Rightarrow var(a) \neq var(b)$, et une fonction de compatibilité à valeurs booléennes $cpt : E \rightarrow \{F, T\}$. (Pour une notation plus simple, nous écrivons $cpt(a, b)$ au lieu de $cpt(\{a, b\})$).

Pour un motif $P = \langle V, A, var, E, cpt \rangle$ et une variable $v \in V$, nous utilisons A_v pour dénoter l'ensemble d'affectations $\{a \in A \mid var(a) = v\}$ à v . Si $cpt(a, b) = T$ alors les deux affectations (points) a, b sont compatibles et $\{a, b\}$ est une arête de compatibilité ; si $cpt(a, b) = F$ alors les deux affectations a, b sont incompatibles et $\{a, b\}$ est une arête d'incompatibilité. Dans un motif, la compatibilité d'une paire de points a, b telle que $var(a) \neq var(b)$ et $(a, b) \notin E$ est indéfinie.

Une instance CSP binaire est un motif $\langle V, A, var, E, cpt \rangle$ dans lequel la compatibilité de chaque paire d'affectations à des variables distinctes est spécifiée par la fonction de compatibilité. La question correspondant à l'instance est : existe-t-il une solution, c'est-à-dire un ensemble d'affectations à toutes les variables dans V entièrement composé de paires compatibles ? La contrainte sur les variables $v_1, v_2 \in V$ est la sous instance à 2 variables $\langle \{v_1, v_2\}, A_{12}, var|_{A_{12}}, E_{12}, cpt|_{E_{12}} \rangle$ dans laquelle $A_{12} = A_{v_1} \cup A_{v_2}$ et $E_{12} = \{\{a, b\} \mid a \in A_{v_1}, b \in A_{v_2}\}$. La contrainte entre les variables v_1 et v_2 dans une instance est non-triviale s'il y a au moins une paire d'affectations incompatibles, c'est-à-dire $a \in A_{v_1}$ et $b \in A_{v_2}$ tels que $cpt(a, b) = F$.

Un motif est un moyen compact de représenter l'ensemble de toutes les instances obtenues en spécifiant arbitrairement la compatibilité de ses paires indéfinies. Deux motifs P et Q sont isomorphes s'ils sont identiques à renommage près (de variables ou d'affectations).

Définition 2. Nous disons qu'un motif P apparaît dans un motif P' (et P' contient P) si P' est isomorphe à un motif Q dans la fermeture transitive des deux opérations suivantes (extension, collage) appliquées à P :

extension $P = \langle V_P, A_P, var_P, E_P, cpt_P \rangle$ est un sous-motif de $Q = \langle V_Q, A_Q, var_Q, E_Q, cpt_Q \rangle$ (et Q une extension de P) : $V_P \subseteq V_Q, A_P \subseteq$

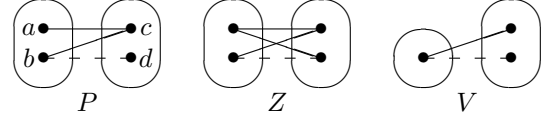


FIG. 1 – Trois motifs

$$A_Q, var_P = var_Q|_{A_P}, E_P \subseteq E_Q, cpt_P = cpt_Q|_{E_P}.$$

collage l'identification (ou collage) de deux points dans $P = \langle V_P, A_P, var_P, E_P, cpt_P \rangle$ transforme P en $Q = \langle V_Q, A_Q, var_Q, E_Q, cpt_Q \rangle$: $\exists a, b \in A_P$ tels que $var_P(a) = var_P(b)$ et $\forall c \in A_P$ tel que $\{a, c\}, \{b, c\} \in E_P$, $cpt_P(a, c) = cpt_P(b, c)$. De plus, $V_P = V_Q, A_Q = A_P \setminus \{b\}$, $var_Q = var_P|_{A_Q}$, $E_Q = E_P \cup \{\{a, x\} \mid \{b, x\} \in E_P\}$ et $cpt_Q(a, x) = cpt_Q(b, x)$ si $\{b, x\} \in E_P$, $cpt_Q(e) = cpt_P(e)$ sinon.

Considérons les trois motifs donnés en Figure 1. Les affectations (points) sont représentés par des points, et les affectations à la même variable v sont groupés ensemble à l'intérieur d'un ovale représentant A_v . Les lignes continues représentent les arêtes de compatibilités et les lignes en pointillés représentent les arêtes d'incompatibilités. Par exemple, P comporte de 4 points $a, b \in A_{v_0}, c, d \in A_{v_1}$ tels que $cpt(a, c) = cpt(b, c) = T$ et $cpt(b, d) = F$. P apparaît dans Z car Z est une extension de P . P apparaît également dans V car V peut être obtenu depuis P par le collage des points a, b .

