Locally Consistent Constraint Satisfaction Problems

Manuel Bodirsky* Daniel Král'†

Abstract

An instance of a constraint satisfaction problem (CSP) is *variable k-consistent* if any subinstance with at most k variables has a solution. For a fixed constraint language $\mathcal{L}$, $\rho_k(\mathcal{L})$ is the largest ratio such that any variable k-consistent instance has a solution that satisfies at least a fraction of $\rho_k(\mathcal{L})$ of the constraints. We provide an expression for the limit $\rho(\mathcal{L}) := \lim_{k\to\infty} \rho_k(\mathcal{L})$, and show that this limit coincides with the corresponding limit for *constraint k-consistent* instances, i.e., instances where all subinstances with at most k constraints have a solution. We also design an algorithm that for an input instance and a given ε either computes a solution that satisfies at least a fraction of $\rho(\mathcal{L})-\varepsilon$ constraints or finds a set of inconsistent constraints whose size only depends on ϵ. Most of our results apply both to weighted and to unweighted instances of the constraint satisfaction problem.

1 Introduction

Constraint satisfaction problems (CSPs) 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 the problem and its variants [2–6, 9–11, 13, 14, 16, 17, 21, 30]. However, sometimes

*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 Theoretical Computer Science (ITI), Faculty of Mathematics and Physics, Charles University, Malostranské náměstí 25, 118 00 Prague 1, Czech Republic. E-mail: `kral@kam.mff.cuni.cz`. Institute for Theoretical computer science is supported as project 1M0545 by Czech Ministry of Education.

not all the constraints of an input instance need to be satisfied, but it suffices to satisfy a large fraction of them. A natural notion in this context is the notion of local consistency. An CSP instance is *variable (constraint) k-consistent* if any subinstance with at most k variables (constraints) has a solution. In this paper, we focus on the effect of local consistency on the quality of an optimal solution with respect to the number of constraints that can be simultaneously satisfied. There is an interesting connection between this question and the notion of tree-duality, which has been studied in graph theory [16] and recently in logic [24].

Let us remark that there are several other notions of local consistency that are different from the notion that we analyze, such as the notion of *k–consistency* introduced by Freuder [13], or the notion of *relational k-consistency* studied by Dechter and van Beek [5]. However, we do not address any of these notions in this paper.

1.1 Previous Results

The notion of local consistency considered in this paper can be traced back to the early 1980's. Lieberherr and Specker [25, 26] 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 the clauses can be satisfied. In their papers, they settled the cases $k = 1, 2, 3$. A simpler proof of their results was later found by Yannakakis [31]. The case $k = 4$ was settled in [22]. There is an interesting connection between this problem and Usiskin numbers [29]. Locally consistent CNF formulas are also discussed in Chapter 20 of the monograph [20].

The asymptotic behavior of locally consistent CNF formulas when k approaches infinity was first addressed by Huang et al. [18] and further studied by Trevisan [28]. Trevisan [28] was the first to define the notion of local consistency for general CSPs with constraints being 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 for any k-consistent input. Note that in [28] negations of the arguments of the constraints are allowed, i.e., the domain is not just a two-element set, but it is the Boolean field.

In this scenario, $\lim_{k\to\infty} \rho_k(\Pi) = 2^{1-\ell}$ for a set Π of all predicates of arity ℓ [28]. 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. [7]. The asymptotic behavior of $\rho_k(\Pi)$ for fixed sets Π of predicates was studied in [23], and $\lim_{k\to\infty} \rho_k(\Pi)$ 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 [7, 8, 23].

1.2 Our Results

In the conference version of this paper, we studied CSPs with a single binary constraint type [1]. We further develop the methods used there to address the problem in full generality and provide an analysis for all constraint languages. In other words, the constraint language can now contain several types of constraints, which are not necessarily binary.

We now briefly summarize our results. Formal definitions of the mentioned quantities are given in Section 2, and the rigorous statements of the achieved results can be found in the subsequent sections.

Let $\mathcal{L}$ be a fixed constraint language. Then $\rho_{v,k}(\mathcal{L})$ denotes the largest ratio such that any variable k-consistent instance has a solution that satisfies at least a fraction of $\rho_{v,k}(\mathcal{L})$ of the constraints. Let $\rho_v(\mathcal{L})$ be the limit of $\rho_{v,k}(\mathcal{L})$ for k tending to infinity. In Section 3, we introduce a parameter $\pi(\mathcal{L})$ of the constraint language $\mathcal{L}$, and later show that $\rho_v(\mathcal{L})$ equals $\pi(\mathcal{L})$. Similarly, we introduce $\rho_{c,k}(\mathcal{L})$ as the largest ratio such that any *constraint* k-consistent instance has a solution that satisfies at least a fraction of $\rho_{c,k}(\mathcal{L})$ of the constraints. We show that the limit of this quantity also equals $\pi(\mathcal{L})$. Unless the language $\mathcal{L}$ contains a unary constraint, the result applies to both weighted and unweighted instances. In case that $\mathcal{L}$ contains a unary constraint, our results apply only to weighted instances.

Next, we develop techniques to find solutions that satisfy many constraints of an instance. In Section 4, we design an efficient algorithm that either constructs a solution of an input instance that satisfies at least the fraction of $\rho_v(\mathcal{L}) - \varepsilon$ of the constraints or finds an inconsistent set of constraints of size bounded by a function that only depends on ε.

In Section 5, we address the difference between weighted and unweighted instances of the CSP. Note that we do not allow to repeat the same constraint in unweighted instances to simulate weights. We show that if the constraint language does not contain a unary constraint, then weighted and unweighted locally consistent problems have the same extremal behavior.

Finally, in Section 6, we use our results to derive the results obtained in the conference version of this paper for constraint languages with a single binary constraint type. Results for such constraint languages are of particular interest, since the constraint satisfaction problem then corresponds to the (directed) graph homomorphism problem. In particular, we prove that $\rho(\mathcal{L}) = 1$ if and only if $\rho(\mathcal{L}) = 1$ if and only if $\mathcal{L}$ has *tree duality*. The concept of tree-duality for (directed) graphs is well-studied in graph theory [16].

2 Notation and Definitions

A *constraint language* is a pair $\mathcal{L} = (D,\mathcal{U})$ such that D is a finite *domain* and $\mathcal{U} = \{U_1, \ldots, U_k\}$ is a finite set of relations on D, i.e., $U_i \subseteq D^{r_i}$ for some positive integer r_i. The value of r_i is the *arity* of the relation U_i. Relations of $\mathcal{U}$ are called *constraint types*.

An *instance* of a constraint satisfaction problem with the constraint language $(D,\mathcal{U})$ is a pair $(X,\mathcal{R})$ such that X is the set of *variables* and $\mathcal{R} = \{R_1, \ldots, R_m\}$ is the set of *constraints*. Each R_i is an ordered r_j-tuple of the (not necessarily distinct) elements of X, and R_i is associated with one of the k constraint types. Inspired by terminology in graph theory, we say that an instance is *simple* if each constraint in $\mathcal{R}$ contains each variable at most once. We say that a mapping $\varphi : X \to D$ *satisfies* a constraint $R_i = (x_1, \ldots, x_{r_j})$ of constraint type U_j if $(\varphi(x_1), \ldots, \varphi(x_{r_j})) \in U_j$. Mappings φ from X to D will be called *solutions* of an instance; an *exact solution* is a solution that satisfies all the constraints of the instance.

