Extending Broken Triangles and Enhanced Value-Merging

Martin Cooper, Achref El Mouelhi, Cyril Terrioux

Extending Broken Triangles and Enhanced Value-Merging

Martin C. Cooper[1], Achref El Mouelhi[2(✉)], and Cyril Terrioux[2]

[1] IRIT, University of Toulouse III, 31062 Toulouse, France
cooper@irit.fr

[2] Aix-Marseille Université, CNRS, ENSAM, Université de Toulon,
LSIS UMR 7296, 13397 Marseille, France
{achref.elmouelhi,cyril.terrioux}@lsis.org

Abstract. Broken triangles constitute an important concept not only for solving constraint satisfaction problems in polynomial time, but also for variable elimination or domain reduction by merging domain values. Specifically, for a given variable in a binary arc-consistent CSP, if no broken triangle occurs on any pair of values, then this variable can be eliminated while preserving satisfiability. More recently, it has been shown that even when this rule cannot be applied, it could be possible that for a given pair of values no broken triangle occurs. In this case, we can apply a domain-reduction operation which consists in merging these values while preserving satisfiability.

In this paper we show that under certain conditions, and even if there are some broken triangles on a pair of values, these values can be merged without changing the satisfiability of the instance. This allows us to define a stronger merging operation and a new tractable class of binary CSP instances. We report experimental trials on benchmark instances.

1 Introduction

Identifying tractable classes constitutes an important research goal in constraint programming. The *broken-triangle property* (BTP) defines a hybrid tractable class [6,7]. This class has some interesting characteristics, both from a theoretical and practical viewpoint: it generalises existing language-based and structural classes and is solved in polynomial time by the algorithm MAC which is omnipresent in constraint solvers [20]. Besides, many extensions of the broken-triangle property have led to the definition of new tractable classes [8,10,11,14,18,19]. Local versions of the BTP have also given rise to novel reduction operations for CSP instances. In particular, in arc-consistent binary CSP instance, if no broken triangle occurs on any pair of values in the domain of a variable, then this variable can be eliminated without changing the satisfiability of the instance [2]. Even when this variable-elimination rule cannot be applied, it can nevertheless happen that no broken triangle occurs on a particular pair of values. In this case, these two values can be merged into a single value without changing the satisfiability of the instance [5]. This domain-reduction operation, known as BT-merging, was found to be applicable in diverse benchmark domains, although extensive experimental trials would seem to indicate that it is not useful, in terms of total solving time, as a preprocessing operation in a general-purpose solver [4].

In the light of these results, in this paper we study a lighter version of BTP-merging which allows the presence of some broken triangles on the pair of values to be merged, thus giving rise to a stronger domain-reduction operation.

In the following section we recall basic definitions and notations used in the rest of the paper. In Sect. 3 we introduce a new generic rule, called m-wBTP, which allows us to merge two values even in the presence of some broken triangles. We then show in Sect. 4 that, for sufficiently large m, this rule is maximal. We go on to show, in Sect. 5, that this merging rule does not allow the elimination of variables. Nevertheless, in Sect. 6 we show that it does allow us to define a tractable class. We also compare m-wBTP with certain other generalisations of BTP, such as k-BTP [8] and WBTP [19]. In Sect. 7 we report experimental trials to evaluate the practical interest of 1-wBTP-merging.

2 Preliminaries

Constraint satisfaction problems (CSPs [17]) are at the heart of numerous applications in Artificial Intelligence and Operations Research. In this paper, we study only binary CSP instances, defined formally as follows:

Definition 1. *A* ***binary CSP instance*** *is a triple $I = (X, D, C)$, where $X = \{x_1, \ldots, x_n\}$ is a finite set of n* ***variables****, $D = \{D(x_1), \ldots, D(x_n)\}$ is a set of* ***domains*** *containing at most d* ***values****, a domain for each variable, and C is a set of* ***binary constraints****. Each constraint $C_{ij} \in C$ is a pair $(S(C_{ij}), R(C_{ij}))$ with:*

- *$S(C_{ij}) = \{x_i, x_j\} \subseteq X$, the* ***scope*** *of the constraint,*
- *$R(C_{ij}) \subseteq D(x_i) \times D(x_j)$, the* ***relation*** *specifying the compatibility of values.*

If the constraint C_{ij} is not defined in C, then we consider C_{ij} to be a universal constraint (i.e. such that $R(C_{ij}) = D(x_i) \times D(x_j)$).

The interaction between the values of each variable through the relations associated to constraints can be represented graphically by a *microstructure* graph [13]. The vertices of this graph are thus the variable-value pairs (x_i, v_i) $(v_i \in D(x_i))$ and the edges are the tuples authorized by the constraints (that is, there is an edge between the vertices (x_i, v_i) and (x_j, v_j) iff $(v_i, v_j) \in R(C_{ij})$). Given a binary instance I, deciding whether I has a *solution* (an assignment $(v_1, \ldots, v_n)$ such that $\forall i$, $v_i \in D(x_i)$ and $\forall i \neq j$, $(v_i, v_j) \in R(C_{ij})$), is well known to be NP-complete. However, by imposing some restrictions on the constraint scopes and/or relations, we can define *tractable* classes of instances which can be solved in polynomial time. The BTP (*Broken Triangle Property*) tractable class, is an important tractable class since it generalises certain previously known classes based exclusively on properties of the constraint scopes or the constraint relations and has been the inspiration for a new branch of research on tractable classes of CSPs based on forbidden patterns [1,4,9,11,18]. The Broken Triangle Property imposes the *absence* of so-called broken triangles. Formally, BTP is defined as follows:

Definition 2 (Broken-Triangle Property [6,7]**).** *Let I be a binary CSP instance with a variable order $<$. A pair of values $v'_k, v''_k \in D(x_k)$ satisfies BTP if, for each pair of variables (x_i, x_j) such that $x_i < x_j < x_k$, $\forall v_i \in D(x_i)$, $\forall v_j \in D(x_j)$, if $(v_i, v_j) \in R(C_{ij})$, $(v_i, v'_k) \in R(C_{ik})$ and $(v_j, v''_k) \in R(C_{jk})$, then either $(v_i, v''_k) \in R(C_{ik})$, or $(v_j, v'_k) \in R(C_{jk})$.*

A variable x_k satisfies BTP if each pair of values in $D(x_k)$ satisfies BTP. An instance satisfies BTP if all its variables satisfy BTP.

This definition can be represented graphically in the microstructure of I as shown in Fig. 1. Throughout this paper, we represent an unauthorized assignment (a tuple which violates the constraint) either by a dashed line or by the absence of a line.

