CDMTCS
Research
Report
Series

Parameterized Circuit Complexity and the W Hierarchy

Rodney G. Downey
Department of Mathematics
Victoria University
Wellington, New Zealand

Michael R. Fellows
Department of Computer Science
Univeristy of Victoria
Victoria, B.C. Canada

Kenneth W. Regan
Department of Computer Science
State University of New York at Buffalo
Buffalo, NY, U.S.A.

CDMTCS-049
August 1997
Centre for Discrete Mathematics and
Theoretical Computer Science

Abstract

A parameterized problem $\langle L, k\rangle$ belongs to $W[t]$ if there exists k' computed from k such that $\langle L, k\rangle$ reduces to the weight-k' satisfiability problem for *weft-t* circuits. We relate the fundamental question of whether the $W[t]$ hierarchy is proper to parameterized problems for constant-depth circuits. We define classes $G[t]$ as the analogues of AC^0 depth-t for parameterized problems, and $N[t]$ by weight-k' existential quantification on $G[t]$, by analogy with $\mathrm{NP} = \exists \cdot \mathrm{P}$. We prove that for each t, $W[t]$ equals the closure under fixed-parameter reductions of $N[t]$. Then we prove, using Sipser's results on the AC^0 depth-t hierarchy, that both the $G[t]$ and the $N[t]$ hierarchies are proper. If this separation holds up under parameterized reductions, then the $W[t]$ hierarchy is proper.

We also investigate the hierarchy $H[t]$ defined by alternating quantification over $G[t]$. By trading weft for quantifiers we show that $H[t]$ coincides with $H[1]$. We also consider the complexity of unique solutions, and show a randomized reduction from $W[t]$ to Unique $W[t]$.

1 Parameterized Problems and the W Hierarchy

Many important and familiar problems have the general form

INSTANCE:	An object x, a number $k \geq 1$.
QUESTION:	Does x have some property Π_k that depends on k?

For example, the NP-complete CLIQUE problem asks: given an undirected graph G and natural number k, does G have a clique of size k? The VERTEX COVER and DOMINATING SET problems ask whether G has a vertex cover, respectively dominating set, of size k. Here k is called the *parameter*.

Formally, a *parameterized language* is a subset of $\Sigma^* \times \mathbf{N}$. A parameterized language A is said to be *fixed-parameter tractable*, and to belong to the class FPT, if there is a polynomial p, a function $f : \mathbf{N} \to \mathbf{N}$, and a Turing machine M such that on any input (x, k), M decides whether $(x, k) \in A$ within $f(k) \cdot p(|x|)$ steps. A is in *strongly uniform* FPT if the function f is computable. Note that if M runs in time polynomial in the length of (x, k) then it meets this condition with f computable. Examples of problems in FPT for which the only f are uncomputable are given in [DF93], while [DF95c] describes *natural* problems in FPT for which the only *known* f are not known to be computable.

The best known method for solving the parameterized CLIQUE problem is the algorithm of Nesetril and Poljak [NP85] that runs in time $O(n^{(\frac{2+\epsilon}{3})k})$, where $2+\epsilon$ represents the exponent on the time for multiplying two $n \times n$ matrices (best known is $2.376\ldots$, see [CW90]). For DOMINATING SET we know of nothing better than the trivial $O(n^{1+k})$-time algorithm that tries all vertex subsets of size k. VERTEX COVER, however, belongs to FPT, via a depth-first search algorithm that runs in time $2^k \cdot O(n)$ (see [DF95c]). Quite a few other NP-complete problems, with natural parameter k, are in FPT via algorithms of time $f(k) \cdot O(n)$ through $f(k) \cdot O(n^3)$, while many others treated in [DF95a] seem to be hard in the manner of CLIQUE and DOMINATING SET. The established way in complexity theory of comparing the hardness of problems is by formulating appropriate notions of *reducibility* and *completeness*. Here the former is provided by

Definition 1.1. A parameterized language A *FPT-many-one reduces* to a parameterized language B, written $A \leq_m^{fpt} B$, if there are a polynomial q, functions $f, g : \mathbf{N} \to \mathbf{N}$, and a Turing machine T such that on any input (x, k), T runs for $f(k) \cdot q(|x|)$ steps and outputs $(x', g(k))$ such that $(x, k) \in A \iff (x', g(k)) \in B$.

