Theory Comput Syst (2010) 46: 311–339
DOI 10.1007/s00224-008-9138-6

W-Hierarchies Defined by Symmetric Gates

**Michael Fellows · Jörg Flum · Danny Hermelin ·
Moritz Müller · Frances Rosamond**

Published online: 13 August 2008

Abstract The classes of the W-hierarchy are the most important classes of intractable problems in parameterized complexity. These classes were originally defined via the weighted satisfiability problem for Boolean circuits. Here, besides the Boolean connectives we consider connectives such as *majority*, *not-all-equal*, and *unique*. For example, a gate labelled by the majority connective outputs TRUE if more than half of its inputs are TRUE. For any finite set $\mathcal{C}$ of connectives we construct the corresponding W($\mathcal{C}$)-hierarchy. We derive some general conditions which guarantee that the W-hierarchy and the W($\mathcal{C}$)-hierarchy coincide levelwise. If $\mathcal{C}$ only contains the majority connective then the first levels of the hierarchies coincide. We use this to show that a variant of the parameterized vertex cover problem, the majority vertex cover problem, is W[1]-complete.

Research supported by the Australian Research Council, Centre in Bioinformatics, and by Fellowships to the Institute of Advanced Studies, Durham University, and to Grey College, Durham.

M. Fellows
Research Unit, University of Newcastle, Newcastle, Australia
e-mail: michael.fellows@newcastle.edu.au

J. Flum · M. Müller (✉)
Mathematics Department, University of Freiburg, Freiburg, Germany
e-mail: moritz.mueller@math.uni-freiburg.de

J. Flum
e-mail: joerg.flum@math.uni-freiburg.de

D. Hermelin
Caesaria Rothschild Institute, University of Haifa, Haifa, Israel
e-mail: danny@cri.haifa.ac.il

F. Rosamond
Parameterized Complexity Research Unit, University of Newcastle, Newcastle, Australia
e-mail: frances.rosamond@newcastle.edu.au

Keywords Parameterized complexity · W-hierarchy · Symmetric gates · Bounded weft circuits

1 Introduction

Parameterized complexity is a refinement of classical complexity theory, in which one takes into account not only the total input length n, but also other aspects of the problem codified as the parameter k. In doing so, one attempts to confine the exponential running time needed for solving many natural problems strictly to the parameter. For example, the classical VERTEX-COVER problem can be solved in $O(2^k \cdot n)$ time, when parameterized by the size k of the solution sought [10] (significant improvements to this algorithm are surveyed in [11]). This running time is practical for instances with small parameter, and in general is far better than the $O(n^k)$ running time of the brute-force algorithm. More generally, a problem is said to be *fixed-parameter tractable* if it has an algorithm running in time $f(k) \cdot p(n)$, where n is the length of the input, k its parameter, f an arbitrary computable function and p a polynomial. Such an algorithm is said to run in *fpt-time*, and FPT denotes the class of all parameterized problems that are fixed-parameter tractable.

Parameterized complexity theory not only provides methods for proving problems to be fixed-parameter tractable but also gives a framework for dealing with apparently intractable problems. There are a great variety of classes of parameterized intractable problems. However, the most important of these classes are the classes W[1], W[2], ... of the W-hierarchy, on top of which there are the classes W[SAT] and W[P],

$$\text{FPT} \subseteq \text{W}[1] \subseteq \text{W}[2] \subseteq \cdots \subseteq \cdots \subseteq \text{W[SAT]} \subseteq \text{W[P]}.$$

These classes were originally defined via the weighted satisfiability problem for Boolean circuits. In this context Boolean circuits are allowed to contain NOT gates, *small* AND and OR gates of fan-in ≤ 2 and *large* AND and OR gates of arbitrary finite fan-in. The *weft* of such a circuit is the maximum number of large gates on a path from an input to the output, its *depth* the maximum number of all gates on a path from an input to the output. The (Hamming) *weight* of an assignment of truth values to the input variables of the circuit is the number of variables set to TRUE. A circuit is *k-satisfiable* if there is an assignment of weight k satisfying it. The *parameterized weighted satisfiability problem* p-WSAT(Ω) for a class Ω of circuits is the problem

p-WSAT(Ω)
Input: A circuit D in Ω and a natural number k.
Parameter: k.
Question: Is D k-satisfiable?

By definition W[t] is the class of parameterized problems reducible to p-WSAT($\Omega_{t,d}$) for some $d \geq 1$, where $\Omega_{t,d}$ is the class of Boolean circuits of weft $\leq t$ and depth $\leq d$. The classes W[SAT] and W[P] are obtained when taking as Ω the class of all propositional formulas and the class of all circuits respectively. Using these definitions one easily verifies that the parameterized independent set problem p-INDEPENDENT-SET (when parameterized by the size of the solution sought) is in W[1] and that the parameterized dominating set problem p-DOMINATING-SET is in W[2]. In fact, these problems are complete problems for W[1] and W[2], respectively.

Some problems suggest an analysis of the weighted satisfiability problem for circuits with other types of gates. For example, let us consider the parameterized problem

p-MAJORITY-VERTEX-COVER
Input: A graph $G = (V, E)$ and $k \in \mathbb{N}$.
Parameter: k.
Question: Is there a set of k vertices in G which covers a majority of the edges of G, i.e., is there $S \subseteq V$ with $|S| = k$ and $|\{e \in E \mid e \cap S \neq \emptyset\}| > |E|/2$?

It is not hard to reduce this problem to the weighted satisfiability problem for majority circuits of weft 1 and depth 2 (see Sect. 6). The gates of such a circuit are (only!) majority gates, that is, they are labelled by the connective *Maj*, which outputs TRUE if more than half of its inputs are TRUE. Such a gate is *small* if it has fan-in ≤ 3. What is the complexity of the weighted satisfiability problem for circuits based *purely* on majority gates? This paper addresses questions of this kind.

Besides the majority connective there are other quite natural connectives. We mention

the *not-all-equal* connective *NAE*	(there are inputs set to TRUE and inputs set to FALSE)
the *unique* connective U	(exactly one input is set to TRUE)
the connective $c_>$	(at most $c - 1$ inputs are set to TRUE).
the connective $c_\leq$	(at least c inputs are set to TRUE)

(here we assume that c is a natural number ≥ 1). We call the connectives $c_>$ and $c_\leq$ the *threshold connectives*.

All these connectives are *symmetric* in the sense that their value depends only on the number of input gates set to TRUE and the number of input gates set to FALSE, that is, their value is invariant under permutations of the inputs. For example, conjunctions $\bigwedge$ and disjunctions $\bigvee$ (of arbitrary fan-in) are symmetric, but the classical binary implication $\rightarrow$ is not. In this paper we consider symmetric gates only.

The connective *NAE* has the property that if we set one input to TRUE and set another to FALSE, then the value of an *NAE*-gate will be TRUE no matter what the truth values of the other inputs are. Similarly, the connectives U, $c_>$, and $c_\leq$ have the property that a “bounded number of inputs determine the value”. The majority connective *Maj* does not have this property. In Sect. 3 we discuss the connectives C which share the *bounded* property. The corresponding bound determines what should be called a *small* C-gate. This allows us to define the W(C)-hierarchy in the same way as the W-hierarchy was defined.

For a given set $\mathcal{C}$ of bounded connectives we ask how the W-hierarchy and the W($\mathcal{C}$)-hierarchy relate. There are various respects in which this question is interesting.

We will see that, if $\mathcal{C}$ satisfies some further property, then the two hierarchies coincide levelwise. Informally, this suggests a certain robustness in the notion of weft: if we replace Boolean connectives by certain others, the weft remains the key parameter governing the parameterized complexity of weighted satisfiability problems.

On the other hand, besides the W-hierarchy there is a huge variety of parameterized intractable classes like those of the M-, the W*-, the $\mathrm{W}^{\mathrm{func}}$- or the A-hierarchy just to mention a few. We hope that the results and techniques presented in this paper turn out to be useful to sort out the relationships of all these classes.

Typical problems complete for W[1] are *antimonotone*. For the parameterized independent set problem p-INDEPENDENT-SET, famously W[1]-complete, antimonotonicity means that an input graph with an independent set of size k also has an independent set of size k' for any $k' \leq k$. We refer to [3] for a precise definition of (anti)monotonicity. So far only a few monotone problems are known to be W[1]-complete [9]. To this short list we add three new ones.

The contents of the paper can be described as follows. For three groups of connectives we study the properties of the classes of the corresponding W-hierarchy.

The first group is formed by the bounded connectives and is dealt with in Sect. 4. We first show how the property of boundedness entails that the W(C)-hierarchy is contained levelwise in the W-hierarchy. We then explore the issue of possible reverse inclusions. The following observation hints on how this can be achieved.

A Π_t Boolean circuit as defined by Sipser [12] consists of t levels of large gates that alternate AND and OR with an AND gate at the top and with the bottom level gates connected to the input variables and their negations. Such a circuit is in Π_t^+ if negations do not occur and in Π_t^- if all variables are negated. It is well-known [5] that:

- if t is even, then p-WSAT(Π_t^+) is complete for W[t];
- if $t > 1$ is odd, then p-WSAT(Π_t^-) is complete for W[t].

Let $t > 1$. Circuits in Π_t^+ (if t is even) or in Π_t^- (if t is odd) can equivalently be written as propositional formulas of the form

$$\neg \bigvee_{i_1} \neg \bigvee_{i_2} \cdots \neg \bigvee_{i_t} Y_{i_1 \ldots i_t} \tag{1}$$

with variables $Y_{i_1 \ldots i_t}$. Let *NOR* be the connective "defined" by the equivalence

$$NOR[Y_1, \ldots, Y_n] \equiv \neg \bigvee_{i \in [n]} Y_i.$$

Then (1) shows that for every connective C that is capable of simulating *NOR* by a circuit of weft 1 and of constant depth, the W(C)-hierarchy will contain the W-hierarchy levelwise for $t > 1$. This applies to the connectives *NAE*, U, and $c_>$. Together with the result mentioned above, it follows that for bounded connectives C capable of simulating *NOR* the W- and the W(C)-hierarchy coincide levelwise for $t > 1$. We shall exhibit a weaker condition ensuring this coincidence for all t.

Our second group of connectives is formed by the threshold connectives $c_\leq$ and is dealt with in Sect. 5. For these the general results of Sect. 4 do not apply and, in fact, the picture we gain is quite different. The parameterized weighted satisfiability problem for such circuits is:

- solvable in polynomial time if the depth is bounded by a constant;
- W[1]-complete if the depth is bounded in terms of the parameter;
- W[SAT]-complete if the depth is logarithmic in the circuit size and the circuit contains only small gates;
- W[P]-complete for circuits of arbitrary depth

(recall that the weighted satisfiability problem for propositional formulas is complete for W[SAT]). The second result is the first of our new examples of monotone W[1]-complete problems.

