# Approximation of Natural W[P]-complete Minimisation Problems is Hard

Kord Eickmeyer, Martin Grohe, Magdalena Grüber
Humboldt Universität Berlin
Institut für Informatik
Unter den Linden 6, 10099 Berlin, Germany
$\{$eickmeye,grohe,grueber$\}$@informatik.hu-berlin.de

## Abstract

*We prove that the weighted monotone circuit satisfiability problem has no fixed-parameter tractable approximation algorithm with constant or polylogarithmic approximation ratio unless* FPT $=$ W[P]. *Our result answers a question of Alekhnovich and Razborov [2], who proved that the weighted monotone circuit satisfiability problem has no fixed-parameter tractable* $2$*-approximation algorithm unless every problem in* W[P] *can be solved by a randomized fpt algorithm and asked whether their result can be derandomized. Alekhnovich and Razborov used their inapproximability result as a lemma for proving that resolution is not automatizable unless* W[P] *is contained in randomized* FPT. *It is an immediate consequence of our result that the complexity theoretic assumption can be weakened to* W[P] $\neq$ FPT.

*The decision version of the monotone circuit satisfiability problem is known to be complete for the class* W[P]. *By reducing them to the monotone circuit satisfiability problem with suitable approximation preserving reductions, we prove similar inapproximability results for all other natural minimisation problems known to be* W[P]*-complete.*

## 1 Introduction

In [2], Alekhnovich and Razborov proved that resolution is not automatizable unless every problem in the parameterized complexity class W[P] can be solved by a randomized fixed-parameter tractable algorithm. Alekhnovich and Razborov asked whether their result can be derandomized; it is our main result that indeed it can be.

### Background from parameterized complexity theory

Let us start by explaining the complexity theoretic assumption in Alekhnovich and Razborov's result. A *parameterization* of a problem is a polynomial time computable function that assigns some *parameter*, usually a positive integer, to each problem instance. An *fpt algorithm* for a parameterized problem is an algorithm that solves the problem in time $f(k)\cdot n^{O(1)}$, where $n$ is the size of the input instance, $k$ the parameter, and $f$ some computable function. FPT denotes the class of all parameterized decision problems that can be solved by an fpt algorithm. The randomized fpt algorithms occurring in Alekhnovich and Razborov's result have a one-sided error with false positives. Let us denote the class of all parameterized problems with such an algorithm by co-RFPT. The complexity class W[P] consists of all parameterized problems that can be solved by a nondeterministic fpt algorithm that uses at most $g(k) \cdot \log n$ nondeterministic bits, where $k, n$ are as above and $g$ is a computable function. (This is not the original definition of W[P], but a characterisation from [6].) W[P] may be viewed as an analogue of NP in the world of parameterized complexity theory, but it is much less robust than NP in the sense that many natural problems in W[P] are neither in FPT nor W[P]-complete. Instead, they are complete for different levels of the so-called W-hierarchy of parameterized complexity classes between FPT and W[P]. Here, we are interested in W[P]-complete problems, the best-known of which is the so called *weighted circuit satisfiability problem* [1, 8]: Decide whether a Boolean circuit has a satisfying assignment of Hamming weight at most $k$. Hence instances of this problem are pairs $(C, k)$, where $C$ is a Boolean circuit and $k$ a positive integer, and the parameter is $k$. The version of this problem for *monotone* circuits is still W[P]-complete. Two more examples of W[P]-complete problems are the *generating set problem*, which asks whether for a binary function $f$ on a finite set $D$ there is a set $B \subseteq D$ of size at most $k$ such that the $f$-closure of $B$ is $D$, and the *linear inequality deletion problem*, which asks whether a set of linear inequalities over the rationals can be made solvable by deleting at most $k$ inequalities from the set. Precise definitions and further examples of W[P]-complete problems can be found in Section 4.

### Parameterized approximability

A key technical step in Alekhnovich and Razborov's proof of the non-automatizability of resolution is a parameterized inapproximability result for the weighted monotone circuit satisfiability problem: They prove that the problem, viewed as a minimisation problem, has no fpt 2-approximation algorithm unless W[P] $\subseteq$ co-RFPT. This means that there is no fpt algorithm that, given a Boolean circuit $C$ and a positive integer $k$ such that $C$ has a satisfying assignment of weight $k$, computes a satisfying assignment for $C$ of Hamming weight at most $2k$. In general, an *fpt approximation algorithm with approximation ratio* $\rho$ for a minimisation problem is an fpt algorithm that, given an instance $x$ of the problem and a positive integer $k$, computes a solution of cost at most $k\cdot\rho(k)$ if instance $x$ has a solution of cost at most $k$. If $x$ has no solution of cost at most $k$, then the output of the algorithm is arbitrary. A similar definition can be given for maximisation problems. Alekhnovich and Razborov's inapproximability result for the weighted monotone circuit satisfiability problem is the first nontrivial parameterized inapproximability result. It was actually proved before the notion of fpt approximability was introduced (independently in [4, 7, 9]). Other known parameterized inapproximability results are either for *nonmonotone* problems such as weighted satisfiability problems for various classes of (nonmonotone) Boolean circuits in [7] or for the independent dominating set problem in [9], or are based on very strong and nonstandard complexity theoretic assumptions [4]. Here we call a minimisation problem *monotone* if the solutions are subsets of some set (such as the set of input gates of the Boolean circuit for the weighted satisfiability problem), and supersets of solutions are also solutions of at least the same cost. For example, the weighted satisfiability problem for monotone circuits and the dominating set problem are monotone, whereas the weighted satisfiability problem for arbitrary circuits and the independent dominating set problem are not. The inapproximability results for nonmonotone problems all use nonmonotonicity in a crucial way by creating instances whose solution space has large gaps. To the best of our knowledge, Alekhnovich and Razborov's result is the only known parameterized inapproximability result for a natural monotone problem. However, Alekhnovich and Razborov's result is based on the slightly nonstandard complexity theoretic assumption W[P] $\not\subseteq$ co-RFPT.

### Our results

We derandomize Alekhnovich and Razborov's inapproximability result for the weighted monotone circuit satisfiability problem and at the same time strengthen it: We prove that, unless W[P] $=$ FPT, the problem has no fpt approximation algorithm with approximation ratio

$$\rho(k) = \exp\left(\log^{\gamma} k\right) \text{ for any } \gamma < \frac{\log 2}{\log 6} \approx 0.387,$$

and hence in particular no fpt approximation algorithm with constant or polylogarithmic approximation ratio. A corollary of our result is that resolution is not automatizable unless W[P] $=$ FPT. Our proof is based on a similar gap-amplification construction as Alekhnovich and Razborov's. However, we do not take the derandomization strategy suggested by Alekhnovich and Razborov in the conclusions of their paper; it would require certain very strong expanders for which still no explicit constructions are known. In our construction, we only need expanders that have very good expansion properties on small sets of vertices. Here "small" means "only depending on the parameter $k$". We give an explicit construction for such expanders based on simple linear algebra over finite fields. The existence of such expanders may be of independent interest and useful in other context as well.

In the second part of the paper, we prove similar inapproximability results as for the weighted monotone circuit satisfiability problem for all other known natural W[P]-complete minimisation problems (at least all such problems known to us; this includes the list of all W[P]-complete minimisation problems appearing in Downey and Fellows' monograph [8]). These results are obtained by suitable approximation preserving reductions to the weighted monotone circuit satisfiability problem. We remark that it is not hard to construct artificial W[P]-complete minimisation problems that are easy to approximate. This follows from a general result from [7].