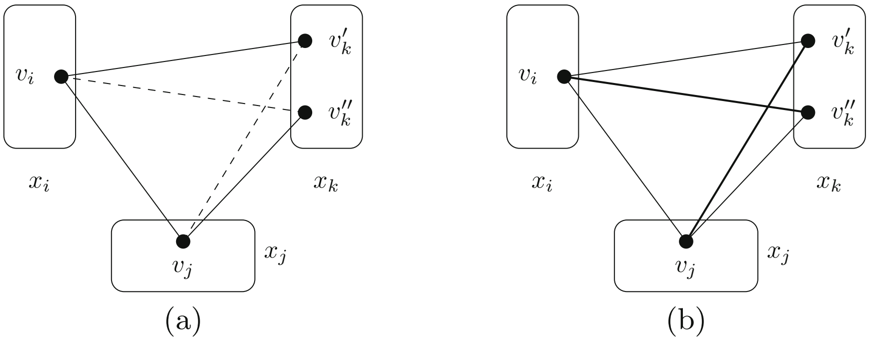

Fig. 1. (a) A broken triangle (v_i, v_j, v'_k, v''_k). (b) The assignments (v_i, v_j, v'_k, v''_k) do not form a broken triangle.

In Fig. 1(a), the CSP instance is not BTP relative to the order $x_i < x_j < x_k$ because the tuples (v_j, v'_k) and (v_i, v''_k) are not authorized. In this example, (v_i, v_j, v'_k, v''_k) constitute a *broken triangle* on the values v'_k and v''_k. Because of this broken triangle, we say that there is a broken triangle on x_k relative to x_i and x_j. On the other hand, if $(v_i, v''_k) \in R(C_{ik})$ or $(v_j, v'_k) \in R(C_{jk})$, as illustrated in Fig. 1(b), then the broken-triangle property is satisfied.

We now define the merging of domain values before recalling the merging operation based on BTP.

Definition 3 [4]**.** ***Merging*** *the values $v'_k, v''_k \in D(x_k)$ in a binary CSP instance I consists of replacing v'_k, v''_k in $D(x_k)$ by a new value v_k which is compatible with all values which are compatible with at least one of the values v'_k or v''_k. A* ***value-merging condition*** *is a polytime-verifiable property such that when it holds on a pair of values $v'_k, v''_k \in D(x_k)$, the CSP instance obtained after merging the values v'_k and v''_k is satisfiable if and only if I was satisfiable.*

In binary CSP instances, the absence of broken triangles on a pair of values is a valid value-merging condition [4]. For example, in Fig. 1(b), the values v'_k and v''_k are mergeable.

3 Weakly Broken Triangles

The absence of broken triangles on a pair of values allows them to be merged while preserving satisfiability. In this section, we show that it is possible to merge certain pairs of values even in the presence of some broken triangles. This idea was inspired by recent work by Naanaa [19] on a new extension of BTP. We call our new property m-wBTP: the parameter m defines the number of variables supporting the *weakly broken triangles*.

3.1 1-wBTP-Merging

We start with the simplest case ($m = 1$) based on a new concept called *weakly broken triangles* supported by one other variable.

Definition 4. *A pair of values $v'_k, v''_k \in D(x_k)$ satisfies* **1-*wBTP*** *if for each broken triangle (v_i, v_j, v'_k, v''_k) with $v_i \in D(x_i)$ and $v_j \in D(x_j)$, there is at least one variable $x_\ell \in X \setminus \{x_i, x_j, x_k\}$ such that: $\forall\ v_\ell \in D(x_\ell)$ if $(v_i, v_\ell) \in R(C_{i\ell})$ and $(v_j, v_\ell) \in R(C_{j\ell})$ then*

- *$(v'_k, v_\ell) \notin R(C_{k\ell})$ and*
- *$(v''_k, v_\ell) \notin R(C_{k\ell})$.*

If this is the case, (v_i, v_j, v'_k, v''_k) is known as a ***weakly broken triangle*** *supported by the variable x_ℓ.*

This definition can be represented by the microstructure graph, as shown in Fig. 2. There is a broken triangle (v_i, v_j, v'_k, v''_k). Since for each value v_ℓ of the variable x_ℓ, v_ℓ is compatible with v_i and v_j and we have $(v'_k, v_\ell) \notin R(C_{k\ell})$ and $(v''_k, v_\ell) \notin R(C_{k\ell})$, this triangle is a weakly broken triangle supported by x_ℓ.

The notion of weakly broken triangles allows us to generalise BTP-merging.

Proposition 1. *In a binary CSP, merging two values $v'_k, v''_k \in D(x_k)$ which satisfy 1-wBTP does not change the satisfiability of an instance.*

Proof. Let I be the original instance and I' the new instance in which v'_k, v''_k have been merged into a new value v_k (which replaces v'_k, v''_k in $D(x_k)$). Clearly, if I is satisfiable then so is I'. Hence, it suffices to show that if I' has a solution s which assigns v_k to x_k, then I also has a solution.

Let s', s'' be two assignments which are identical to s except that s' assigns v'_k to x_k and s'' assigns v''_k to x_k. Suppose, for a contradiction, that neither s' nor s'' is a solution to I. Then there are two variables $x_i, x_j \in X \setminus \{x_k\}$ such that $(s(x_i), v'_k) \notin R(C_{ik})$ and $(s(x_j), v''_k) \notin R(C_{jk})$. Since s is a solution to I' assigning v_k to x_k, we must have $(s(x_i), v''_k) \in R(C_{ik})$ and $(s(x_j), v'_k) \in R(C_{jk})$.

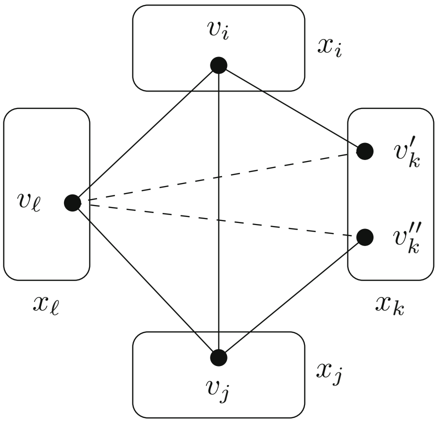

Fig. 2. A triangle which is weakly broken since $(v'_k, v_\ell) \notin R(C_{k\ell})$ and $(v''_k, v_\ell) \notin R(C_{k\ell})$.

Obviously, we also have $(s(x_i), s(x_j)) \in R(C_{ij})$ since s is a solution to I'. So $(s(x_i), s(x_j), v'_k, v''_k)$ is a broken triangle in I.

By the definition of 1-wBTP, there is a variable $x_\ell \in X \setminus \{x_i, x_j, x_k\}$ such that $\forall v_\ell \in D(x_\ell)$ if $(s(x_i), v_\ell) \in R(C_{i\ell})$ and $(s(x_j), v_\ell) \in R(C_{j\ell})$ then

- $(v'_k, v_\ell) \notin R(C_{k\ell})$ and
- $(v''_k, v_\ell) \notin R(C_{k\ell})$.