Finally, in Sect. 6 we deal with the majority connective *Maj*. We show that the W-hierarchy is contained levelwise in the W(*Maj*)-hierarchy. While we conjecture that $W[t] \subsetneq W[t](\mathit{Maj})$ for $t > 1$, we can show that the first levels coincide. This gives our second example of a monotone W[1]-complete problem. As a corollary we get a third: p-MAJORITY-VERTEX-COVER, a majority version of the parameterized vertex cover problem, is W[1]-complete.

We close in Sect. 7 with a discussion of a key question that we have failed to settle, namely, how $W[t]$ and $W[t](\mathit{Maj})$ relate for $t > 1$. As a starting point we show that certain majority versions of the W[2]-complete parameterized dominating set and hitting set problems have the same complexity and are contained in W[2](*Maj*).

2 Preliminaries

The set of natural numbers (that is, nonnegative integers) is denoted by $\mathbb{N}$. For a natural number n let $[n] := \{1, \dots, n\}$.

For detailed introductions to parameterized complexity theory the reader should consult one of the monographs [5, 8, 11] or the recent surveys [6]. A *parameterized problem* P is a subset of $\Sigma^* \times \mathbb{N}$, where Σ is a finite alphabet and Σ^* is the set of finite strings over Σ. If $(x, k) \in \Sigma^* \times \mathbb{N}$ is an instance of P, then x is the *input* and k the *parameter*. P is *fixed-parameter tractable* if there is an algorithm that decides whether $(x, k) \in P$ in time $f(k) \cdot p(|x|)$, where f is an arbitrary computable function, and p is a polynomial (here $|x|$ denotes the length of the string x).

Let $P \subseteq \Sigma^* \times \mathbb{N}$ and $P' \subseteq (\Sigma')^* \times \mathbb{N}$ be parameterized problems.

An *fpt-reduction*, from P to P' is a mapping $R : \Sigma^* \times \mathbb{N} \to (\Sigma')^* \times \mathbb{N}$ such that:

1. For all $(x, k) \in \Sigma^* \times \mathbb{N}$ we have: $(x, k) \in P \iff R(x, k) \in P'$.
2. There exists a computable function $f : \mathbb{N} \to \mathbb{N}$ such that R is computable in time $f(k) \cdot |x|^c$ for some $c \in \mathbb{N}$.
3. There exists a computable function $g : \mathbb{N} \to \mathbb{N}$ such that for all $(x, k) \in \Sigma^* \times \mathbb{N}$, say with $R(x, k) = (x', k')$, we have $k' \leq g(k)$.

We write $P \leq^{\text{fpt}} P'$ if there is an fpt-reduction from P to P' and we write $P \equiv^{\text{fpt}} P'$ if $P \leq^{\text{fpt}} P'$ and $P' \leq^{\text{fpt}} P$.

3 The W-Hierarchy for Bounded Connectives

In this section we lay down the framework of our investigations. We introduce a notion of a symmetric connective and say what it means for it to be bounded. For a set of bounded connectives we define the classes of the corresponding W-hierarchy.

Definition 1 A *symmetric connective* C (abbreviated *connective*) is a function that maps any pair $(m,n) \in \mathbb{N} \times \mathbb{N}$ with $m+n \geq 1$ to a value in $\{1,0\}$ and can be computed in time polynomial in $(m+n)$.

We interpret $C(m,n)=1$ as "C outputs a '1' (= TRUE), if it gets as input m-many '1's and n-many '0's." Examples of connectives are the *(big) conjunction* $\bigwedge$, *(big) disjunction* $\bigvee$, the *not-all-equal* connective *NAE*, the *unique* connective U, for $c \in \mathbb{N}$ the *c-threshold* connectives $c_{>}$ and $c_{\leq}$, the *majority* connective *Maj* defined by [1]

$$\begin{aligned}
\textstyle\bigwedge(m,n)=1 &\iff n=0;\\
\textstyle\bigvee(m,n)=1 &\iff m\neq 0;\\
\mathit{NAE}(m,n)=1 &\iff m\geq 1 \text{ and } n\geq 1;\\
U(m,n)=1 &\iff m=1;\\
c_{>}(m,n)=1 &\iff m<c;\\
c_{\leq}(m,n)=1 &\iff c\leq m;\\
\mathit{Maj}(m,n)=1 &\iff m>n;\\
\mathit{NOR}(m,n)=1 &\iff m=0.
\end{aligned}$$

Let $\mathcal{C}$ be a finite set of connectives (throughout this paper $\mathcal{C}$ will always denote a *finite* set of connectives). A $\mathcal{C}$*-circuit* D is a finite connected acyclic directed graph with multiple edges; mostly we will call the vertices of D *gates*. Each gate of D of positive fan-in is labelled with a symbol $C \in \mathcal{C}$. We then call it a C-gate. As usual we call a *NOR*-gate of fan-in one a $\neg$-gate.

Gates with fan-in zero are the *input gates*; they are labeled with *variables* $X_1, X_2, \ldots$ or with the *Boolean constants* $\top$ and $\bot$. We only consider circuits having exactly one gate of fan-out zero, the *output gate*. We let $X, Y, Y_1, Y_2, \ldots, Z$ denote variables. We denote by CIRC($\mathcal{C}$) the class of all $\mathcal{C}$-circuits. A *Boolean* circuit is a $\{NOR, \bigwedge, \bigvee\}$-circuit.

The *size* $\|D\|$ of a $\mathcal{C}$-circuit D is defined as the sum of the number of gates of D and the number of edges of D.

An *assignment* or *valuation* for a $\mathcal{C}$-circuit D is a mapping $\mathcal{V}$ from a set of variables containing all the variables occurring in D to $\{1,0\}$. In the obvious bottom-up way one defines the value $\mathcal{V}(g)$ for any gate g of D:

If g is labelled by a variable X, then $\mathcal{V}(g) := \mathcal{V}(X)$.
If g is labelled by $\top$, then $\mathcal{V}(g) := 1$.
If g is labelled by $\bot$, then $\mathcal{V}(g) := 0$.

[1] Whenever we write $C(m,n)$ we assume that $m+n \geq 1$.

If g is labelled by C and has entries from the gates $(g_i)_{i\in I}$ (where a gate g' occurs exactly m times in the enumeration $(g_i)_{i\in I}$ if there are exactly m edges from g' to g), then

$$\mathcal{V}(g) := C\big(|\{i \in I \mid \mathcal{V}(g_i) = 1\}|, |\{i \in I \mid \mathcal{V}(g_i) = 0\}|\big).$$

The assignment *satisfies* the circuit if its value for the output gate is 1. A $\mathcal{C}_1$-circuit D_1 and a $\mathcal{C}_2$-circuit D_2 are *equivalent*, $D_1 \equiv D_2$, if they are satisfied by the same assignments defined on all variables occurring in D_1 or D_2.

The *weight* of an assignment is the number of variables set to 1. A circuit D is *k-satisfiable* (where $k \in \mathbb{N}$), if there is an assignment for the input variables of D of weight k satisfying D. For a set Ω of $\mathcal{C}$-circuits, the *parameterized weighted satisfiability problem* p-WSAT(Ω) *for circuits in* Ω is the following problem:

p-WSAT(Ω)
Input: A circuit $D \in \Omega$ and $k \in \mathbb{N}$.
Parameter: k.
Question: Is D k-satisfiable?

A $\mathcal{C}$-circuit where all gates, besides the output gate, have fan-out one is a *$\mathcal{C}$-formula*. They constitute the formulas of *propositional logic with connectives from $\mathcal{C}$*. Thus these formulas can be viewed as the strings obtained from the propositional variables $X_1, X_2, \ldots$ and the constants $\top$ and $\bot$ by finitely many applications of the rule:

If $n \in \mathbb{N}$ and $\alpha_1, \ldots, \alpha_n$ are $\mathcal{C}$-formulas and $C \in \mathcal{C}$, then $C[\alpha_1, \ldots, \alpha_n]$ is a formula.

We denote by FORM($\mathcal{C}$) the class of all $\mathcal{C}$-formulas.

Definition 2 A connective C is *bounded* if there is $b \in \mathbb{N}$ such that for all m, n, $n' \in \mathbb{N}$ with $n, n' \geq b$

$$C(m,n) = C(m,n') \quad \text{and} \quad C(n,m) = C(n',m).$$

We then say that C is *b-bounded*. Clearly, if C is b-bounded and $b < b'$, then C is b'-bounded. The smallest b such that C is b-bounded is denoted by $b(C)$.

Example 3 The reader will easily verify that

$$b(NOR) = 1, \qquad b\Big(\bigwedge\Big) = 1, \qquad b\Big(\bigvee\Big) = 1, \qquad b(NAE) = 1, \qquad b(U) = 2,$$
$$b(c_{>}) = c \quad \text{and} \quad b(c_{\leq}) = c.$$

Maj is not bounded.

Weft and the W-Hierarchy Let $\mathcal{C}$ be a set of bounded connectives. Let D be a $\mathcal{C}$-circuit. A C-gate of D is *small* if it has fan-in less than $2(b(C)+1)$; otherwise it is *large*. A circuit is *small* if it only contains small gates. The *weft* of D is the maximum number of large gates on any path from the input gates of D to its output gate.

The *depth* of D is the maximum number of (large and small) gates of positive fan-in on any path from the input gates of D to its output gate. We set

$$\Omega_{t,d}(\mathcal{C}) := \{D \mid D \text{ a } \mathcal{C}\text{-circuit of weft} \leq t \text{ and depth} \leq d\}.$$

By definition, for $t \geq 1$, W[t] is the class of parameterized problems fpt-reducible to p-WSAT($\Omega_{t,d}(\{NOR, \bigwedge, \bigvee\})$) for some $d \in \mathbb{N}$.[2] The classes W[1], W[2], ... constitute the W-hierarchy. Furthermore:

- W[SAT] is the class of parameterized problems that are fpt-reducible to the problem p-WSAT(FORM($\{NOR, \bigwedge, \bigvee\}$));
- W[P] is the class of parameterized problems that are fpt-reducible to the problem p-WSAT(CIRC($\{NOR, \bigwedge, \bigvee\}$)).

Therefore we define:

Definition 4 Let $\mathcal{C}$ be a class of connectives.

- W[SAT]($\mathcal{C}$) is the class of parameterized problems that are fpt-reducible to the problem p-WSAT(FORM($\mathcal{C}$));
- W[P]($\mathcal{C}$) is the class of parameterized problems fpt-reducible to the problem p-WSAT(CIRC($\mathcal{C}$)).

Let $\mathcal{C}$ be a class of bounded connectives.

- For $t \geq 1$, W[t]($\mathcal{C}$) is the class of parameterized problems fpt-reducible to p-WSAT($\Omega_{t,d}(\mathcal{C})$) for some $d \in \mathbb{N}$. The classes W[1]($\mathcal{C}$), W[2]($\mathcal{C}$), ... constitute the W($\mathcal{C}$)-*hierarchy*.

