Locally Consistent Constraint Satisfaction Problems with Binary Constraints

Manuel Bodirsky * Daniel Král'† ‡

Abstract

An instance of a constraint satisfaction problem is k-consistent if any k constraints of it can be simultaneously satisfied. We focus on constraint languages with a single binary constraint. In this case, the constraint satisfaction problem is equivalent to the question whether there is a homomorphism from an input digraph G to a fixed target digraph H. The instance corresponding to G is k-consistent if every subgraph of G of size at most k is homomorphic to H. Let $\rho_k(H)$ be the largest ρ such that every k-consistent G contains a subgraph G' of size at least $\rho||E(G)||$ that is homomorphic to H. The ratio $\rho_k(H)$ reflects the fraction of constraints of a k-consistent instance that can be always satisfied. We determine $\rho_k(H)$ for all digraphs H that are not acyclic and show that $\lim_{k\to\infty} \rho_k(H) = 1$ if H has tree duality. For the latter case we design an efficient algorithm that computes in linear time for a given input graph G and $\varepsilon > 0$ either a homomorphism from almost entire graph G to H or a subgraph of G of bounded size that is not homomorphic to H.

*Humboldt-Universität zu Berlin, Institut für Informatik, Abteilung Algorithmen und Komplexität I, Unter den Linden 6, 10099 Berlin, Germany. E-mail: `bodirsky@informatik.hu-berlin.de`.

†Institute for Mathematics, Technical University Berlin, Strasse des 17. Juni 136, D-10623 Berlin, Germany. E-mail: `kral@math.tu-berlin.de`. The author is a postdoctoral fellow at TU Berlin within the framework of the European training network COMBSTRU.

‡Department of Applied Mathematics and Institute for Theoretical Computer Science (ITI), Faculty of Mathematics and Physics, Charles University, Malostranské náměstí 25, 118 00 Praha 1, Czech Republic. E-mail: `kral@kam.mff.cuni.cz`. Institute for Theoretical computer science is supported as project 1M0021620808 by Czech Ministry of Education.

1 Introduction

Constraint satisfaction problems form an important computational model for problems arising in many areas of computer science. This is witnessed by an enormous interest in the computational complexity of various variants of constraint satisfaction problems [1, 2, 8, 9, 10, 11, 23]. However, sometimes not all the constraints need to be satisfied but it suffices to satisfy a large fraction of them. In order to maximize this fraction, the input can usually be pruned at the beginning by removing small sets of contradictory constraints so that the input instance is usually "locally" consistent. Formally, an instance of the constraint satisfaction problem is *k-consistent* if any k constraints can be simultaneously satisfied.

A similar notion of local consistency can also be defined in terms of variables instead of constraints: an instance is k-consistent if the values of any k variables can be chosen so that any constraint on only these variables is satisfied. Our results also hold for this version of local consistency.

Both the notions of local consistency mentioned before differ fundamentally from the notion of *k-consistency* introduced by Freuder [11], and the notion of *relational k-consistency* studied by Dechter and van Beek [4]. There, an CSP instance is called k-consistent if every solution for the constraints on only $k-1$ variables can be extended to a solution to another variable in the instance.

1.1 History of Locally Consistent CSPs

The notion of local consistency considered in this paper can be traced back to the early 1980's. Lieberherr and Specker [19, 20] studied the corresponding problem for CNF formulas: they require that any k clauses of a given formula can be satisfied and asked what fraction of all the clauses of the formula can be satisfied. In their papers, they settled the case $k = 1, 2, 3$. A simpler proof of their results was later found by Yannakakis [24]. The case $k = 4$ was settled in [17] (exploring an interesting connection to so-called Usiskin's numbers [21]). Locally consistent CNF formulas can also be found in Chapter 20 of [16].

Huang et al. [14] and Trevisan [22] resolved the asymptotic behavior of locally consistent CNF formulas as k approaches infinity. Trevisan [22] was the first to define the notion of local consistency for CSPs with constraints that can be represented by Boolean predicates. For a set Π of Boolean constraints, let $\rho_k(\Pi)$ be the maximum ρ such that a fraction of at least

ρ constraints can be satisfied in any k-consistent input. Note that we now allow negations in the arguments of the constraints (the domain is not just a two-element set, but it is the Boolean field). If Π is the set of all the predicates of arity ℓ, then $\lim_{k\to\infty} \rho_k(\Pi) = 2^{1-\ell}$ [22]. The ratios $\rho_k(\Pi)$, $k \geq 1$, for a set Π consisting of a single predicate of arity at most three were determined by Dvořák et al. [6]. The asymptotic behavior for sets Π of predicates was studied in [18] where $\lim_{k\to\infty} \rho_k(\Pi)$ was expressed as the minimum of a certain functional on a convex set of polynomials derived from Π. Efficient algorithms for locally consistent CSPs with constraints that are Boolean predicates were also designed [6, 7, 18].