The reduction is *strongly uniform* if f is computable. Then (strongly uniform) FPT is closed downward under (strongly uniform) FPT reductions. Note that g is computable, and the parameter $k' = g(k)$ in the reduction does not depend on x.

For the completeness notion, Downey and Fellows [DF95a] defined a natural hierarchy of classes of parametrized languages

$$\text{FPT} \subseteq W[1] \subseteq W[2] \subseteq W[3] \subseteq \ldots \subseteq W[poly], \tag{1}$$

and showed that the parameterized version of CLIQUE is complete for $W[1]$ under FPT reductions, while that of DOMINATING SET is complete for $W[2]$. This gives a sense in which DOMINATING SET is apparently harder than CLIQUE. The formal definition of the W hierarchy is deferred to the next section, but the main idea can be seen by examining the logical definitions of CLIQUE and DOMINATING SET. For each k, the language of graphs with a clique of size k is defined by the existential formula

$$\phi_k := (\exists u_1 \ldots u_k) : \bigwedge_{i,j \leq k} E(u_i, u_j),$$

where $E(\cdot,\cdot)$ formalizes the adjacency relation for graphs. By contrast, the language of graphs with a dominating set of size k is requires two blocks of like quantifiers to define in first-order logic, such as by the Σ_2 formula

$$\psi_k := (\exists u_1 \ldots u_k) : \ (\forall v) \bigvee_{i \leq k} (v = u_i \ \vee \ E(v, u_i)).$$

Both problems are about searching for a set of vertices of size k that satisfy the condition following the ':', but in ψ_k the condition is more complex, because it has the extra quantifier over vertices v. Put another way, once candidate vertices have been assigned to $u_1, \ldots, u_k$, the condition for CLIQUE is entirely "local" in a sense studied for parameterized languages in [Reg89], while that for DOMINATING SET requires a "global" reference to other parts of the graph. Some parameterized problems on graphs have conditions that make several alternating first-order quantifications over the graph, and are known to belong to $W[t]$ only for higher values of t. Other problems have conditions that are not first-order definable at all, and some of these are complete for $W[poly]$ (see [ADF95, DFHKW94]). Intuitively, the question

Does $W[1] = W[2]$?

asks whether a local check of a fixed-size substructure can do the same work as a global check. The question

For all t, does $W[t] = W[2]$?

asks whether the simple check over vertices v in the $W[2]$-complete DOMINATING SET problem suffices to verify any condition that is definable by circuits of bounded *weft* t. Similarly, if $W[poly] = W[2]$ then fixed-parameter many-one reductions have an enormous power to simplify the checking of properties. Note that "k-slices" of VERTEX COVER have logical definitions of form similar to that of ψ_k and yet are fixed-parameter tractable. The parameterized versions of the NP-complete problems PERFECT CODE, SUBSET SUM, and SUBSET PRODUCT (see [GJ79, DF95a, FK93]) are known to belong to $W[2]$ and to be hard for $W[1]$, and are equivalent to each other under FPT reductions.

Earlier work [ADF95, DF93, DF95a] noted that if the W hierarchy is proper, or so long as FPT $\neq W[poly]$, then P $\neq$ NP. The paper [DF93] constructed a recursive oracle relative to which P $\neq$ NP and yet $W[poly]$ = FPT, so the above questions are in a sense stronger than P =?NP. Our results in this paper provide some evidence for a positive answer to the question,

Are all classes in (1) distinct?