Définition 3. Si P est un motif, $CSP(\overline{P})$ dénote l'ensemble d'instances CSP binaires Q dans lesquelles P n'apparaît pas. Le motif P est traitable s'il y a un algorithme de complexité polynomiale pour résoudre $CSP(\overline{P})$; P est intraitable si $CSP(\overline{P})$ est NP-Complexe.

2 Opérations de prétraitement

Cette section décrit certaines opérations de simplification de complexité polynomiale qui peuvent s'appliquer à une instance CSP $\langle V, A, var, E, cpt \rangle$. Si pour une variable v , A_v est un singleton $\{a\}$, alors l'élimination d'une variable à une seule valeur correspond à faire l'affectation a et consiste à éliminer v de V et éliminer a de A avec toutes les affectations b qui sont incompatibles avec a .

La cohérence d'arc [1] consiste à éliminer de A toutes les affectations a pour lesquels il y a une

variable $v \neq \text{var}(a)$ dans V telle que $\forall b \in A_v$, $\text{cpt}(a, b) = F$.

Si $\text{var}(a) = \text{var}(b)$ et pour toutes les variables $v \neq \text{var}(a)$, $\forall c \in A_v$, $\text{cpt}(a, c) = T \Rightarrow \text{cpt}(b, c) = T$, alors nous pouvons éliminer a de A par *substitution de voisinage*, car dans chaque solution dans laquelle a apparaît, nous pouvons remplacer a par b [12, 6]. Ni l'élimination d'une variable à une seule valeur, ni la cohérence d'arc, ni la substitution de voisinage ne peut introduire le motif interdit P quand l'opération est appliquée à une instance $\text{CSP}(\bar{P})$. Pour simplifier nos preuves, nous supposons à partir de maintenant que nous avons appliqué ces trois opérations jusqu'à convergence à toutes les instances CSP.

Nous considérons maintenant deux nouvelles opérations de simplification. Ce sont des opérations de simplification qui peuvent s'appliquer à certaines instances CSP. Nous pouvons toujours effectuer la fusion de deux variables v_1, v_2 dans une instance CSP en une seule variable v dont l'ensemble des affectations est le produit cartésien des ensembles d'affectations à v_1 et à v_2 . Sous certaines conditions, nous n'avons pas besoin de garder tous les éléments du produit cartésien et, en effet, le nombre total d'affectations en fait diminue.

Définition 4. *Considérons une instance CSP $\langle V, A, \text{var}, E, \text{cpt} \rangle$ avec $v_1, v_2 \in V$. Supposons qu'il y ait une fonction de fusion $f : A_{v_1} \rightarrow A_{v_2}$, telle que $\forall u \in A_{v_1}$, à chaque fois que u est dans une solution S , il y a une solution S' contenant à la fois u et $f(u)$. Alors nous pouvons effectuer la fusion simple de v_2 et v_1 pour créer une nouvelle variable fusionnée v dont le nombre d'affectations $|A_v| = |A_{v_1}|$. L'instance qui en résulte est $\langle V', A', \text{var}', E', \text{cpt}' \rangle$ définie par $V' = (V \setminus \{v_1, v_2\}) \cup \{v\}$, $A' = A \setminus A_{v_2}$, $\text{var}'(u) = \text{var}(u)$ pour tout $u \in A' \setminus A_{v_1}$ et $\text{var}'(u) = v$ pour tout $u \in A_{v_1}$, $E' = \{(p, q) \in \binom{A'}{2} \mid \text{var}'(p) \neq \text{var}'(q)\}$, $\text{cpt}'(p, q) = \text{cpt}(p, q)$ si $p, q \in A' \setminus A_{v_1}$, $\text{cpt}'(u, q) = \text{cpt}(u, q) \wedge \text{cpt}(f(u), q)$ pour tout $u \in A_{v_1}$ et tout $q \in A' \setminus A_{v_1}$.*

Définition 5. *Considérons une instance CSP $\langle V, A, \text{var}, E, \text{cpt} \rangle$ avec $v_1, v_2 \in V$ et une valeur charnière $a \in A_{v_1}$. Supposons qu'il y ait une fonction de fusion $f : A_{v_1} \setminus \{a\} \rightarrow A_{v_2}$, telle que $\forall u \in A_{v_1} \setminus \{a\}$, à chaque fois que u est dans une solution S , il y a une solution S' contenant à la fois u et $f(u)$. Alors nous pouvons effectuer la fusion complexe de v_2 et v_1 pour créer une nouvelle variable fusionnée v dont le nombre*