1.2 Our Contribution

In this paper, we initiate the study of locally consistent CSPs on larger finite domains. We consider the simplest case in this setting where all the constraints are of the same binary relation. The relation can be described by a directed graph H whose vertices correspond to the elements of the domain and where two vertices are joined by an arc if the ordered pair of the elements corresponding to them is contained in the relation. Similarly, the input can also be described by a directed graph G: the vertices of G correspond to the variables and the arcs to the given constraints. There is a satisfying assignment for the input if and only if G is homomorphic to H, i.e., there is a mapping $h : V(G) \to V(H)$ such that $h(u)h(v) \in E(H)$ for every $uv \in E(G)$.

The notion of local consistency translates to the language of digraph homomorphisms as follows: a graph G corresponds to an k-consistent input if every subgraph of G of size at most k, i.e., with at most k edges, is homomorphic to H. The ratio $\rho_k(H)$ denotes the largest ρ such that for any G corresponding to a k-consistent input, there is a mapping $h : V(G) \to V(H)$ preserving at least $\rho||G||$ arcs of G. Note that the version of local consistency defined in terms of variables instead of constraints also easily translates to the graph terminology: for that we require that each subgraph G' of order at most k is homomorphic to H. The corresponding ratios are then denoted by $\rho^v_k(H)$.

In our considerations, we can restrict to directed graphs H that are cores. A *core* is a directed graph H that does not have a homomorphism to a proper subgraph of H. It is well-known that any directed graph contains a unique (up to isomorphism) subgraph H' such that H' is a core and H is homomorphic to H'. Obviously, $\rho_k(H) = \rho_k(H')$ and $\rho^v_k(H) = \rho^v_k(H')$.

We show that if H contains a directed cycle (or a loop), then $\rho_k(H)$ and $\rho_k^v(H)$ coincide and they are equal to the fractional relative density $\delta'_{\mathrm{rel}}(H)$ of H as defined in Section 2. For such directed graphs H we design a simple linear time algorithm that finds a mapping $h : V(G) \to V(H)$ that preserves at least $\delta'_{\mathrm{rel}}(H) \cdot ||G||$ arcs of G.

In the rest of the paper we focus on acyclic directed graphs H. Using the notion of tree duality from [13] we show that $\lim_{k\to\infty} \rho_k(H) = 1$ for all orientations of a path and all acyclic tournaments. In general, the equality $\lim_{k\to\infty} \rho_k(H) = 1$ holds for all directed graphs H that have tree duality. Unfortunately, there are acyclic directed graphs H that do not have tree duality (there are even examples that are orientations of trees)—we discuss the cases that we were not able to settle in a satisfactory way in the concluding Section 5 where we also mention possible generalizations for CSPs with larger constraint language.

2 Target Graphs with Cycles or Loops

In this section, we focus on the binary relations (constraints) such that the corresponding target graph H contains a loop or a directed cycle. Note that this includes the case when the binary relation is symmetric.

For a directed graph H, we define the *fractional relative density* of H as follows:

$$\delta'_{\mathrm{rel}}(H) = \max_{x:V(H)\to\langle 0,1\rangle \mid \sum\limits_{v\in V(H)} x(v)=1} \sum_{uv\in E(H)} x(u)\cdot x(v)$$

where the maximum is taken over all functions $x : V(H) \to \langle 0, 1\rangle$ such that the sum of $x(v)$ is equal to one. In particular, if H contains a loop, then $\delta'_{\mathrm{rel}}(H) = 1$ (if the loop is incident to a vertex v, set $x(v) = 1$ and $x(v') = 0$ for $v' \neq v$). This notion of density is similar to that of *relative density* as used e.g. in [15]:

$$\delta_{\mathrm{rel}}(H) = \max_{\emptyset\neq H'\subseteq H} \frac{||H'||}{|H'|^2}.$$

The two notions are different in general. Consider a directed graph H depicted in Figure 1. The graph H is obtained by replacing each edge of K_5 by a bigon and removing two non-incident arcs. The relative density of H is $\delta_{\mathrm{rel}}(H) = 18/25 = 0.720$ but its fractional relative density is $\delta'_{\mathrm{rel}}(H) = 88/121 \approx 0.727$ (set $x(v) = 3/11$ or the vertex incident with 8 arcs and

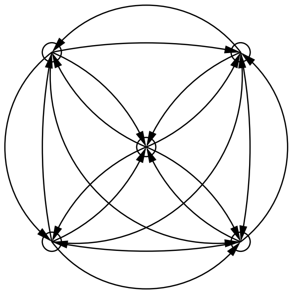

Figure 1: A digraph with different relative density and fractional relative density.