As $s(x_\ell)$ is compatible with $s(x_i)$ and $s(x_j)$, it cannot be compatible with either v'_k or v''_k. It follows that $s(x_\ell)$ is not compatible with v_k, which implies that s is not a solution to I'. But this contradicts our initial hypothesis. Thus, this merging rule preserves satisfiability. □

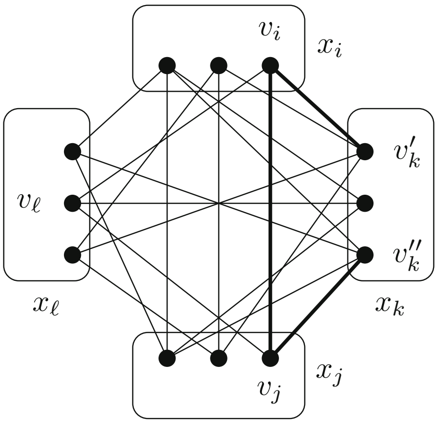

Fig. 3. A CSP instance in which all values are arc consistent (in bold, the weakly broken triangle).

At first sight, there appears to be an obvious link between this definition and arc consistency [16]. Indeed, imposing that the tuples (v'_k, v_ℓ) and (v''_k, v_ℓ) are unauthorized seems to imply that the goal is to render the values v'_k and v''_k arc-inconsistent. But the example in Fig. 3 shows that this is not always the case.

Indeed, although the two values $v'_k, v''_k \in D(x_k)$ in this figure satisfy 1-wBTP, establishing arc consistency deletes no values (and obviously no tuples). Thus, arc consistency does not delete the broken triangle (v_i, v_j, v'_k, v''_k).

3.2 m-wBTP-Merging

In Definition 4, thanks to the supporting variable(s) x_ℓ, merging values on which there are only weakly broken triangles leaves the satisfiability of the instance invariant. In terms of the microstructure, the variable x_ℓ prevents the creation of a new clique in the microstructure of size n (i.e. a new solution) which did not exist before merging. This principle can clearly be extended to m variables $(m \leq n-3)$.

An assignment $(v_{\ell_1}, \ldots, v_{\ell_m}) \in D(x_{\ell_1}) \times \ldots \times D(x_{\ell_m})$ is a **partial solution** if it satisfies all constraints C_{ij} such that $\{x_i, x_j\} \subseteq \{x_{\ell_1}, \ldots, x_{\ell_m}\}$.

Definition 5. *A pair of values $v'_k, v''_k \in D(x_k)$ satisfies m-**wBTP** where $m \leq n-3$ if for each broken triangle (v_i, v_j, v'_k, v''_k) with $v_i \in D(x_i)$ and $v_j \in D(x_j)$, there is a set of $r \leq m$ **support variables** $\{x_{\ell_1}, \ldots, x_{\ell_r}\} \subseteq X \setminus \{x_i, x_j, x_k\}$ such that for all $(v_{\ell_1}, \ldots, v_{\ell_r}) \in D(x_{\ell_1}) \times \ldots \times D(x_{\ell_r})$, if $(v_{\ell_1}, \ldots, v_{\ell_r}, v_i, v_j)$ is a partial solution, then there is $\alpha \in \{1, \ldots, r\}$ such that $(v_{\ell_\alpha}, v'_k), (v_{\ell_\alpha}, v''_k) \notin R(C_{\ell_\alpha k})$. We say that x_{ℓ_α} is the **shield variable** for this partial solution.*

Figure 4 shows two configurations illustrating Definition 5. In the first, the pair of values $v'_k, v''_k \in D(x_k)$ satisfies 2-wBTP because for the unique partial solution $(v_{\ell_\sigma}, v_{\ell_\gamma}, v_i, v_j)$ we have $(v_{\ell_\sigma}, v'_k), (v_{\ell_\sigma}, v''_k) \notin R(C_{\ell_\sigma k})$. In the second, there is no partial solution on the set of variables $\{x_{\ell_\sigma}, x_{\ell_\gamma}, x_i, x_j\}$; thus $v'_k, v''_k \in D(x_k)$ trivially satisfies 2-wBTP.

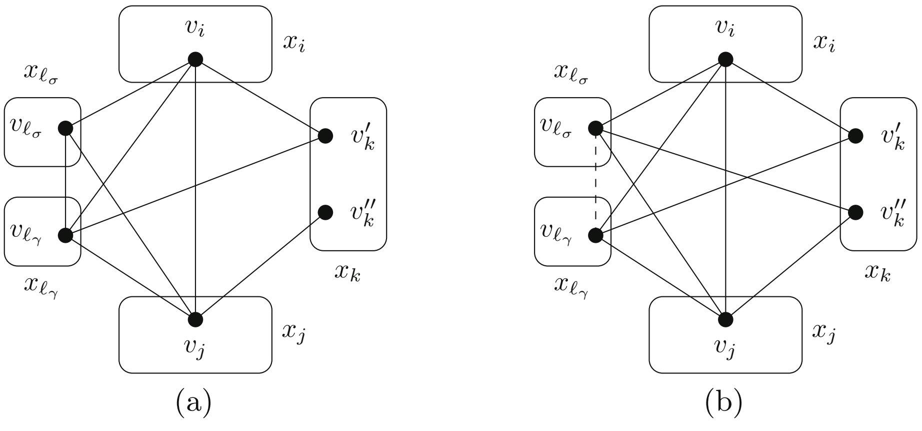

Fig. 4. Two different cases of two values v'_k and v''_k which satisfy 2-wBTP.

We now generalise Proposition 1 to the merging of values satisfying m-wBTP.

Proposition 2. *In a binary CSP, merging two values $v'_k, v''_k \in D(x_k)$ which satisfy m-wBTP does not change the satisfiability of an instance.*

Proof. Let I be the original instance and I' the new instance in which v'_k, v''_k have been merged into a new value v_k (which replaces v'_k, v''_k in $D(x_k)$). Clearly, if I is satisfiable then so is I'. Hence, it suffices to show that if I' has a solution s which assigns v_k to x_k, then I also has a solution.

Let s', s'' be two assignments which are identical to s except that s' assigns v'_k to x_k and s'' assigns v''_k to x_k. Suppose, for a contradiction, that neither s' nor s'' is a solution to I. Then there are two variables $x_i, x_j \in X \setminus \{x_k\}$ such that $(s(x_i), v'_k) \notin R(C_{ik})$ and $(s(x_j), v''_k) \notin R(C_{jk})$. Since s is a solution to I' assigning v_k to x_k, we must have $(s(x_i), v''_k) \in R(C_{ik})$ and $(s(x_j), v'_k) \in R(C_{jk})$. We also have $(s(x_i), s(x_j)) \in R(C_{ij})$ since s is a solution to I. So $(s(x_i), s(x_j), v'_k, v''_k)$ is a broken triangle in I.