## 2 Preliminaries

In this paper $\mathbb{N}$ denotes the natural numbers (positive integers), $\mathbb{R}$ the real numbers, and $\mathbb{R}_{>1}$ the real numbers greater than 1.

We think of *(Boolean) circuits* as being directed acyclic graphs, where each node of in-degree $> 1$ is labelled as and-node or as or-node, each node of in-degree $1$ is labelled as negation node and all nodes with in-degree $0$ are input nodes. Furthermore one node with out-degree $0$ is labelled as output node. The *size of a circuit* $C$ is the total number of nodes and edges and is denoted by $|C|$. Given an assignment $a \in \{0,1\}^n$ for a circuit $C$ with $n$ input nodes, we say that $a$ *satisfies* $C$ if the value computed by $C$ on input $a$ is 1. The *(Hamming) weight* of an assignment $a$ is the number of 1-entries of $a$ and if there is an assignment $a$ of weight $k$ that satisfies $C$ we say that $C$ is *$k$-satisfiable*. A Boolean circuit is *monotone* if it does not contain any negation nodes. CIRC denotes the class of all Boolean circuits and CIRC$^+$ the class of all monotone Boolean circuits. The central problem studied in this paper is the following minimisation problem:

| MIN-WSAT(CIRC$^+$) | |
|---|---|
| *Input:* | A monotone circuit $C$ with $n$ input nodes. |
| *Solutions:* | All satisfying assignments $a \in \{0,1\}^n$. |
| *Cost:* | The weight of a satisfying assignment $a$. |
| *Goal:* | min. |

For a given circuit $C$ we denote the minimum weight of a satisfying assignment for $C$ by $\min(C)$.

Parameterized approximability is a relaxed notion of classical approximability. Intuitively, an fpt approximation algorithm is an algorithm whose running time is fpt for the parameter "cost of the solution" and whose approximation ratio only depends on the parameter and not on the size of the input. Hence every polynomial time approximation algorithm with constant approximation ratio is an fpt approximation algorithm, but an approximation algorithm with approximation ratio $\log n$, where $n$ denotes the input size, is not. We will only give the definitions related to fpt approximability for minimisation problems, but it is straightforward to adapt them to maximisation problems.

**Definition 1.** Let $\rho : \mathbb{N} \to \mathbb{R}_{>1}$ be a computable function. An *fpt approximation algorithm* for an NP minimisation problem $O$ (over some alphabet $\Sigma$) with approximation ratio $\rho$ is an algorithm $\mathbb{A}$ with the following properties:

1. $\mathbb{A}$ expects inputs $(x,k) \in \Sigma^* \times \mathbb{N}$. For every input $(x,k) \in \Sigma^* \times \mathbb{N}$ such that there exists a solution for $x$ of cost at most $k$, the algorithm $\mathbb{A}$ computes a solution for $x$ of cost at most $k \cdot \rho(k)$. For inputs $(x,k) \in \Sigma^* \times \mathbb{N}$ without solution of cost at most $k$, the output of $\mathbb{A}$ can be arbitrary.

2. There exists a computable function $f$ such that the running time of $\mathbb{A}$ on input $(x,k)$ is bounded by $f(k) \cdot |x|^{O(1)}$.

In our inapproximability results, we will work with a weaker notion of approximability where an algorithm is only required to compute the cost of an optimal solution rather than an actual solution; this notion was called *cost approximability* in [7]. It will be convenient to define cost approximability in terms of certain decision problems associated with the optimisation problems. Instances of the *standard decision problem* associated with a minimisation problem $O$ are pairs $(x,k)$, where $x$ is an instance of $O$ and $k$ a natural number, and the problem is to decide if $\min(x) \le k$. Taking $k$ as parameter, we obtain the *standard parameterization* of the minimisation problem $O$. We define cost approximability in terms of a gap version of the standard decision problem:

**Definition 2.** Let $O$ be an NP minimisation problem over the alphabet $\Sigma$ and let $\rho : \mathbb{N} \to \mathbb{R}_{>1}$ be a computable function. Then a decision algorithm $\mathbb{A}$ is an *fpt cost approximation algorithm* for $O$ with *approximation ratio* $\rho$ if it is an fpt algorithm satisfying the following conditions for all inputs $(x,k) \in \Sigma^* \times \mathbb{N}$ such that there exists at least one solution for $x$:

1. If $k \ge \min(x) \cdot \rho(\min(x))$, then $\mathbb{A}$ accepts $(x,k)$.

2. If $k < \min(x)$, then $\mathbb{A}$ rejects $(x,k)$.

(For instances $x$ with no valid solution, the algorithm $\mathbb{A}$ can be assumed to reject $(x,k)$ for all $k \in \mathbb{N}$.)

It is easy to see that fpt approximability implies fpt cost approximability with the same ratio (cf. [7]).

## 3 Inapproximability of the weighted monotone circuit satisfiability problem

We start by constructing certain unbalanced bipartite graphs with good expansion properties, which we will need later.

**Lemma 3.** *For any $\epsilon > 0$, integer $t \ge 2$, and $K_{\max} \in \mathbb{N}$, we set*

$$d := \left\lceil \frac{K_{\max} \cdot (t-1)}{2\epsilon} \right\rceil . \tag{1}$$

*Then for any prime power $q > d$ we can explicitly construct a bipartite graph $G = (V,E)$ with left degree $d$, left vertex set $L$ of size $q^t$, and right vertex set $R$ of size $dq$, such that*

$$\forall W \subseteq L, |W| \le K_{\max} : \quad |\Gamma(W)| \ge (1-\epsilon)d\,|W| ,$$

*where $\Gamma(W) = \{r \in R \mid E(v,r) \text{ for some } v \in W\}$ is the set of neighbours of $W$.*

*Thus, $G$ is a $(K_{\max}, \epsilon)$-lossless expander: For small ($\le K_{\max}$ elements) sets of left vertices we have only very few collisions, resulting in nearly lossless expansion. Moreover, the left degree of our expander depends only on $K_{\max}$ and $\epsilon$, not on $q$.*

*Proof.* We give an explicit construction. Suppose that $q > d$ is a prime power. For the vertex sets $L$ and $R$, we choose

$$L := \mathbb{F}_q^t \qquad \text{and} \qquad R := [d] \times \mathbb{F}_q ,$$

where $[d] = \{1, \ldots, d\}$ and $\mathbb{F}_q$ is the Galois field with $q$ elements. We pick vectors $\mathbf{u}_1, \ldots, \mathbf{u}_d \in \mathbb{F}_q^t$ such that any $t$ of them are linearly independent, for example

$$\mathbf{u}_i := \left(1, x_i, \ldots, x_i^{\,t-1}\right)^{\mathrm{T}} ,$$

where $x_1, \ldots, x_d$ are pairwise different elements of $\mathbb{F}_q$. We connect the vertex $\mathbf{v} \in L$ to the vertices

$$\left(1, \mathbf{v}^{\mathrm{T}}\mathbf{u}_1\right), \quad \ldots, \quad \left(d, \mathbf{v}^{\mathrm{T}}\mathbf{u}_d\right).$$