$x(v) = 2/11$ for the remaining vertices). However, the two notions coincide in the case that H corresponds to a symmetric binary relation [5]. Note that in this case both the target and the input graph can be viewed as an undirected graph (each bigon is replaced by a single undirected edge).

Proposition 1. *If a directed graph H corresponds to a symmetric binary relation, then its fractional relative density $\delta'_{\mathrm{rel}}(H)$ and its relative density $\delta_{\mathrm{rel}}(H)$ are the same. Moreover, $\delta_{\mathrm{rel}}(H) = 1$ if H contains a loop and otherwise $\delta_{\mathrm{rel}}(H) = 1 - 1/\omega(H)$ where $\omega(H)$ denotes the size of the largest subset A of vertices of H such that any two distinct vertices of A are joined by an arc.*

Proof. If H contains a loop, then $\delta_{\mathrm{rel}}(H) = \delta'_{\mathrm{rel}}(H) = 1$. In the rest, we assume that H has no loops. Let $x : V(H) \to \langle 0, 1\rangle$ be the function such that $\delta'_{\mathrm{rel}}(H) = \sum_{uv\in E(H)} x(u) \cdot x(v)$, $\sum_{v\in V(H)} x(v) = 1$ and the support of x is minimal. We show that $uv \in E(H)$ for any two vertices u and v contained in the support of x.

Assume the opposite and let u and v be two non-adjacent vertices such that $x(u), x(v) > 0$. Let $X_u = 2\sum_{uw\in E(H)} x(w)$ and $X_v = \sum_{uw\in E(H)} x(w)$. By symmetry, we can assume that $X_u \le X_v$. Consider the following labeling x':

$$x'(w) = \begin{cases} x(u) - \min\{x(u), x(v)\} & \text{if } w = u, \\ x(v) + \min\{x(u), x(v)\} & \text{if } w = v, \text{ and} \\ x(w) & \text{otherwise.} \end{cases}$$

Since the vertices u and v are non-adjacent in H, the following holds:

$$\sum_{uv\in E(H)} x'(u)\cdot x'(v) - \sum_{uv\in E(H)} x(u)\cdot x(v) = 2(X_v - X_u)\min\{x(u),x(v)\}.$$

It follows that $X_u = X_v$ from $\delta'_{\mathrm{rel}}(H) = \sum_{uv\in E(H)} x(u)\cdot x(v)$. Hence, $\delta'_{\mathrm{rel}}(H) = \sum_{uv\in E(H)} x'(u)\cdot x'(v)$. Since $X_u = X_v$, the configuration is again symmetric with respect to u and v and we may assume that $x(u) < x(v)$. Consequently, $x'(u) = 0$ and the support of x' is a subset of the support of x that contradicts the choice of x. We conclude that the support of x induces a complete graph in H.

It is an easy exercise in the mathematical analysis to show that if $\delta'_{\mathrm{rel}}(H) = \sum_{uv\in E(H)} x(u)\cdot x(v)$, then $x(u) = 1/k$ where k is the size of the support of x. Hence, $\delta'_{\mathrm{rel}}(H) = 1 - 1/\omega(H)$. Since it always holds that $\delta_{\mathrm{rel}}(H) \le \delta'_{\mathrm{rel}}(H)$, it follows that $\delta_{\mathrm{rel}}(H) = \delta'_{\mathrm{rel}}(H)$. □

We first observe that the fractional relative density $\delta'_{\mathrm{rel}}(H)$ of H is a lower bound on $\rho_k(H)$ (even if H is acyclic):

Proposition 2. *Let H be a directed graph. The following inequality holds for every $k \ge 1$:*

$$\rho_k(H) \ge \delta'_{\mathrm{rel}}(H)\ .$$

Moreover, there exists a deterministic algorithm that for any directed graph G finds a mapping $h : V(G) \to V(H)$ that preserves at least $\delta'_{\mathrm{rel}}(H)\cdot||G||$ arcs of G. The running time of the algorithm is linear in the size of G (if H is fixed).

Proof. Let $x : V(H) \to \langle 0,1\rangle$ be the function such that $\sum_{v\in V(H)} x(v) = 1$ and $\delta'_{\mathrm{rel}}(H) = \sum_{uv\in E(H)} x(u)\cdot x(v)$. Consider a mapping $h : V(G) \to V(H')$ that maps each vertex of G to a vertex v of H with probability $x(v)$. The probability that an arc of G is mapped to an arc of H is exactly $\sum_{uv\in E(H)} x(u)\cdot x(v) = \delta'_{\mathrm{rel}}(H)$. Hence, the expected number of arcs of G mapped to the arcs of H is $\delta'_{\mathrm{rel}}(H)\cdot||G||$. The mapping h can be found deterministically using the derandomization method based on conditional expectations as described in [24]. The running time of the algorithm remains linear in the size of G. □

In the proof of the next theorem, Markov's inequality and Chernoff's inequality are used to bound the probability of large deviations from the expected value. The reader is referred to [12] for further exposition.