A *weighted instance* is an instance such that each constraint R_i is assigned a non-negative *weight* $w(R_i)$. The *total weight* w_0 of an instance is the sum of the weights of all its constraints. The weight $w(\varphi)$ of a solution $\varphi : X \to D$ is the sum of the weights $w(R_i)$ of the satisfied constraints R_i, i.e., those constraints $(x_1, \ldots, x_{r_j})$ with $(\varphi(x_1), \ldots, \varphi(x_{r_j})) \in U_j$. An *optimum solution* is a solution of maximum weight. Finally, if all the constraints are assigned the same weight, the instance is called *uniform*; uniform instances correspond to unweighted ones.

An important class of constraint satisfaction problems are those corresponding to graph homomorphism problems. If G and H are graphs, then a *homomorphism* from G to H is a mapping $\varphi : V(G) \to V(H)$ where $\varphi(u)\varphi(v) \in E(H)$ for every edge $uv \in E(G)$. The same definition applies both to undirected and directed graphs. The problem of deciding an existence of a homomorphism to a graph H corresponds to the constraint language $(D,\mathcal{U})$ where $D = V(H)$ and $\mathcal{U}$ consists of a single binary constraint type U_1 such that $(u, v) \in U_1$ if and only if $uv \in E(H)$. The instance $(X,\mathcal{R})$ corresponding to an input graph G consists of $|V(G)|$ variables x_v

corresponding to vertices $v \in V(G)$, and $\mathcal{R}$ contains a binary constraint (x_u, x_v) for every edge $uv \in E(G)$. It is easy to see that exact solutions of the constructed instance of CSP are in one-to-one correspondence with homomorphisms from G to H.

As in the case of graphs and graph homomorphisms, one may view general constraint satisfaction problems as homomorphism problems for relational structures [21]. Let $(X, \mathcal{R})$ and $(X', \mathcal{R}')$ be two instances with the same constraint language $\mathcal{L}$. A mapping $\psi : X \to X'$ is a *homomorphism* from $(X, \mathcal{R})$ to $(X', \mathcal{R})$ if for every constraint $(x_1, \ldots, x_r) = R_i \in \mathcal{R}$ of type U_i, there is a constraint $R'_i = (\psi(x_1), \ldots, \psi(x_r)) \in \mathcal{R}'$ of type U_i. If such a homomorphism exists, we say that $(X, \mathcal{R})$ is *homomorphic* to $(X', \mathcal{R})$. It is not hard to see that if $(X, \mathcal{R})$ is homomorphic to $(X', \mathcal{R}')$ and $(X', \mathcal{R}')$ has an exact solution φ', then $(X, \mathcal{R})$ has also an exact solution—set $\varphi(x) = \varphi'(\psi(x))$ for every $x \in X$.

In this paper, we address locally consistent constraint satisfaction problems. An instance $(X, \mathcal{R})$ of a constraint satisfaction problem is *variable k-consistent* if $(X', \mathcal{R}')$ has an exact solution for every set X' of at most k elements of X and $\mathcal{R}'$ containing all the constraints of $\mathcal{R}$ that include only the variables of X'. Similarly, an instance is *constraint k-consistent* if $(X, \mathcal{R}')$ has an exact solution for every subset $\mathcal{R}' \subseteq \mathcal{R}$ with at most k elements.

Let $\mathcal{L}$ be a constraint language. We define $\rho^w_{v,k}(\mathcal{L})$ to be the largest ratio α such that every vertex k-consistent instance with the language $\mathcal{L}$ with total weight w_0 has a solution of weight at least αw_0. Similarly, $\rho^w_{c,k}(\mathcal{L})$ is the largest such ratio for constraint k-consistent instances. Analogously, we define ratios $\rho_{v,k}(\mathcal{L})$ and $\rho_{c,k}(\mathcal{L})$ for uniform instances with the language $\mathcal{L}$ (these ratios correspond to the unweighted case). Finally, we define the notation for the limit values:

$$\rho^w_v(\mathcal{L}) = \lim_{k\to\infty} \rho^w_{v,k}(\mathcal{L}) \ .$$

Similarly, we use $\rho^w_c(\mathcal{L})$, $\rho_v(\mathcal{L})$ and $\rho_c(\mathcal{L})$.

An important notion in our considerations is the concept of the *set structure* for a constraint language [4, 11]. It is the counterpart of set graphs studied in the area of graph homomorphisms [16]. If $\mathcal{L} = (D, \mathcal{U})$ is a constraint language, then the *set structure* $2^{\mathcal{L}}$ of $\mathcal{L}$ is an instance of $\mathcal{L}$ with $2^{|D|} - 1$ variable x_A, each corresponding to a non-empty subset $A \subseteq D$. An r_i-tuple $(x_{A_1}, \ldots, x_{A_{r_i}})$ forms a constraint of type $U_i \in \mathcal{U}$ if it satisfies the following: for every j, $1 \le j \le r_i$, and every $y_j \in A_j$, there exist $y_{j'} \in A_{j'}$,

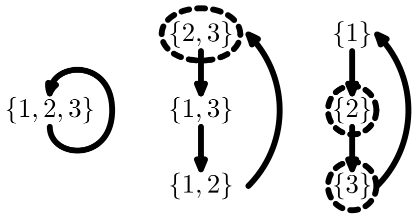

Figure 1: An example of the set structure of a constraint language. The domain of the language is $D = \{1, 2, 3\}$, and the language contains two relations R_1 and R_2, where R_1 is the unary relation $\{x \mid x \neq 1\}$, and R_2 is the binary relation $\{[x, y] \mid x = y + 1 \pmod 3\}$. The relation R_1 is depicted by dashed circles and R_2 by arrows in the figure.

$1 \leq j' \leq r_i$ and $j' \neq j$, such that $(y_1, \ldots, y_{r_i}) \in U_i$. Note that the set structure $2^{\mathcal{L}}$ is in general not simple. An example of the set structure of a constraint language can be found in Figure 1. The importance of the concept of the set structure arises from the fact that if the set structure $2^{\mathcal{L}}$ has an exact solution, then the CSP for $\mathcal{L}$ can be solved in polynomial time [11, 17].

In our proofs, we often use probabilistic arguments involving Markov's inequality and Chernoff's inequality. We recall these two well-known results from probability theory for the reader's convenience, and refer to [15] for further details.

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

$$\mathrm{Prob}(X \geq \alpha) \leq \frac{E}{\alpha} .$$

Proposition 2. *Let X be a random variable for the sum of N zero-one independent random variables each of which is equal to 1 with probability p. Then the following holds for every $0 < \delta \leq 1$:*

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

3 Upper Bounds on ρ

In this section, we prove upper bounds on the limits $\rho_v^w(\mathcal{L})$, $\rho_c^w(\mathcal{L})$, $\rho_v(\mathcal{L})$ and $\rho_c(\mathcal{L})$. (And in the next section we will prove the matching lower bounds.) To achieve our goal, we have to construct a locally consistent instance with constraint language $\mathcal{L}$. One way of proving that a given instance is locally consistent is to show that it is homomorphic to $2^{\mathcal{L}}$ and that it does not have short cycles. A *cycle* of an instance $(X, \mathcal{R})$ is a cyclic sequence of constraints $R_1, \ldots, R_k \in \mathcal{R}$, $k \geq 1$, with arity at least two such that any two consecutive constraints share at least one variable. In case that $k = 1$ we additionally require that at least two variables in R_1 coincide. The number k of constraints is called the *length* of the cycle. Cycles of length 1 are also called *loops*. Note that no cycle can contain a unary constraint. A cycle is *minimal* if no proper subset of its constraints forms a cycle. A CSP instance $(X, \mathcal{R})$ is *acyclic* if it does not contain a cycle. In particular, acyclic instances do not contain loops.