Clearly, for a class $\mathcal{C}$ of bounded connectives, we have

$$\mathrm{W}[1](\mathcal{C}) \subseteq \mathrm{W}[2](\mathcal{C}) \subseteq \cdots \subseteq \mathrm{W}[\mathrm{SAT}](\mathcal{C}) \subseteq \mathrm{W}[\mathrm{P}](\mathcal{C}).$$

4 Comparing the W-Hierarchy and the W($\mathcal{C}$)-Hierarchy for Classes $\mathcal{C}$ of Bounded Connectives

In this section we show that the W-hierarchy and the W($\mathcal{C}$)-hierarchy coincide levelwise for a set of bounded connectives $\mathcal{C}$ satisfying some further conditions. First we prove for such a $\mathcal{C}$ that the W($\mathcal{C}$)-hierarchy is contained levelwise in the W-hierarchy. We use two facts holding for a bounded connective C:

- For $m, n \geq b(C)$ we have $C(m,n) = C(b(C), b(C))$ (compare the definition of a bounded connective).
- A C-gate with fan-in less than $2 \cdot (b(C) + 1)$ can be simulated by a small Boolean circuit of constant depth (by functional completeness of $\{\neg, \wedge, \vee, \}$).

[2]Note that in [5] the depth is defined without taking into account the $\neg$-gates. Clearly, this difference has no effect on the definition of W[t].

Proposition 5 *Let $\mathcal{C}$ be a set of bounded connectives. There is a $d_0 \in \mathbb{N}$ and a polynomial time algorithm assigning to every $\mathcal{C}$-circuit D an equivalent Boolean circuit D' such that*:

- *if $D \in \Omega_{t,d}(\mathcal{C})$, then $D' \in \Omega_{t,d\cdot d_0}$;*
- *if D is a $\mathcal{C}$-formula, then D' is a Boolean propositional formula.*

Proof For $C \in \mathcal{C}$ an assignment $\mathcal{V}$ satisfies a C-formula of the form $C[Y_1, \dots, Y_n]$ if and only if:

- there is an $r < b(C)$ such that $C(r, n-r) = 1$ and $\mathcal{V}$ satisfies exactly r of the Y_i, or
- there is an $s < b(C)$ such that $C(n-s, s) = 1$ and $\mathcal{V}$ satisfies exactly $n-s$ of the Y_i, or
- $C(b(C), b(C)) = 1$ and $\mathcal{V}$ satisfies at least $b(C)$ of the Y_i and at least $b(C)$ of the Y_i are not satisfied by $\mathcal{V}$.

We find Boolean formulas $\gamma_1, \gamma_2, \gamma_3$ "expressing these cases"—namely:

$$\gamma_1 := \bigvee_{r<b(C), C(r,n-r)=1} \Bigg(\bigvee_{1\le i_1<\dots<i_r\le n} (Y_{i_1} \wedge \dots \wedge Y_{i_r}) \wedge \bigwedge_{1\le j_1<\dots<j_{r+1}\le n} (\neg Y_{j_1} \vee \dots \vee \neg Y_{j_{r+1}}) \Bigg).$$

γ_2 is defined similarly. γ_3 is $\perp$ in case $C(b(C), b(C)) = 0$ and else

$$\gamma_3 := \bigvee_{1\le i_1<\dots<i_{b(C)}\le n} (Y_{i_1} \wedge \dots \wedge Y_{i_{b(C)}}) \wedge \bigvee_{1\le j_1<\dots<j_{b(C)}\le n} (\neg Y_{j_1} \wedge \dots \wedge \neg Y_{j_{b(C)}}).$$

Here and later we omit $\bigvee \emptyset$ and $\bigwedge \emptyset$. Then we have

$$C[Y_1, \dots, Y_n] \equiv (\gamma_1 \vee \gamma_2 \vee \gamma_3). \tag{2}$$

One easily verifies that there is a $d_0 \in \mathbb{N}$ such that for every $C \in \mathcal{C}$:

- if $n < 2(b(C)+1)$, then the Boolean formula $(\gamma_1 \vee \gamma_2 \vee \gamma_3)$ in (2) can be chosen of weft 0 and depth $\le d_0$;
- if $n \ge 2(b(C)+1)$, then the Boolean formula $(\gamma_1 \vee \gamma_2 \vee \gamma_3)$ in (2) can be chosen of weft 1 and depth $\le d_0$.

Let D be a $\mathcal{C}$-circuit. For $C \in \mathcal{C}$ we replace, bottom-up, each C-gate in D according to the equivalence (2), thus obtaining an equivalent Boolean circuit D'. Clearly, the mapping $D \mapsto D'$ is computable in polynomial time and maps formulas to formulas. If $D \in \Omega_{t,d}(\mathcal{C})$, then we have $D' \in \Omega_{t,d\cdot d_0}$. □

Corollary 6 *If $\mathcal{C}$ is a set of bounded connectives, then*

$$\mathrm{W}[t](\mathcal{C}) \subseteq \mathrm{W}[t] \quad \textit{for all } t \geq 1, \qquad \mathrm{W[SAT]}(\mathcal{C}) \subseteq \mathrm{W[SAT]}, \qquad \mathrm{W[P]}(\mathcal{C}) \subseteq \mathrm{W[P]}.$$

For the last inclusion of this corollary we do not need the boundedness of the connectives in $\mathcal{C}$, that is:

Proposition 7 *If $\mathcal{C}$ is a set of connectives, then* $\mathrm{W[P]}(\mathcal{C}) \subseteq \mathrm{W[P]}$.

Proof The following nondeterministic algorithm decides the $\mathrm{W[P]}(\mathcal{C})$-complete problem p-WSAT(CIRC($\mathcal{C}$)): On input (D, k) the algorithm first guesses k input variables of D (by guessing $k \cdot \log \|D\|$ bits) and then in time polynomial in $\|D\|$ checks whether the assignment setting exactly these variables to TRUE satisfies D. By the machine characterization of [4] this algorithm witnesses that p-WSAT(CIRC($\mathcal{C}$)) $\in$ W[P]. □

We turn to the converse inclusions. We look for conditions on $\mathcal{C}$ that allow us to find a weft-preserving translation from Boolean circuits to $\mathcal{C}$-circuits.

Definition 8 Let C be a connective.

- C is *monotone* if and only if $C(m,n) \leq C(m+r, n-r)$ for all $m, n \in \mathbb{N}$ and all $r \in [n]$.
- C is $\bigvee$*-closed* if and only if there are $m, n \in \mathbb{N}$ such that for all $r \geq 1$ and all $s \in [r]$

$$C(m, n+r) \neq C(m+s, n+r-s).$$

We say that a class $\mathcal{C}$ of connectives is *monotone* if and only if all connectives in $\mathcal{C}$ are monotone. $\mathcal{C}$ is $\bigvee$*-closed* if and only if at least one connective in $\mathcal{C}$ is $\bigvee$-closed.

Example 9

(a) The connectives $c_{\leq}$ and *Maj* are monotone.
(b) The connectives *NAE*, U, and $c_{>}$ are not monotone as seen by taking

$$\begin{aligned} m &= 1, & r &= 1, & &\text{and} & n &= 1 & &\text{for } \mathit{NAE}; \\ m &= 1, & r &= 1, & &\text{and} & n &= 1 & &\text{for } U; \\ m &= c-1, & r &= 1, & &\text{and} & n &= 1 & &\text{for } c_{>}. \end{aligned}$$

(c) The connectives $\bigvee$, *NAE*, U, $c_{>}$, and $c_{\leq}$ are $\bigvee$-closed as seen by taking

$$\begin{aligned} m &= 0 & &\text{and} & n &= 0 & &\text{for } \bigvee \\ m &= 0 & &\text{and} & n &= 1 & &\text{for } \mathit{NAE}; \\ m &= 1 & &\text{and} & n &= 0 & &\text{for } U; \\ m &= c-1 & &\text{and} & n &= 0 & &\text{for } c_{>}; \\ m &= c-1 & &\text{and} & n &= 0 & &\text{for } c_{\leq}. \end{aligned}$$

Lemma 10 *Let C be a bounded connective. Then*:

1. *if C is $\bigvee$-closed, then there are $m, n \in \mathbb{N}$ with $m + n < 2b(C)$ such that for all $r \geq 1$ and all $s \in [r]$ we have $C(m, n + r) \neq C(m + s, n + r - s)$;*
2. *if C is not monotone, then there are $m, n, r \in \mathbb{N}$ with $m + n + r \leq 2b(C)$ such that $C(m + r, n) < C(m, n + r)$.*

Proof Let C be bounded. To prove (1) assume that C is $\bigvee$-closed. Choose $m, n \in \mathbb{N}$ such that for all $r \geq 1$ and all $s \in [r]$ we have $C(m, n + r) \neq C(m + s, n + r - s)$. In case $n > b(C)$ we can replace it by $b(C)$, so we can assume $n \leq b(C)$. But $m < b(C)$ as otherwise for sufficiently large r

$$C(m, n + r) = C(m + 1, n + r) = C(m + 1, n + r - 1)$$

contradicting the choice of m and n.

To prove (2) assume that C is not monotone. It follows from Definition 8 that there are $m, n, r \in \mathbb{N}$ such that $C(m + r, n) < C(m, n + r)$. We have to show that we can find such m, n, r with $m + n + r \leq 2b(C)$. In fact, if both $m, n \geq b(C)$, then $C(m + r, n) = C(m, n + r)$. Assume now that $m < b(C)$ and $n \geq b(C)$. Then we can replace n by $b(C)$. If $m + r \leq b(C)$, we are done. If $m + r > b(C)$, then we replace r by $b(C) - m$. The remaining cases are treated similarly. □

Proposition 11 *Let $\mathcal{C}$ be a set of bounded connectives which is $\bigvee$-closed and not monotone. There is a $d_0 \in \mathbb{N}$ (we can even choose $d_0 = 3$) and a polynomial time algorithm assigning to every Boolean circuit D an equivalent $\mathcal{C}$-circuit D' such that:*

- *if $D \in \Omega_{t,d}$, then $D' \in \Omega_{t,d\cdot d_0}(\mathcal{C})$;*
- *if D is a Boolean propositional formula, then D' is a $\mathcal{C}$-formula.*

Proof Choose a connective $C \in \mathcal{C}$ which is not monotone. By Lemma 10(2) there are $m, n, r \in \mathbb{N}$ with $m + n + r \leq 2b(C)$ such that

$$0 = C(m + r, n) \neq C(m, n + r) = 1.$$

Then we get:

$$\neg Y \quad \text{and} \quad C[\underbrace{\top, \ldots, \top}_{m}, \underbrace{Y, \ldots, Y}_{r}, \underbrace{\bot, \ldots, \bot}_{n}] \quad \text{are equivalent.} \tag{3}$$

This shows how we can replace $\neg$-gates of Boolean circuits by C-gates; moreover, the occurrence of C in (3) is small as $m + n + r \leq 2b(C) < 2(b(C) + 1)$.