The values v'_k and v''_k satisfy m-wBTP, so, by definition, there is a set of $r \leq m$ variables $\{x_{\ell_1}, \ldots, x_{\ell_r}\} \subseteq X \setminus \{x_i, x_j, x_k\}$ such that for all $(v_{\ell_1}, \ldots, v_{\ell_r}) \in D(x_{\ell_1}) \times \ldots \times D(x_{\ell_r})$, if $(v_{\ell_1}, \ldots, v_{\ell_r}, v_i, v_j)$ is a partial solution, then there is $\alpha \in \{1, \ldots, r\}$ such that $(v_{\ell_\alpha}, v'_k), (v_{\ell_\alpha}, v''_k) \notin R(C_{\ell_\alpha k})$.

Since s is a solution of the instance I', $(s(x_{\ell_1}), \ldots, s(x_{\ell_r}), s(x_i), s(x_j))$ is necessarily a partial solution, so there is $\alpha \in \{1, \ldots, r\}$ such that we have $(s(x_{\ell_\alpha}), v'_k)$, $(s(x_{\ell_\alpha}), v''_k) \notin R(C_{\ell_\alpha k})$, which implies $(s(x_{\ell_\alpha}), v_k) \notin R(C_{\ell_\alpha k})$. This is a contradiction since s is a solution of the instance I' with $s(x_k) = v_k$.

We can deduce that m-wBTP-merging preserves satisfiability. □

The BTP-merging rule [4] can be seen as 0-wBTP-merging since it is based on zero support variables. The following proposition establishes the link between the different versions of merging based on BTP.

Proposition 3. *In an n-variable binary CSP, if a pair of values $v'_k, v''_k \in D(x_k)$ satisfies m-wBTP then it satisfies $(m+1)$-wBTP (for $0 \leq m \leq n-4$).*

The BTP-merging rule generalises both neighbourhood substitution [12] and virtual interchangeability [15]. As m-wBTP-merging generalises BTP-merging for all $m \geq 0$, the following result follows immediately:

Corollary 1. *m-wBTP-merging generalises neighbourhood substitution and virtual interchangeability.*

Besides the fact that m-wBTP-merging preserves satisfiability, it is also possible to reconstruct in polynomial time all solutions to the original instance I from the solutions from an instance I' obtained by applying a sequence of m-wBTP-mergings. What is more, the reconstruction of a solution to I from a solution to I' can be achieved in time which is linear in the size of I. It suffices to apply the same algorithm as in the case of BTP-merging [4].

4 A Maximal Value-Merging Condition

It is well known that any pair of values which satisfies BTP can be merged while preserving satisfiability [4]. We have shown that a pair of values which does not satisfy BTP can nevertheless be merged while preserving satisfiability if this pair satisfies m-wBTP. Thus, in an obvious sense, BTP-merging is not a maximal value-merging condition. A value-merging condition is *maximal* if the merging of any other pair of values not respecting the condition necessarily leads to a modification of the satisfiability of some instance. In this section, we show that m-wBTP is a maximal value-merging condition when $m = n - 3$.

Theorem 1. *In an unsatisfiable n-variable binary CSP instance, there is no pair of values not satisfying m-wBTP for $m = n - 3$ and which can be merged while preserving satisfiability.*

Proof. Let I be an unsatisfiable n-variable CSP instance and let $v'_k, v''_k \in D(x_k)$ be a pair of values which does not satisfy m-wBTP for $m = n - 3$. By the definition of m-wBTP-merging, there is a broken triangle (v_i, v_j, v'_k, v''_k), with $v_i \in D(x_i)$ and $v_j \in D(x_j)$, and there is $(v_{\ell_1}, \ldots, v_{\ell_m}) \in D(x_{\ell_1}) \times \ldots \times D(x_{\ell_m})$, where $\{x_{\ell_1}, \ldots, x_{\ell_m}\} = X \setminus \{x_i, x_j, x_k\}$, such that $(v_{\ell_1}, \ldots, v_{\ell_m}, v_i, v_j)$ is a partial solution and for all $\alpha \in \{1, \ldots, m\}$ we have $(v_{\ell_\alpha}, v'_k) \in R(C_{\ell_\alpha k})$ or $(v_{\ell_\alpha}, v''_k) \in R(C_{\ell_\alpha k})$.

We have a broken triangle, and so: $(v_i, v''_k) \notin R(C_{ik})$, $(v_j, v'_k) \notin R(C_{jk})$, $(v_i, v'_k) \in R(C_{ik})$ and $(v_j, v''_k) \in R(C_{jk})$. We also have, for all $\ell \in \{\ell_1, \ldots, \ell_m\}$:

- $(v_\ell, v'_k) \in R(C_{\ell k})$ or
- $(v_\ell, v''_k) \in R(C_{\ell k})$.

After merging, and by definition of merging, the new merged value v_k satisfies $(v_\ell, v_k) \in R(C_{\ell k})$ for all $\ell \in \{\ell_1, \ldots, \ell_m\} \cup \{i, j\}$. We obtain a solution given by $v_{\ell_1}, \ldots, v_{\ell_m}, v_i, v_j$ and v_k. Thus, we have introduced a solution which did not exist in the original instance since $(v_i, v''_k) \notin R(C_{ik})$ and $(v_j, v'_k) \notin R(C_{jk})$. It follows that the merging of any pair of values which does not satisfy m-wBTP does not preserve satisfiability. □

A valid value-merging condition has to guarantee that an unsatisfiable instance does not become satisfiable after merging. We can therefore deduce the following corollary.

Corollary 2. $(n - 3)$*-wBTP is a maximal value-merging condition.*

5 wBTP and Variable Elimination

BTP allows value-merging [4], variable elimination [2,3] and the definition of a tractable class [7]. There are several distinct generalisations of BTP according to the desired property. m-wBTP is a generalisation of BTP which allows us to reduce the size of domains via value-merging. m-wBTP is a less restrictive condition than BTP and thus allows more mergings than BTP. On the other hand, this gain in the number of mergings is counterbalanced by the fact that m-wBTP does not allow variable elimination.

In [2], it was shown that, for a given variable x_k of an arc-consistent binary CSP instance I, if there is no broken triangle on any pair of values of $D(x_k)$, then eliminating the variable x_k from I preserves satisfiability. We now show that this is not the case for m-wBTP when $m > 0$.