Proposition 3. *Let X be a non-negative random variable with the expected value E. Then the following holds for every $\alpha \ge 1$:*

$$\mathrm{Prob}(X \ge \alpha) \le \frac{E}{\alpha} .$$

Proposition 4. *If X is a random variable equal to the sum of N independent random zero-one variables such that each of them equals one with the probability p, then the following holds for every $0 < \delta \le 1$:*

$$\mathrm{Prob}(X \ge (1+\delta)pN) \le e^{-\frac{\delta^2 pN}{3}} \quad \textit{and} \quad \mathrm{Prob}(X \le (1-\delta)pN) \le e^{-\frac{\delta^2 pN}{2}} .$$

The converse inequality of Proposition 2 is true if the target graph H contains a loop or a directed cycle.

Theorem 1. *If H is a directed graph that is not acyclic, then $\rho_k(H)$ is equal to $\delta'_{\mathrm{rel}}(H)$ for every $k \ge 1$.*

Proof. Fix $k \ge 1$ and ε, $0 < \varepsilon \le 1/2$. Let n be a sufficiently large integer. We construct a directed graph G of order n such that every subgraph of G of size at most k can be mapped to H, but every mapping $h : V(G) \to V(H)$ preserves at most $(\delta'_{\mathrm{rel}}(H) - \varepsilon)||G||$ arcs of G.

We first consider a random directed graph G_0 and we later prune it to obey all the constraints that G should satisfy. Let G_0 be a random graph of order n in which the arc from u to v, $u \ne v$, is included with probability $n^{-1+1/2k}$. The arcs are included to G_0 mutually independently. G_0 contains no loops. Since the expected number of arcs of G_0 is $n(n-1)n^{-1+1/2k} = n^{1/2k}(n-1)$, Proposition 4 implies that the probability that the number of arcs of G_0 is smaller than $(1-\varepsilon/4)n^{1+1/2k}$ does not exceed $1/4$ if n is sufficiently large.

Next, we estimate the number of (not necessarily consistently oriented) cycles of G_0. The expected number of bigons of G_0 is $\binom{n}{2}n^{-2+2/2k} \le n^{1/k}$. The expected number of cycles of G_0 of length ℓ, $3 \le \ell \le k$, is at most $n^\ell 2^\ell (n^{-1+1/2k})^\ell \le 2^\ell n^{1/2}$. By Proposition 3, the number of such bigons and cycles of length at most k does not exceed $4 \cdot k2^k n^{1/2}$ with probability at least $1/4$. Hence, if n is sufficiently large (recall that k is fixed), the number of arcs contained in bigons and cycles of length at most k is bounded by $\varepsilon n/4$ with probability at least $3/4$.

Let us now consider a mapping $h : V(G_0) \to V(H)$. Set $x(v) := |h^{-1}(v)|/n$ for $v \in V(H)$. By the definition of $\delta'_{\mathrm{rel}}(H)$, the following holds:

$$\sum_{uv \in E(H)} x(u)x(v) \le \delta'_{\mathrm{rel}}(H) \tag{1}$$

The expected number of arcs of G_0 preserved by h can be estimated using (1) as follows:

$$\sum_{uv\in E(H)} |h^{-1}(u)|\cdot|h^{-1}(v)|n^{-1+1/2k} =$$

$$\sum_{uv\in E(H)} x(u)x(v)n^{1+1/2k} \le \delta'_{\rm rel}(H)n^{1+1/2k}.$$

By Proposition 4, the probability that the number of arcs preserved by h exceeds $(1+\varepsilon/4)\delta'_{\rm rel}(H)n^{1+1/2k}$ is at most $e^{-\frac{\varepsilon^2\delta'_{\rm rel}(H)n^{1+1/2k}}{48}}$. Since there are $|V(H)|^n$ possible choices of the mapping h, and since the target graph H, the integer k and the real ε are fixed, the probability that there exists a mapping $h: V(G_0)\to V(H)$ that preserves more than $(1+\varepsilon/4)\delta'_{\rm rel}(H)n^{1+1/2k}$ arcs of G_0 does not exceed $1/4$ if n is sufficiently large.