Furthermore, $\mathcal{C}$ is $\bigvee$-closed. Hence there are $C' \in \mathcal{C}$ and $m, n \in \mathbb{N}$ such that for all $r \geq 1$ and all $s \in [r]$

$$C'(m, n + r) \neq C'(m + s, n + r - s).$$

We have that, if $C'(m, n + r) = 1$, then

$$\neg \bigvee_{i \in [r]} Y_i \quad \text{and} \quad C'[\underbrace{\top, \ldots, \top}_{m}, Y_1, \ldots, Y_r, \underbrace{\bot, \ldots, \bot}_{n}] \quad \text{are equivalent} \tag{4}$$

and, if $C'(m, n+r) = 0$, then

$$\bigvee_{i\in[r]} Y_i \quad \text{and} \quad C'[\underbrace{\top,\dots,\top}_{m}, Y_1,\dots,Y_r, \underbrace{\bot,\dots,\bot}_{n}] \quad \text{are equivalent.} \tag{5}$$

By Lemma 10(1) we can assume that $m+n < 2b(C)$. Then for $r = 2$ it follows that $m+n+r < 2b(C)+2$ and hence the formulas on the right hand side are small.

From (3), (4), and (5) one gets the desired translation $D \mapsto D'$. □

Corollary 12 *Let $\mathcal{C}$ be a set of bounded connectives which is $\bigvee$-closed and not monotone. Then*

$$\mathrm{W}[t] \subseteq \mathrm{W}[t](\mathcal{C}) \quad \text{for all } t \geq 1, \qquad \mathrm{W[SAT]} \subseteq \mathrm{W[SAT]}(\mathcal{C}), \qquad \mathrm{W[P]} \subseteq \mathrm{W[P]}(\mathcal{C}).$$

Remark 13 The preceding proof together with Example 9 shows that if for some $c \geq 1$ we have $c_> \in \mathcal{C}$, we can obtain an equivalent $\mathcal{C}$-circuit not containing the constant $\bot$. Furthermore, if $c = 1$ the equivalent $\mathcal{C}$-circuit can be chosen such that it contains neither $\bot$ nor $\top$.

Putting Corollary 6 and Corollary 12 together we obtain the main result of this section, namely:

Theorem 14 *Let $\mathcal{C}$ be a set of bounded connectives which is not monotone and $\bigvee$-closed. Then*

$$\mathrm{W}[t] = \mathrm{W}[t](\mathcal{C}) \quad \textit{for all } t \geq 1, \qquad \mathrm{W[SAT]} = \mathrm{W[SAT]}(\mathcal{C}), \qquad \mathrm{W[P]} = \mathrm{W[P]}(\mathcal{C}).$$

Again, for the last equality, we do not need the boundedness of the connectives.

Corollary 15 *Let $\mathcal{C}$ be one of the following classes*:

- $\{NOR, \bigwedge, \bigvee, NAE, U, c_{1>}, \dots, c_{s>}, c'_{1\leq}, \dots, c'_{r\leq}\}$, *where* $s, r \in \mathbb{N}$ *and* $c_1, \dots, c_s, c'_1, \dots, c'_r \geq 1$, *or*
- $\{NAE\}$ *or* $\{U\}$ *or* $\{c_>\}$, *where* $c \geq 1$.

Then the W*-hierarchy and the* $\mathrm{W}(\mathcal{C})$*-hierarchy coincide levelwise and the classes* W[SAT] *and* $\mathrm{W[SAT]}(\mathcal{C})$ *and the classes* W[P] *and* $\mathrm{W[P]}(\mathcal{C})$ *coincide.*

Pure Circuits In the Boolean context we can do without the constants $\top$ and $\bot$. A similar result holds for some of the connectives considered so far.

Definition 16 A $\mathcal{C}$-circuit is *pure* if and only if it does not contain the constants $\top$ and $\bot$. By $\Pi_{t,d}(\mathcal{C})$ we denote the class of all pure circuits in $\Omega_{t,d}(\mathcal{C})$.

In the following, for $\mathcal{C} = \{C\}$, we often write C-circuit for $\mathcal{C}$-circuit and use analogous conventions for other notions.

Theorem 17 *Let $c \geq 1$ and let $\mathcal{C}$ be a set of bounded connectives containing one of the connectives NAE, U, or $c_>$. Then, for sufficiently large d, the problem p-*WSAT*$(\Pi_{t,d}(\mathcal{C}))$ is* W$[t]$*-complete under fpt-reductions.*

Proof Let $C \in \{NAE, U, c_>\}$ and $t, d \geq 1$. By Corollary 15 it suffices to show that there is a d' such that

$$p\text{-WSAT}(\Omega_{t,d}(C)) \quad \text{is fpt-reducible to} \quad p\text{-WSAT}(\Pi_{t,d'}(C)).$$

First let $C = NAE$. Let (D, k) be an instance of p-WSAT$(\Omega_{t,d}(NAE))$. Let $Y_1, \dots, Y_n$ be the variables of D. We may assume that $n > k$. First we get rid of $\bot$ by replacing it everywhere by $NAE[Y_1, Y_1]$ thus obtaining a circuit D' with $D \equiv D'$. Let Z be a new variable and

$$\alpha := NAE[Y_1, \dots, Y_n, Z].$$

Clearly every assignment of weight $k+1$ satisfies α. Let $D'\frac{Z}{\top}$ be the circuit obtained from D' by replacing all occurrences of $\top$ by Z. Note that the two occurrences of NAE displayed in

$$NAE\left[\alpha, NAE\left[\alpha, D'\frac{Z}{\top}, Z\right]\right]$$

are small ones, while the one in α will be large in general. Hence the weft of this circuit is $\leq \max\{1, \text{weft}(D)\}$. Moreover, we have:

$$\begin{aligned} & NAE[\alpha, NAE[\alpha, D'\tfrac{Z}{\top}, Z]] \text{ is } k+1\text{-satisfiable} \\ \iff & \quad \text{there is an assignment of weight } k+1 \text{ not satisfying } NAE[\alpha, D'\tfrac{Z}{\top}, Z] \\ \iff & \quad \text{there is an assignment of weight } k+1 \text{ satisfying } D'\tfrac{Z}{\top} \text{ and } Z \\ \iff & \quad D' \text{ is } k\text{-satisfiable.} \end{aligned}$$

Thus $(D, k) \mapsto (NAE[\alpha, NAE[\alpha, D'\frac{Z}{\top}, Z]], k+1)$ yields the desired reduction.

Now let $C = U$. Let (D, k) be an instance of p-WSAT$(\Omega_{t,d}(U))$. We may assume that D contains a variable X. Replacing, if necessary, $\bot$ by $U(X, X)$, we can furthermore assume that $\bot$ does not occur in D. Again let Z be a new variable and let $D\frac{Z}{\top}$ be the circuit obtained from D by replacing all occurrences of $\top$ by Z. We consider the circuit

$$D' := U\left[D\frac{Z}{\top}, U\left[D\frac{Z}{\top}, Z\right], U\left[D\frac{Z}{\top}, Z\right]\right]. \tag{6}$$

We have

$$\begin{aligned} & D' \text{ is } k+1\text{-satisfiable} \\ \iff & \quad \text{there is an assignment of weight } k+1 \text{ satisfying } D\tfrac{Z}{\top} \\ & \quad \text{and not satisfying } U[D\tfrac{Z}{\top}, Z] \\ \iff & \quad \text{there is an assignment of weight } k+1 \text{ satisfying } D\tfrac{Z}{\top} \text{ and } Z \\ \iff & \quad D \text{ is } k\text{-satisfiable.} \end{aligned}$$

As the additional occurrences of U in D' displayed in (6) are small, $(D,k) \mapsto (D', k+1)$ is the desired reduction.

Finally let $C = c_>$. Let (D,k) be an instance of p-WSAT$(\Omega_{t,d}(c_>))$. Again by Proposition 5, Proposition 11, and Remark 13 we can assume that $\bot$ does not occur in D and in case $c=1$ we already get our claim. So assume that $c>1$. Let Y be a variable in D (clearly, we can assume that D contains at least one variable). Then the circuit D' obtained from D replacing every occurrence of $\top$ by $c_> Y$ yields the desired equivalent circuit of the same weft. □

5 The Threshold Connectives $c_\leq$

From Sect. 3, we know that the threshold connectives of the form $c_\leq$ with $c \geq 1$ are monotone and bounded.

By the first property, if an assignment $\mathcal{V}$ satisfies a $c_\leq$-circuit, then so does every extension of $\mathcal{V}$, that is, every assignment $\mathcal{V}'$ such that $\mathcal{V}(Y) \leq \mathcal{V}'(Y)$ for all variables Y in the domain of $\mathcal{V}$. By the second property, we get from Corollary 6

$$W[t](c_\leq) \subseteq W[t]$$

for all $t \geq 1$. However, we can even show that $W[t](c_\leq)$ is contained in PTIME.

Furthermore, we exhibit classes of $c_\leq$-circuits such that the weighted satisfiability problem is W[1]-complete, W[SAT]-complete and W[P]-complete.

Theorem 18 *Let $c,d \geq 1$. There is a polynomial time algorithm that, given a $c_\leq$-circuit D of depth $\leq d$ and $k \in \mathbb{N}$ decides whether D is k-satisfiable.*

The following notion will be helpful for the proof of this and also of later results.

Definition 19 Let D be a $c_\leq$-circuit. A *thin subcircuit D' of D* is a $c_\leq$-circuit satisfying:

- the directed graph D' is a subgraph of D and every gate is labelled in D' as in D;
- the output gate of D is contained in D';
- every gate of D' has fan-in $= c$.

The proof of the following facts is immediate:

Lemma 20 *Let D be a $c_\leq$-circuit.*

1. *Every thin subcircuit of D has at most $(1+c)^d$ gates and at most $(1+c)^d$ edges, where d is the depth of D.*
2. *Let $k \in \mathbb{N}$ be less than or equal to the number of variables occurring in D. Then the following are equivalent:*
   - *D is k-satisfiable;*
   - *D has a thin subcircuit with at most k variables.*

Proof of Theorem 18 Assume $c, d \geq 1$. Let D be a $c_{\leq}$-circuit of depth d and $k \in \mathbb{N}$. Furthermore let ℓ be the maximum fan-in of the gates of D. Then D contains at most $\binom{\ell}{c}^d$ thin subcircuits. Thus an exhaustive search for a thin subcircuit with at most k variables runs in polynomial time. This yields the desired algorithm by the previous lemma. □

A W[P]-*Complete Problem* The previous result shows that the W-hierarchy and the W($c_{\leq}$)-hierarchy do not coincide levelwise; however, W[P] and W[P]($c_{\leq}$) coincide as shown by Theorem 21 below.

In the following proofs we shall use the equivalence

$$\bigvee_{i \in [n]} Y_i \equiv c_{\leq}[\underbrace{Y_1, \ldots, Y_1}_{c \text{ times}}, \ldots, \underbrace{Y_n, \ldots, Y_n}_{c \text{ times}}] \tag{7}$$

and for $c > 1$ the equivalence

$$(Y_1 \wedge Y_2) \equiv c_{\leq}[\underbrace{Y_1, \ldots, Y_1}_{c-1 \text{ times}}, Y_2]. \tag{8}$$

Recall that by Definition 2, a $c_{\leq}$-gate is small if it has fan-in less than $2 \cdot (c+1)$. Note that the occurrence of $c_{\leq}$ in (8) is small and that for $n = 2$ the occurrence of $c_{\leq}$ in (7) is small, too.