Proposition 4. *Given a variable x_k of an arc-consistent binary CSP instance I, even if each pair of values in $D(x_k)$ satisfies m-wBTP, where $m \geq 1$, eliminating variable x_k can change the satisfiability of I.*

Proof. Let I be the binary CSP instance defined on four variables $x_1, \ldots, x_4$ with $D(x_i) = \{0, 1, 2\}$ $(i = 1, \ldots, 4)$ and the following constraints: $x_1 = x_2$, $x_2 = x_3$, $x_3 = x_1$, $x_1 = (x_4 + 1) \bmod 3$, $x_2 = (x_4 - 1) \bmod 3$, $x_3 = x_4$. This instance is arc-consistent. There are three partial solutions $(0, 0, 0)$, $(1, 1, 1)$ and $(2, 2, 2)$ on variables x_1, x_2, x_3, but I does not have a solution. Therefore, the elimination of variable x_4 does not preserve the satisfiability of the instance (see Fig. 5).

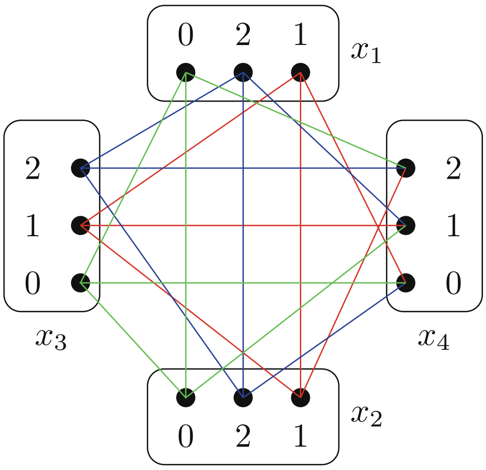

Fig. 5. An unsatisfiable CSP instance in which each pair of values in $D(x_4)$ satisfies 1-wBTP but the elimination of x_4 introduces three solutions.

Let x_i, x_j, x_ℓ be the variables x_1, x_2, x_3 (in any order). There are three broken triangles (v_i, v_j, v'_4, v''_4) on the variables x_i, x_j, x_4 (the weakly broken triangles are represented by three different colours in Fig. 5): in each of these broken triangles, we have $v_i = v_j$. For each of these broken triangles, there is exactly one partial solution of the form (v_ℓ, v_i, v_j) on the variables x_ℓ, x_i, x_j because we necessarily have $v_\ell = v_i = v_j$. By the choice of constraints, the values v_ℓ, v_i, v_j are compatible with three different values in $D(x_4)$. We can deduce that (v_ℓ, v'_4), $(v_\ell, v''_4) \notin R(C_{\ell 4})$ since, by the definition of a broken triangle, each of the values v'_4, v''_4 is compatible with one of the values v_i, v_j. Thus, each pair of values $v'_4, v''_4 \in D(x_4)$ satisfies 1-wBTP.

We have exhibited an instance I such that each pair of values in $D(x_4)$ satisfies 1-wBTP, but eliminating the variable x_4 changes the satisfiability of I. For values of $m > 1$, it suffices to add $m - 1$ other non-constrained variables to the instance I. □

In the instance I in the proof of Proposition 4, each pair of values in the domain $D(x_4)$ satisfies 1-wBTP. However, after having performed the merging of two values, the two remaining values no longer satisfy 1-wBTP and cannot be merged.

6 wBTP and Tractability

In order to compare m-wBTP and other generalisations of BTP defining tractable classes, we extend the definition of m-wBTP in a natural way to instances.

Definition 6. *Given a constant $m \leq n-3$, a binary CSP instance I with a variable-order $<$ satisfies m-wBTP relative to this order if for all variables x_k, each pair of values in $D(x_k)$ satisfies m-wBTP in the sub-instance of I on variables $x_i \leq x_k$.*

A lighter version of BTP, called k-BTP, which allows certain broken triangles, has recently been defined [8]. Binary CSP instances which satisfy both strong k-consistency and k-BTP constitute a tractable class.

Definition 7 (k-BTP [8]). *A binary CSP instance I satisfies the k-BTP property for a given k $(2 \leq k < n)$ relative to a variable order $<$ if, for all subsets of variables $x_{i_1}, x_{i_2}, \ldots, x_{i_{k+1}}$ such that $x_{i_1} < x_{i_2} < \cdots < x_{i_{k+1}}$, there is at least one pair of variables $(x_{i_j}, x_{i_{j'}})$ with $1 \leq j < j' \leq k$ such that there is no broken triangle on x_{k+1} relative to x_{i_j} and $x_{i_{j'}}$.*

Unfortunately, and unlike m-wBTP, the k-BTP property cannot be used for merging values when k is strictly greater than 2 (we recall that 2-BTP = BTP). As an example, the instance of Fig. 6(a) satisfies 3-BTP. To see this, observe that there is no broken triangle on x_k relative to x_i and x_ℓ. But, if we merge v'_k and v''_k, this CSP instance becomes satisfiable whereas it was not initially. Therefore, k-BTP (for k strictly greater than 2) is not a valid value-merging condition. We can also note that k-BTP ($k > 2$) and m-wBTP ($m > 0$) are incomparable, since it can happen that m-wBTP-merging can authorize more broken triangles than k-BTP. For example, the instance in Fig. 6(b) satisfies 1-wBTP but not 3-BTP: there are broken triangles on the variable x_k for each pair of other variables, but in each case the fourth variable is a support variable.

Naanaa has given two other generalisations of BTP which define tractable classes [18,19]. It has been shown [8] that the notion of directional rank $k-1$ [18] strictly generalises k-BTP. We can deduce that the example of Fig. 6(a) has directional rank 2, which shows that directional rank k (for $k \geq 2$) cannot be used to merge values (knowing that the case $k = 1$ corresponds to BTP).

The notion WBTP [19] inspired our definition of 1-wBTP, but is different. We first give the definition of WBTP before showing that it can be seen as a strictly stronger condition than 1-wBTP (and thus leads to less mergings).

Definition 8 (WBTP [19]). *A binary CSP instance equipped with an order $<$ on its variables satisfies WBTP (Weak Broken Triangle Property) if for each triple of variables $x_i < x_j < x_k$ and for all $v_i \in D(x_i)$, $v_j \in D(x_j)$ such that $(v_i, v_j) \in R(C_{ij})$, there is a variable $x_\ell < x_k$ such that when $v_\ell \in D(x_\ell)$ is compatible with v_i and v_j, then $\forall v_k \in D(x_k)$,*

$$(v_\ell, v_k) \in R(C_{\ell k}) \Rightarrow ((v_i, v_k) \in R(C_{ik}) \ \wedge \ (v_j, v_k) \in R(C_{jk}))$$