An instance is homomorphic to the set structure if and only if the so-called *arc-consistency procedure* detects a contradictory set of constraints in the instance [4, 12]. Together with the well-known fact that the arc-consistency procedure solves acyclic instances of the constraint satisfaction problem (see [6, 14] for stronger results on bounded tree-width instances), this implies the following lemma.

Lemma 3. *Let $(X, \mathcal{R})$ be an acyclic CSP instance with constraint language $\mathcal{L}$. If $(X, \mathcal{R})$ is homomorphic to $2^{\mathcal{L}}$, then $(X, \mathcal{R})$ has an exact solution.*

Next, we show how Lemma 3 can be applied to show that an instance with no short cycles is locally consistent:

Lemma 4. *Let $(X, \mathcal{R})$ be an instance of CSP($\mathcal{L}$). If $(X, \mathcal{R})$ is homomorphic to $2^{\mathcal{L}}$ and $(X, \mathcal{R})$ does not have a cycle of length at most k, then $(X, \mathcal{R})$ is both variable and constraint k-consistent.*

Proof. First, we show that $(X, \mathcal{R})$ is constraint k-consistent. Let $R_1, \ldots, R_k \in \mathcal{R}$ be k constraints from the instance. Since $(X, \mathcal{R})$ does not have a cycle of length at most k, the instance $(X, \mathcal{R}')$ with $\mathcal{R}' = \{R_1, \ldots, R_k\}$ is acyclic. It is easy to observe that $(X, \mathcal{R}')$ is homomorphic to $2^{\mathcal{L}}$, since $(X, \mathcal{R})$ is homomorphic to $2^{\mathcal{L}}$, and since $\mathcal{R}' \subseteq \mathcal{R}$. By Lemma 3, there is an exact solution of $(X, \mathcal{R}')$.

Next, we show that $(X, \mathcal{R})$ is variable k-consistent. Let $X' \subseteq X$ be any subset of X with k variables, and let $\mathcal{R}' \subseteq \mathcal{R}$ be the set of all the constraints of $(X,\mathcal{R})$ that contain only the variables of X. If $(X',\mathcal{R}')$ contains a cycle, it also contains a minimal cycle. The length of a minimal cycle cannot exceed $|X'| = k$. Since $(X,\mathcal{R})$ does not have a cycle of length at most k, $(X',\mathcal{R}')$ is acyclic. Again, we know that $(X,\mathcal{R}')$ is homomorphic to $2^{\mathcal{L}}$ and it has an exact solution by Lemma 3. □

Our bounds on $\rho_v^w(\mathcal{L})$, $\rho_c^w(\mathcal{L})$, $\rho_v(\mathcal{L})$, and $\rho_c(\mathcal{L})$ are given in terms of the following parameter $\pi(\mathcal{L})$ of a constraint language $\mathcal{L} = (D,\mathcal{U})$.

Definition 1. *The following quantity is denoted by* $\pi(\mathcal{L})$*.*

$$\min_{w} \max_{p: X\times D\to[0,1]} \sum_{R_i=(x_1,\dots,x_{r_i})\in\mathcal{R}} w(R_i)\cdot \sum_{(d_1,\dots,d_{r_i})\in U_j} \prod_{k=1}^{r_i} p(x_{A_k}, d_k) \qquad (1)$$

where the minimum is taken over all weight functions $w : \mathcal{R} \to [0,\infty)$ *on the constraints of the set structure* $2^{\mathcal{L}} = (X,\mathcal{R})$ *with total weight* $w_0 = 1$*, the maximum is taken over all functions* $p : X \times D \to [0,1]$ *that satisfy*

$$\sum_{d\in D} p(x,d) = 1$$

for every variable $x \in X$ *of the set structure* $2^{\mathcal{L}}$*. Finally,* U_j *is the constraint type of the constraint* R_i*.*

It is easy to verify that both the minimum and the maximum in (1) are attained: for fixed w, the sum over all constraints in the set structure in (1) is a continuous function in p which is defined on a compact set. Hence, the maximum is attained. Moreover, the maximum is again a continuous function in w, and since w is also from a compact set, the minimum is attained as well. Note that the inner quantity of the expression (1) equals the expected weight of a solution $\varphi : X \to D$ of $2^{\mathcal{L}}$ that assigns a value $d \in D$ to the variable $x \in X$ with probability $p(x,d)$. Hence, $\pi(\mathcal{L})$ is equal to the maximum expected weight of a solution φ for the "worst" constraint weights w.

Also observe that $2^{\mathcal{L}} = (X,\mathcal{R})$ for a constraint language $\mathcal{L} = (D,\mathcal{U})$ has an exact solution if and only if $\pi(\mathcal{L}) = 1$. Indeed, if $2^{\mathcal{L}}$ has an exact solution $\varphi : X \to D$, then the maximum in (1) is attained for

$$p(x,d) = \begin{cases} 1 & \text{if } \varphi(x) = d\text{, and} \\ 0 & \text{otherwise,} \end{cases}$$

since the product $\prod_{k=1}^{r_i} p(x_{A_k}, d_k)$ in the definition equals one for all the constraints $R_i \in \mathcal{R}$, and thus the inner part of (1) also equals $w_0 = 1$. Hence, $\pi(\mathcal{L}) = 1$. On the other hand, if $\pi(\mathcal{L}) = 1$, then consider the weight function w that assigns $1/m$ to each constraint of $\mathcal{R}$ where $m = |\mathcal{R}|$. Since $\pi(\mathcal{L}) = 1$, there exists a function p in (1) such that the expected weight of a solution $\varphi : X \to D$ of $2^{\mathcal{L}}$ that assigns a variable $x \in X$ a value $d \in D$ with probability $p(x, d)$ is equal to w_0. It follows that $2^{\mathcal{L}}$ has an exact solution.

We are now ready to state the main theorem of this section.

Theorem 5. *Let $\mathcal{L} = (D, \mathcal{U})$ be a constraint language. The following holds:*

$$\rho_v^w(\mathcal{L}) \leq \pi(\mathcal{L}) \text{ and } \rho_c^w(\mathcal{L}) \leq \pi(\mathcal{L}) .$$

If $\mathcal{L}$ does not contain any unary constraint type, we also have

$$\rho_v(\mathcal{L}) \leq \pi(\mathcal{L}) \text{ and } \rho_c(\mathcal{L}) \leq \pi(\mathcal{L}) .$$

Proof. For a fixed integer $k \geq 1$ and a positive real $0 < \varepsilon < 1/10$, we construct a simple weighted instance $(X, \mathcal{R})$ with total weight w_0 that is both variable and constraint k-consistent, all the non-unary constraints of $\mathcal{R}$ have weight one, and every solution φ has weight at most $(\pi(\mathcal{L}) + \varepsilon)w_0$. The existence of such instances yields all the statements of the theorem.