Theorem 21 *Let $c > 1$. Then p-*WSAT(CIRC($c_{\leq}$)) *is* W[P]-*complete under fpt-reductions. Furthermore, the weighted satisfiability problem of pure $\{c_{\leq}\}$-circuits containing no large gates, is* W[P]-*complete under fpt-reductions.*

Proof Let $c > 1$. As the connective $c_{\leq}$ is bounded, Proposition 5 shows that the problem p-WSAT(CIRC($c_{\leq}$)) is in W[P].

For hardness we use the fact that the weighted satisfiability problem of *positive* Boolean circuits with no large gates is W[P]-complete [1]. Using (7) and (8) we can translate such circuits into pure $c_{\leq}$-circuits containing no large gates. □

A W[1]-*Complete Problem* In order to describe weighted satisfiability problems for $c_{\leq}$-circuits that are complete for W[1], we have to consider instances (D, k) where the depth of D depends on the parameter k (this is reminiscent of the defining problems for the classes of the W*-hierarchy, cf. [7]).

Let $f : \mathbb{N} \to \mathbb{N}$ be a computable function. The *weighted satisfiability problem* p-WSAT($c_{\leq}, f$) *of $c_{\leq}$-circuits of depth f* is given by

p-WSAT($c_{\leq}, f$)
Input: $k \in \mathbb{N}$ and a $c_{\leq}$-circuit D of depth $\leq f(k)$.
Parameter: k.
Question: Does $\mathcal{C}$ have a satisfying assignment of weight k?

We want to show:

Theorem 22 *Let $c > 1$ and f be a computable function such that $f(k) \geq 2 + 2\log k$ for all $k \in \mathbb{N}$. Then p-*WSAT$(c_{\leq}, f)$ *is* W[1]*-complete under fpt-reductions.*

We obtain part of the theorem by:

Lemma 23 *Let f be a computable function. Then p-*WSAT$(c_{\leq}, f)$ *is in* W[1].

Proof By Lemma 20, the following nondeterministic algorithm $\mathbb{A}$ solves the problem p-WSAT$(c_{\leq}, f)$. Given $k \in \mathbb{N}$ and a $\{c_{\leq}\}$-circuit D of depth $\leq f(k)$ the algorithm $\mathbb{A}$ first guesses at most $(1+c)^{f(k)}$ gates and edges of D and then in time depending only on the parameter (and the constant c) checks that they constitute a thin subcircuit of D with at most k variables. To finish the proof we have two options. The reader familiar with the characterization of W[1] in terms of nondeterministic random access machines [4] will easily see that the algorithm $\mathbb{A}$ can be simulated by a program for such a machine. The second option: It is not hard, using the algorithm $\mathbb{A}$, to construct an fpt-reduction of p-WSAT$(c_{\leq}, f)$ to the parameterized short halting problem for nondeterministic single-tape Turing machines, a problem in W[1] (see [2]). □

To obtain a proof of the W[1]-hardness of p-WSAT$(c_{\leq}, f)$ we reduce to it a variant of the parameterized clique problem.

Let p-MULTICOLOURED-CLIQUE be the problem:

p-MULTICOLOURED-CLIQUE

Input:	A graph $G = (V, E)$, a number $k \in \mathbb{N}$, and a function $h : V \to [k]$.
Parameter:	k.
Question:	Does G contain a clique $C \subseteq V$ of size k with $h(C) = [k]$?

Here $h(C) := \{h(a) \mid a \in C\}$ for $C \subseteq V$. We call sets C with $h(C) = [k]$ *colourful*. We refer to h as a *colouring* and to the elements of $[k]$ as *colours*. The vertices of colour $i \in [k]$ are those in the set $\{a \in V \mid h(a) = i\}$.

Although we believe the following result to be known we are not aware of a reference and therefore we include the simple proof.

Lemma 24 *p-*MULTICOLOURED-CLIQUE *is* W[1]*-complete.*

Proof To show membership in W[1] we observe that an instance (G, k, h) of p-MULTICOLOURED-CLIQUE is a "yes"-instance if and only if the graph G' has a clique of size k, where we obtain G' from G by deleting all edges between vertices of the same colour. This defines an fpt-reduction to the W[1]-complete problem p-CLIQUE.

To show the W[1]-hardness we reduce from p-CLIQUE. Let $G = (V, E)$ be a graph and $k \in \mathbb{N}$. Let the graph $G' = (V', E')$ have set of vertices $V' := V \times [k]$ and an edge between (u, i) and (v, j) if and only if $\{u, v\} \in E$ and $i \neq j$. The colouring h' of G' is the projection to the second component. It is easy to verify that $(G, k) \mapsto (G', k, h)$ is an fpt-reduction. □

Lemma 25 *Let $c > 1$ and f be a computable function such that $f(k) \geq 2 + 2\log k$ for all $k \in \mathbb{N}$. Then there is a fpt reduction from p-MULTICOLOURED-CLIQUE to p-WSAT(pure $c_{\leq}$, f).*

Proof Let (G, k, h) be an instance of p-MULTICOLOURED-CLIQUE with $G = (V, E)$. We may assume that for each two colours there is at least one edge in G between vertices of these colours (otherwise, (G, k, h) is a "no"-instance). First we construct a Boolean circuit D such that

$$(G, k, h) \in p\text{-MULTICOLOURED-CLIQUE} \iff D \text{ is } k\text{-satisfiable}. \qquad (9)$$

It is important to pay attention to the form of the circuit D, as this will be important later. The circuit D has variables X_v for $v \in V$. The input level of D consists of k blocks, where the ith block contains the vertices v of colour i, $v \in h^{-1}(i)$; vertex v is labelled by X_v. The first level contains gates arranged in $\binom{k}{2}$ blocks, one for each pair $\{i, j\}$ of different colours. The $\{i, j\}$th block contains a gate $g_{\{u,v\}}$ for each edge $\{u, v\} \in E$ with $h(\{u, v\}) = \{i, j\}$. Each such gate $g_{\{u,v\}}$ gets inputs from g_u and g_v and is labelled by $\wedge$. The second level contains, for each pair $\{i, j\}$ of different colours, a gate $g_{\{i,j\}}$ labelled by $\bigvee$ that receives inputs from all the gates in the $\{i, j\}$th block from the first level. Finally we want to take the conjunction of all these $\binom{k}{2}$ $\bigvee$-gates from the second level. We do this by adding $\lceil \log \binom{k}{2} \rceil$ levels containing $\leq \binom{k}{2}$ gates of fan-in 2, all labelled by $\wedge$. Hence, D has weft 1 and depth $\leq 2 + \lceil \log k(k-1) \rceil$.

By the construction, every assignment satisfying all $\bigvee$-gates of D must set at least one variable in $h^{-1}(i)$ to TRUE for all $i \in [k]$. Hence, one easily verifies the equivalence (9).

Finally, we pass from the Boolean circuit D to an equivalent $c_{\leq}$-circuit D' of the same weft and depth using the equivalences (7) and (8). □

Proof of Theorem 22 Immediate from the Lemmas 23, 24 and 25. □

Remark 26 The very same argument shows that Theorem 22 remains true if we restrict the problem p-WSAT($c_{\leq}$, f) to *pure* $c_{\leq}$-circuits.

A W[SAT]-*Complete Problem* The weighted satisfiability problem for arbitrary $\{c_{\leq}\}$-circuits turns out to be W[P]-complete (see Theorem 21). To obtain a W[1]-complete problem, we considered circuits where the depth was bounded in terms of the parameter; for W[SAT] we have to consider $\{c_{\leq}\}$-circuits where the depth is logarithmic in the circuit size. We used the lower case letter f for functions bounding the depth in terms of the parameter; we shall use the capital letter F for functions bounding the depth in terms of the circuit size. Recall that a *small* circuit is a circuit without large gates. We set

p-WSAT(small $c_{\leq}$; F)
Input: $k \in \mathbb{N}$ and a small $\{c_{\leq}\}$-circuit D of depth $\leq F(\|D\|)$.
Parameter: k.
Question: Does D have a satisfying assignment of weight k?

We aim at:

Theorem 27 *For sufficiently large d the problem p-*WSAT(small $c_{\leq}$; $d \cdot \log n$) *is* W[SAT]*-complete under fpt-reductions.*

Here, by p-WSAT(small $c_{\leq}$; $d \cdot \log n$) we mean p-WSAT(small $c_{\leq}$; F) for $F(n) := d \cdot \log n$.

The theorem is proven with the following lemmas.

Lemma 28 *For all $d \in N$ we have p-*WSAT(small $c_{\leq}$; $d \cdot \log n$) $\in$ W[SAT].

Proof Let $d \in \mathbb{N}$. Furthermore let D be a small $\{c_{\leq}\}$-circuit with depth $\leq d \cdot \log \|D\|$. Bottom up, we define for every gate g a Boolean propositional formula α_g equivalent to the induced subcircuit of D with output gate g. For gates g of fan-in 0 we let α_g be its label. If the gate g has fan-in r and receives incoming edges from the gates $g_1, \dots, g_r$ with $1 \leq r < 2(c+1)$ we let

$$\alpha_g := \bigvee_{1 \leq i_1 < \cdots < i_c \leq r} (\alpha_{g_{i_1}} \wedge \ldots \wedge \alpha_{g_{i_c}}). \tag{10}$$

Then $\alpha_{g_o} \equiv D$ for the output gate g_o of D.

Clearly, the time needed to obtain α_{g_o} is polynomial in $\|\alpha_{g_o}\|$. As $r < 2(c+1)$ one easily verifies that there is a constant M (depending on c) such that $\|\alpha_{g_o}\| \leq M^{d \cdot \log \|D\|}$, a value polynomial in $\|D\|$. □

For the hardness proof we need the following "positive version" [14] of a result due to Spira [13].

Lemma 29 *There is a $d_0 \in \mathbb{N}$ and a polynomial time algorithm assigning to every small positive Boolean formula α an equivalent small positive Boolean formula α' of depth $\leq d_0 \cdot \log \|\alpha'\|$.*

Using the fact that the weighted satisfiability problem of small positive Boolean formulas is W[SAT]-complete [5, Theorem 13.7] we obtain from the previous lemma:

Lemma 30 *There is a $d_0 \in \mathbb{N}$ such that the weighted satisfiability problem of small positive Boolean formulas α of depth less than $d_0 \cdot \log \|\alpha\|$ is* W[SAT]*-complete under fpt-reductions.*

Replacing (small) $\wedge$-gates and $\vee$-gates by small $c_{\leq}$-gates according to (7) and (8), we get:

Lemma 31 *Let $c > 1$. For sufficiently large $d \in \mathbb{N}$ the problem p-*WSAT(small $c_{\leq}$; $d \cdot \log n$) *is* W[SAT]*-hard under fpt-reductions.*

Proof of Theorem 27: Immediate by Lemma 28 and Lemma 31. □

6 The Majority Connective

Recall from Sect. 3 that the majority connective *Maj* is defined so that

$$Maj[Y_1, \dots, Y_n]$$

gets the value TRUE if and only if more than half of the Y_i's have the value TRUE. We have seen that *Maj* is monotone and not bounded.