We conclude based on the discussions in the previous paragraphs that the following holds with a positive probability:

1. G_0 contains at least $(1-\varepsilon/4)n^{1+1/2k}$ arcs,

2. the size of the set E of the arcs contained in bigons or cycles of length at most k in G_0 does not exceed $\varepsilon n/4$, and

3. every mapping $h: V(G_0)\to V(H)$ preserves at most $(1+\varepsilon/4)\,\delta'_{\rm rel}(H)n^{1+1/2k}$ arcs of G_0.

Therefore, there exists a graph G_0 with the above three properties. The final graph G is obtained from G_0 by removing the arcs contained in the set E.

We argue that every mapping $h: V(G)\to V(H)$ preserves at most $(\delta'_{\rm rel}(H)+\varepsilon)||G||$ arcs of G: the size $||G||$ of G is at least $(1-\varepsilon/2)n^{1+1/2k}$. Since every mapping $h: V(G_0)\to V(H)$ preserves at most $(1+\varepsilon/4)\delta'_{\rm rel}(H)n^{1+1/2k}$ arcs of G_0 and G is a subgraph of G_0, every mapping $h: V(G)\to V(H)$ also preserves at most $(1+\varepsilon/4)\delta'_{\rm rel}(H)n^{1+1/2k}$ arcs. We infer the following bound on the fraction of arcs of G preserved by h (recall that $\varepsilon\le 1/2$):

$$\frac{(1+\varepsilon/4)\delta'_{\rm rel}(H)n^{1+1/2k}}{||G||} \le \frac{1+\varepsilon/4}{1-\varepsilon/2}\delta'_{\rm rel}(H) \le (1+\varepsilon)\delta'_{\rm rel}(H)\ .$$

Next, we show that any subgraph of G of size at most k is homomorphic to H. Let G' be such a subgraph of G. Since the size of G' is at most k, G'

is an orientation of a forest. Hence, there is a homomorphism from G' to any directed cycle. In particular, there is a homomorphism from G' to H, because H contains a loop or a directed cycle,

Since we constructed for every $\varepsilon > 0$ a graph G that corresponds to an instance of CSP that is k-consistent and the fraction of arcs preserved by any mapping $h : V(G) \to V(H)$ does not exceed $\delta'_{\rm rel}(H) + \varepsilon$, it follows that $\rho_k(H) \le \delta'_{\rm rel}(H)$. The opposite inequality follows from Proposition 2. $\square$

By Proposition 1 and Theorem 1, the following holds for symmetric binary relations:

Corollary 2. *Let H be a directed graph corresponding to a symmetric binary relation $\mathcal{R}$ on a set X. The following holds for every $k \ge 1$:*

$$\rho_k(H) = \begin{cases} 1 & \text{if there exists } a \in X \text{ such that } [a,a] \in \mathcal{R}, \text{ and} \\ 1 - 1/\ell & \text{otherwise,} \end{cases}$$

where ℓ is the size of the largest set $A \subseteq X$ such that $[a, a'] \in \mathcal{R}$ for any two distinct elements a and a' of A.

Since every subgraph of order at most k of the graph G obtained in the proof of Theorem 1 is an orientation of a forest, it can be mapped to H and thus we can conclude the following:

Corollary 3. *If H is a directed graph that is not acyclic, then $\rho^v_k(H)$ is equal to $\delta'_{\rm rel}(H)$ for every $k \ge 1$.*

3 Graph Homomorphisms and Tree Duality

A key ingredient to our algorithm in the next section is the notion of tree duality. A directed graph H *has tree duality* if a graph G is homomorphic to H if and only if every directed tree homomorphic to G is also homomorphic to H. It is not hard to see that every orientation of a simple path or every acyclic tournament has tree duality. Feder and Vardi [10] and Hell, Nešetřil and Zhu [13] observed that if H has tree duality, then the H-coloring problem (the decision problem whether a given graph is homomorphic to H) can be solved in polynomial time by a simple procedure called arc-consistency. We explain their approach in more detail.

An equivalent definition of having tree duality uses the notion of set graphs. For a directed graph H, let 2^H be the graph whose vertices are all

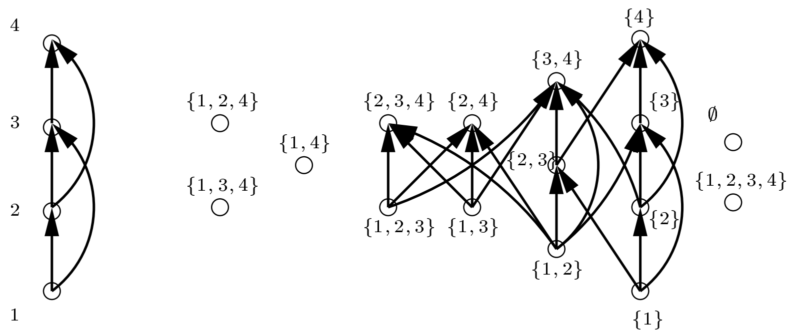

Figure 2: An example of a digraph H with tree duality and its set graph.

the $2^{|H|}$ subsets of vertices of H and two subsets U and V are joined by an arc if the following holds: for every vertex $u \in U$, there exists a vertex $v \in V$ such that uv is an arc of H, and for every vertex $v \in V$, there exists a vertex $u \in U$ such that uv is an arc of H. The graph 2^H is called the *set graph* of H. It can be shown that H has tree duality if and only if the graph 2^H is homomorphic to H [3, 10]. An example of a set graph can be found in Figure 2.