d'affectations $|A_v| = |A_{v_1}| + |A_{v_2}| - 1$. L'instance qui en résulte est $\langle V', A', \text{var}', E', \text{cpt}' \rangle$ définie par $V' = (V \setminus \{v_1, v_2\}) \cup \{v\}$, $A' = A \setminus \{a\}$, $\text{var}'(u) = \text{var}(u)$ pour tout $u \in A' \setminus (A_{v_1} \cup A_{v_2})$ et $\text{var}'(u) = v$ pour tout $u \in (A_{v_1} \setminus \{a\}) \cup A_{v_2}$, $E' = \{(p, q) \in \binom{A'}{2} \mid \text{var}'(p) \neq \text{var}'(q)\}$, $\text{cpt}'(p, q) = \text{cpt}(p, q)$ si $p, q \in A' \setminus (A_{v_1} \cup A_{v_2})$, $\text{cpt}'(u, q) = \text{cpt}(u, q) \wedge \text{cpt}(f(u), q)$ pour tout $u \in A_{v_1} \setminus \{a\}$ et tout $q \in A' \setminus (A_{v_1} \cup A_{v_2})$, $\text{cpt}'(p, q) = \text{cpt}(a, q) \wedge \text{cpt}(p, q)$ pour tout $p \in A_{v_2}$ et tout $q \in A' \setminus (A_{v_1} \cup A_{v_2})$.

Lemme 1. *Si I est une instance CSP et I' le résultat d'une fusion (simple ou complexe) de deux variables de I , alors I' est soluble ssi I est soluble.*

Démonstration. Nous ne donnons la preuve que dans le cas d'une fusion complexe, car une fusion simple peut être vue comme un cas particulier. Parmi les affectations dans le produit cartésien de A_{v_1} et A_{v_2} , il est suffisant, afin de préserver la solubilité, de ne garder que ceux de la forme (a, q) avec $q \in A_{v_2}$ ou de la forme $(u, f(u))$ avec $u \in A_{v_1} \setminus \{a\}$. Pour compléter la preuve, il suffit de remarquer que dans A' nous utilisons $q \in A_{v_2}$ pour représenter la paire d'affectations (a, q) et $u \in A_{v_1} \setminus \{a\}$ pour représenter $(u, f(u))$. \square

La fusion préserve la solubilité et le nombre total d'affectations diminue d'au moins 1 (en fait, par $|A_{v_2}|$ dans le cas d'une fusion simple). Cependant, lors de la résolution d'instances $I \in \text{CSP}(\bar{P})$, une opération de fusion ne sera utile que si elle n'introduit pas le motif interdit P .

3 Réduction et motifs intraitables

Dans un motif P , un point a qui n'est lié que par une seule arête de compatibilité au reste de P est un *point pendant*. Si une instance archérente I ne contient pas le motif P alors elle ne contient pas le motif P' qui est équivalent à P dans lequel le point pendant a et l'arête de compatibilité correspondante ont été supprimés. Ainsi, pour identifier les motifs traitables nous n'avons besoin d'étudier que les motifs sans points pendants.

Définition 6. *Nous disons qu'un motif P peut être réduit à un motif P' , et que P' est une réduction de P , si P' est isomorphe à un motif Q dans*

la fermeture transitive des trois opérations extension, collage et élagage appliquées à P , où élagage est l'opération suivante :

élagage éliminer un point pendant et son arête de compatibilité correspondante de P transforme P en Q .

Le lemme suivant découle immédiatement des définitions.

Lemme 2. Soient P et Q deux motifs, tels que P peut être réduit à Q . Soit I une instance CSP satisfaisant la cohérence d'arc. Si Q apparaît dans I , alors P apparaît aussi dans I . Si Q est traitable, alors P est traitable. Si P est intraitable, alors Q est intraitable.

Il s'ensuit que nous n'avons besoin d'étudier que les motifs qui ne peuvent pas être réduits à un motif traitable connu et qui ne sont pas la réduction d'un motif intraitable connu. Dans le reste de cette section nous prouvons quelques résultats qui sont essentiels pour la preuve de la dichotomie pour motifs à 2 contraintes que nous donnons dans la section suivante.

Lemme 3. Soit P un motif tel qu'une contrainte dans P contient deux arêtes d'incompatibilité distinctes qui ne peuvent pas être collées (par identification de points). Alors P est intraitable.

Démonstration. Soit P un motif tel qu'une contrainte de P contient deux arêtes d'incompatibilité non collables. Soit SAT1 l'ensemble d'instances SAT avec au plus une apparition de chaque variable dans chaque clause. SAT1 est trivialement équivalente à SAT qui est NP-Complet [5]. Il suffit de donner une réduction polynomiale de SAT1 vers $\text{CSP}(\overline{P})$. Nous supposons que nous avons une instance SAT1 $I = \{V, S\}$ avec V un ensemble de variables $\{v_1, v_2, \dots, v_n\}$ et S un ensemble de clauses $\{C_1, C_2, \dots, C_k\}$ tel que chaque clause C_i est une disjonction de c_i littéraux $l_i^1 \vee \dots \vee l_i^{c_i}$. Nous créons l'instance CSP I' suivante :