In this section we show that the W(*Maj*)-hierarchy contains the W-hierarchy levelwise and that the first levels coincide.

Because *Maj* is not bounded, the notion of small *Maj*-gate is not defined so far. As

$$(Y_1 \wedge Y_2) \equiv Maj[Y_1, Y_2], \tag{11}$$

$$(Y_1 \vee Y_2) \equiv Maj[\top, Y_1, Y_2], \tag{12}$$

it seems to be natural to identify *small Maj*-gates with *Maj*-gates of fan-in less than or equal to three.

Furthermore, we have:

$$\bigvee_{i\in[n]} Y_i \equiv Maj\Big[Y_1, \dots, Y_n, \underbrace{\top, \dots, \top}_{n \text{ times}}\Big]; \tag{13}$$

$$\bigwedge_{i\in[n]} Y_i \equiv Maj\Big[Y_1, \dots, Y_n, \underbrace{\bot, \dots, \bot}_{n-1 \text{ times}}\Big]. \tag{14}$$

Finally, let the variables $Y_1, \dots, Y_n$ be pairwise distinct and $i \in [n]$. Then an easy computation shows that for assignments of weight k to these variables, where $2k \le n$, the formulas

$$\neg Y_i \quad \text{and} \quad Maj\Big[Y_1, \dots, Y_{i-1}, Y_{i+1}, \dots, Y_n, \underbrace{\top, \dots, \top}_{n-1-2k+2 \text{ times}}\Big] \tag{15}$$

are equivalent. In general, the *Maj*-gate in the formula on the right hand side will be large.

Theorem 32 W[P] = W[P](*Maj*).

Proof We get W[P] ⊆ W[P](*Maj*) by (11)–(15) and the fact that on the Boolean side we can restrict ourselves to instances (D, k), where

- $\neg$-gates only appear immediately above input variables in D;
- $2k \le n$, where n is the number of input variables of D.

The inclusion W[P](*Maj*) ⊆ W[P] follows from Proposition 7. □

Theorem 33 *For all* $t \ge 1$

$$W[t] \subseteq W[t](Maj).$$

We prove this result in three steps: in the first step, we prove the claim for even t (Lemma 34); in the second step, for odd $t > 1$ (Lemma 35); finally, we treat the special case of $t = 1$ (Lemma 36). Recall from Sect. 1 the definition of the classes Π_t^+ and Π_t^- of circuits (in fact, of propositional formulas).

Lemma 34 *Let t be even. Then p-WSAT$(\Pi_t^+) \in \mathrm{W}[t](Maj)$.*

Proof Let t be even. Recall that Π_t^+-formulas have the form

$$\bigwedge_{i_1} \bigvee_{i_2} \bigwedge_{i_3} \cdots \bigvee_{i_t} Y_{\bar{\imath}}$$

with variables $Y_{\bar{\imath}}$. We obtain an equivalent formula in $\Omega_{t,t}(Maj)$ by inductively replacing all $\bigvee$ and $\bigwedge$ according to (13) and (14), respectively. This gives a reduction of p-WSAT(Π_t^+) to p-WSAT$(\Omega_{t,t}(Maj))$ and hence the former problem is in $\mathrm{W}[t](Maj)$. □

Lemma 35 *Let $t > 1$ be odd. Then p-WSAT$(\Pi_t^-) \in \mathrm{W}[t](Maj)$.*

Proof Let $t > 1$ be odd. Let $\alpha \in \Pi_t^-$ and $k \in \mathbb{N}$, say

$$\alpha = \bigwedge_{i_1} \bigvee_{i_2} \bigwedge_{i_3} \cdots \bigvee_{i_{t-1}} \beta_{\bar{\imath}} \tag{16}$$

with formulas $\beta_{\bar{\imath}} \in \Pi_1^-$, i.e., each $\beta_{\bar{\imath}}$ is a large conjunction of negated variables. Let n be the number of variables in α.

In a first step, we replace these $\beta_{\bar{\imath}}$'s as follows (later we will replace the $\bigwedge$ and the $\bigvee$ as we did it in the previous proof). Let $X_1^{\bar{\imath}}, \dots, X_{n_{\bar{\imath}}}^{\bar{\imath}}$ be the variables of α which do *not* occur in $\beta_{\bar{\imath}}$. We may assume that $n_{\bar{\imath}} \geq 2k$ for all $\bar{\imath}$. This assumption can be enforced by the following preprocessing in fpt-time. If some $n_{\bar{\imath}}$ is smaller than $2k$ this means that $\beta_{\bar{\imath}}$ is the conjunction of the negations of more than $n - 2k$ of the n variables. For each weight k assignment to the remaining variables, we check whether combining it with 0's for the other variables satisfies α. The number of these assignments is bounded in terms of the parameter k. If we find a satisfying assignment, then (α, k) is a "yes"-instance, otherwise we know that no weight k-assignment satisfying α satisfies $\beta_{\bar{\imath}}$. We then delete $\beta_{\bar{\imath}}$ from α.

We set

$$\gamma_{\bar{\imath}} := Maj\Big(X_1^{\bar{\imath}}, \dots, X_{n_{\bar{\imath}}}^{\bar{\imath}}, \underbrace{\top, \dots, \top}_{n_{\bar{\imath}}-2k+1 \text{ times}}\Big) .$$

Note that $n_{\bar{\imath}} - 2k + 1$ is the smallest number m such that $k + m > (n_{\bar{\imath}} + m)/2$. This means that an assignment $\mathcal{V}$ to the variables of α of weight k satisfies $\gamma_{\bar{\imath}}$ if and only if $\mathcal{V}$ satisfies k variables not occurring in $\beta_{\bar{\imath}}$, that is, if and only if $\mathcal{V}$ satisfies $\beta_{\bar{\imath}}$. Thus with respect to assignments of weight k the formula in (16) and the formula

$$\bigwedge_{i_1} \bigvee_{i_2} \bigwedge_{i_3} \cdots \bigvee_{i_{t-1}} \gamma_{\bar{\imath}}$$

are equivalent. We can replace the $\bigwedge$ and the $\bigvee$ as we did it in the previous proof using (14) and (13). □

In order to prove the claim of Theorem 33 for $t = 1$ we introduce the parameterized problem

p-MAJORITY-VERTEX-COVER
Input: A graph $G = (V, E)$ and $k \in \mathbb{N}$.
Parameter: k.
Question: Is there a set of k vertices in G which covers a majority of the edges of G, i.e., is there $S \subseteq V$ with $|S| = k$ and $|\{e \in E \mid e \cap S \neq \emptyset\}| > |E|/2$?

Even though the parameterized vertex cover problem is fixed-parameter tractable, we will show the W[1]-completeness of this variant.

Lemma 36 *The parameterized problem* p-MAJORITY-VERTEX-COVER *is*

(a) *contained in* W[1](*Maj*), *and*
(b) W[1]-*hard under fpt-reductions*.

Proof Let (G, k) be an instance of p-MAJORITY-VERTEX-COVER. We construct a *Maj*-circuit D of weft 1 and depth 2 such that

$$(G, k) \in p\text{-MAJORITY-VERTEX-COVER} \iff D \text{ is } k\text{-satisfiable.} \tag{17}$$

Let $G = (V, E)$ and $n = |V|$ and $m = |E|$. The set V together with an additional gate will be the set of input gates of D; vertex v is labelled by a variable X_v and the additional gate by $\top$. Beyond these input gates we have a level of *Maj*-gates of fan-in 3, one for each edge $e \in E$. The gate associated with the edge $e = \{u, v\}$ is connected to the gates u, v and the gate labelled by $\top$. Note that all these *Maj*-gates are small and each of them is satisfied by an assignment if and only if the edge associated with this gate is incident to at least one vertex selected by the assignment. Finally the circuit D has as output a further *Maj*-gate which receives its input from all the small *Maj*-gates.

One easily verifies that (17) holds and thus $(G, k) \mapsto (D, k)$ is an fpt-reduction from p-MAJORITY-VERTEX-COVER to p-WSAT($\Omega_{1,2}$(*Maj*)); hence the former problem is in W[1](*Maj*).

We prove part (b) by reducing the W[1]-complete parameterized independent set problem p-INDEPENDENT-SET to p-MAJORITY-VERTEX-COVER.

Let (G, k) be an instance of p-INDEPENDENT-SET. We may assume that $k + 1 < n/4$, since otherwise the trivial brute-force algorithm runs in fpt-time. We construct an equivalent instance (G', k') of p-MAJORITY-VERTEX-COVER as follows. Let $n = |V|$ and for $v \in V$ let $d^G(v)$ denote the number of vertices adjacent to v in G. The graph $G' = (V', E')$ is constructed from G in two steps. First, for every $v \in V$, we add $n - 1 - d(v)$ new vertices, which are connected only to v and hence have degree one. Let m_0 be the number of edges of the graph obtained so far. Clearly

$$m_0 \geq \frac{n \cdot (n-1)}{2}. \tag{18}$$

We choose the smallest $s \in \mathbb{N}$ such that

$$k \cdot (n-1) + s > \frac{m_0 + s}{2}. \tag{19}$$

In the second step we add a vertex v^* and s further vertices to our graph and make v^* adjacent to the s many new vertices, which thus all have degree one. This finishes the construction of the graph G'. Note that G' has $m_0 + s$ many edges. We set $k' := k + 1$. We show that

$$(G, k) \in p\text{-Independent-Set} \iff (G', k') \in p\text{-Majority-Vertex-Cover}.$$

Assume first that S is an independent set of G of size k. Then $S \cup \{v^*\}$ covers $k \times (n-1) + s$ edges, which by (19) is more than half of the edges of G'.

Conversely, let S' be a subset of k' vertices in G', which cover more than $(m_0 + s)/2$ edges in G'. Then $v^* \in S'$, since all other vertices have degree at most $n-1$ and therefore at most $(k+1) \cdot (n-1)$ edges can be covered otherwise. However, as $k + 1 < n/4$ we get $(k+1) \cdot (n-1) < n \cdot (n-1)/4 \leq m_0/2$ (the last inequality holding by (18)); therefore, at most half of the edges would be covered. We set $S := S' \setminus \{v^*\}$. Thus $|S| = k$ and by the choice of s the set S must cover in G' at least $k \cdot (n-1)$ edges. As the vertices in V have degree at most $n-1$ in G' and the vertices in $V' \setminus (V \cup \{v^*\})$ have degree one, we see that $S \subseteq V$. Moreover, in order to cover $k \cdot (n-1)$ edges, S must be an independent set of G. □

Proof of Theorem 33: Immediate by Lemmas 34–36, as we know that p-WSat(Π_t^+) is W[t]-complete for even t and p-WSat(Π_t^-) is W[t]-complete for odd $t > 1$. □

While we conjecture that W[t] ⊂ W[t](Maj) for $t > 1$, we can show:

Theorem 37 W[1] = W[1](Maj).