We now describe the *arc-consistency procedure* studied already in [11]. At the beginning, each vertex v of an input graph G is assigned the set of all the vertices of the target graph H. The set assigned to v after i steps of the algorithm is denoted by $\ell_i(v)$ and the initial sets are denoted by $\ell_0(v)$, i.e., $\ell_0(v) = V(H)$ for each $v \in V(G)$. At the i-th step, a vertex w of H is removed from the set assigned to v if G contains an arc vv' such that H does not contain an arc ww' for any $w' \in \ell_{i-1}(v')$ or G contains an arc $v'v$ such that H does not contain an arc $w'w$ for any $w' \in \ell_{i-1}(v')$. We say such an arc vv' *violated* at the i-th step. The procedure terminates when there are no further changes in the sets assigned to the vertices of G. The number of steps of the procedure never exceeds $|G| \cdot |H|$. The entire procedure can be implemented so that its running time is linear in $|G| + ||G||$ when the target graph H is fixed (and when the assignments ℓ_i at each step are implicitly represented).

Let $\ell(v)$ be the final set of the vertices of H assigned to a vertex v of the input graph G. If there exists a vertex v of G such that $\ell(v) = \emptyset$, then there is no homomorphism from G to H. On the other hand, if $\ell(v) \neq \emptyset$ for all $v \in V(G)$, then the mapping $h : V(G) \to V(2^H)$ such that $h(v) := \ell(v)$ is a homomorphism from G to the set graph of H. If H has tree duality, then the set graph 2^H is homomorphic to H. Consequently, in this case G is homomorphic to H. It is well-known that that if H has tree duality, then the arc-consistency procedure is a polynomial-time algorithm for the H-coloring problem [3, 10, 13].

4 Target Graphs with Tree Duality

In this section, we focus on binary relations such that the corresponding directed graph H has tree duality. Though the decision problem whether all the constraints can be satisfied can be solved in polynomial time if H has tree duality, the corresponding problem to maximize the number of satisfied constraints can be hard: consider a graph H consisting of a single oriented edge. The problem whether a graph G is homomorphic to H can be solved in polynomial time. On the other hand, if G is the directed graph obtained from an undirected graph G_0 by replacing each edge by a bigon, then the maximum number of arcs that can be preserved by a mapping from G to H is equal to the size of the maximum cut of G_0. Hence, the problem to maximize the number of preserved arcs (constraints) is NP-hard.

In this section, we show that if H has tree duality, then $\lim_{k\to\infty} \rho_k(H) = 1$, and we design an algorithm that either finds a good mapping from G to H or detects a subgraph of G of bounded size that is not homomorphic to H.

Theorem 4. *If H is a directed graph that has tree duality, then the following holds:*

$$\lim_{k\to\infty} \rho_k(H) = 1.$$

Moreover, there exists an algorithm that given an input graph G and $\varepsilon > 0$ either finds a mapping $h : V(G) \to V(H)$ that preserves at least $(1-\varepsilon)\cdot||G||$ arcs of G or detects a subgraph of G of size at most $|H|^{\lceil 2|H|/\varepsilon\rceil}$ that is not homomorphic to H. The running time of the algorithm is linear in $|G|+||G||$ if the target graph H is fixed.

Proof. We first describe the algorithm from the statement of the theorem. The algorithm invokes the arc-consistency procedure for the first $\lceil 2|H|/\varepsilon\rceil$ steps and constructs the assignments ℓ_i for $i = 0, \ldots, \lceil 2|H|/\varepsilon\rceil$. It then distinguishes two cases. The first case is that there exists a vertex v of