Notice that no two vertices of $L$ can have more than $t-1$ neighbours in common, because any $t$ of the vectors $u_1, \ldots, u_d$ form a basis of $\mathbb{F}_q^t$. Therefore, if $W \subseteq L$ has at most $K_{\max}$ elements, then

$$\begin{aligned} |\Gamma(W)| &= \left| \bigcup_{\mathbf{v} \in W} \Gamma(\mathbf{v}) \right| \\ &\geq \sum_{\mathbf{v} \in W} |\Gamma(\mathbf{v})| - \sum_{\mathbf{v} \neq \mathbf{w} \in W} |\Gamma(\mathbf{v}) \cap \Gamma(\mathbf{w})| \\ &\geq d\,|W| - \binom{|W|}{2}(t-1) \\ &\geq d\,|W| \left(1 - \frac{(t-1)\,|W|^2}{2d\,|W|}\right) \\ &\geq d\,|W|\,(1-\epsilon) \end{aligned}$$

by our choice of $d$. □

**Remark 4.** In Lemma 3 it is crucial that the left degree $d$ does not depend on $q$. This is because we want the position of the gap in the circuit $\pi(C,k,\delta)$ which we will construct in Lemma 5 to depend only on $k$ and $\delta$, but not on $n$.

In [5], Capalbo et al. gave a construction of lossless expanders, but there the left degree grows polylogarithmically in $L/R$, the quotient of the number of left and right vertices, which is $O(q)$ in our case. The benefit of their expanders is that $K_{\max} = \epsilon L$ grows linear with the number of left vertices. The expanders constructed by Guruswami et al. [11] are even more unbalanced than our expanders ($R = \operatorname{polylog}(L)$), but with a degree polylogarithmic in $L$.

These expanders can also be seen as error correcting codes or as a family of $d$-element subsets of $[dq]$ such that any two subsets have small intersection. Nisan and Wigderson [14] constructed a system of $q$-element subsets of $[q^2]$, such that any two sets intersect in at most $\log q$ elements. They essentially use Reed-Solomon-Codes over $\mathbb{F}_q$, the same construction is used, for example, in [3] to devise a randomised query scheme for storing subsets. Again, the size of the sets grows with $q$, so we can not use this construction here.

**Lemma 5.** *Given $k \in \mathbb{N}$, $\delta > 1$, and a monotone circuit $C$ with $n$ inputs, we can deterministically construct a circuit $\pi(C,k,\delta)$ such that*

$$\min(C) \leq k \quad \Rightarrow \quad \min(\pi(C,k,\delta)) \leq \alpha(k,\delta) \qquad (2)$$

*and*

$$\min(C) \geq k+1 \quad \Rightarrow \quad \min(\pi(C,k,\delta)) \geq \delta\alpha(k,\delta). \quad (3)$$

*Here, $\alpha(k,\delta)$ depends only on $k$ and $\delta$, and $\pi(C,k,\delta)$ has size $g(k)n\,|C|$ for some computable function $g$ and can be computed by an fpt algorithm.*

*Proof.* We construct the circuit $\pi(C,k,\delta)$ by starting with a copy of $C$, below which we add layers of copies of $C$ as shown in Figure 1. Each layer achieves a certain gap amplification, while only increasing the number of inputs by a factor depending only on $k$. To be precise, the layers have the following properties:

(a) Layer $\ell$ is a monotone circuit with $I_\ell$ inputs and $O_\ell$ outputs, where

$$O_1 := n, \qquad d_\ell O_\ell \leq I_\ell < 2d_\ell O_\ell, \qquad O_{\ell+1} := I_\ell.$$

Here, $d_\ell$ is a constant to be specified later which depends only on $k$ and $\ell$. We will use the notation $D_\ell := d_1 \cdot d_2 \cdots d_\ell$ for the product of the first $\ell$ of these constants (with $D_0 := 1$).

(b) If $\min(C) \leq k$, then for any set $S$ of $D_{\ell-1}k^\ell$ outputs of layer $\ell$ there is an assignment of weight $D_\ell k^{\ell+1}$ to the inputs of that layer such that (at least) all the outputs in $S$ are satisfied.

(c) If, on the other hand, $\min(C) \geq k+1$, then there is no assignment of weight less than $D_\ell(k^{\ell+1} + (\ell+1)k^\ell)$ to the inputs of layer $\ell$ which satisfies $D_{\ell-1}(k^\ell + \ell k^{\ell-1})$ or more of the outputs of that layer.

(d) For fixed $k$,the size of layer $\ell$ as a circuit depends linearly on $n \cdot |C|$ (so it is quadratic in the size of $C$).

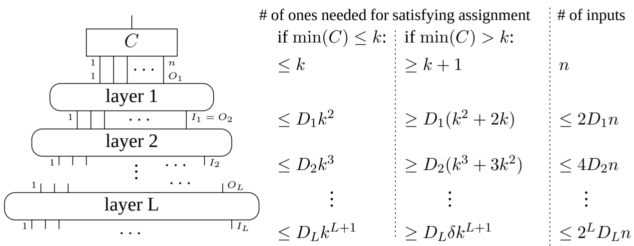


**Figure 1. The overall structure of $\pi(C,k,\delta)$**

We choose

$$L := \lceil (\delta-1)k - 1 \rceil$$

and see by descending down the layers using property (c) that if $\min(C) \geq k+1$ we need at least

$$D_L \cdot \left(k^{L+1} + (L+1)k^L\right) \geq D_L \cdot \delta \cdot k^{L+1}$$

many ones to satisfy $\pi(C,k,\delta)$, while in the case $\min(C) \leq k$ we need only $D_L k^{L+1}$ many by property (b). Thus, both (2) and (3) are satisfied, with $\alpha(k,\delta) := D_L k^{L+1}$.

It remains to describe the construction of the individual layers (cf. Figure 2). Each of the $O_\ell$ outputs of layer $\ell$ is connected to a copy of $C$. These have a total of $n \cdot O_\ell$ inputs.

We let $\tilde{O}_\ell$ be the least power of two greater than or equal to $O_\ell$, so that $O_\ell \leq \tilde{O}_\ell < 2O_\ell$. By induction, property (a) implies that $O_\ell \geq n$ for all $\ell$, so we get

$$n \cdot O_\ell \leq O_\ell^2 \leq \tilde{O}_\ell^2.$$

We use Lemma 3 with parameters

$$t := 2,$$

$$K_{\max} := K_\ell := D_{\ell-1}(k^\ell + \ell k^{\ell-1})(k+1),$$

$$\epsilon := \epsilon_\ell := \frac{\ell k^{\ell-1}}{(k^\ell + \ell k^{\ell-1})(k+1)}$$

to construct a bipartite expander with $\tilde{O}_\ell^2$ left vertices, left degree $d_\ell$ defined as in (1) and $d_\ell \tilde{O}_\ell$ right vertices. For each of the right vertices we introduce an input of layer $\ell$. We view the $n \cdot O_\ell$ inputs of the copies of $C$ as (a subset of the) left vertices of this expander and connect each of them to the conjunction of $d_\ell$ of the inputs of layer $\ell$.

This construction obviously satisfies properties (a) and (d). To see that (b) also holds, we assume that $\min(C) \leq k$. Then there exists a satisfying assignment of weight $\leq k$ for $C$, so if we are given a subset $S$ of $D_{\ell-1}k^\ell$ outputs of layer $\ell$ it suffices to satisfy $k \cdot D_{\ell-1}k^\ell$ many of the and-gates in that layer. But these are connected to at most $d_\ell$ inputs each, so there is an assignment of weight $D_\ell k^{\ell+1}$ to the inputs of layer $\ell$ such that all outputs in $S$ are satisfied.