Let N be a sufficiently large integer that we fix later. Let further $(X_0, \mathcal{R}_0)$ be the set structure $2^{\mathcal{L}}$, let $w_{\mathcal{L}}$ be the weight function that minimizes the value of (1), and let $w_{\mathcal{L},0} = 1$ be the total weight of the constraints of $2^{\mathcal{L}}$. Recall that X_0 contains $2^{|D|} - 1$ variables x_A, each corresponding to a non-empty subset $A \subseteq D$ (see Definition 1). The instance $(X, \mathcal{R})$ that we are going to construct contains $N \cdot (2^{|D|} - 1)$ variables $x_{A,k}$ for $k = 1, \ldots, N$. If $\mathcal{R}_0$ contains a constraint $R_i = (x_{A_1}, \ldots, x_{A_{r_i}})$ of type U_i with arity $r_i \geq 2$, then we include the r_i-tuple of variables $(x_{A_1,k_1}, \ldots, x_{A_{r_i},k_{r_i}})$ as a constraint of type U_i and weight one to $\mathcal{R}$ with probability $w_{\mathcal{L}}(R_i)N^{1+1/2k-r_i}$. Such constraints $(x_{A_1,k_1}, \ldots, x_{A_{r_i},k_{r_i}})$ are said to *correspond* to R_i. If $\mathcal{R}_0$ contains a constraint $R_i = (x_A)$ of type U_i, then $\mathcal{R}$ contains a constraint $R_i = (x_{A,k})$ of type U_i and weight $w_{\mathcal{L}}(R_i)N^{1/2k}$ with probability one, for $k = 1, \ldots, N$.

Let $(X, \mathcal{R})$ be the instance that we obtain in this way, and let w_0 be the total weight of the constraints of $(X, \mathcal{R})$. The instance $(X, \mathcal{R})$ is almost surely neither simple nor variable or constraint k-consistent. We show that $(X, \mathcal{R})$ does not have a solution of weight at least $(\pi(\mathcal{L}) + \varepsilon/2)w_0$ with high probability, and later show that $(X, \mathcal{R})$ can be pruned to be locally consistent with removing only a small fraction of the constraints. An analogous argument applies to unary constraint types.

We now prove that if N is sufficiently large, then

$$\mathrm{Prob}(w_0 \geq (1-\varepsilon/4)w_{\mathcal{L},0}N^{1+1/2k}) \geq 7/8\,.$$

We apply Chernoff's inequality (Proposition 2) separately to every set of constraints of $(X,\mathcal{R})$ corresponding to the same constraint $R_i \in \mathcal{R}_0$ of $2^{\mathcal{L}}$. If $r_i \geq 2$ is the arity of R_i, then there are N^{r_i} constraints that can be included to $\mathcal{R}$ and each of them is included with probability $w_{\mathcal{L}}(R_i)N^{1+1/2k-r_i}$. Hence, the probability that the total weight of such constraints is less then $(1-\varepsilon/4)w_{\mathcal{L}}(R_i)N^{1+1/2k}$ is at most

$$e^{-\frac{\varepsilon^2 w_{\mathcal{L}}(R_i)N^{1+1/2k}}{32}}\,.$$

Since there are finitely many constraints $R_i \in \mathcal{R}_0$, there is with probability at least $7/8$ no constraint $R_i \in \mathcal{R}_0$ such that the weight of constraints corresponding to R_i is less than $(1-\varepsilon/4)w_{\mathcal{L}}(R_i)N^{1+1/2k}$ (if N is sufficiently large). Since $w_{\mathcal{L},0} = \sum_{R_i \in \mathcal{R}_0} w_{\mathcal{L}}(R_i)$, the total weight w_0 of all the constraints of $(X,\mathcal{R})$ is at least $(1-\varepsilon/4)w_{\mathcal{L},0}N^{1+1/2k}$ with probability at least $7/8$.

Next, we show that with probability at least $7/8$, every solution φ of $(X,\mathcal{R})$ has weight at most $(\pi(\mathcal{L})+\varepsilon/4)w_{\mathcal{L},0}N^{1+1/2k}$. For simplicity, we first assume that $\mathcal{L}$ does not have unary constraint types. Let us fix a solution $\varphi : X \to D$ for the rest of this paragraph. We define a function $p : X_0 \times D \to [0,1]$ as follows: the value $p(x_A,d)$ is equal to the number of variables $x_{A,k}$ with $\varphi(x_{A,k}) = d$ divided by N. We infer from the construction of $(X,\mathcal{R})$ that for every constraint $R_i = (x_{A_1},\ldots,x_{A_{r_i}}) \in \mathcal{R}_0$ of type $U_i \in \mathcal{U}$ with arity $r_i \geq 2$, there are

$$N^{r_i} \sum_{(d_1,\ldots,d_{r_i})\in U_i} \prod_{k=1}^{r_i} p(x_{A_k},d_k)$$

constraints satisfied by φ that can be included to $(X,\mathcal{R})$. We will call such constraints *good*. Moreover, we additionally mark $\varepsilon\pi(\mathcal{L})N^{r_i}/8$ constraints corresponding to R_i to be good (if there are not enough additional constraints, mark as many of them as possible – the reader is welcome to check that our arguments work smoothly in this case, too). Since each good constraint is included to $(X,\mathcal{R})$ with probability $w_{\mathcal{L}}(R_i)N^{1+1/2k-r_i}$, the expected number of good constraints corresponding to R_i is

$$w_{\mathcal{L}}(R_i)N^{1+1/2k}\left(\frac{\varepsilon\pi(\mathcal{L})}{8} + \sum_{(d_1,\ldots,d_{r_i})\in U_i} \prod_{k=1}^{r_i} p(x_{A_k},d_k)\right)\,.$$

Hence, the probability that the number of good constraints corresponding to R_i is greater than

$$(1+\varepsilon/8)w_{\mathcal{L}}(R_i)N^{1+1/2k}\left(\frac{\varepsilon\pi(\mathcal{L})}{8}+\sum_{(d_1,\ldots,d_{r_i})\in U_i}\prod_{k=1}^{r_i}p(x_{A_k},d_k)\right) \qquad (2)$$

is at most

$$e^{-\frac{\varepsilon^2 w_{\mathcal{L}}(R_i)N^{1+1/2k}\left(\varepsilon\pi(\mathcal{L})/8+\sum_{(d_1,\ldots,d_{r_i})\in U_i}\prod_{k=1}^{r_i}p(x_{A_k},d_k)\right)}{192}} \le e^{-\frac{\varepsilon^3\pi(\mathcal{L})w_{\mathcal{L}}(R_i)N^{1+1/2k}}{1536}}$$

by Chernoff's inequality (Proposition 2). Since the weight function $w_{\mathcal{L}}$ is fixed, the value of $\pi(\mathcal{L})$ is constant independently from the instance, and there are only finitely many constraints in $\mathcal{R}_0$, the probability that there exists $R_i \in \mathcal{R}_0$ such that the number of good constraints corresponding to R_i is greater than the above expression is exponentially small in $N^{1+1/2k}$.

Since there are at most $|D|^{(2^{|D|}-1)\cdot N} = e^{O(N)}$ choices of φ, there exists a sufficiently large N such that the number of good constraints corresponding to each $R_i \in \mathcal{R}_0$ is with probability at least 7/8 bounded from above by (2) for every solution φ. Summation over all $R_i \in \mathcal{R}_0$ yields that the total number of satisfied constraints is at most

$$(1+\varepsilon/8)N^{1+1/2k}\sum_{R_i=(x_1,\ldots,x_{r_i})\in\mathcal{R}} w_{\mathcal{L}}(R_i)\cdot\sum_{(d_1,\ldots,d_{r_i})\in U_j}\prod_{k=1}^{r_i}p(x_{A_k},d_k)$$
$$+\,(1+\varepsilon/8)\frac{\varepsilon\pi(\mathcal{L})}{8}N^{1+1/2k}\sum_{R_i\in\mathcal{R}} w_{\mathcal{L}}(R_i)\ .$$

By the definition of $\pi(\mathcal{L})$, the number of satisfied constraints is at most

$$(1+\varepsilon/8)N^{1+1/2k}\pi(\mathcal{L})w_{\mathcal{L},0}+\frac{\varepsilon\pi(\mathcal{L})}{8}w_{\mathcal{L},0}N^{1+1/2k} =$$
$$(1+\varepsilon/4)N^{1+1/2k}\pi(\mathcal{L})w_{\mathcal{L},0}$$

with probability at least 7/8. An analogous conclusion can be obtained if the language L might contain unary constraint types.