Proof By the previous lemma, we have to show that W[1](Maj) ⊆ W[1]. We start with an observation. For any Maj-circuit D let Min(D) be the set of ⊆-minimal assignments $\mathcal{V}$ satisfying D (we identify an assignment $\mathcal{V}$ with the set of variables it sets to 1).

As majority gates are monotone, we know that

$$D \equiv \bigvee_{\mathcal{V} \in \mathrm{Min}(D)} \bigwedge \mathcal{V}. \tag{20}$$

Now fix $d \geq 1$. We show that p-WSat($\Omega_{1,d}(Maj)$) ∈ W[1], which yields our claim.

Let $D \in \Omega_{1,d}(Maj)$ and let us first assume that the weft 1 of D comes from a single large majority gate at the output. Let us call such circuits *simple*. Assume that the output gate of such a simple $D \in \Omega_{1,d}(Maj)$ has fan-in ℓ and receives incoming edges from gates $g_1, \ldots, g_\ell$. We denote by $D_1, \ldots, D_\ell$ the subcircuits of D with output gates $g_1, \ldots, g_\ell$, respectively. Then the D_is are small *Maj*-circuits of depth $< d$; in particular, they have $\le 1 + 3^{d-1}$ gates. Therefore there are constants $c', c'' \in \mathbb{N}$ depending only on d such that for each D_i we know

$$|\mathrm{Min}(D_i)| \le c'' \quad \text{and} \quad |\mathcal{V}| \le c' \quad \text{for each } \mathcal{V} \in \mathrm{Min}(D_i).$$

An assignment $\mathcal{V}$ satisfies the circuit D if and only if

$$N(\mathcal{V}) := |\{i \in [\ell] \mid \mathcal{V} \text{ satisfies } D_i\}| > \ell/2.$$

According to (20) the assignment $\mathcal{V}$ satisfies D_i if and only if $\mathcal{V}$ is a superset of some $\mathcal{V}' \in \mathrm{Min}(D_i)$. Let $\mathcal{V}^{[\le c']}$ be the set of assignments of weight $\le c'$ which are subsets of $\mathcal{V}$, i.e. $\mathcal{V}^{[\le c']} := \{\mathcal{V}' \subseteq \mathcal{V} \mid |\mathcal{V}'| \le c'\}$. Hence, again by monotonicity,

$$N(\mathcal{V}) = \Bigg| \bigcup_{\mathcal{V}' \in \mathcal{V}^{[\le c']}} \underbrace{\{i \in [\ell] \mid \mathcal{V}' \in \mathrm{Min}(D_i)\}}_{=:\mathrm{Sat}(\mathcal{V}')} \Bigg|.$$

Let $\mathcal{V}^{[\le c']}$ contain r elements, say, $\mathcal{V}^{[\le c']} = \{\mathcal{V}_1, \ldots, \mathcal{V}_r\}$. Applying the inclusion-exclusion principle we get

$$\begin{aligned} N(\mathcal{V}) &= \sum_{j=1}^{r} (-1)^{j+1} \cdot \sum_{i_1 < \cdots < i_j} \left|\mathrm{Sat}(\mathcal{V}_{i_1}) \cap \cdots \cap \mathrm{Sat}(\mathcal{V}_{i_j})\right| \\ &= \sum_{j=1}^{c''} (-1)^{j+1} \cdot \sum_{i_1 < \cdots < i_j} \left|\mathrm{Sat}(\mathcal{V}_{i_1}) \cap \cdots \cap \mathrm{Sat}(\mathcal{V}_{i_j})\right|, \end{aligned}$$

where the second equality holds since $|\mathrm{Min}(D_s)| \le c''$ for all $s \in [\ell]$, so that the intersection of more than c'' many $\mathrm{Sat}(\mathcal{V}_i)$ will be the empty set.

Let m be the number of variables of D. Then there are at most $m^{c'}$ assignments of weight $\le c'$ to these variables and hence there are at most $O(m^{c' \cdot c''})$ intersections of the form $\mathrm{Sat}(\mathcal{V}_{i_1}) \cap \cdots \cap \mathrm{Sat}(\mathcal{V}_{i_j})$, where $j \in [c'']$ and $\mathcal{V}_1, \ldots, \mathcal{V}_j$ are distinct assignments of weight $\le c'$. Thus this number is polynomial in the size of D.

Hence, the following nondeterministic algorithm $\mathbb{A}$ solves the weighted satisfiability problem for simple circuits in $\Omega_{1,d}(Maj)$. Let (D, k) be an instance with simple D. First $\mathbb{A}$ computes all these intersections just mentioned and stores all the numbers $|\mathrm{Sat}(\mathcal{V}_1) \cap \cdots \cap \mathrm{Sat}(\mathcal{V}_j)|$. Then $\mathbb{A}$ nondeterministically guesses an assignment $\mathcal{V}$ of weight k and using the stored information computes $N(\mathcal{V})$ in time bounded in terms of the parameter k. If $N(\mathcal{V}) > \ell/2$ (as above, ℓ is the fan-in of the output gate of D), then $\mathbb{A}$ accepts, and rejects otherwise.

In the case where D is not simple, D is equivalent to a Boolean combination of simple subcircuits such that this combination has no large Boolean gates and is of size bounded in terms of d. In this case the algorithm $\mathbb{A}$, in its first phase, in addition computes and stores all assignments of the output nodes of the simple subcircuits that lead to an assignment satisfying D and then essentially proceeds as in the preceding case.

To finish the proof, again there are two options. The reader familiar with the characterization of W[1] in terms of nondeterministic random access machines will easily see that the algorithm $\mathbb{A}$ can be simulated by a program for such a machine. The second option: it is not hard using algorithm $\mathbb{A}$ to construct an fpt-reduction of p-WSAT($\Omega_{1,d}(Maj)$) to the parameterized short halting problem for nondeterministic single-tape Turing machines, a problem in W[1]. □

Corollary 38 *p-MAJORITY-VERTEX-COVER is* W[1]*-complete under fpt-reductions.*

Pure Majority Circuits In this paragraph we ask whether we can restrict ourselves to pure circuits when considering weighted satisfisfiability problems for *Maj*-circuits.

We always can eliminate the Boolean constant $\top$ as follows: Let D be a *Maj*-circuit containing an input gate labelled with $\top$. We may assume that there is exactly one such gate. Let Z be a new variable. Let D' be the *Maj*-circuit obtained from D by replacing the label $\top$ by Z and by adding on top of D the output gate of D', a new (small) *Maj*-gate of in-degree two connected to the output gate of D and to the input gate now labelled by Z. Then, for any $k \in \mathbb{N}$, we have that D is k-satisfiable if and only if D' is $k+1$-satisfiable. The circuit D' has the same weft as D and a depth increased by one.

We can also eliminate the occurrence of $\bot$ in D but thereby increasing the weft by one: Let $Z_1, \ldots, Z_n$ be the variables of D and let k be such that $2k \le n$ (we can assume this without loss of generality when considering k-satisfiability). Then we can replace every occurrence of $\bot$ by $Maj[Z_1, \ldots, Z_n]$ without changing k-satisfiability.

In view of Theorem 32 we have shown:

Theorem 39 *The weighted satisfiability problem for pure Maj-circuits is* W[P]*-complete under fpt-reductions.*

In the proof of part (a) of Lemma 36 we presented a reduction of the parameterized problem p-MAJORITY-VERTEX-COVER to p-WSAT($\Omega_{1,2}(Maj)$). For instances of p-MAJORITY-VERTEX-COVER we obtained circuits in $\Omega_{1,2}(Maj)$ containing $\top$ but not containing $\bot$. Hence, by the preceding observation:

Theorem 40 *The weighted satisfiability problem for pure circuits in $\Omega_{1,3}(Maj)$ is* W[1]*-complete under fpt-reductions.*

Fix $t > 1$. We do not know whether for some $d \in \mathbb{N}$ the weighted satisfiability problem for pure circuits in $\Omega_{t,d}(Maj)$ is W[t](Maj)-complete under fpt-reductions. However, we can show W[t]-hardness of the problem:

Proposition 41 *For all $t > 1$, the weighted satisfiability problem for pure circuits in $\Omega_{t,t+1}(Maj)$ is* W[t]*-hard under fpt-reductions.*

Proof Let $t > 1$ and let α be in $\Pi^+_{t,1}$ or in $\Pi^-_{t,1}$ depending on whether t is even or odd respectively. We can assume that α does not contain $\top$ and $\bot$. The construction in the proofs of Lemmas 34 and 35 produces a majority formula in $\Omega_{t,t}(Maj)$ containing both $\top$ and $\bot$. We have already seen how to move to a formula in $\Omega_{t,t+1}(Maj)$ without $\top$. The occurrences of $\bot$ are due to the simulation of big conjunctions according to the equivalence (14). We now show an alternative way to simulate big conjunctions without using $\bot$.

Let ℓ be the maximum fan-in of any gate in α. Let $Y_1, \ldots, Y_{\ell\cdot(k+1)-1}$ be new variables. Let $\bigwedge_{i\in[\ell']}\alpha_i$ with $\ell' \le \ell$ be a conjunction in α. Then $\bigwedge_{i\in[\ell']}\alpha_i$ has a satisfying assignment of weight k if and only if so does the formula

$$Maj\Big[\underbrace{\alpha_1,\ldots,\alpha_1}_{k+1\text{ times}},\underbrace{\alpha_2,\ldots,\alpha_2}_{k+1\text{ times}},\ldots,\underbrace{\alpha_{\ell'},\ldots,\alpha_{\ell'}}_{k+1\text{ times}},Y_1,\ldots,Y_{\ell'\cdot(k+1)-1}\Big].$$

Why? An assignment satisfying all $\alpha_1, \ldots, \alpha_{\ell'}$ clearly satisfies this formula. Conversely if a weight k assignment does not satisfy all $\alpha_1, \ldots, \alpha_{\ell'}$ then the above formula has at most $(\ell'-1)(k+1)$ satisfied arguments plus possibly some from the new variables, but at most k. In total no more than $(\ell'-1)(k+1)+k$ arguments are satisfied and this is less than half the total number of arguments, which is $2\ell'(k+1)-1$.

Let $\beta \in \Omega_{t,t+1}(Maj)$ be the pure majority circuit obtained in this way. Then an assignment of weight k to the variables of β satisfies β if and only if the restriction to the variables of α satisfies α.