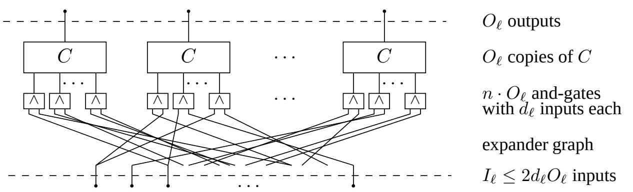


**Figure 2. Layer $\ell$ of $\pi(C, k, \delta)$**

If, on the other hand, there is no assignment of weight $\leq k$ which satisfies $C$, then for $D_{\ell-1}(k^\ell + \ell k^{\ell-1})$ of the output gates of layer $\ell$ to be satisfied, at least

$$(k+1)D_{\ell-1}(k^\ell + \ell k^{\ell-1}) = K_\ell$$

many of the and-gates in that layer must be satisfied. By the expansion property of our wiring, any set of $K_\ell$ and-gates is connected to at least

$$(1 - \epsilon_\ell)d_\ell K_\ell = D_\ell(k^{\ell+1} + (\ell+1)k^\ell)$$

many inputs of the layer, therefore no satisfying assignment of weight less than this number can exist and (c) is proved. □

**Theorem 6** (Main Theorem)**.** *There exists no fpt cost approximation algorithm for* MIN-WSAT(CIRC$^+$) *with ratio*

$$\rho(k) = \exp\left(\log^\gamma k\right), \quad \textit{where } \gamma < \frac{\log 2}{\log 6} \approx 0.387,$$

*unless* W[P] $=$ FPT.

*Proof of the Main Theorem.* We use Lemma 5 to reduce the standard parameterization $p$-WSAT(CIRC$^+$) to its approximation variant.

We first show that if there were a constant fpt cost approximation algorithm for MIN-WSAT(CIRC$^+$), this could be used to solve $p$-WSAT(CIRC$^+$) by an fpt algorithm. Say $\mathbb{A}$ is such an algorithm with constant approximation ratio $\rho(k) = c$ for all $k$. Given a circuit $C$ and a parameter $k$, we wish to decide whether or not $C$ is $k$-satisfiable. We use Lemma 5 with $\delta = c + 1$ to obtain a circuit $C'$, and run algorithm $\mathbb{A}$ on $(C', c \cdot \alpha)$, where $\alpha = \alpha(k, \delta)$ is as in the lemma (note that it can easily be computed from $\delta$ and $k$).

Now, if $\min(C) \leq k$, then $\min(C') \leq \alpha$, so $c \cdot \min(C') \leq c \cdot \alpha$ and the algorithm accepts. If, on the other hand, $\min(C) > k$, then $\min(C') \geq \delta \cdot \alpha > c \cdot \alpha$, so in this case the algorithm rejects. In summary, we have gained an fpt algorithm for $p$-WSAT(CIRC$^+$). Because this problem is W[P]-hard, it follows that W[P] $=$ FPT.

We sharpen this result, starting from an fpt cost approximation algorithm for MIN-WSAT(CIRC$^+$) with approximation ratio

$$\rho(k) = \exp\left(\log^\gamma k\right), \qquad \gamma < \frac{\log 2}{\log 6}.$$

Given a circuit $C$ and a parameter $k$, we seek to find a circuit $C''$ such that

$$\min(C'') \begin{cases} \leq \alpha & \text{if } \min(C) \leq k, \\ > \rho(\alpha) \cdot \alpha & \text{if } \min(C) > k. \end{cases}$$

The problem is that the construction of Lemma 5 does not only increase the gap, but at the same time also increases its position. Here we need a gap size that grows with the position of the gap.

We choose $\beta$ such that

$$6^\gamma < \beta < 6^{\log 2 / \log 6} = 2$$

and set

$$\delta_0 = 2^{\frac{1}{2-\beta}}.$$

Then we use Lemma 5 to obtain a circuit $C'$ with a gap at position $\alpha_0$ of relative size $\delta_0$ for some $\alpha_0$. This means that either there is a satisfying assignment of weight at most $\alpha_0$, or any satisfying assignment has weight at least $\delta_0 \cdot \alpha_0$.

The overall structure of $C''$ is similar to the construction of Lemma 5, see Figure 3. Suppose that before level $\ell$ of the construction, we have a gap of relative size $\delta_{\ell-1}$ at position $\alpha_{\ell-1}$. As before, each layer looks like in Figure 2, with $C'$ instead of $C$ and the parameters for the expander chosen as follows:

$$\epsilon_\ell := \frac{1}{2}, \qquad K_\ell := (\delta_{\ell-1} \cdot \alpha_{\ell-1})^2 \qquad \text{and} \qquad t = 2.$$

By (1), the expander therefore has left degree $d_\ell = K_\ell$. If $\delta_{\ell-1} > \rho(\alpha_{\ell-1})$, we would not need another layer, so in particular we may assume that $\delta_{\ell-1} \leq \alpha_{\ell-1}$. Layer $\ell$ moves the gap to

$$\alpha_\ell = \alpha_{\ell-1}^2 \cdot d_\ell = \alpha_{\ell-1}^2 \cdot (\delta_{\ell-1}\alpha_{\ell-1})^2 \leq \alpha_{\ell-1}^6,$$

while increasing its size to

$$\delta_\ell \geq (1-\epsilon_\ell)\delta_{\ell-1}^2 = \delta_{\ell-1}^\beta \cdot \left(\frac{1}{2}\delta_{\ell-1}^{2-\beta}\right) \geq \delta_{\ell-1}^\beta,$$

where the last inequality follows from $\delta_{\ell-1} \geq \delta_0$, which is easily seen by induction, and our choice of $\delta_0$.

We see that after layer $\ell$ the gap is at position

$$\alpha_\ell \leq \alpha_0^{6^\ell}$$

and has relative size

$$\delta_\ell \geq \delta_0^{\beta^\ell}.$$

The total number $L$ of layers must be big enough to satisfy

$$\delta_0^{\beta^L} > \rho(\alpha_0^{6^L}) = \exp\left(\beta'^L \cdot \log^\gamma \alpha_0\right),$$

where $\beta' := 6^\gamma < \beta$, and thus we need

$$\left(\frac{\beta}{\beta'}\right)^L > \frac{\log^\gamma \alpha_0}{\log \delta_0},$$

and set

$$L := \left\lceil \frac{\log\log^\gamma \alpha_0 - \log\log \delta_0}{\log \beta - \log \beta'} \right\rceil,$$

which for fixed $\gamma$ and $\beta$ depends only on $k$. The size of the resulting circuit can be estimated as in Lemma 5. $\square$

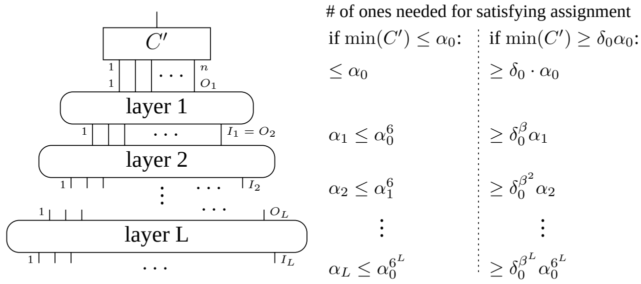


**Figure 3. Refined gap amplification.**

## 4 Further inapproximability results

In this section, we go through the list of all known W[P]-complete minimisation problems and show that all of them are hard to approximate. To deal with the monotone problems in this list, we introduce a suitable notion of gap preserving reduction and show that MIN-WSAT(CIRC$^+$) can be reduced to the problems. For the nonmonotone problems, we obtain even stronger inapproximability results by reduction from MIN-WSAT(CIRC), which was proved to be fpt inapproximable to any ratio in [7].