- $n + k$ variables v'_1, \dots, v'_{n+k} .
- $\forall v'_i$ avec $1 \leq i \leq n$, deux points " v_i " et " $\overline{v_i}$ " dans $A_{v'_i}$.
- $\forall v'_i$ avec $n + 1 \leq i \leq n + k$, c_{i-n} points $l_{i-n}^1, \dots, l_{i-n}^{c_{i-n}}$ dans $A_{v'_i}$.
- $\forall 1 \leq i \leq k, \forall 1 \leq j \leq c_i$, une arête d'incompatibilité entre le point $l_i^j \in A_{v'_{n+i}}$ et l'apparition dans $A_{v'_1}, \dots, A_{v'_n}$ du littéral $\overline{l_i^j}$.

Par construction, I' a une solution si et seulement si I a une solution. De plus, chaque fois qu'une arête d'incompatibilité apparaît dans une contrainte C , cette contrainte C est entre une variable CSP v'_i représentant la variable SAT1 v_i et une variable CSP v'_{n+j} représentant la clause SAT1 C_j . Comme v_i apparaît au plus une fois dans C_j , alors il n'y a qu'une seule arête d'incompatibilité dans C . Donc I' ne contient pas le motif P . Donc nous avons réduit SAT1 à $\text{CSP}(\overline{P})$. \square

Définition 7. Etant donné un motif $P = \langle V, A, var, E, cpt \rangle$, une variable $v \in V$, et un point $a \in A_v$, nous disons que a est explicitement compatible (respectivement explicitement incompatible) s'il y a un point $b \in A$ tel que a est compatible avec b (respectivement tel que a est incompatible avec b).

Le lemme suivant s'ensuit de la définition de collage.

Lemme 4. Soit P un motif dans lequel aucune paire de points n'est collable. Alors pour chaque variable v dans P , il n'y a au plus qu'un point dans A_v qui n'est pas explicitement incompatible.

Lemme 5. Soit Z le motif sur deux variables v et v' (donné en Figure 1), avec des points $a, b \in A_v$ et des points $c, d \in A_{v'}$ tels que a est compatible avec c et d , b est compatible avec c et incompatible avec d . Z est intraitable.

Démonstration. Comme 3-COLOURING est NP-complet [13], il suffit de donner une réduction polynomiale depuis 3-COLOURING vers $\text{CSP}(\overline{Z})$, l'ensemble d'instances CSP dans lesquelles le motif Z n'apparaît pas.

Définissons la contrainte $R_{s,t} \subseteq \{1, 2, 3\}^2$ par

$$R_{s,t} = \{(u, v) \mid (u = s \wedge v = t) \vee (u \neq s \wedge v \neq t)\}$$

Il est facile de vérifier que $R_{s,t}$ ne contient pas le motif Z . Considérons le gadget à 5 variables avec les variables v_i, v_j, u_1, u_2, u_3 , chacune avec pour domaine $\{1, 2, 3\}$, et avec les contraintes $R_{k,k}$ sur les variables (v_i, u_k) ($k = 1, 2, 3$) et les contraintes $R_{1+(k \bmod 3), k}$ sur les variables (u_k, v_j) ($k = 1, 2, 3$). L'effet combiné de ces six contraintes est simplement d'imposer la contrainte $v_i \neq v_j$. Toute instance $\langle V, E \rangle$ de 3-COLOURING, vers $V = \{1, \dots, n\}$, peut être réduite à une instance de $\text{CSP}(\overline{Z})$ avec pour variables v_1, \dots, v_n en plaçant une copie

de ce gadget entre chaque paire de variables (v_i, v_j) telles que $\{i, j\} \in E$. Cette réduction est clairement polynomiale. \square

Lemme 6. *Tout motif P avec 2 contraintes sur 3 variables dans lequel les deux contraintes contiennent une arête d'incompatibilité et deux arêtes partageant un point mais non collables est intraitable.*

Démonstration. Nous allons réduire CSP à $\text{CSP}(\overline{P})$. Soit I une instance CSP. Pour chaque (v, w) dans I tel qu'il y a une contrainte non triviale entre v et w , nous introduisons deux nouvelles variables v' et w' telles que le domaine de v' est le même que le domaine de v , le domaine de w' est le même que le domaine de w . Nous ajoutons des contraintes d'égalité entre v et v' , et entre w et w' , et nous ajoutons entre v' et w' la même contrainte qu'il y avait entre v et w . Toutes les autres contraintes impliquant v' ou w' sont triviales. Nous remplaçons aussi la contrainte entre v et w par une contrainte triviale. Soit I' l'instance obtenue après que toutes ces opérations ont été effectuées sur I . Par construction, I' a une solution si et seulement si I a une solution.