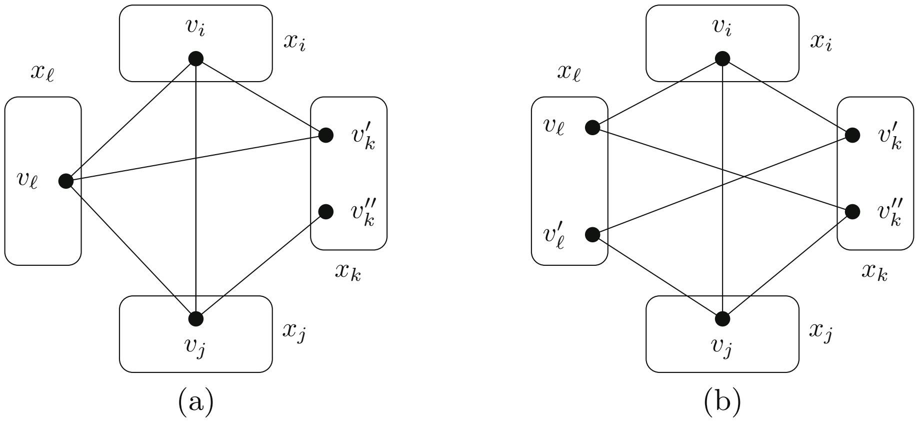

Fig. 6. (a) An instance which does not satisfy 1-wBTP but does satisfy 3-BTP, for the variable ordering $x_\ell < x_i < x_j < x_k$. (b) An instance which satisfies 1-wBTP but does not satisfy 3-BTP, for the variable ordering $x_\ell < x_i < x_j < x_k$.

Proposition 5. *If a binary CSP instance equipped with an order $<$ on its variables satisfies WBTP, then it satisfies 1-wBTP for each pair of values in the domain of the last variable (relative to the order $<$).*

Proof. Suppose that the binary CSP instance I satisfies WBTP for the variable order $<$ and let x_k be the last variable of I according to this order. Suppose, for a contradiction, that I does not satisfy 1-wBTP on a pair of values $v'_k, v''_k \in D(x_k)$. Then, by the definition of 1-wBTP, there is a broken triangle (v_i, v_j, v'_k, v''_k) with $v_i \in D(x_i)$, $v_j \in D(x_j)$, $v'_k, v''_k \in D(x_k)$ such that there is no variable $x_\ell \in X \setminus \{x_i, x_j, x_k\}$ such that $\forall\ v_\ell \in D(x_\ell)$ compatible with v_i and v_j, we have $(v_\ell, v'_k), (v_\ell, v''_k) \notin R(C_{\ell k})$.

But WBTP guarantees the existence of a variable $x_\ell < x_k$ such that $\forall v_\ell \in D(x_\ell)$ compatible with v_i and v_j, we have $\forall v_k \in D(x_k)$,

$$(v_\ell, v_k) \in R(C_{\ell k}) \Rightarrow ((v_i, v_k) \in R(C_{ik})\ \wedge\ (v_j, v_k) \in R(C_{jk}))$$

The existence of the broken triangle (v_i, v_j, v'_k, v''_k) implies that $x_\ell \notin \{x_i, x_j\}$ and so $x_\ell \in X \setminus \{x_i, x_j, x_k\}$. On the other hand, since (v_i, v_j, v'_k, v''_k) is a broken triangle,

$$(v_i, v_k) \in R(C_{ik}) \wedge (v_j, v_k) \in R(C_{jk})$$

is false for $v_k \in \{v'_k, v''_k\}$. We can deduce that $(v_\ell, v'_k), (v_\ell, v''_k) \notin R(C_{\ell k})$, a contradiction. □

Imposing WBTP is strictly stronger than imposing 1-wBTP. WBTP imposes a condition on each value $v_k \in D(x_k)$ relative to the *same variable* x_ℓ, whereas 1-wBTP (for each pair of values $v'_k, v''_k \in D(x_k)$) imposes an equivalent condition but for which the variable x_ℓ can vary according to the values v'_k, v''_k. The instance in Fig. 7 satisfies 1-wBTP but not WBTP because:

- only variable x_{ℓ_2} supports (v_i, v_j, v'_k, v''_k),
- only variable x_{ℓ_1} supports (v_i, v_j, v'_k, v'''_k),

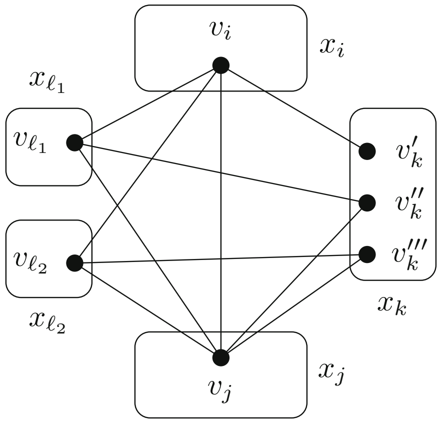

Fig. 7. An instance that satisfies 1-wBTP but not WBTP.

Therefore, there is no variable which supports at the same time the broken triangles (v_i, v_j, v'_k, v''_k) and (v_i, v_j, v'_k, v'''_k).

WBTP defines a tractable class [19]. We now show that this is also true for m-wBTP.

Definition 9. *Let I be a m-wBTP binary CSP instance on variables $x_1, \ldots, x_n$ ordered by $<$.*

- *The **BT-variable set** B_k of x_k is the set of the variables $x_i < x_k$ such that there is a broken triangle on x_k relative to x_i (and some other variable $x_j < x_k$).*
- *A **shield set** S_k of x_k is a set of variables $x_\ell < x_k$ such that for each broken triangle (v_i, v_j, v'_k, v''_k) on x_k relative to variables $x_i, x_j < x_k$, each partial solution $(v_{\ell_1}, \ldots, v_{\ell_r}, v_i, v_j)$ of its support variables, has a shield variable $x_{\ell_\alpha} \in S_k$.*
- *The **BT-width** of x_k is the smallest value of $|B_k \cap S_k|$ among all shield sets S_k of x_k. The BT-width of I is the maximum BT-width of its variables.*

Observe that for constants b and m, it is possible to determine in polynomial time whether a given instance (with a fixed variable order) has BT-width less than or equal to b (by exhaustive search). The BT-width provides an upper bound on the minimum level of consistency required to solve an instance, as demonstrated by the following theorem.

Theorem 2. *If a m-wBTP binary CSP instance I has BT-width b and is strong directional* $\max(2, b+1)$*-consistent, then I has a solution.*

Proof. Let I be a binary CSP instance which has BT-width b and is directional $(b+1)$-consistent. For simplicity of presentation, we suppose that the variable order is $x_1 < \ldots < x_n$. We suppose that it has a partial solution $\sigma = (v_1, \ldots, v_{k-1})$ on the variables $(x_1, \ldots, x_{k-1})$. We will show that this partial solution can be extended to a partial solution on $(x_1, \ldots, x_k)$. The base case of the induction is easily seen to be true, since by arc consistency there is necessarily a partial solution on the first two variables.