### 4.1 Monotone Problems

We need the definition of fpt reductions (cf. [8, 10]):

**Definition 7.** Let $Q \subseteq \Sigma^* \times \mathbb{N}$, $Q' \subseteq (\Sigma')^* \times \mathbb{N}$ be parameterized problems. A function $R : \Sigma^* \times \mathbb{N} \to \Sigma'^* \times \mathbb{N}$ is an *fpt reduction* from $Q$ to $Q'$ if

1. for all $x \in \Sigma^*$ and parameters $k \in \mathbb{N}$ it holds that $(x, k) \in Q$ if and only if $R(x, k) \in Q'$,

2. $R$ is computable by an fpt algorithm with parameter $k$,

3. there exists a computable function $g : \mathbb{N} \to \mathbb{N}$ such that for all inputs $(x, k) \in \Sigma^* \times \mathbb{N}$ with $R(x, k) = (x', k')$ it holds $k' \leq g(k)$.

Recall that the *standard parameterization* of a minimisation problem $O$ is the problem of deciding for instances $(x, k)$ whether $\min(x) \leq k$. Here $x$ is an instance of $O$ and $k \in \mathbb{N}$ is the parameter.

**Definition 8.** Let $\delta : \mathbb{N} \to \mathbb{R}_{>1}$ be a function and let $O$ be a minimisation problem. An instance $(x, k)$ of the standard parameterization of $O$ is a *$\delta$-gap instance* of $O$ if

$$\text{either} \quad \min(x) \leq k, \qquad \text{or} \quad \min(x) \geq k\delta(k).$$

**Definition 9.** Let $\delta, \delta'$ be functions from $\mathbb{N}$ to $\mathbb{R}_{>1}$ and let $O$ and $O'$ be minimisation problems over the alphabets $\Sigma$ and $\Sigma'$. A function $R : \Sigma^* \times \mathbb{N} \to (\Sigma')^* \times \mathbb{N}$ is a $(\delta, \delta')$-*gap-preserving reduction* from $O$ to $O'$ if

1. $R$ is an fpt reduction from the standard parameterization of $O$ to the standard parameterization of $O'$,

2. for every $\delta$-gap instance $(x, k)$ of $O$ the instance $R(x, k)$ is a $\delta'$-gap instance of $O'$.

The proof of the Main Theorem directly implies the following

**Lemma 10.** *There exists an fpt reduction that translates arbitrary instances $(C, k)$ of $p$-WSAT(CIRC$^+$) into $\delta$-gap instances $(C', \alpha(k))$ of $p$-WSAT(CIRC$^+$) with $\alpha$ being some computable function and with*

$$\delta(k) > \exp(\log^\gamma k), \qquad \textit{where } \gamma < \frac{\log 2}{\log 6} \approx 0.387.$$

The first W[P]-complete minimisation problem we consider here is the *chain reaction closure* problem:

| MIN-CHAIN-REACTION-CLOSURE | |
|---|---|
| *Input:* | A digraph $G = (V, E)$. |
| *Solutions:* | All sets $S \subseteq V$ such that the chain reaction closure of $S$ is $V$, where the chain reaction closure of $S$ is the smallest superset $S' \subseteq V$ of $S$ such that $v, v' \in S'$ and $(v, w), (v', w) \in E$ imply $w \in S'$. |
| *Cost:* | $\lvert S \rvert$. |
| *Goal:* | min. |

**Lemma 11.** *For every gap function $\delta : \mathbb{N} \to \mathbb{R}_{>1}$ there exists a $(\delta, \delta')$-gap-preserving reduction from* MIN-WSAT(CIRC$^+$) *to* MIN-CHAIN-REACTION-CLOSURE *with $\delta'(k) := \delta(\lfloor \frac{k}{2} \rfloor)$ for all $k \in \mathbb{N}$.*

*Proof.* We adapt the fpt reduction from $p$-WSAT(CIRC$^+$) to $p$-CHAIN-REACTION-CLOSURE described in [1].

Let $(C, k)$ be a $\delta$-gap instance of $p$-WSAT(CIRC$^+$). We assume that every node in $C$ has fan-in at most two, which we can always achieve with at most a linear increase in the size of the circuit.

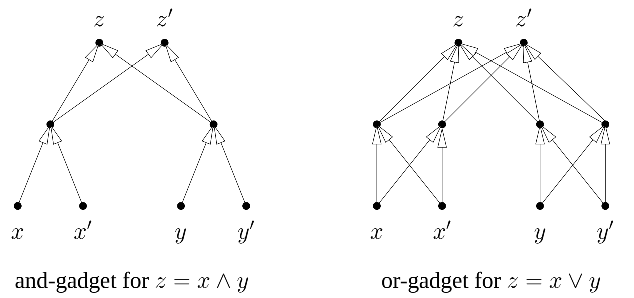


**Figure 4. The gadgets used for the** CHAIN-REACTION-CLOSURE **problem.**

The idea of the proof is as follows: We construct a digraph from the circuit by replacing each node $x$ of the circuit by two vertices $x, x'$ and replacing and-gates and or-gates by the gadgets shown in Figure 4. Note that the gadgets enforce both vertices $x, x'$ of an input pair to be chosen, as choosing just one of the two has no effect on the chain reaction closure. The resulting graph $G''$ has the following property: An assignment $a$ for $C$ is a satisfying assignment if and only if the chain reaction closure of the set $S = \{x, x' \mid a(x) = 1\}$ contains the two vertices corresponding to the output node of $C$. To make sure that the chain reaction closure of $S$ contains all vertices of the graph, in a second step of the reduction we connect the pair of vertices corresponding to the output node to all pairs of vertices corresponding to input nodes. Then once the vertices corresponding to the output node are in the chain reaction closure, all other vertices will get in. Hence the resulting digraph $G'$ has the following property: An assignment $a$ for $C$ is a satisfying assignment if and only if the chain reaction closure of the set $S = \{x, x' \mid a(x) = 1\}$ contains all vertices of $G'$. However, it could still be that there is some other, smaller set of vertices not corresponding to input nodes whose chain reaction closure contains all vertices. Actually, the chain reaction closure of the two vertices corresponding to the output node already contains all vertices. To prevent this, we take many copies of all vertices except those corresponding to input nodes. Here "many" is $(k+1)$ if we just want to have an fpt reduction, and $\lceil k\delta(k) \rceil$ if we want a $(\delta, \delta')$-gap preserving reduction. Furthermore, we have to adapt the connection between vertices corresponding to output nodes in all these copies and vertices corresponding to input nodes to assure that only the latter have significant impact in a solution for $G$. For every pair of vertices that corresponds to one of the $n$ input nodes, we add a conjunction over all vertices that correspond to output nodes in the $\lceil k\delta(k) \rceil$ copies and connect the two outputs of this conjunction with the corresponding pair. To realize this conjunction we use a binary tree of and-gadgets with $\lceil \frac{\lceil k\delta(k) \rceil}{2} \rceil$ leaves.