We conclude that with probability at least 3/4, the weight of the optimum solution of $(X,\mathcal{R})$ is at most

$$(1+\varepsilon/4)N^{1+1/2k}\pi(\mathcal{L})w_{\mathcal{L},0}$$

and the total weight w_0 of the constraints of $(X, \mathcal{R})$ is at least

$$(1 - \varepsilon/4) N^{1+1/2k} w_{\mathcal{L},0} \ .$$

Our next goal is to show that, with high probability, we can remove all cycles of length at most k from the instance $(X, \mathcal{R})$ without significantly decreasing the total weight of all the constraints. Observe now that the instance $(X, \mathcal{R})$ is homomorphic to $(X_0, \mathcal{R}_0)$ through the homomorphism $\psi : X \to X_0$ defined as $\psi(x_{A,i}) = x_A$. Hence, the image of a cycle of $(X, \mathcal{R})$ through ψ is a cycle of $(X_0, \mathcal{R}_0)$ (of the same or shorter length). Fix a cycle $R_1, \ldots, R_\ell$ of $(X_0, \mathcal{R}_0)$ of length $\ell \leq k$. Since R_i and R_{i+1} share at least one variable, the number of possible preimages of the cycle is at most $N^{\sum_{i=1}^{\ell} r_i - \ell}$ where r_i is the arity of R_i. Consider one of these preimages $R'_1, \ldots, R'_\ell$. The probability of including R'_i to $(X, \mathcal{R})$ is

$$w_{\mathcal{L}}(R_i) N^{1+1/2k-r_i} \leq w_{\mathcal{L},0} N^{1+1/2k-r_i} \ .$$

Hence, the expected number of cycles of $(X, \mathcal{R})$ that are preimages of the cycle $R_1, \ldots, R_\ell$ is at most

$$N^{\sum_{i=1}^{\ell} r_i - \ell} \cdot \prod_{i=1}^{\ell} w_{\mathcal{L},0} N^{1+1/2k-r_i} = w_{\mathcal{L},0}^{\ell} N^{\ell/2k} \leq w_{\mathcal{L},0}^{\ell} N^{1/2} \ .$$

If M is the number of constraints of $\mathcal{R}_0$, then the number of cycles of $(X_0, \mathcal{R}_0)$ of length at most k does not exceed $(M+1)^k$ and the expected number of cycles of $(X, \mathcal{R})$ is thus at most

$$(M+1)^k w_{\mathcal{L},0}^{k} N^{1/2} \ .$$

Hence, Markov's inequality (Proposition 1) implies that the total weight of constraints of $(X, \mathcal{R})$ contained in a cycle of length at most k is at most (if N is sufficiently large)

$$2k(M+1)^k w_{\mathcal{L},0}^{k+1} N^{1/2} \leq \frac{\varepsilon}{4} w_{\mathcal{L},0} N^{1+1/2k}$$

with probability at least 1/2. The inequality holds, since for fixed k all terms in the expression (including ϵ) except for N are constant.

Remove now all the constraints of $(X, \mathcal{R})$ that are contained in a cycle of length at most k. Let $(X', \mathcal{R}')$ be the resulting instance and let w'_0 be its total weight. We infer from our previous claims that with probability at least 1/4, the weight of an optimum solution of $(X', \mathcal{R}')$ does not exceed

$$(1 + \varepsilon/4)N^{1+1/2k}\pi(\mathcal{L})w_{\mathcal{L},0}$$

and the total weight w_0' of the constraints of $(X', \mathcal{R}')$ is at least

$$(1 - \varepsilon/2)N^{1+1/2k}w_{\mathcal{L},0} .$$

Hence, with probability at least 1/4, the weight of an optimum solution of $(X', \mathcal{R}')$ is at most

$$\frac{1 + \varepsilon/4}{1 - \varepsilon/2}\pi(\mathcal{L})w_0' \leq (1 + \varepsilon)\pi(\mathcal{L})w_0' .$$

This inequality holds since we have assumed that $\varepsilon < 1/10$. Because the instance $(X', \mathcal{R}')$ does not contain a cycle of length at most k and is homomorphic to $2^{\mathcal{L}}$, it is vertex and constraint k-consistent by Lemma 4. □

4 Algorithmic Results

In this section, we design our linear-time algorithm for a fixed constraint language $\mathcal{L}$, and prove the lower bounds on the limits.

Theorem 6. *Let $\mathcal{L} = (D, \mathcal{U})$ be a fixed constraint language. There exists an algorithm that for an input instance $(X, \mathcal{R})$ with total weight w_0 and a given real number $\varepsilon > 0$,*

- *either constructs a solution φ of weight $(\pi(\mathcal{L}) - \varepsilon)w_0$, or*
- *finds a set of at most $f(\varepsilon)$ constraints of $\mathcal{R}$ that cannot be simultaneously satisfied*

where $f(\varepsilon)$ is a function that only depends on ε. The running time of the algorithm is linear in $|X| + |\mathcal{R}|$ and polynomial in $1/\varepsilon$ (for a fixed constraint language $\mathcal{L}$).

Proof. Let us briefly describe the main steps of the algorithm and we then focus on each step separately. The parameter r used in the description of the algorithm is the maximum arity of a constraint of $\mathcal{R}$.

- First, we construct sets $\mathcal{R}_i \subseteq \mathcal{R}$, $i = 1, \ldots, 2|D|r/\varepsilon$, such that the instance $(X, \mathcal{R} \setminus \mathcal{R}_i)$ is homomorphic to $2^{\mathcal{L}}$ for every i. If the algorithm fails to construct the sets, then it exhibits a set of at most $f(\varepsilon)$ inconsistent constraints.

- Next, we remove the set $\mathcal{R}_i$ with the least weight from the instance. We show that this decreases the total weight of the instance by at most $\varepsilon w_0/2$.

- We compute a function $p : X \times D \to [0, 1]$ such that if $x \in X$ is assigned a value $d \in D$ with probability $p(x, d)$, then the expected weight of satisfied constraints is at least $(\pi(\mathcal{L}) - \varepsilon)w_0$.

- Finally, using standard derandomization techniques, we find a solution $\varphi : X \to D$ of weight at least $(\pi(\mathcal{L}) - \varepsilon)w_0$.

Here and hereafter, we assume for simplicity that ε is the inverse of an integer.

Let us start with the first step. In addition to the constraint sets $\mathcal{R}_i$, we construct functions $\psi_i : X \to 2^D$ such that the mapping that assigns $x \in X$ the variable $x_{\psi_i(x)}$ of $2^{\mathcal{L}}$ is a homomorphism from $(X, \mathcal{R} \setminus \mathcal{R}_i)$ to $2^{\mathcal{L}}$. Initially, set $\psi_1(x) = D$ for every $x \in X$. Once ψ_i has been constructed, set $\psi_{i+1} = \psi_i$ and add the constraints $(x_1, \ldots, x_k) \in \mathcal{R}$ of type U_j to $\mathcal{R}_i$ where for some $d_{j'} \in \psi_i(x_{j'})$, there is no choice of $d_1, \ldots, d_k$ with $(d_1, \ldots, d_k) \in U_j$ and $d_{j''} \in \psi_i(x_{j''})$ for $j'' \neq j$. If there are such elements $d_{j'} \in \psi_i(x_{j'})$, the value $d_{j'}$ is also removed from the set $\psi_{i+1}(x_j)$. It is straightforward to verify that the mapping assigning $x \in X$ the variable $x_{\psi_i(x)}$ of $2^{\mathcal{L}}$ is a homomorphism from $(X, \mathcal{R} \setminus \mathcal{R}_i)$ to $2^{\mathcal{L}}$ unless $\psi_i(x) = \emptyset$ for some x.