Let B_k be the set of the BT-variables of x_k and let S_k be a shield set of x_k such that $|B_k \cap S_k| \leq b$. By directional $(b+1)$-consistency, any partial solution on the variables $B_k \cap S_k$ can be extended to variable x_k. Therefore $\exists v_k \in D(x_k)$ such that

$$\forall x_i \in B_k \cap S_k,\ (v_i, v_k) \in R(C_{ik}) \tag{1}$$

Denote by $B_k(\sigma)$ the variables $x_i \in B_k$ such that there is a broken triangle of the form (v_i, v_j, v'_k, v''_k) on x_k (where v_i, v_j are assignments from σ). Similarly, let $S_k(\sigma)$ be the variables of S_k which shield such broken triangles. Let $N_k(\sigma)$ be the variables $x_i < x_k$ such that $x_i \notin B_k(\sigma)$. The sub-instance of I on variables $N_k(\sigma) \cup \{x_k\}$ has no broken triangles on x_k. Therefore, $\exists u_k \in D(x_k)$ such that $(v_i, u_k) \in R(C_{ik})$ for all $x_i \in N_k(\sigma)$ [7]. If $N_k(\sigma) = \emptyset$, then u_k is simply an arbitrary element of $D(x_k)$. We will show that one of $(v_1, \ldots, v_{k-1}, v_k)$ or $(v_1, \ldots, v_{k-1}, u_k)$ is a partial solution.

Suppose, for a contradiction, that this is not the case. Then $\exists x_i, x_j < x_k$ such that $(v_i, u_k) \notin R(C_{ik})$ and $(v_j, v_k) \notin R(C_{jk})$. We must have $x_i \in B_k(\sigma)$ and $x_j \notin B_k \cap S_k$. Since $x_i \in B_k(\sigma)$, there is a broken triangle (v_i, v_h, v'_k, v''_k) on x_k with $(v_i, v'_k) \in R(C_{ik})$. This broken triangle must have a shield variable $x_\ell \in S_k(\sigma)$. If $x_\ell \in N_k(\sigma)$, then $(v_\ell, u_k) \in R(C_{\ell k})$. We also have $(v_\ell, v_i) \in R(C_{ik})$ (by the definition of a partial solution) and $(v_\ell, v'_k) \notin R(C_{\ell k})$ (by definition of a support variable). Since, by assumption, $(v_i, u_k) \notin R(C_{ik})$, we have a broken triangle (v_i, v_ℓ, v'_k, u_k) which is impossible since $x_\ell \in N_k(\sigma)$ and hence cannot participate in such a broken triangle. So the shield variable x_ℓ belongs to $B_k(\sigma) \cap S_k(\sigma)$. By (1), we have $(v_\ell, v_k) \in R(C_{\ell k})$. Suppose now that $(v_i, v_k) \notin R(C_{ik})$. Then we have a broken triangle (v_i, v_ℓ, v'_k, v_k). This broken triangle must have a shield variable x_m. By the same argument as for x_ℓ, we can deduce that $x_m \in B_k(\sigma) \cap S_k(\sigma)$. However, this contradicts (1) since we have $(v_m, v_k) \notin R(C_{mk})$ (since x_m is a shield variable of the broken triangle (v_i, v_ℓ, v'_k, v_k)). It follows that $(v_i, v_k) \in R(C_{ik})$. Indeed, we have shown

$$\forall x_i \in B_k(\sigma),\ (v_i, u_k) \notin R(C_{ik}) \ \Rightarrow\ (v_i, v_k) \in R(C_{ik}) \tag{2}$$

Now, if $x_j \in N_k(\sigma)$ we have a broken triangle (v_i, v_j, v_k, u_k), which is in contradiction with the definition of $N_k(\sigma)$, so we must have $x_j \in B_k(\sigma)$. Now, by (2) and our assumption that $(v_j, v_k) \notin R(C_{jk})$, we can deduce that $(v_j, u_k) \in R(C_{jk})$. We then have a broken triangle (v_i, v_j, v_k, u_k). This broken triangle must have a shield variable x_p. By definition of a shield variable, we must have $(v_p, v_k), (v_p, u_k) \notin R(C_{pk})$. But this is impossible since $x_p \in N_k(\sigma) \Rightarrow (v_p, u_k) \in R(C_{pk})$ and, by (2), $x_p \in B_k(\sigma) \Rightarrow (v_p, u_k) \in R(C_{pk}) \vee (v_p, v_k) \in R(C_{pk})$.

This contradiction shows that one of $(v_1, \ldots, v_{k-1}, v_k)$ or $(v_1, \ldots, v_{k-1}, u_k)$ is a partial solution. By induction, I has a solution. □

Naanaa showed that a binary arc-consistent CSP instance satisfying WBTP always has a solution [19]. We can observe that this is a special case of Theorem 2 since a WBTP instance has BT-width of 1.

An open question is whether it is possible to determine, in polynomial time, the existence of some variable order for which a given instance has BT-width b even for $b = 1$.

7 Experimental Results

In order to test the applicability of our merging rules, and in particular 1-wBTP-merging, we carried out an experimental study on all the binary benchmark instances of the 2008 international CSP solver competition[1] (a total of 3,795 instances). The 1-wBTP-merging algorithm is similar to the algorithm for BTP-merging [5]. More specifically, given a variable x_k, we check for each pair of values $v'_k, v''_k \in D(x_k)$ if these two values are mergeable by 1-wBTP-merging. Once a broken triangle on v'_k, v''_k is found, we search over the other $n - 3$ variables to see if there exists a variable x_ℓ which supports this broken triangle. If we find one, we continue the search for other broken triangles; if not, the test is finished for these two values. Finally, if there are no broken triangles or only weakly broken triangles on the pair v'_k, v''_k, we merge them. We do not attempt to maximize the number of merges since we know that this is an NP-hard problem, even in the case of BTP-merging [4]. We implemented the two merging algorithms to be tested (BTP-merging and 1-wBTP-merging) in C++ within our own CSP library. The experiments were performed on 8 Dell PowerEdge M820 blade servers with two processors (Intel Xeon E5-2609 v2 2.5 GHz and 32 GB of memory) under Linux Ubuntu 14.04.