G and $i = 0, \ldots, \lceil 2|H|/\varepsilon \rceil$ such that $\ell_i(v) = \emptyset$. Let i be the smallest index with this property. For every $w \in V(H)$, there exists a step of the algorithm when w was removed from the set assigned to v because an edge incident to v was violated. For $w \in V(H)$, consider such an edge $v_w v$ and the corresponding step i_w. Note that $i_w < i$. Now, for every $w' \in V(H)$ missing in $\ell_{i_w}(v_w)$, consider the step when w' was removed from the sets assigned to v_w. At this point note that $\ell_{i_w}(v_w) \neq \emptyset$ by the choice of i. We obtain new sets of arcs of G that were violated before the i_w-th step and that caused vertices w' to be removed from the set assigned to v_w. Continue in this way unless the considered sets assigned to the vertices of G are equal to $V(H)$. The procedure terminates because the numbers i_w of steps decrease. Since $i \leq \lceil 2|H|/\varepsilon \rceil$, the number of violated arcs obtained in this way does not exceed the following:

$$|H|+|H|(|H|-1)+|H|(|H|-1)^2+\cdots+|H|(|H|-1)^{\lceil 2|H|/\varepsilon \rceil -1} \leq |H|^{\lceil 2|H|/\varepsilon \rceil} .$$

This set of arcs contains a subgraph of G that is not homomorphic to H.

The remaining case is that $\ell_i(v) \neq \emptyset$ for every $v \in V(G)$ and $i = 0, \ldots, \lceil 2|H|/\varepsilon \rceil$. Let E_i be the set of the arcs of G violated at the i-th step, $i = 1, \ldots, \lceil 2|H|/\varepsilon \rceil$. Since each edge can be violated at most $2|H|$ times (at each step when the edge is violated, the size of the set assigned to one of its end-vertices decreases), the sum $|E_1| + \cdots + |E_{\lceil 2|H|/\varepsilon \rceil}|$ does not exceed $2|H| \cdot ||G||$. In particular, there exists $i = 1, \ldots, \lceil 2|H|/\varepsilon \rceil$ such that $|E_i| \leq \varepsilon ||G||$. Consider now a mapping $h' : V(G) \to V(2^H)$ defined as $h'(v) := \ell_{i-1}(v)$. All the arcs that might not be preserved by h' are contained in E_i. Since 2^H is homomorphic to H, there is a homomorphism $h : V(G) \to V(H)$ that preserves all the arcs of G except for those of E_i. This finishes the description and the analysis of the algorithm. The bound on the running time of our algorithm follows from the discussions in Section 3.

Since the algorithm finds for a $|H|^{\lceil 2|H|/\varepsilon \rceil}$-consistent input a mapping that preserves a fraction of at least $(1-\varepsilon)$ arcs, $\rho_{|H|^{\lceil 2|H|/\varepsilon \rceil}}(H) \geq 1 - \varepsilon$. Hence, $\lim_{k \to \infty} \rho_k(H) = 1$. □

Since $\rho^v_{2k}(H) \geq \rho_k(H)$ and $\rho(H)$ is non-decreasing in k, Theorem 4 implies the following:

Corollary 5. *If H is a directed graph that has tree duality, then the following holds:*

$$\lim_{k \to \infty} \rho^v_k(H) = 1.$$

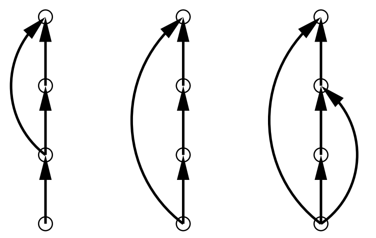

Figure 3: The three smallest digraphs H for which $\lim_{k\to\infty} \rho_k(H)$ is unknown.

Moreover, there exists an algorithm that given an input graph G and $\varepsilon > 0$ either finds a mapping $h : V(G) \to V(H)$ that preserves at least $(1-\varepsilon) \cdot ||G||$ arcs of G or detects a subgraph of G of order at most $1 + |H|^{\lceil 2|H|/\varepsilon \rceil}$ that is not homomorphic to H. The running time of the algorithm is linear in $|G| + ||G||$ if the target graph H is fixed.

5 Directions for Future Research

The main interest in locally consistent CSPs comes from the question how much it helps that the input is locally consistent. This is reflected by the behavior of $\rho_k(H)$ as a function of k for a fixed directed graph H. In case that the corresponding graph H contains a loop or a directed cycle, we have seen that the assumption on local consistency does not help at all. On the other hand, if H has tree duality, this assumption helps a lot. We were not able to settle the case when H is acyclic but does not have tree duality. In Figure 3, the reader can find the three smallest directed graphs H for which we were not able to compute the limit $\lim_{k\to\infty} \rho_k(H)$.

In the papers on locally consistent CSPs with constraints being Boolean predicates [6, 7, 18], the authors also addressed the weighted versions of the problems. Let us mention that all our results, in particular Theorems 1 and 4, Corollaries 2, 3 and 5, hold for the weighted versions of the problems, too. The reader is welcome to check him/her/itself that the proofs translate to this setting.