Nous supposons maintenant que nous avons trois variables v_0, v_1 et v_2 dans I' telles qu'il y a des contraintes non triviales entre v_0 et v_1 et entre v_0 et v_2 . Par construction, au moins une de ces contraintes est une contrainte d'égalité. Par définition, un point dans une contrainte d'égalité est compatible avec un et un seul point. Ainsi, P ne peut pas apparaître sur v_0, v_1 et v_2 . Cette réduction polynomiale de CSP vers $\text{CSP}(\overline{P})$ montre que P est intraitable. \square

4 Dichotomie pour les motifs à 2 contraintes

Soit $\mathcal{T} = \{T_1, T_2, T_3, T_4, T_5\}$ l'ensemble de motifs donnés en Figure 2.

Aucun motif de \mathcal{T} ne peut être réduit à un autre motif de \mathcal{T} . Comme nous allons le montrer, chaque T_i définit une classe traitable d'instances CSP binaires. Par exemple, T_4 définit une classe d'instances qui inclut en tant que sous-ensemble propre toutes les instances avec contraintes zéro-une-toutes [7], une généralisation des clauses 2SAT aux logiques multi valuées. Comme les motifs traitables à

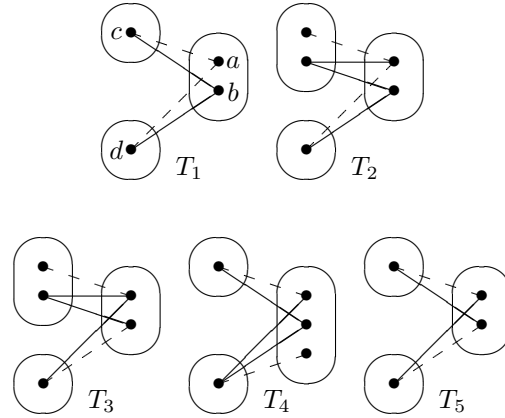


FIG. 2 – L'ensemble \mathcal{T} des motifs traitables

2 contraintes sur 4 variables sont nécessairement composés de deux motifs triviaux à 1 contrainte, nous restreignons notre attention aux motifs à 2 contraintes sur 3 variables.

Théorème 1. *Soit P un motif à deux contraintes sur trois variables. Alors P est traitable si et seulement si P est réductible à un des motifs de \mathcal{T} .*

Démonstration. \Rightarrow : D'après le Lemme 3, nous savons que nous n'avons besoin d'étudier que les motifs avec au plus une arête d'incompatibilité dans chaque contrainte. Si l'une des contraintes ne contient aucune arête d'incompatibilité du tout, alors le motif est réductible par collage et/ou élagage à un motif avec une seule contrainte. Il n'est pas difficile de montrer que tous les motifs traitables à 1 contrainte sont réductibles à un des motifs de \mathcal{T} . Donc nous pouvons supposer à partir de maintenant qu'il y a exactement une arête d'incompatibilité ($a \in A_{v_0}, b \in A_{v_1}$) entre v_0 et v_1 , et aussi exactement une arête d'incompatibilité ($c \in A_{v_0}, d \in A_{v_2}$) entre v_0 et v_2 . Le "squelette" d'arêtes d'incompatibilité d'un motif traitable irréductible peut ainsi prendre deux formes selon si $a = c$ (squelette de type 1) ou $a \neq c$ (squelette de type 2).

D'après le Lemme 4 nous savons que $|A_v| \leq 2$ pour chaque variable v avec un seul point explicitement incompatible, et que $|A_v| \leq 3$ pour chaque variable v avec deux points explicitement incompatibles. Nous savons d'après les Lemmes 5 et 6 que tout motif à 2 contraintes sur 3 variables contenant Z ou dans lequel les deux contraintes contiennent chacune une arête d'incompatibilité et deux arêtes de compatibilité non collables est intraitable. Nous

savons que nous avons deux squelettes d'incompatibilité possibles à étudier, chacun impliquant un nombre maximum de points apparaissant dans le motif. Par une recherche exhaustive sur tous les motifs, nous pouvons déduire que tous les motifs traitables avec un squelette d'arêtes d'incompatibilité de type 1 sont réductibles par extension, collage et élagage à l'un de T_1 ou T_2 , et que tous les motifs traitables avec un squelette d'arêtes d'incompatibilité de type 2 sont réductibles à l'un de T_3 , T_4 ou T_5 . Donc les seuls motifs traitables irréductibles possibles sont T_1, \dots, T_5 .

⇐ : Nous donnons maintenant les preuves de polynomialité pour tous les motifs de \mathcal{T} .

Preuve de la polynomialité de T_1 : Considérons une instance de $\text{CSP}(\overline{T_1})$.