In order to simplify our further exposition let us also define a function $\Psi_i : X \to 2^{\mathcal{R}}$, $i = 0, \ldots, 2|D|r/\varepsilon$. With this function we want to represent for each variable x a set of constraints that implies that in any exact solution the value for x is from $\psi_i(x)$. Initially, $\Psi_0 = \emptyset$. Whenever a value $d_{j'}$ is removed from $\psi_i(x_j)$ because the assignment $x_j = d_{j'}$ cannot be extended to a constraint R_j at some stage of the algorithm, $\Psi_i(x_j)$ is defined to be the union of $\{R_j\}$ and the set $\Psi_{i-1}(x_{j''})$ for all the variables $x_{j''}$ contained in R_j. If the value $d_{j'}$ is removed from $\psi_i(x_j)$ because of several constraints, the union is taken only for one such constraint. If several variables are removed from $\psi_i(x_j)$ at the same time, the union is taken for a single (arbitrarily chosen) constraint for each removed variable. If there is no value removed from $\psi_i(x_j)$, $\Psi_i(x_j)$ is equal to $\Psi_{i-1}(x_j)$.

We show that the constraints contained in $\Psi_{i-1}(x_j)$ force the value of a variable x_j to be one of the values contained in $\psi_i(x_j)$, i.e., any exact solution of the instance $(X, \Psi_{i-1}(x_j))$ assigns x_j a value contained in $\psi_i(x_j)$. The proof is by induction on i. If $i = 1$, there is nothing to prove since $\psi_1(x_j) = D$. For $i > 1$, if $\psi_i(x_j) = \psi_{i-1}(x_j)$, then $\Psi_{i-1}(x_j) = \Psi_{i-2}(x_j)$ and the claim follows by induction. If $\psi_i(x_j) \subset \psi_{i-1}(x_j)$, then there is a value $d_j \in \psi_{i-1}(x_j)$ that was removed from it because of a constraint $R_{j'} \in \mathcal{R}$. The constraints contained in the sets $\Psi_{i-2}(x_{j''})$ for variables contained in $R_{j'}$ force the values of these variables to be in $\psi_{i-1}(x_{j''})$, however, this is incompatible with assigning the value d_j to x_j. Hence, the union of $\Psi_{i-2}(x_{j''})$ and $\{R_{j'}\}$ force the variable $x_{j''}$ not to be assigned the value d_j. We conclude that the constraints contained in $\Psi_{i-1}(x_j)$ force the value of x_j to be one of the elements contained in $\psi_i(x_j)$.

If there exist i and x such that $\psi_i(x) = \emptyset$, then the constraints contained in $\Psi_{i-1}(x)$ restrict the values of x to those of $\psi_i(x) = \emptyset$, i.e., the subinstance $(X, \Psi_{i-1}(x))$ has no exact solution. The algorithm returns the set $\Psi_{i-1}(x)$ of inconsistent constraints—next, we bound its size.

We prove that a set $\Psi_i(x)$ contains at most $(|D|r+1)^i$ constraints by induction on i. The bound clearly holds for $i = 0$ since the sets $\Psi_0(x)$ are empty. Each set $\Psi_i(x)$ is a union of at most $|D|r$ different sets $\Psi_{i-1}(x')$ with at most $|D|$ additional constraints, i.e., its size does not exceed

$$|\Psi_i(x)| \leq |D| + |D|r(|D|r+1)^{i-1} \leq (|D|r+1)^i \ .$$

Hence, the size of the set of inconsistent constraints returned by the algorithm does not exceed $(|D|r+1)^{2|D|r/\varepsilon}$, which is a function that only depends on ε (recall that the constraint language $\mathcal{L}$ is fixed). Let us remark that the bound on the size of the sets of inconsistent constraints can be decreased by a finer analysis. In the actual implementation of this step of the algorithm, we do not compute the sets $\Psi_i(x)$ explicitly, however, for each removal of a value $d \in D$ from $\psi_i(x)$, we mark which of the input constraints caused the removal of d and compute the set $\Psi_i(x)$ (in linear time in the input size) only when it is supposed to be returned.

Suppose now that the algorithm does not output a set of contradictory constraints, i.e., we assume that $\psi_i(x) \neq \emptyset$ for all $i = 1, \ldots, 2|D|r/\varepsilon$ and $x \in X$. Observe that each constraint $R_j \in \mathcal{R}$ is contained in at most $|D|r$ sets $\mathcal{R}_i$: each time a constraint R_j is included into a set $\mathcal{R}_i$, the cardinality of the set $\psi_i(x)$ of one of the variables x contained in R_j is decreased by one. Since R_j contains at most r variables and the sets $\psi_1(x)$ contain $|D|$ elements each, R_j is included in at most $|D|r$ sets $\mathcal{R}_j$. Hence, the total sum of weights of constraints contained in all the sets $\mathcal{R}_i$ is at most $|D|rw_0$. Since there are $2|D|r/\varepsilon$ sets $\mathcal{R}_i$, there exists a set $\mathcal{R}_{i_0}$ such that the sum of the weights of the constraints contained in $\mathcal{R}_{i_0}$ is at most $\varepsilon w_0/2$. Fix such an index i_0 for the rest of the proof.

Recall that the instance $(X, \mathcal{R} \setminus \mathcal{R}_{i_0})$ is homomorphic to $2^{\mathcal{L}}$ and the homomorphism maps a variable $x \in X$ to x_A for $A = \varphi_{i_0}(x)$. Next, we define values $p(A, d)$ for non-empty subsets $A \subseteq D$ and $d \in D$ such that $\sum_{d \in D} p(A, d) = 1$ for every A. Let $E(p)$ be the expected weight of the satisfied constraints if each variable $x \in X$ is assigned randomly and independently a value $d \in D$ with probability $p(\varphi_{i_0}(x), d)$. Let E_0 be the maximum value of E under the constraints that $0 \leq p(A, d) \leq 1$ and $\sum_{d \in D} p(A, d) = 1$. Since the number of variables on which the function $E(p(A, d))$ depends is finite, it is possible to find, in time polynomial in $1/\varepsilon$, a function p_0 such that $E(p_0)$ is at least $(1 - \varepsilon/2)E_0$. By the definition of $\pi(\mathcal{L})$, we have that $E_0 \geq \pi(\mathcal{L})w_0'$ where w_0' is the total weight of constraints of $(X, \mathcal{R} \setminus \mathcal{R}_{i_0})$. Hence, the value of $E(p_0)$ is at least

$$(1 - \varepsilon/2)\pi(\mathcal{L})w_0' \geq (1 - \varepsilon/2)^2 \pi(\mathcal{L})w_0 \geq (\pi(\mathcal{L}) - \varepsilon)w_0 \ .$$

In particular, if the value of each variable $x \in X$ is set to be $d \in D$ with probability $p_0(\varphi_{i_0}(x), d)$, then the expected weight of satisfied constraints is at least $(\pi(\mathcal{L}) - \varepsilon)w_0$.

The final stage of the algorithm (once the function p_0 has been constructed) lies in derandomizing the choices of values of the variables $x \in X$. A standard technique of derandomization using conditional expectations can be applied in this scenario. We are not going to describe the details of the entire derandomization process and refer to standard literature on the subject, e.g., [27]. Let us just remark that the derandomization can be done in time linear in the size of the input. □

There is an immediate corollary of Theorems 5 and 6.