More formally, the resulting digraph $G$ consists of

- $n$ pairs of vertices $(v_1, v_1'), \ldots, (v_n, v_n')$ corresponding to the $n$ input nodes of $C$.

- $\lceil k\delta(k) \rceil$ identical subgraphs $G_1, \ldots, G_{\lceil k\delta(k) \rceil}$ such that each subgraph arises from taking $C$ without the input nodes and replacing every node with two new vertices, every and-gate with an and-gadget and every or-gate with an or-gadget. Each subgraph $G_i$ is connected with all the pairs $(v_j, v_j')$ for $1 \le j \le n$ in the same way as $C$ connects gates with input nodes.

- $n$ identical subgraphs $H_1, \ldots, H_n$ where each subgraph $H_j$ consists of a complete binary tree of and-gates with $\lceil \frac{\lceil k\delta(k) \rceil}{2} \rceil$ leaves such that all the pairs of vertices at the leaves are connected to all the pairs of vertices corresponding to the outputs of $G_1, \ldots, G_{\lceil k\delta(k) \rceil}$. For every $1 \le j \le n$ the pair of vertices in $H_j$ that corresponds to the root of the tree is connected with the pair $(v_j, v_j')$.

The overall structure of $G$ is sketched in Figure 5. The re-

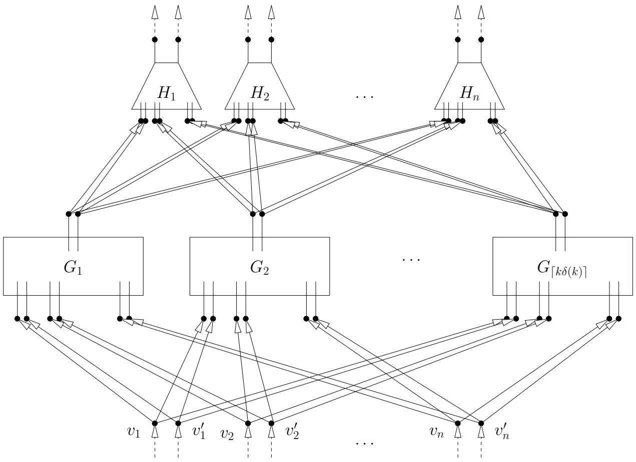


**Figure 5. The overall structure of the directed graph $G$ constructed in Lemma 11. The dashed lines indicate edges between the pair of output vertices of $H_j$ and the pair $(v_j, v_j')$ for $1 \leq j \leq n$.**

sulting digraph $G$ has the desired property that $C$ has a satisfying assignment of weight $k$ if and only if $G$ has a set of $2k$ vertices whose chain reaction closure contains all vertices (see [1] for details).

To see that the given reduction preserves gaps, we assume that $(C, k)$ is a $\delta$-gap instance of MIN-WSAT(CIRC$^+$) and that $(G, 2k)$ is no $\delta'$-gap instance of MIN-CHAIN-REACTION-CLOSURE. Then there exists a set $S \subseteq V$ of size $k'^*$, $2k < k'^* < 2k\delta'(2k) = 2k\delta(k)$ whose chain reaction closure in $G$ is $V$. Note that the graph $G$ was designed such that only pairs of vertices $(x, x')$ in $S$ can have any impact on the chain reaction closure of $S$ in $G$. Therefore we can always delete any single vertices in $S$ and assume that only pairs of vertices are chosen. As we chose the number of copies big enough, there is a part of $G$ that corresponds to a copy of $C$ and that contains no vertex pair in $S$. This implies that there have to be pairs of vertices in $S$ which correspond to input nodes of $C$ and cause all vertices corresponding to output nodes in all copies of $C$ to be in the closure. These "input nodes" in $S$ define a satisfying assignment for $C$ of weight $w$ with $k < w \leq \frac{k'^*}{2} < k\delta(k)$, contradicting the assumption that $(C, k)$ is a $\delta$-gap instance. □

**Corollary 12.** *There exists no fpt cost approximation algorithm for* MIN-CHAIN-REACTION-CLOSURE *with ratio*

$$\rho(k) = \exp\left(\log^{\tilde{\gamma}} k\right), \qquad \textit{where } \tilde{\gamma} < \frac{\log 2}{\log 6} \approx 0.387,$$

*unless* W[P] $=$ FPT.

*Proof.* We reduce $p$-WSAT(CIRC$^+$) to the approximation variant of $p$-CHAIN-REACTION-CLOSURE. Assume there exists an fpt cost approximation algorithm $\mathbb{A}$ for MIN-CHAIN-REACTION-CLOSURE with approximation ratio $\rho(k) = \exp\left(\log^{\tilde{\gamma}} k\right)$ for all $k \in \mathbb{N}$. We chose $\gamma \in (\tilde{\gamma}, \log 2/\log 6)$. Given an input $(C, k)$ for $p$-WSAT(CIRC$^+$), we first use an analogous construction as described in the proof of the Main Theorem. In fpt time (with parameter $k$) we get a circuit $C'$ of size $|C'| = f(k) \cdot |C|^{O(1)}$ for some computable function $f$, such that

> $(C', \alpha(k))$ is a $\delta$-gap instance for some $\alpha : \mathbb{N} \to \mathbb{N}$ and $\delta : \mathbb{N} \to \mathbb{R}_{>1}$,
>
> and $\rho(2\alpha(k)) < \delta(\alpha(k))$.

This last condition is satisfiable as $\tilde{\gamma} < \gamma$ implies $\log(2k)^{\tilde{\gamma}} < \log k^{\gamma}$ for big $k$.

Using the gap-preserving reduction described in Lemma 11 from MIN-WSAT(CIRC$^+$) to MIN-CHAIN-REACTION-CLOSURE on the $\delta$-gap instance $(C', \alpha(k))$, we get a $\delta'$-gap instance $(G, 2\alpha(k))$ of $p$-CHAIN-REACTION-CLOSURE with $\delta'(2\alpha(k)) = \delta(\alpha(k))$. We run $\mathbb{A}$ on $(G, \rho(2\alpha(k)) \cdot 2\alpha(k))$. If $\min(G) \leq 2\alpha(k)$, then $\rho(\min(G)) \min(G) \leq \rho(2\alpha(k))2\alpha(k)$ and $\mathbb{A}$ accepts. If, on the other hand, $\min(G) \geq \delta'(2\alpha(k))2\alpha(k) > \rho(2\alpha(k))2\alpha_L$, then $\mathbb{A}$ rejects. This way we could solve the W[P]-complete problem $p$-WSAT(CIRC$^+$) in fpt time, therefore such an algorithm $\mathbb{A}$ can not exist unless W[P] $=$ FPT. □

In the following we list further W[P]-complete problems, for which there exist gap-preserving reductions from MIN-WSAT(CIRC$^+$) or MIN-CHAIN-REACTION-CLOSURE. This implies inapproximability results for these problems analogous to Corollary 12.