Lemme 7. *Considérons une fusion (simple ou complexe) de deux variables v, v' dans une instance de $\text{CSP}(\overline{T_1})$. Supposons qu'à chaque fois que (a, a') et (b, b') sont des paires de points fusionnés durant cette fusion, telles que $a \neq b \in A_v$ et $a' \neq b' \in A_{v'}$, alors soit a et b' sont incompatibles ou b et a' sont incompatibles. Alors le motif T_1 ne peut pas être introduit par cette fusion.*

Démonstration. Par définition de la fusion (simple ou complexe), la seule façon d'introduire T_1 est quand les deux points dans la variable de droite de T_1 sont créés par la fusion de paires de points (a, a') et (b, b') telles que les compatibilités des points $a, b \in A_v$ et $a', b' \in A_{v'}$ avec les deux autres points c, d de T_1 sont données par : $\text{cpt}(c, a) = \text{cpt}(d, a') = F$, $\text{cpt}(c, b) = \text{cpt}(c, b') = \text{cpt}(a, a') = \text{cpt}(b, b') = \text{cpt}(d, b) = \text{cpt}(d, b') = T$. Maintenant, si a et b' étaient incompatibles, alors T_1 était déjà présent sur les points c, a, b, b' dans l'instance d'origine, et ne peuvent donc pas être introduits par la fusion. De même, si b et a' étaient incompatibles, alors T_1 était déjà présent sur les points b, a', b', d dans l'instance d'origine. □

Il est possible de montrer qu'après un prétraitement comprenant l'application d'opérations de fusion comme décrites dans le Lemme 7 jusqu'à convergence, nous avons une instance avec deux ensembles de variables E et $F = V \setminus E$ telle que :

- il n'y a pas de contrainte non triviale entre v et v' si $v, v' \in E$ ou $v, v' \in F$.

- $\forall v \in F, \forall f \in A_{v'}, f$ est incompatible avec des points dans A_v pour une et une seule variable $v' \in E$. De plus, f est incompatible avec tous les points de $A_{v'}$ sauf un.
- La seule contrainte non triviale possible entre une variable $v \in F$ et une variable $v' \in E$ est de la forme suivante : il y a un point $b \in A_v$ incompatible avec tous les points de $A_{v'}$ sauf un, et $\forall c \in A_v$ avec $c \neq b, c$ est compatible avec tous les points de $A_{v'}$.

Le nombre total d'affectations diminue quand nous fusionnons des variables, donc le nombre total de fusions qui peuvent être effectuées est linéaire par rapport à la taille de l'instance d'origine.

Nous appelons SATNUF (pour SAT Non binaire Une seule Foix) le problème suivant : un ensemble de variables $V = \{v_1, v_2, \dots, v_n\}$, un ensemble de valeurs $A = \{a_1, a_2, \dots, a_n\}$, et un ensemble de clauses $C = \{C_1, C_2, \dots, C_f\}$ tels que : chaque clause est une disjonction de littéraux, un littéral étant dans ce cas de la forme $v_i = a_j$, et $\forall i, j, p, q ((v_i = a_j) \in C_p) \wedge ((v_i = a_j) \in C_q) \Rightarrow p = q$.

Lemme 8. *$\text{CSP}(\overline{T_1})$ peut être réduit à SATNUF en temps polynomial.*

Démonstration. Nous supposons que nous avons une instance CSP binaire de $\text{CSP}(\overline{T_1})$ et prétraitée comme décrit ci-dessus. Nous savons que les contraintes non triviales entre les variables $v \in F$ et $v' \in E$ sont toutes de la forme $v = b \Rightarrow v' = a$. De plus, chaque affectation $v = b$ d'une valeur à une variable apparaît dans exactement une de ces contraintes. Pour tout $v \in F$, nous pouvons remplacer l'ensemble de telles contraintes $v = b_i \Rightarrow v_i = a_i$, pour toutes les valeurs b_i dans le domaine de v , par la clause $(v_1 = a_1) \vee \dots \vee (v_d = a_d)$. Il ne reste plus qu'à prouver qu'aucun littéral n'apparaît dans deux clauses distinctes. Supposons que nous avons un littéral $v_1 = a$ qui apparaît dans deux clauses distinctes. Alors il doit y avoir deux contraintes $v_2 = b \Rightarrow v_1 = a$ et $v_3 = c \Rightarrow v_1 = a$ et avec $v_1 \in E, v_2 \neq v_3 \in F$. Soit $a' \neq a$ un point de A_{v_1} . Alors b et c sont tous les deux incompatibles avec a' mais compatible avec a . Or c'est précisément le motif interdit. Cette contradiction montre que $\text{CSP}(\overline{T_1})$ peut être réduit à SATNUF. □

Une instance CSP peut être vue comme une instance WCSP avec des fonctions de

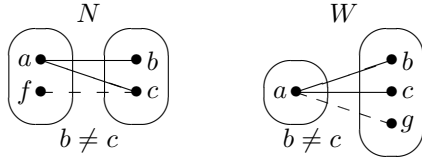


FIG. 3 – Deux gadgets

coût à valeurs dans $\{0, \infty\}$. En tant que fonctions de coût, les contraintes de SATNUF sont convexes, et les clauses sont non intersectantes. Donc, d'après [11], SATNUF est résoluble en temps polynomial, et par conséquent T_1 est traitable.