We also compare the structure of the W hierarchy to that of the polynomial hierarchy. Our larger purpose is to examine how the W hierarchy can be characterized in ways that are important to other aspects of complexity theory.

We make the following progress on the above questions: First, each class $W[t]$ is shown to be definable via existential quantification on the class of parameterized languages recognizable by polynomial-sized circuits of constant *weft* t, analogous to the way NP is defined by existential quantification on P. The circuits we obtain are actually AC^0 circuits of depth t except for extra layers of gates of fan-in 2, and providing also for parameterization, we call them $G[t]$ circuits. In symbols we have $W[t] = \langle N[t] \rangle$, where $N[t] = \exists \cdot G[t]$. Also $W[poly] = N[poly] =_{\text{def}} \exists \cdot G[poly]$, without the closure notation. Then we show that not only is the $G[t]$ hierarchy proper, but more importantly the $N[t]$ hierarchy is proper. Thus among the three "elements" of the W hierarchy, namely parameterized languages, circuit weft, and FPT-reductions, only the last can be responsible for any collapse. We explain how these results rule out any "normal" argument for collapse of the W hierarchy, and give this as evidence that the hierarchy doesn't collapse.

Second, by analogy with the polynomial hierarchy, we define for each t a hierarchy $H[t]$ using alternating $\forall\cdot$ and $\exists$ quantification over $G[t]$. The hierarchy over $G[1]$ contains all levels of all of the hierarchies: For all $t > 1$, $H[t]$ equals $H[1]$.

Natural fixed-parameter analogues of the BP$\cdot$ and $\oplus$ operators on complexity classes can also be defined, and all of this raises questions about the relationships between classes defined by these operators. For example, it would be interesting to know if

$$\exists\cdot G[t] = N[t] \subseteq \mathrm{BP}\cdot\oplus G[t]$$

holds, which would be an analog of the Valiant-Vazirani lemma NP $\subseteq$ BP$\cdot\oplus$P. Although this remains an open problem, we show by similar techniques that there is a randomized reduction of $W[t]$ to Unique $W[t]$.

2 Parameterized Circuit Complexity and the $W[t]$ Classes

Boolean circuits are said to be of *mixed type* if they may contain both *small* gates of fan-in ≤ 2 and *large* AND and/or OR gates of unbounded fan-in. We consider only *decision circuits*; i.e., those with a single output gate. The *weft* of such a circuit is the maximum number of large gates on a path from an input to the output. The n inputs are labeled by variables $x_1, \ldots, x_n$, and the *Hamming weight* $wt(x)$ of an assignment $x \in \{0,1\}^n$ equals the number of bits that are set to 1. The circuit is *monotone* if it has no NOT gates, and *anti-monotone* if all wires from an input go to a NOT gate, and these are the only NOT gates in the circuit. A pure Σ_t circuit as defined by Sipser [Sip83] consists of t *levels* of large gates that alternate $\wedge$ and $\vee$ with a single $\vee$ gate at the top (i.e., the output), and with the bottom-level gates connected to the input gates $x_1, \ldots, x_n$ and their negations $\bar{x}_1, \ldots, \bar{x}_n$. A pure Π_t circuit is similarly defined with a large $\wedge$ gate at the output. In both cases, "pure" means that the circuit has no small gates. A Boolean expression is the same as a circuit in which each gate has fan-out 1. We call a Boolean expression *t-normalized* if it forms a pure Π_t circuit. For $t = 2$ this is the same as an expression in conjunctive normal form. For $t = 3$ this is product-of-sums-of-products (P-o-S-o-P) form; for $t = 4$ this is P-o-S-o-P-o-S form, and so on.

For all constants $h, t > 0$, the parameterized WEIGHTED CIRCUIT SATISFIABILITY problem is defined by:

$WCS(t,h)$	
INSTANCE:	A circuit C of weft t and overall depth $t+h$.
PARAMETER:	k.
QUESTION:	Does C accept some input of Hamming weight exactly k?