Corollary 7. *Let $\mathcal{L} = (D, \mathcal{U})$ be a constraint language. Then the following equalities hold.*

$$\rho_v^w(\mathcal{L}) = \rho_c^w(\mathcal{L}) = \pi(\mathcal{L})$$

If $\mathcal{L}$ does not contain unary constraint types, then

$$\rho_v(\mathcal{L}) = \rho_c(\mathcal{L}) = \pi(\mathcal{L}) \ .$$

5 Unweighted and Weighted Instances

We have seen that $\rho_v^w(\mathcal{L}) = \rho_v(\mathcal{L})$ and $\rho_c^w(\mathcal{L}) = \rho_c(\mathcal{L})$ if the constraint language $\mathcal{L}$ does not contain unary constraint types. It turns out that this is not just a coincidence, but the following stronger statement holds.

Theorem 8. *Let $\mathcal{L} = (D,\mathcal{U})$ be a fixed constraint language without unary constraint types. For every $k \geq 1$, the following holds:*

$$\rho_{v,k}^w(\mathcal{L}) = \rho_{v,k}(\mathcal{L}) \text{ and } \rho_{c,k}^w(\mathcal{L}) = \rho_{c,k}(\mathcal{L}) \ .$$

Proof. We focus on proving that $\rho_{v,k}^w(\mathcal{L}) = \rho_{v,k}(\mathcal{L})$; the proof for the constraint k-consistent instances is analogous. Since it is obvious that $\rho_{v,k}^w(\mathcal{L}) \leq \rho_{v,k}(\mathcal{L})$, we have to show that $\rho_{v,k}^w(\mathcal{L}) \geq \rho_{v,k}(\mathcal{L})$. Fix $\varepsilon > 0$. By the definition of $\rho_{v,k}^w(\mathcal{L})$, there exists a weighted variable k-consistent instance $(X_0, \mathcal{R}_0)$ with the weight function w that does not have solution of weight greater than $(\rho_{v,k}^w + \varepsilon/2)w_0$ where w_0 is the total weight of the constraints of $(X_0, \mathcal{R}_0)$. Our goal is to construct an unweighted instance $(X, \mathcal{R})$ with m constraints that does not have a solution satisfying more than $(\rho_{v,k}^w - \varepsilon)m$ of its constraints. Since the construction is similar to the one presented in the proof of Theorem 5, we decided to provide all the details only where the two proofs differ, and sketch the arguments where they are analogous.

Let $X_0 = \{x_1, \ldots, x_n\}$ and let N be a sufficiently large integer. The instance $(X, \mathcal{R})$ contains $n \cdot N$ variables $x_{i,j}$, $1 \leq i \leq n$ and $1 \leq j \leq N$. For every constraint $R_k = (x_{i_1}, \ldots, x_{i_r}) \in \mathcal{R}$ of type U_k with $r' \leq r$ distinct variables, a constraint $(x_{i_1,j_1}, \ldots, x_{i_r,j_r})$ is included to $\mathcal{R}$ with probability $w(R_k)N^{2-r'}$; we require that if a single variable, e.g., x_{i_1} appears several times in R_k, then all its occurrences are replaced by the same variable x_{i_1,j_1}.

Observe that the constructed instance $(X, \mathcal{R})$ is homomorphic to the original instance $(X_0, \mathcal{R}_0)$ (just map a variable $x_{i,j} \in X$ to $x_i \in X_0$). Hence, if $X' \subseteq X$ is a set of at most k variables of X, the subinstance $(X', \mathcal{R}')$ of $(X, \mathcal{R})$ where $\mathcal{R}'$ are those constraints of $\mathcal{R}$ that contain only the variables of X, is homomorphic to a subinstance of $(X_0, \mathcal{R}_0)$ with at most k variables. Since the subinstance of $(X_0, \mathcal{R}_0)$ has an exact solution, the subinstance $(X', \mathcal{R}')$ has also an exact solution. In particular, the instance $(X, \mathcal{R})$ is variable k-consistent.

An application of Chernoff's bound (Proposition 2) yields that with probability at least 3/4 the number of constraints of $(X, \mathcal{R})$ is at least

$(1-\varepsilon/8)w_0 \cdot N^2$. The details of the proof are analogous to the proof of Theorem 5.

Fix one out of the $|D|^{nN}$ possible solutions $\varphi : X \to D$ of $(X, \mathcal{R})$. Let $p(x_i, d)$ for $x_i \in X_0$ and $d \in D$ be the number of variables $x_{i,j}$, $1 \le j \le N$, with $\varphi(x_{i,j}) = d$ divided by N. By the choice of $(X_0, \mathcal{R}_0)$, the expected weight of constraints of $\mathcal{R}_0$ that are satisfied when each variable $x_i \in X_0$ is assigned randomly (and independently of other variables of X_0) a value $d \in D$ with probability $p(x_i, d)$ is at most the weight of an optimum solution, which is at most $(\rho^w_{v,k}(\mathcal{L}) + \varepsilon/2)w_0$. If p is the probability that a constraint $R_k \in \mathcal{R}_0$ is satisfied, there are $pN^{r'}$ constraints R corresponding to R_k that can be included to $\mathcal{R}$ and that are satisfied by φ.

Another application of Chernoff's bound yield that with probability at most $e^{-\Theta(N^2)}$, the number of constraints corresponding to R_k that are satisfied by φ and included to $\mathcal{R}$ is greater than $(1+\varepsilon/8)pw(R_k)N^2$. Hence, the total number of constraints satisfied by φ is greater than $(1+\varepsilon/8)(\rho^w_{v,k}(\mathcal{L})+\varepsilon/2)w_0N^2$ with probability at most $e^{-\Theta(N^2)}$. Since there are $|D|^{nN}$ choices of φ, the total number of constraints satisfied by any solution φ, is at most $(1+\varepsilon/8)(\rho^w_{v,k} + \varepsilon/2)w_0N^2$ with probability at most 7/8 if N is sufficiently large.

We conclude that there exists a variable k-consistent instance $(X, \mathcal{R})$ with at least $(1-\varepsilon/8)w_0N^2$ constraints and the number of constraints that can be satisfied by a solution φ is at most

$$(1+\varepsilon/8)(\rho^w_{v,k} + \varepsilon/2)w_0N^2 \le (\rho^w_{v,k} + \varepsilon)m$$

where m is the number of constraints of $(X, \mathcal{R})$. Since such an instance exists for every $\varepsilon > 0$, it follows that $\rho_{v,k} \le \rho^w_{v,k}$. $\square$

Let us comment on one difference between the proof of Theorem 5 and the proof of Theorem 8: in the former proof, we needed to prune the random instance to remove constraints that contain the same variable several times. Such constraints could spoil the acyclicity of the constructed instance and thus its consistency. However, in the latter proof, the constructed instance is always homomorphic to a locally consistent instance $(X_0, \mathcal{R}_0)$ and thus locally consistent, too. Such an argument cannot be used in the former proof, since $2^{\mathcal{L}}$ need not be locally consistent.

6 Graph Homomorphisms

In this section, we discuss how our results on general CSPs relate to the results obtained in our conference paper [1] on CSPs corresponding to graph homomorphisms. In the following, we state all the results for directed graphs. However, analogous results for undirected graphs follow directly from the results on directed graphs. As discussed in Section 2, the corresponding CSPs have a constraint language that contains a single binary constraint type. Throughout this section, $\mathcal{L}_H$ stands for the constraint language obtained from a (directed) graph H through the construction described in Section 2.