Preuve de la polynomialité de T_2 : Il est possible de montrer qu'après prétraitement, toute instance de $\text{CSP}(\overline{T_2})$ se décompose en sous-instances indépendantes qui soit sont fonctionnelles [4] soit satisfont la propriété de gagnant ex-aequo ("Joint-winner property") [10], et par conséquent sont résolubles en temps polynomial.

Preuve de la polynomialité de T_3 : Considérons une instance de $\text{CSP}(\overline{T_3})$.

Soit N le gadget donné en Figure 3 : deux variables v_0 et v_1 avec des points $a, f \in A_{v_0}$, $b, c \in A_{v_1}$, avec $b \neq c$, tels que a est compatible à la fois avec b et c , et f est incompatible avec c . Supposons que N apparaît dans l'instance et soit d un point dans A_{v_2} , avec $v_2 \neq v_0, v_1$. Si d est compatible avec c mais pas avec b , alors nous avons le motif interdit T_3 . Donc si c est compatible avec un point hors de A_{v_0} , alors b est aussi compatible avec le même point.

Soit S une solution contenant c . Soit e le point de S dans A_{v_0} . Si e est compatible avec b , alors nous pouvons remplacer c par b dans S tout en maintenant la correction de la solution, car tous les points de l'instance hors de A_{v_0} qui sont compatibles avec c sont aussi compatibles avec b .

Si e n'est pas compatible avec b , alors les arêtes $\{b, e\}$, $\{e, c\}$ et $\{c, a\}$ forment le gadget N . Donc, par notre argument précédent, si e est compatible avec un point hors de A_{v_1} , alors a est aussi compatible avec le même point. Nous pouvons donc remplacer c par b et e par a dans S tout en maintenant la correction de la solution, car tous les points de l'instance hors de A_{v_0} qui sont compatibles avec c sont aussi compatibles avec b et tous les points de l'ins-

tance hors de A_{v_1} qui sont compatibles avec e sont aussi compatibles avec a . Donc si une solution contient c , alors il y a une autre solution contenant b . Ainsi nous pouvons supprimer c tout en préservant la solubilité.

Donc à chaque fois que le gadget N est présent, nous pouvons supprimer l'un de ses points et ainsi éliminer N . Le gadget N est un motif traitable connu car interdire N revient à dire que toutes les contraintes sont soit triviales soit des bijections. Donc s'il n'est pas présent, alors l'instance est traitable. Il s'ensuit que le motif T_3 est traitable.

Preuve de la polynomialité de T_4 : Considérons une instance de $\text{CSP}(\overline{T_4})$.

Soit W le gadget donné en Figure 3 : deux variables v_0 et v_1 telles que $a \in A_{v_0}$, $b, c, g \in A_{v_1}$, avec $b \neq c$, a compatible avec b et c , et a incompatible avec g . Supposons que nous avons W dans l'instance.

Soit f un point dans A_{v_2} , avec $v_2 \neq v_0, v_1$. Si f est compatible avec b et incompatible avec c (ou compatible avec c et incompatible avec b), alors nous avons le motif interdit T_4 . Donc, tous les points de l'instance hors de $A_{v_0} \cup A_{v_1}$ ont la même compatibilité avec b et c .

Si tous les points de A_{v_0} qui sont compatibles avec b sont aussi compatibles avec c , alors tous les points de l'instance compatibles avec b sont aussi compatibles avec c et on peut supprimer b par la substitution de voisinage. Donc, nous pouvons supposer qu'il existe $d \in A_{v_0}$ tel que d est compatible avec b et incompatible avec c .

Soit S une solution qui contient c . Soit e le point de S dans v_0 . Si e est compatible avec b , alors on peut remplacer c par b dans S tout en conservant la correction de la solution, car b et c ont la même compatibilité avec tous les points de l'instance hors de $A_{v_0} \cup A_{v_1}$. Si e n'est pas compatible avec b , alors les arêtes $\{b, e\}$, $\{b, a\}$ et $\{b, d\}$ constituent le gadget W . Donc, par l'argument ci-dessus, a et d ont la même compatibilité avec tous les points de l'instance hors de $A_{v_0} \cup A_{v_1}$. De même, les arêtes $\{c, d\}$, $\{c, a\}$ et $\{c, e\}$ constituent le gadget W . Donc a et e ont la même compatibilité avec tous les points de l'instance hors de $A_{v_0} \cup A_{v_1}$. Ainsi d et e ont la même compatibilité avec tous les points de l'instance hors de $A_{v_0} \cup A_{v_1}$. Donc on peut remplacer c par b et e par d dans S tout en conservant la correction de la solution, car b, c ont la même compatibilité avec tous les

points de l'instance hors de $A_{v_0} \cup A_{v_1}$ et e, d ont la même compatibilité avec tous les points de l'instance hors de $A_{v_0} \cup A_{v_1}$.