It follows that α has a satisfying weight k assignment to its variables if and only if β has a satisfying weight k assignment to its variables. Why? To see necessity extend an assignment to the variables of α by mapping all new variables $Y_1, Y_2, \ldots$ to 0 and use the above equivalence. Conversely, a satisfying weight k assignment for β restricts to one of weight at most k satisfying α by the above equivalence; but observe that α is monotone and hence has a satisfying assignment of weight k if and only if it has a satisfying assignment of weight at most k. □

7 Open Problems

We have seen (cf. Corollary 6) that in revisiting the original definition of the W-hierarchy by means of Boolean circuits (*large* and *small*), we can explore the concept of *circuit weft* (or more simply, *large gate depth*, in the context of overall bounded depth) by considering gates labelled by arbitrary bounded connectives. The majority connective is not bounded, nevertheless the first level of the corresponding hierarchy coincides with W[1]. For higher levels we only know that $W[t] \subseteq W[t](Maj)$. There are various open questions related to the question whether these classes are distinct. For example, let us consider the majority versions of the W[2]-complete parameterized dominating set problem p-DS and of the hitting set problem p-HS.

p-MAJ-DS
Input: A graph G and $k \in \mathbb{N}$.
Parameter: k.
Question: Does G have a set of k vertices dominating the majority of vertices?

p-MAJ-HS

Input: A hypergraph H and $k \in \mathbb{N}$.

Parameter: k.

Question: Does H have a set of k vertices hitting the majority of hyperedges?

Here a set S of vertices in a graph $G = (V, E)$ *dominates* a vertex u if $u \in S$ or there is a $v \in S$ such that $\{u, v\} \in E$. A set S of vertices in a hypergraph $H = (V, E)$ *hits* an hyperedge $e \in E$ if $S \cap e \neq \emptyset$.

What we know about the relationship between these problems is indicated by:

Theorem 42

(a) p-DS $\le^{\mathrm{fpt}}$ p-MAJ-DS $\equiv^{\mathrm{fpt}}$ p-MAJ-HS.
(b) p-MAJ-HS $\in$ W[2](Maj).

Proof (a) p-DS $\le^{\mathrm{fpt}}$ p-MAJ-DS: Let (G, k) be an instance of p-DS and $G = (V, E)$ with $n := |V|$. Let G' be the graph obtained from G by adding $n - 1$ isolated vertices. Then for $k \le |V|$

$$(G, k) \in p\text{-DS} \quad \Longleftrightarrow \quad (G', k) \in p\text{-MAJ-DS}.$$

In fact, if S is a dominating set of G of size k, then S dominates n vertices of G' and hence the majority of vertices. Conversely, if a set S of vertices of size k dominates the majority of vertices of G' and contains ℓ of the added isolated vertices, then at most ℓ vertices of G are not dominated by S. Replacing in S the ℓ isolated vertices by these ones we obtain a dominating set of G of size $\le k$.

p-MAJ-DS $\le^{\mathrm{fpt}}$ p-MAJ-HS: Let $G = (V, E)$ be a graph. For every $v \in V$ let v^* be a copy of v. Let $H = (V', E')$ be the hypergraph with

$$V' := V \cup \{v^* \mid v \in V\} \quad \text{and} \quad E' := \{N^*(v) \mid v \in V\},$$

where

$$N^*(v) := \{v, v^*\} \cup \{u \in V \mid \{u, v\} \in E\}.$$

We added the points v^* to ensure that $|E'| = |V|$. We show that

$$(G, k) \in p\text{-MAJ-DS} \quad \Longleftrightarrow \quad (H, k) \in p\text{-MAJ-HS}.$$

Note that a set S of vertices of G dominates a vertex $v \in V$ if and only if it hits $N^*(v)$ in the hypergraph H. This yields the direction from left to right. Conversely, assume that a subset S' of V' hits the majority of hyperedges in H. Then so does the set obtained from S' by replacing vertices of the form v^* by v. This yields the other direction.

p-MAJ-HS $\le^{\mathrm{fpt}}$ p-MAJ-DS: The proof is similar, though a little bit more involved, to that presented in [8, Example 2.7] for the non-majority versions of the problems.

Let (H,k) with $H=(V,E)$ be an instance of p-MAJ-HS. We may assume that $|V|\geq k$ and that E contains at least two nonempty hyperedges. Let $n:=|V|$ and $m:=|E|$. We introduce the graph

$$G=(V\cup V^*\cup E, E_1\cup E_2),$$

where $V^*:=\{v^*\mid v\in V\}$ is a disjoint copy of V. Furthermore, $E_1:=\{\{v,e\}\mid v\in V, e\in E, v\in e\}$ and E_2 contains edges between all pairs of distinct vertices of V, that is, $E_2:=\{\{v,w\}\mid v,w\in V,\ v\neq w\}$. Thus G has $2n+m$ vertices and the vertices in V^* are isolated. Then

$$(H,k)\in p\text{-MAJ-HS} \iff (G,k)\in p\text{-MAJ-DS}, \tag{21}$$

which yields our claim. First observe that for $S\subseteq V$ we have:

$$\begin{aligned} &S \text{ hits more than } m/2 \text{ (i.e., the majority) of hyperedges of } H\\ \iff\quad &S \text{ dominates more than } n+m/2 \text{ (i.e., the majority) of vertices of } G. \end{aligned}$$

This yields the direction from left to right in (21). We call sets of vertices of G dominating the majority of vertices of G *good*. For the other direction in (21), let S be a good set of vertices of G of size k. If $S\cap V$ hits more than half of the hyperedges in H, we are done (recall that $|V|\geq k$). (In particular, by the equivalence above, this holds if $S\subseteq V$.) Otherwise, as long as this is not the case and $S\cap V^*\neq\emptyset$, we replace every vertex of V^* in S by some vertex of V contained in a hyperedge of H, which is not hit by $S\cap V$ so far. We may thus assume that $S\subseteq V\cup E$ and S is good. If then $S\cap V$ does not hit more than half of the hyperedges of H, we further change S in order to achieve $S\subseteq V$: Assume $e\in E\cap S$. We show that we can replace e in S by a vertex of V. The vertex e of G only has edges to the elements of e. Therefore, for every $v\in e$, the set $S_v:=(S\setminus\{e\})\cup\{v\}$ is good, too. If $v\notin S$ for some $v\in e$, then the corresponding S_v has cardinality k, and we replace S by S_v. If $e\subseteq S$, then we add to $S\setminus\{e\}$ a vertex $v\in V$ from some hyperedge of H not hit so far (here, in case $e=\emptyset$ we need the assumption that E contains at least two nonempty hyperedges).

(b) p-MAJ-HS $\in$ W[2](*Maj*): Let $H=(V,E)$ be a hypergraph. We construct a *Maj*-circuit D with input variables Y_v for $v\in V$ such that for every subset S of V we have

$$\begin{aligned} &S \text{ hits the majority of hyperedges of } H\\ \iff\quad &\text{the assignment } \{Y_v\mid v\in S\} \text{ satisfies } D. \end{aligned}$$

The output gate of D is a large majority gate. For every hyperedge $e\in E$ the output gate receives an input edge from a gate labelled by $\bigvee$, which itself has incoming edges from the input gates labelled by Y_v with $v\in e$. As the $\bigvee$-gates can be replaced by majority gates according to (13), the circuit D is (equivalent to) a *Maj*-circuit. □

We close by mentioning two open problems in connection with the previous theorem explicitly:

- Is p-MAJ-DS $\leq^{\text{fpt}}$ p-DS? Is p-MAJ-DS contained in some level W[t] of the W-hierarchy?
- Is p-MAJ-HS hard under fpt-reductions for the class W[2](*Maj*)?

8 Summary

Originally, the W-hierarchy is defined by weighted satisfiability problems for certain Boolean circuits by stepwise increasing their weft. This definition makes sense for different choices of connectives. For various sets of not necessarily Boolean connectives $\mathcal{C}$ we introduced a W($\mathcal{C}$)-hierarchy and compared it with the original W-hierarchy.

We studied the hierarchies for three groups of connectives $\mathcal{C}$. For the first group, sets of bounded connectives satisfying some further property, we showed that the levels of the W-hierarchy and the W($\mathcal{C}$)-hierarchy coincide levelwise.

As a second group of connectives we studied threshold connectives $c_{\leq}$ and showed that their weighted satisfiability problem is W[1]-complete for circuits whose depth is bounded in terms of the parameter, W[SAT]-complete for small circuits whose depth is logarithmic in the input size and W[P]-complete when no restriction is imposed.

Finally, we studied the majority connective and showed first that the corresponding W(*Maj*)-hierarchy contains the W-hierarchy levelwise and second that the first levels coincide. The last result implies that a majority version of the parameterized vertex cover problem is W[1]-complete. We conjecture that $\mathrm{W}[t] \subsetneq \mathrm{W}[t](\mathit{Maj})$ for $t > 1$.

Acknowledgements We would like to thank the anonymous reviewers. Their detailed comments have significantly improved our presentation.

References

1. Abrahamson, K.A., Downey, R.G., Fellows, M.R.: Fixed-parameter tractability and completeness IV: on completeness for W[P] and PSPACE analogs. Ann. Pure Appl. Log. **73**, 235–276 (1995)
2. Cai, L., Chen, J., Downey, R.G., Fellows, M.R.: On the parameterized complexity of short computation and factorization. Arch. Math. Log. **36**, 321–337 (1997)
3. Chen, Y., Flum, J.: The parameterized complexity of maximality and minimality problems. Ann. Pure Appl. Log. **151**, 22–61 (2007)
4. Chen, Y., Flum, J., Grohe, M.: Machine-based methods in parameterized complexity theory. Theor. Comput. Sci. **339**, 167–199 (2005)
5. Downey, R.G., Fellows, M.R.: Parameterized Complexity. Springer, Berlin (1999)
6. Downey, R.G., Fellows, M., Langston, M.A. (guest eds.): Special Issue on Parameterized Complexity, Part I and Part II. Comput. J. **51**(1, 3) (2008)
7. Downey, R.G., Fellows, M.R., Taylor, U.: The parameterized complexity of relational database queries and an improved characterization of W[1]. In: Bridges, D.S., Calude, C., Gibbons, P., Reeves, S., Witten, I.H. (eds.) Combin. Complex. Log. **39**, 194–213 (1996)
8. Flum, J., Grohe, M.: Parameterized Complexity Theory. Springer, Berlin (2006)
9. Marx, D.: Parameterized complexity of independence and domination on geometric graphs. In: Proceedings of the Second International Workshop on Parameterized and Exact Computation (IWPEC). LNCS, vol. 4169, pp. 154–165. Springer, Berlin (2006)
10. Melhorn, K.: Data Structures and Algorithms, Vol. 2: NP-Completeness and Graph Algorithms. EATCS Monographs on Theoretical Computer Science. Springer, Berlin (1984)
11. Niedermeier, R.: Invitation to Fixed-Parameter Algorithms. Oxford University Press, London (2006)

12. Sipser, M.: Borel sets and circuit complexity. In: Proc. of the 15th Annual ACM Symposium on the Theory of Computing, pp. 61–69 (1983)
13. Spira, P.M.: On time hardware complexity tradeoffs for Boolean functions. In: Proceedings of the Fourth Hawaii International Symposium on System Sciences, pp. 525–527 (1971)
14. Wegener, I.: Relating monotone formula size and monotone depth of Boolean functions. Inf. Process. Lett. **16**, 41–42 (1983)