One of the main results of [1] is that $\rho_v(\mathcal{L}_H) = \rho_c(\mathcal{L}_H) = 1$ if and only if the graph H has tree duality. A graph H has *tree duality* iff a graph G is homomorphic to H when every directed tree homomorphic to G is also homomorphic to H. Examples of graphs with tree duality include orientations of paths or acyclic tournaments. The set structure 2^H of a graph H is again a graph, and is called the *set graph* of H. An equivalent characterization of tree duality asserts that a directed graph H has tree duality if and only if its set graph 2^H is homomorphic to H [4,11].

The following is then an immediate consequence of Corollary 7.

Theorem 9. *Let H be a directed graph. The equalities*

$$\rho_v(\mathcal{L}_H) = 1 \text{ and } \rho_c(\mathcal{L}_H) = 1$$

hold if and only if H has tree duality.

The other types of CSP problems addressed in [1] were those corresponding to graphs H with a directed cycle. If W is the vertex set of a directed cycle of H, the instance $2^{\mathcal{L}_H}$ contains a constraint (x_W, x_W) (in graph theory notation, we say that the set graph 2^H contains a loop at the vertex x_W). Hence, it is possible to consider a weight function w that assigns weight one to (x_W, x_W) and zero weight to the remaining constraints of $2^{\mathcal{L}_H}$. For such a weight function, the expression that is maximized in (1) is equal to the *fractional relative density* of H as defined in [1]:

$$\delta'_{\mathrm{rel}}(H) = \max_{p:V(H)\to[0,1]} \sum_{uv\in E(H)} p(u)\cdot p(v) \tag{3}$$

where the maximum is taken over all functions $p : V(H) \to [0,1]$ such that the sum of $p(v)$ is equal to one. Let p_0 be the function for which the maximum is attained in (3). Since the inner part of (1) is equal to $\delta'_{\rm rel}(H)$ for a particular choice of w, $\pi(\mathcal{L}_H) \le \delta'_{\rm rel}(H)$. On the other hand, if $p(x,v) = p_0(v)$ for every variable x of $2^{\mathcal{L}_H}$, the most inner sum of (1) is equal to $\delta'_{\rm rel}(H)$ for every constraint of $2^{\mathcal{L}}$ and thus the entire expression is equal to $w_0\delta'_{\rm rel}(H)$. We conclude that $\pi(\mathcal{L}_H) = \delta'_{\rm rel}(H)$.

The following result from [1] now follows from Corollary 7:

Theorem 10. *If H is a directed graph that contains at least one directed cycle, then*

$$\rho_v(\mathcal{L}_H) = \rho_c(\mathcal{L}_H) = \delta'_{\rm rel}(H) \ .$$

If the constraint language $\mathcal{L}_H$ consists of a single symmetric binary relation, i.e., the graph H contains an arc uv for every arc vu and vice versa, then the following holds (note that such languages correspond to undirected graphs).

Corollary 11. *If H is a symmetric directed graph, then*

$$\rho_v(\mathcal{L}_H) = \rho_c(\mathcal{L}_H) = 1$$

if H contains a loop, and otherwise

$$\rho_v(\mathcal{L}_H) = \rho_c(\mathcal{L}_H) = 1/\omega$$

where ω is the order of the largest clique of H.

7 Concluding Remarks

It is natural to ask whether Expression (1) in Definition 1 can be simplified. It does not seem to be the case. The maximization over all functions p, the innermost sum and the product correspond to the maximization, the sum and the product in Expression (3) of relative density, respectively. In [1], we discussed that the definition of relative density does not seem to be replaceable by a simpler concept (e.g., by the relative density as defined in [19]). It also seems that the weight function w cannot be avoided, because this function allows to distinguish the relevant part of the set structure $2^{\mathcal{L}}$.

An issue that we were not able to settle is whether the equalities $\rho_v(\mathcal{L}) = \rho_v^w(\mathcal{L})$ and $\rho_c(\mathcal{L}) = \rho_c^w(\mathcal{L})$ also hold for constraint languages that contain unary constraints. Under the assumption that no repetitions of constraints in instances are allowed (which is a reasonable assumption since otherwise we can simulate the constraint weights by including the same constraint into an instance several times), we are not aware of any partial result in this direction. We conjecture that the equalities $\rho_v^w(\mathcal{L}) = \rho_v(\mathcal{L})$ and $\rho_c^w(\mathcal{L}) = \rho_c(\mathcal{L})$ do not hold in general.

We would also like to make a few remarks on constraint languages with an infinite number of constraint types (but a finite domain). Such constraint languages include in particular CNF formulas with unbounded clause size (clauses are viewed as constraints). Theorems 5 and 6 translate smoothly to this setting. In (1), we replace the minimum with the infimum and require the weight function w to be non-zero for only a finite number of constraints of $\mathcal{R}$. Note that the maximum in (1) is always attained, since for every choice of w, the CSP instance is finite. If $\pi(\mathcal{L})$ is defined in this way, the proofs of Theorems 5 and 6 can be altered as follows. We consider the weight function $w_{\mathcal{L}}$ that is ε-close to $\pi(\mathcal{L})$ instead of that which minimizes (1), and construct a k-consistent instance with weight w'_0 and with an optimum solution of weight at most $(1+\varepsilon)2\pi(\mathcal{L})w'_0$. This yields the statements of both Theorem 5 and Theorem 6. On the other hand, we do not know whether Theorem 7 holds in this setting, too. The main obstacle is that the bound on the size of a set of inconsistent constraints involves the maximum arity of a constraint of an input instance, which may not be bounded.

Acknowledgments. The authors would like to thank Pavol Hell for his comments on tree duality and related notions.

References

[1] M. Bodirsky, D. Král': Locally Consistent Constraint Satisfaction Problems with Binary Constraints, In: Proc. 31th Workshop on Graph-Theoretic Concepts in Computer Science (WG'05), LNCS vol. 3787, Springer-Verlag Berlin (2005), 295–306.

[2] A. Bulatov, A. Krokhin, P. G. Jeavons: Classifying the complexity of constraints using finite algebras. SIAM Journal on Computing 34 (2005), 720-742.

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

[4] 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.

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

[6] Rina Dechter, Judea Pearl: Tree Clustering for Constraint Networks. Artif. Intell. 38(3) (1989) 353–366.

[7] 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.

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

[9] 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.

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

[11] 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.

[12] T. Feder, M. Vardi: The computational structure of monotone monadic SNP and constraint satisfaction: A study through Datalog and group theory, SIAM Journal on Computing 28 (1999), 57–104.

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

[14] Eugene C. Freuder: Complexity of K-Tree Structured Constraint Satisfaction Problems. AAAI 1990: 4-9.

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

[16] P. Hell, J. Nešetřil: Graphs and Homomorphisms. Oxford University Press, 2004.

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

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

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

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

[21] Ph. G. Kolaitis, M. Y. Vardi: Conjunctive-query containment and constraint satisfaction. In: Proc. of PODS'98 205-213.

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

[23] D. Král', O. Pangrác: An Asymptotically Optimal Linear-Time Algorithm for Locally Consistent Constraint Satisfaction Problems, Theoretical Computer Science 348(2–3)(2005), 187–206.

[24] B. Larose, C. Loten, C. Tardif: A characterisation of first-order constraint satisfaction problems, 10 pages, accepted for the Twenty-First Annual IEEE Symposium on Logic in Computer Science (LICS), 2006.

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

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

[27] R. Motwani, P. Raghavan: Randomized Algorithms. Cambridge University Press, 1995.

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

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

[30] 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.

[31] 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.