Then for all $t \geq 1$, $W[t]$ may be defined to be the class of parameterized languages A such that for some h, $A \leq_m^{fpt} WCS(t,h)$ (see [DF95a]). Also $W[poly]$ equals the class of problems that FPT many-one reduce to the problem WCS with no restriction on depth or weft. WCS is the parameterized version of the standard NP-complete CIRCUIT SATISFIABILITY problem, of which SAT is the specialization to the case where the circuit is a Boolean formula (in conjunctive normal form). An interesting aspect of the $W[\cdot]$ theory is that more-extreme special cases of the parameterized versions remain complete. For all $t \geq 2$ define:

WEIGHTED t-NORMALIZED BOOLEAN EXPRESSION SATISFIABILITY $\mathit{WBES}(t)$
INSTANCE: A t-normalized Boolean expression E.
PARAMETER: k.
QUESTION: Is there some assignment a of Hamming weight exactly k such that $E(a) = \mathit{true}$?

MONOTONE $\mathit{WBES}(t)$ ($\mathit{MWBES}(t)$)
Restriction of $\mathit{WBES}(t)$ to instances E that are *monotone*.

ANTI-MONOTONE $\mathit{WBES}(t)$ ($\mathit{AWBES}(t)$)
Restriction of $\mathit{WBES}(t)$ to instances E that are *anti-monotone*.

For $t = 1$, also define $\mathit{AWBES}(1,1)$ to be the restriction of $\mathit{WCS}(t,1)$ to instances consist of a single large AND gate, with input from a layer of binary OR gates, with the OR gates connected to negated inputs only.

Theorem 2.1 ([DF95a]) *(a) For all* even $t \geq 2$, *$\mathit{MWBES}(t)$ is complete for $W[t]$ under $\leq_m^{fpt}$. Hence so is $\mathit{WBES}(t)$.*

(b) For all odd $t \geq 3$, *the problem $\mathit{AWBES}(t)$ is complete for $W[t]$ under $\leq_m^{fpt}$. Hence so is $\mathit{WBES}(t)$.*

(c) The problem $\mathit{AWBES}(1,1)$ is complete for $W[1]$ under $\leq_m^{fpt}$.

For $t = 1$, the extra level of small OR gates is necessary (unless $W[1] = \mathrm{FPT}$) [DF95b]. The methods there and in Section 4 in [ADF95] remove this layer of small gates from earlier completeness proofs for odd $t \geq 3$.

We point out one important aspect of FPT reductions that strongly governs the *size* of the objects one can produce. Suppose $A \leq_m^{fpt} \mathit{WCS}(t,h)$, and take the polynomial q and functions $f, g : \mathbf{N} \to \mathbf{N}$ from Definition 1.1. Since T on input (x,k) must run in time $f(k)q(n)$ ($n = |x|$), the circuits $C_{x,k}$ it produces have size polynomial in n for fixed k, and most importantly, the exponent of the polynomial is independent of k. Let $n' = f(k)q(n)$ and $k' = g(k)$, the latter being the Hamming weight parameter for $C_{x,k}$ and independent of x.

Definition 2.1. A *parametric connection* is a function $\alpha : (N \times N) \to (N \times N) : (n,k) \mapsto (n',k')$, a polynomial q, and arbitrary functions $f, g : N \to N$ with $n' = f(k)q(n)$ and $k' = g(k)$. A parametric connection is *nice* if $g(k)$ is recursive and α can be computed in time $h(k)p(n)$ where h is an arbitrary function and p is a polynomial.

To economize on notation we write $n, k, n', k', n'', k'', \ldots$ to indicate that the first four quantities represent one parametric connection, the third through sixth another, and so on. The connection relation is transitive. This notion enables us to define circuit complexity directly for parameterized problems:

Definition 2.2. A *parameterized family* of circuits is a bi-indexed family of circuits $\mathcal{F} = \{ C_{n,k} \}$

such that each $C_{n,k}$ has n inputs and size at most n', where n' is part of a connection with n, k. We say that such a family is *FPT-uniform* if there is a algorithm to produce the circuit $C_{n,k}$ in time $O(n')$.