The ultimate goal is to settle the behavior of locally consistent CSPs with more types of constraints and with constraints of arbitrary arity. The approach to CSPs for binary constraints based on graphs with tree duality applies to all constraint languages that admit a set function, even if the constraint language contains several constraint types. Note that this class of computational problems contains many previously known tractable families of problems including Horn, constant, and ACI problems [3]. However, we have little knowledge of the behavior for constraint languages without tree duality.

Acknowledgement

The authors are indebted to Zdeněk Dvořák for discussions on relative density of graphs and to Pavol Hell for his comments on tree duality.

References

[1] A. Bulatov, A. Krokhin, P. Jeavons: The Complexity of Maximal Constraint Languages. In: Proc. of the 33rd Symposium on the Theory of Computation, STOC (2001) 667–674.

[2] S. Cook, D. Mitchell: Finding Hard Instances of the Satisfiability Problem: A Survey. In: Satisfiability Problem: Theory and Applications. DIMACS Series in DMTCS Vol. 35 AMS (1997).

[3] V. Dalmau, J. Pearson: Closure Functions and Width 1 Problems. In: Proc. of 5th International Conferences on Principles and Practice of Constraint Programming, CP, LNCS Vol. 1713, Springer-Verlag Berlin (1999) 159–173.

[4] R. Dechter, P. van Beek: Local and Global Relational Consistency. Theor. Comput. Sci. 173 (1997) 283–308.

[5] Z. Dvořák, personal communication.

[6] Z. Dvořák, D. Král', O. Pangrác: Locally Consistent Constraint Satisfaction Problems. In: Proc. of the 31st International Colloquium on Automata, Languages and Programming, ICALP, LNCS Vol. 3142, Springer-Verlag Berlin (2004) 469–480.

[7] Z. Dvořák, D. Král', O. Pangrác: Locally Consistent Constraint Satisfaction Problems, to appear in Theor. Comput. Sci.

[8] D. Eppstein: Improved Algorithms for 3-coloring, 3-edge-coloring and Constraint Satisfaction. In: Proc. of the 12th ACM-SIAM Symposium on Discrete Algorithms, SODA (2001) 329–337.

[9] T. Feder, R. Motwani: Worst-case Time Bounds for Coloring and Satisfiability Problems. J. Algorithms 45(2) (2002) 192-201.

[10] T. Feder, M. Vardi: Monotone monadic SNP and constraint satisfaction. In: Proc. of the 25th Symposium on the Theory of Computation, STOC (1993) 612–622.

[11] E. C. Freuder: A sufficient condition for backtrack-free search. J. ACM 29 (1982) 24–32.

[12] T. Hagerup, Ch. Rüb: A guided tour Chernoff bounds. Inform. Process. Letters 33 (1989) 305–308.

[13] P. Hell, J. Nešetřil, X. Zhu: Duality and polynomial testing of tree homomorphisms. Trans. Amer. Math. Soc. 348(4) (1996) 1281–1297.

[14] M. A. Huang, K. Lieberherr: Implications of Forbidden Structures for Extremal Algorithmic Problems. Theor. Comput. Sci. 40 (1985) 195–210.

[15] S. Janson, T. Łuczak, A. Ruciński: Random Graphs. Wiley & Sons, New York (2000).

[16] S. Jukna: Extremal Combinatorics with Applications in Computer Science. Springer, Heidelberg (2001).

[17] D. Král': Locally Satisfiable Formulas. In: Proc. of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA). SIAM (2004) 323-332.

[18] D. Král', O. Pangrác: An Asymptotically Optimal Linear-Time Algorithm for Locally Consistent Constraint Satisfaction Problems, submitted.

[19] K. Lieberherr, E. Specker: Complexity of Partial Satisfaction. J. of the ACM, 28(2) (1981) 411–422.

[20] K. Lieberherr, E. Specker: Complexity of Partial Satisfaction II. Technical Report 293, Dept. of EECS, Princeton University (1982).

[21] Z. Usiskin: Max-min Probabilities in the Voting Paradox. Ann. Math. Stat. 35 (1963) 857–862.

[22] L. Trevisan: On Local versus Global Satisfiability, to appear in SIAM J. Disc. Math. A preliminary version available as ECCC report TR97-12.

[23] G. J. Woeginger: Exact Algorithms for NP-hard Problems: A Survey. In: Proc. 5th International Workshop Combinatorial Optimization—Eureka, You Shrink. LNCS Vol. 2570. Springer-Verlag Berlin (2003) 185-207.

[24] M. Yannakakis: On the Approximation of Maximum Satisfiability. J. Algorithms 17 (1994) 475–502. A preliminary version appeared in Proc. of the 3rd Annual ACM-SIAM Symposium on Discrete Algorithms, SODA (1992) 1–9.