| MIN-$t$-THRESHOLD-STARTING-SET | |
|---|---|
| *Input:* | A digraph $G = (V, E)$. |
| *Solutions:* | All $t$-starting sets $V' \subseteq V$ of $G$, where a $t$-starting set is a set of vertices $V' \subseteq V$ such that beginning with pebbles on the vertices of $V'$ and subsequently placing pebbles on every vertex in $V$ that has at least $t$ incoming arcs from pebbled vertices eventually every vertex of $G$ is pebbled. |
| *Cost:* | $\lvert V'\rvert$. |
| *Goal:* | min. |

**MIN-GENERATING-SET**

| | |
|---|---|
| *Input:* | A finite set $D$ and a binary function $f$ on $D$. |
| *Solutions:* | All generating sets $B \subseteq D$ of $f$, where $B$ is a generating set for $f$ if the smallest superset $B' \subseteq D$ of $B$ such that $b, b' \in B'$ implies $f(b, b') \in B'$ is $D$. |
| *Cost:* | $\lvert B\rvert$. |
| *Goal:* | min. |

**MIN-AXIOM-SET**

| | |
|---|---|
| *Input:* | A finite set $A$ and a binary relation $R$ consisting of pairs $(B, a)$ with $B \subseteq A$ and $a \in A$. |
| *Solutions:* | All subsets $B$ of $A$ such that the closure of $B$ under $R$ is $A$, where the closure of $B$ under $R$ is the smallest superset $B' \subseteq A$ of B such that $(C, a) \in R$ and $C \subseteq B'$ imply $a \in B'$. |
| *Cost:* | $\lvert B\rvert$. |
| *Goal:* | min. |

**MIN-DEGREE-3-SUBGRAPH-ANNIHILATOR**

| | |
|---|---|
| *Input:* | A graph $G = (V, E)$. |
| *Solutions:* | All sets $S \subseteq V$ of vertices such that removing $S$ from $G$ results in a remaining graph that has no subgraph of minimum degree 3. |
| *Cost:* | $\lvert S\rvert$. |
| *Goal:* | min. |

**MIN-LINEAR-INEQUALITY-DELETION**

| | |
|---|---|
| *Input:* | A system $S$ of linear inequalities over the rationals. |
| *Solutions:* | All subsets $S' \subseteq S$ of inequalities such that deleting $S'$ from $S$ results in a remaining system that is solvable. |
| *Cost:* | $\lvert S'\rvert$. |
| *Goal:* | min. |

Given a Boolean formula $\varphi$ in CNF with $n$ variables and a partial assignment $a \in \{0,1\}^{|S|}$ for a subset $S$ of the variables, an *unravelling step* is defined as extending $a$ by setting the value of literal $\lambda$ to 1 if $\lambda$ occurs in a clause where all other literals already have value 0. $\varphi$ is said to *unravel* under $a$ if, after a finite number of unravelling steps, every clause in $\varphi$ contains a literal with value 1. This unravelling procedure can be modified accordingly for an arbitrary Boolean formula $\varphi$.

**MIN-INDUCED-SAT(3CNF)**

| | |
|---|---|
| *Input:* | A 3-CNF formula $\varphi$. |
| *Solutions:* | All sets $S$ of variables in $\varphi$ such that there exists an assignment $a \in \{0,1\}^{|S|}$ to these variables that causes $\varphi$ to unravel. |
| *Cost:* | $\lvert S\rvert$. |
| *Goal:* | min. |

**MIN-INDUCED-SAT**

| | |
|---|---|
| *Input:* | A Boolean formula $\varphi$. |
| *Solutions:* | All sets $S$ of variables in $\varphi$ such that there exists an assignment $a \in \{0,1\}^{|S|}$ to these variables that causes $\varphi$ to unravel. |
| *Cost:* | $\lvert S\rvert$. |
| *Goal:* | min. |

**Lemma 13.** *1. For every gap function $\delta : \mathbb{N} \to \mathbb{R}_{>1}$ there exists a $(\delta, \delta')$-gap-preserving reduction from* MIN-WSAT(CIRC$^+$) *to*

*(i)* MIN-$t$-THRESHOLD-STARTING-SET *with $\delta'(k) := \delta(\lfloor \frac{k}{t} \rfloor)$ for all $k \in \mathbb{N}$,*

*(ii)* MIN-GENERATING-SET *with $\delta'(k) := \delta(\lfloor \frac{k}{2} \rfloor)$ for all $k \in \mathbb{N}$,*

*(iii)* MIN-AXIOM-SET *with $\delta' := \delta$,*

*(iv)* MIN-DEGREE-3-SUBGRAPH-ANNIHILATOR *with $\delta' := \delta$,*

*(v)* MIN-LINEAR-INEQUALITY-DELETION *with $\delta' := \delta$.*

*2. For every gap function $\delta : \mathbb{N} \to \mathbb{R}_{>1}$ there exists a gap-preserving reduction from* MIN-CHAIN-REACTION-CLOSURE *to*

*(vi)* MIN-INDUCED-SAT(3CNF) *with $\delta' := \delta$,*

*(vii)* MIN-INDUCED-SAT *with $\delta' := \delta$.*

A proof of this lemma will appear in the full version of the paper.

## 4.2 Nonmonotone problems

We first consider the minimisation version of the W[P]-complete weighted satisfiability problem for general circuits:

**MIN-WSAT(CIRC)**

| | |
|---|---|
| *Input:* | A circuit $C$ with $n$ input nodes. |
| *Solutions:* | All satisfying assignments $a \in \{0,1\}^n$. |
| *Cost:* | The weight of a satisfying assignment $a$. |
| *Goal:* | min. |

By transforming arbitrary instances $(C,k)$ of $p$-WSAT(CIRC) into instances $(C',k')$, where $C'$ either has a satisfying assignment of weight $k'$ if and only if $C$ has a satisfying assignment of weight $k$ or no satisfying assignments at all, it was shown in [7] that the MIN-WSAT(CIRC) is hard to approximate:

**Lemma 14.** *[7] Let $\rho : \mathbb{N} \to \mathbb{R}_{>1}$ be an arbitrary computable function. Then for* MIN-WSAT(CIRC) *there exists no fpt cost approximation algorithm with ratio $\rho$ unless* W[P] $=$ FPT.

We use the same trick to complete our list of inapproximability results. We will show that there exist fpt reductions from arbitrary instances $(C,k)$ of MIN-WSAT(CIRC) to instances $(x,k')$ of the following problems such that either $x$ has a solution of size exactly $k'$ if and only if $C$ has a satisfying assignment of weight $k$ or $x$ has no solutions at all. This directly implies that these problems are hard to approximate.

| MIN-BOUNDED-NTM-HALT | |
|---|---|
| *Input:* | A nondeterministic Turing machine $\mathbb{M}$ and $n \in \mathbb{N}$ in unary. |
| *Solutions:* | All runs of $\mathbb{M}$ that accept the empty string in at most $n$ steps. |
| *Cost:* | The number of nondeterministic steps in such an accepting run. |
| *Goal:* | min. |

| MIN-PLANAR-WSAT(CIRC) | |
|---|---|
| *Input:* | A planar circuit $C$ (that is $C$ is a graph without crossing edges) with $n$ input nodes. |
| *Solutions:* | All satisfying assignments $a \in \{0,1\}^n$. |
| *Cost:* | The weight of a satisfying assignment $a$. |
| *Goal:* | min. |