The idea of bounded Hamming weight in the weighted circuit satisfiability problems has been very successful in classifying many problems to belong to, and be complete for, the $W[t]$ classes [BFH94, DF95a, DF95b, DFHKW94, FK93]. We suspect that it is really central in fixed-parameter theory. We bring this idea down to tractable parameterized problems, and then use it in a notion of limited nondeterminism.

Definition 2.3. $G[t]$ (Uniform $G[t]$) is the class of parameterized languages $L \subseteq \Sigma^* \times N$ for which there is a parameterized (uniform) family of weft t circuits $\mathcal{F} = \{ C_{n,k} \}$ such that for all x and k, with $n = |x|$, $(x,k) \in L \iff C_{n,k}(x) = 1$. If there is no restriction on the circuit weft, then we obtain the class of parameterized languages $G[poly]$. Monotone $G[t]$ and Uniform Monotone $G[t]$ are defined in exactly the same way for monotone circuit families.

Proposition 2.2 *Uniform* $G[poly] = \mathrm{FPT}$.

Proof. If a parameterized language L is in Uniform $G[poly]$ then membership of (x,k) in L, $|x| = n$, can be decided in the right amount of time $O(n')$ by generating the circuit $C_{n,k}$ and evaluating it on input x. The converse also holds by imitating the usual proof that languages in P have polynomial-sized circuits. □

Thus the classes Uniform $G[t]$ contain problems that are all fixed-parameter tractable. Now we can build upon them in much the same way that NP is definable by bounded existential quantification over P. NP uses a polynomial length bound, while our classes $N[t]$ use bounds on Hamming weight.

Definition 2.4. (a) For any class $\mathcal{C}$ of parameterized languages, $\exists \cdot \mathcal{C}$ stands for the class of parameterized languages A such that for some $B \in \mathcal{C}$ there are nice parametric connections (n, k, n', k', n'', k'') giving for all (x,k), $(x,k) \in A \iff (\exists y \in \Sigma^{n'})[wt(y) = k' \ \wedge \ (xy, k'') \in B]$. (Here $n = |x|$, $n' = |y|$, and $n'' = n + n'$.)

(b) For all $t \geq 1$, $N[t]$ stands for $\exists \cdot$ Uniform-$G[t]$, and $N[poly]$ stands for $\exists \cdot$ Uniform-$G[poly]$.

In a corresponding way, we can define "bounded weight" versions of the other familiar class operators $\forall$, $\oplus$, and BP. Combining the latter two formally, we have that a language A belongs to $\mathrm{BP} \cdot \oplus \cdot \mathcal{C}$ if there exists $B \in \mathcal{C}$ and nice connections giving for all (x,k), $(x,k) \in A \Longrightarrow$

$$\Pr_{y \in \{0,1\}^{n'}, wt(y)=k'}[\|\{ z \in \{0,1\}^{n''} : wt(z) = k'' \ \wedge \ (xyz, k''') \in B \}\| \text{ is odd}] > 3/4,$$

while $(x,k) \notin A \Longrightarrow \Pr[\ldots] < 1/4$. If the latter probability is zero (i.e., we have one-sided error), then we write $A \in \mathrm{RP} \cdot \oplus \cdot G[t]$.

Definition 2.5. If $\mathcal{C}$ is any class of parameterized languages, then by $< \mathcal{C} >$ we denote the parameterized languages that are reducible to a language in $\mathcal{C}$, and refer to this as the *FPT-closure* of $\mathcal{C}$.

3 A Computational Characterization of W Classes

Despite the obvious success of the W hierarchy as a classification mechanism for concrete parameterized problems, the classes $W[t]$ often seem a bit strange. One of the central issues is that they do not seem to embody any "computational mechanism" but are rather defined by reducibility to a particular problem, WEIGHTED t-NORMALIZED SATISFIABILITY. The main theorem of this section gives a more computational characterization of $W[t]$.