Comme dans la preuve de polynomialité de T_3 , nous avons montré que si une solution contient c , alors il y a une autre solution contenant b . Ainsi nous pouvons supprimer c . Par conséquent, à chaque fois que le gadget W est présent, nous pouvons supprimer l'un de ses points. Le gadget W est un motif traitable connu car interdire W revient à dire que toutes les contraintes sont zéro-une-toutes [7]. Donc s'il n'est pas présent, l'instance est traitable. Par conséquent le motif T_4 est traitable.

Preuve de la polynomialité de T_5 : Le motif T_5 est un sous-motif du motif du triangle-cassé BTP , un motif traitable connu [9] sur trois contraintes. Donc le motif T_5 est traitable. \square

5 Conclusion

Nous avons démontré une dichotomie pour des classes d'instance CSP binaires définies par l'interdiction de motifs à 2 contraintes. Cela nous a permis d'identifier de nouvelles classes traitables, comprenant, par exemple, une nouvelle généralisation des contraintes zéro-une-toutes. Une voie de recherche future est d'examiner les généralisations possibles des cinq classes traitables définies par les motifs interdits T_1, \dots, T_5 , en remplaçant les contraintes binaires par des contraintes k -aires ($k > 2$) ou en ajoutant d'autres contraintes aux motifs.

Nous avons aussi identifié des opérations de prétraitement, basées sur la fusion de deux variables, qui pourraient s'appliquer à des instances CSP quelconques.

Références

- [1] Bessière, C., Régin, J.-C., Yap, R. H. C. and Zhang, Y., An optimal coarse-grained arc consistency algorithm, *Artificial Intelligence*, 165 (2) (2005) 165–185.
- [2] Bulatov, A., Jeavons, P. and Krokhin, A., Classifying the complexity of constraints using finite algebras, *SIAM Journal on Computing*, 34 (3) (2005) 720–742.
- [3] Bulatov, A. A., A dichotomy theorem for constraint satisfaction problems on a 3-element set, *J. of the ACM*, 53(1) (2006) 66–120.
- [4] Cohen, D. A., Cooper, M. C., Green, M. J. and Marx, D., On guaranteeing polynomially bounded search tree size, *CP'11*, LNCS 6876 (2011) 160–171.
- [5] Cook, S. A., The Complexity of Theorem-Proving Procedures, *STOC : Proceedings of the ACM symposium on Theory of computing* (1971) 151–158.
- [6] Cooper, M. C. Fundamental properties of neighbourhood substitution in constraint satisfaction problems, *Artificial Intelligence* 90 (1997) 1–24.
- [7] Cooper, M. C., Cohen, D. A. and Jeavons, P. G. Characterising tractable constraints, *Artificial Intelligence* 65 (2) (1994) 347–361.
- [8] Cooper, M. C. and Escamocher, G., A dichotomy for 2-constraint forbidden CSP patterns, *CoRR* abs/1201.3868 (2012).
- [9] Cooper, M. C., Jeavons, P. G., and Salamon, A. Z., Generalizing constraint satisfaction on trees : Hybrid tractability and variable elimination, *Artificial Intelligence*, 174 (9–10) (2010) 570–584.
- [10] Cooper, M. C. and Živný, S., Hybrid tractability of valued constraint problems, *Artificial Intelligence*, 175 (9-10) (2011) 1555–1569.
- [11] Cooper, M. C. and Živný, S., Hierarchically nested convex VCSP, *CP 2011*, LNCS 6876 (2011) 187–194.
- [12] Freuder, E. C., Eliminating interchangeable values in constraint satisfaction problems, *AAAI-91*, Anaheim, CA (1991) 227–233.
- [13] Garey, M. R. and Johnson, D. S., *Computers and Intractability : A Guide to the Theory of NP-Completeness*, W.H. Freeman (1979).
- [14] Grohe, M., The complexity of homomorphism and constraint satisfaction problems seen from the other side, *J. of the ACM*, 54 (1) (2007) 1–24.
- [15] Marx, D., Can You Beat Treewidth?, *Theory of Computing*, 6 (1) (2010) 85–112.
- [16] Marx, D., Tractable hypergraph properties for constraint satisfaction and conjunctive queries, *STOC '10* (2010) 735–744.