**Lemma 15.** *Let $\rho : \mathbb{N} \to \mathbb{R}_{>1}$ be an arbitrary computable function. Then there is no fpt cost approximation algorithm with ratio $\rho$ for any of the following problems unless* W[P] $=$ FPT*:*

*(i)* MIN-BOUNDED-NTM-HALT,

*(ii)* MIN-PLANAR-WSAT(CIRC),

*Proof.* (i) Adapting the fpt reduction described in [10], for a given circuit $C$ and $k \in \mathbb{N}$, we construct a nondeterministic Turing machine $M_{(C,k)}$ whose alphabet contains a symbol for every input node of $C$ (among other symbols). $M(C,k)$ first nondeterministically guesses $k$ inputs of $C$ which are set to 1 in $k$ steps and then simulates $C$ to compute the value of the output node for this $k$-weighted input in $p(|C|)$ steps for some polynomial $p$. If this computed value is 1, then $M_{(C,k)}$ accepts the empty string, otherwise $M_{(C,k)}$ enters some endless loop. Then $(M_{(C,k)}, k + p(|C|), k)$ is an instance of the standard parameterization $p$-BOUNDED-NTM-HALT such that $M_{(C,k)}$ either accepts after $k + p(|C|)$ steps, including exactly $k$ nondeterministic steps if and only if $C$ is $k$-satisfiable, or $M_{(C,k)}$ has no accepting run at all.

(ii) As in the proof of Lemma 14, for any given circuit $C$ and $k \in \mathbb{N}$ we can construct an instance $(C',k')$ of the standard parameterization $p$-WSAT(CIRC) such that $C'$ either has a satisfying assignment of weight $k'$ if and only if $C$ has a satisfying assignment of weight $k$ or no satisfying assignments at all. Now we use the normal fpt reduction from $p$-WSAT(CIRC) to $p$-PLANAR-WSAT(CIRC) that provides a planar gadget of constant size for every crossing in the circuit $C'$ [1]. The resulting instance $(C'',k')$ has the desired properties.

□

Another W[P]-complete nonmonotone minimisation problem that can be proved fpt inapproximable to any ratio in the same way is MIN-HAMMING-DISTANCE, introduced in [12].

The technique that we use to prove the inapproximability of MIN-WSAT(CIRC) can also be applied to the maximisation version of the weighted satisfiability problem (“Find a satisfying assignment of maximum Hamming weight for a Boolean circuit.”) and its restriction to planar circuits. Unless FPT $=$ W[P], both of these maximisation problems are not fpt approximable to any ratio $\rho$ such that $\liminf_{k\to\infty} k/\rho(k) = \infty$. Note that there is an asymmetry between minimisation and maximisation problems; for the latter we need the assumption $\liminf_{k\to\infty} k/\rho(k) = \infty$ for any reasonable approximation ratio. Another W[P]-complete maximisation problem that we can prove to be fpt inapproximable to any ratio $\rho$ such that $\liminf_{k\to\infty} k/\rho(k) = \infty$ is the problem of finding an independent set of maximum size in the intersection of a matroid and a greedoid (see [13] for an exact definition). The proof works by creating gap-instances such that the ratio of gap size and gap position is arbitrarily big in the parameter $k$.

## 5 Conclusions

We have proved that all known natural W[P]-minimisation problems have no fpt approximation algorithm with approximation ratio $\exp(\log^\gamma k)$ for some constant $\gamma$ between 0 and 1 (which might depend on the problem). We conjecture that they actually have no fpt approximation algorithm with any approximation ratio $\rho$. (In the terminology of [7], they are not *fpt approximable*.) Our methods do not seem strong enough to prove this.

There are not many maximisation problems known to be W[P]-complete. Those that are not *antimonotone*, such as the matroid-greedoid intersection problem [13], can easily be dealt with by the methods of Section 4.2. However, for the maximisation version of the weighted satisfiability problem for antimonotone circuits ("Find a satisfying assignment of maximum Hamming weight for an antimonotone Boolean circuit."), we do not know whether it is fpt approximable. This may seem surprising at first because of the duality between the antimonotone maximisation and the monotone minimisation problem. Note, however, that the parameter changes from $k$ to $n - k$ under this duality, and this spoils any direct fpt reduction between the two problems. Actually, it is not unusual for "dual" problems to have completely different parameterized complexities. Nevertheless, we conjecture that the maximisation version of the weighted satisfiability problem for antimonotone circuits is also hard to approximate.

Finally, let us mention that it is still open whether natural optimisation problems of lower parameterized complexity than W[P], such as maximum independent set or minimum dominating set, are fpt approximable.

**Acknowledgements.** We would like to thank Dieter van Melkebeek, Peter Bro Miltersen and Alexander Razborov, as well as three anonymous referees, for helpful comments on an earlier version of this paper.

## References

[1] K. A. Abrahamson, R. G. Downey, and M. R. Fellows. Fixed-parameter tractability and completeness IV: On completeness for W[P] and PSPACE analogues. *Annals of Pure and Applied Logic*, 73:235–276, 1995.

[2] M. Alekhnovich and A. Razborov. Resolution is not automatizable unless W[P] is tractable. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, pages 210–219, 2001.

[3] H. Buhrman, P. B. Miltersen, J. Radhakrishnan, and S. Venkatesh. Are bitvectors optimal? In *STOC '00: Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 449–458, New York, NY, USA, 2000. ACM.

[4] L. Cai and X. Huang. Fixed-parameter approximation: Conceptual framework and approximability results. In *Proceedings of the 2nd International Workshop on Parameterized and Exact Computation*, volume 4169 of *LNCS*, pages 96–108. Springer-Verlag, 2006.

[5] M. Capalbo, O. Reingold, S. Vadhan, and A. Wigderson. Randomness conductors and constant-degree lossless expanders. In *CCC '02: Proceedings of the 17th IEEE Annual Conference on Computational Complexity*, page 15, Washington, DC, USA, 2002. IEEE Computer Society.

[6] Y. Chen, J. Flum, and M. Grohe. Machine-based methods in parameterized complexity theory. *Theoretical Computer Science*, 339:167–199, 2005.

[7] Y. Chen, M. Grohe, and M. Grüber. On parameterized approximability. In *Proceedings of the 2nd International Workshop on Parameterized and Exact Computation*, volume 4169 of *LNCS*, pages 109–120. Springer-Verlag, 2006.

[8] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer-Verlag, New York, 1999.

[9] R. G. Downey, M. R. Fellows, and C. McCartin. Parameterized approximation algorithms. In *Proceedings of the 2nd International Workshop on Parameterized and Exact Computation*, volume 4169 of *LNCS*, pages 121–129. Springer-Verlag, 2006.

[10] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer-Verlag, Berlin Heidelberg, 2006.

[11] V. Guruswami, C. Umans, and S. Vadhan. Unbalanced expanders and randomness extractors from parvaresh-vardy codes. *ccc*, 00:96–108, 2007.

[12] B. Manthey and R. Reischuk. The intractability of computing the hamming distance. *Theoretical Computer Science*, 337(1-3):331–346, 2005.

[13] T. Mielikäinen and E. Ukkonen. The complexity of maximum matroid-greedoid intersection and weighted greedoid maximization. *Discrete Applied Mathematics*, 154(4):684–691, 2006.

[14] N. Nisan and A. Wigderson. Hardness vs. randomness. In *29th Annual Symposium on Foundations of Computer Science*, pages 2–11, 1988.