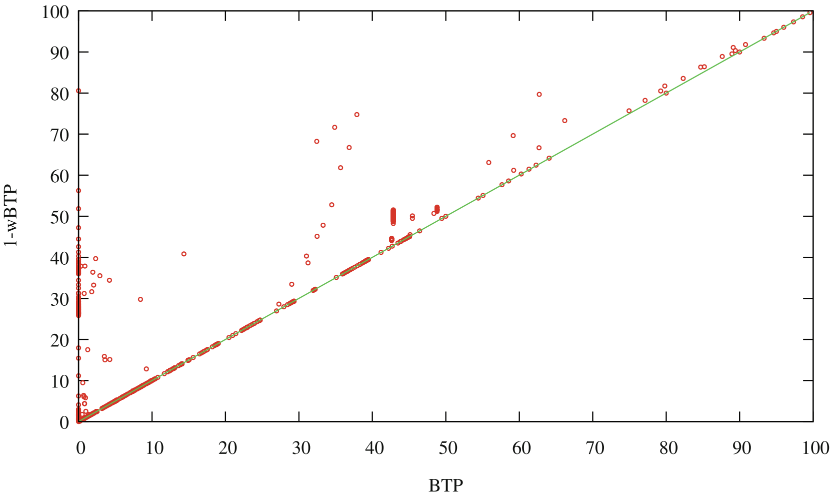

Fig. 8. Comparisons of the percentages of values merged by BTP and 1-wBTP.

For each benchmark instance, we performed BTP-merging and 1-wBTP-merging until convergence with a timeout of one hour. In all, we obtained results for 2,535 out of the 3,795 benchmarks and we succeeded in merging at least one

[1] See http://www.cril.univ-artois.fr/CPAI08 for more details.

Table 1. Experimental results on benchmarks.

Family	#benchmarks	#values	BTP-merging	1-wBTP-merging
BH-4-4	10	674	322	348
BH-4-7	20	2 102	883	932
ehi-85	98	2 079	891	1 045
ehi-90	100	2 205	945	1 100
graph-coloring/school	8	4 473	104	104
graph-coloring/sgb/book	26	1 887	534	534
os-taillard-4	30	2 932	1 820	1 978
rlfapScens	1	8 004	341	1 211
rlfapScensMod	6	8 788	2 415	5 169
subs	9	1 479	40	517
langford-2	22	879	0	233
langford-3	20	1 490	0	554
langford-4	16	1 784	0	504
queenAttacking	7	2 196	0	36

pair of values for 1,001 of these instances. In Table 1, the column #benchmarks shows the number of benchmark instances for which the test finished within the one-hour timeout. The column #values indicates the average total number of values in these benchmarks. The columns BTP-merging and 1-wBTP-merging give the number of merges performed respectively by BTP-merging and 1-wBTP-merging. In Fig. 8, we compare the percentages of domain reduction by BTP-merging and 1-wBTP-merging instance by instance. If, for the majority of instances, the results are comparable, we can observe that for certain instances, 1-wBTP merges significatively more values than BTP. This is notably the case for the instances in the `langford-*` family for which 1-wBTP merges from 25 to 80 % of the values whereas BTP does not merge any.

8 Conclusion

In this paper we have studied value-merging conditions in binary CSP instances, based on a generalisation of BTP. We proposed a family of definitions based on the notion of a weakly broken triangle, which is a broken triangle supported by one or more variables in order to preserve satisfiability after merging.

We have shown that m-wBTP together with different levels of consistency defines a family of tractable classes. Possible links with bounded treewidth are worth investigating. From a practical point of view, it would be interesting to investigate the influence of the order in which merges are performed on the total number of merges. We know that finding the best order in which to perform m-wBTP-merging operations is NP-hard even in the case $m = 0$ [4].

References

1. Carbonnel, C., Cooper, M.C.: Tractability in constraint satisfaction problems: a survey. Constraints **21**(2), 115–144 (2016)
2. Cohen, D.A., Cooper, M.C., Escamocher, G., Živný, S.: Variable elimination in binary CSP via forbidden patterns. In: Proceedings of IJCAI (2013)
3. Cohen, D.A., Cooper, M.C., Escamocher, G., Živný, S.: Variable and value elimination in binary constraint satisfaction via forbidden patterns. J. Comput. Syst. Sci. **81**(7), 1127–1143 (2015)
4. Cooper, M.C., Duchein, A., El Mouelhi, A., Escamocher, G., Terrioux, C., Zanuttini, B.: Broken triangles: from value merging to a tractable class of general-arity constraint satisfaction problems. Artif. Intell. **234**, 196–218 (2016)
5. Cooper, M.C., El Mouelhi, A., Terrioux, C., Zanuttini, B.: On broken triangles. In: O'Sullivan, B. (ed.) CP 2014. LNCS, vol. 8656, pp. 9–24. Springer, Heidelberg (2014)
6. Cooper, M.C., Jeavons, P., Salamon, A.: Hybrid tractable CSPs which generalize tree structure. In: Proceedings of ECAI, pp. 530–534 (2008)
7. Cooper, M.C., Jeavons, P., Salamon, A.: Generalizing constraint satisfaction on trees: hybrid tractability and variable elimination. Artif. Intell. **174**, 570–584 (2010)
8. Cooper, M.C., Jégou, P., Terrioux, C.: A microstructure-based family of tractable classes for CSPs. In: Pesant, G. (ed.) CP 2015. LNCS, vol. 9255, pp. 74–88. Springer, Heidelberg (2015)
9. Cooper, M.C., Živný, S.: The power of arc consistency for CSPs defined by partially-ordered forbidden patterns. In: Proceedings of LICS (2016)
10. El Mouelhi, A., Jégou, P., Terrioux, C.: Hidden tractable classes: from theory to practice. In: Proceedings of ICTAI, pp. 437–445 (2014)
11. El Mouelhi, A., Jégou, P., Terrioux, C.: A hybrid tractable class for non-binary CSPs. Constraints **20**(4), 383–413 (2015)
12. Freuder, E.C.: Eliminating interchangeable values in constraint satisfaction problems. In: Proceedings of AAAI, pp. 227–233 (1991)
13. Jégou, P.: Decomposition of domains based on the micro-structure of finite constraint satisfaction problems. In: Proceedings of AAAI, pp. 731–736 (1993)
14. Jégou, P., Terrioux, C.: The extendable-triple property: a new CSP tractable class beyond BTP. In: Proceedings of AAAI, pp. 3746–3754 (2015)
15. Likitvivatanavong, C., Yap, R.H.C.: Many-to-many interchangeable sets of values in CSPs. In Proceedings of SAC, pp. 86–91 (2013)
16. Mackworth, A.K.: Consistency in networks of relations. Artif. Intell. **8**, 99–118 (1977)
17. Montanari, U.: Networks of constraints: fundamental properties and applications to picture processing. Artif. Intell. **7**, 95–132 (1974)
18. Naanaa, W.: Unifying and extending hybrid tractable classes of CSPs. J. Exp. Theor. Artif. Intell. **25**(4), 407–424 (2013)
19. Naanaa, W.: Extending the broken triangle property tractable class of binary CSPs. In: Proceedings of the 9th Hellenic Conference on Artificial Intelligence, SETN, pp. 3:1–3:6 (2016)
20. Sabin, D., Freuder, E.C.: Contradicting conventional wisdom in constraint satisfaction. In: Proceedings of ECAI, pp. 125–129 (1994)