Theorem 3.1 *For all $t \geq 1$, $W[t] = \langle N[t] \rangle$.*

To see what is interesting about this theorem, consider the special case of $t = 2$ and the $W[2]$-complete parameterized problem DOMINATING SET. The original criterion for showing DOMINATING SET to be in $W[2]$ requires constructing, for each graph G and positive integer k, a weft 2 circuit C_G that accepts a weight k input vector iff G has a k-element dominating set. The point is that for each graph G we construct a *different* circuit, thus perhaps $2^{\binom{n}{2}}$ different circuits for graphs of order n for a fixed value of k. By contrast, to show that DOMINATING SET belongs to the FPT-closure of $N[t]$, we must refer all of the graphs of order n (for a fixed value of k) to a *single* circuit $C_{n',k'}$. The input to $C_{n',k'}$ consists of the concatenation xy of a string x representing G and a string y representing the $k \log n$ bits of nondeterminism. For this particular instance our proof must devise a bi-indexed family of weft 2 circuits, each circuit $C_{n',k'}$ of which is "universal" for the dominating set problem for graphs of order n and for the parameter k. These "universal circuits" resemble programmable logic arrays.

Proof. Assume first that $t \geq 2$ and that t is even. Let L be a parameterized language in $W[t]$. We can assume without loss of generality that the reduction showing membership of L in $W[t]$ maps (x, k) to (C_x, k') where:

1. C_x is a t-normalized circuit
2. C_x has n' inputs
3. C_x has exactly n'' gates on each level other than the input and output levels (achievable by padding)
4. k', n' and n'' are described by nice parametric connections.

Let the gates (including inputs) of C_x be described by the set

$$\{g[s, i] : 0 \leq s \leq t,\ 1 \leq i \leq n''\}.$$

Here the level of the gate is indicated by the first index. Note that on level t only one gate (the output) is important (the padding is just a notational convenience). We may assume the output gate is $g[t, 1]$.

We consider the following uniform circuit family $\mathcal{F}_L = \{\, C_{m,k'} \,\}$, $m = t(n'')^2 + n'$. (To arrange for $\mathcal{F}_L$ to have one circuit for each possible pair of indices, simply pad with nonaccepting empty circuits for index pairs not of the indicated form.)

The circuit $C_{m,k'}$ is described as follows. There are $2t + 1$ levels of gates $L_0, ..., L_{2t}$. The inputs to the circuit constitute level 0. The gate sets are described as follows:

$$L_0 = \{a_X[s, i, j] : 1 \leq s \leq t,\ 1 \leq i \leq n'',\ 1 \leq j \leq n''\} \cup \{a_Y[i] : 1 \leq i \leq n'\},$$

and for $s = 1, ..., t$,

$$L_{2s} = \{c[2s,i] : 1 \leq i \leq n''\}$$

$$L_{2s-1} = \{b[2s-1,i,j] : 1 \leq i \leq n'',\ 1 \leq j \leq n''\}$$

According to our assumption that t is even, we assign the following logic functions to these gates: for $s = 1, ..., 2t$ the gates of L_s are $\wedge$ gates if s is congruent to 0 or 1 mod 4; the other levels are $\vee$ gates.

The gates in the circuit $C_{m,k'}$ are connected as follows.

1. For $s = 1, ..., t$ the gate $c[2s,i]$ receives input from each of the gates $b[2s-1,i,j]$ for $j = 1, ..., n''$.

2. For s odd, $2 \leq s \leq t$, the gate $b[2s-1,i,j]$ computes the Boolean expression $c[2(s-1),j] \wedge a_X[s,i,j]$.

3. For s even, $2 \leq s \leq t$, the gate $b[2s-1,i,j]$ computes the Boolean expression $c[2(s-1),j] \vee \neg a_X[s,i,j]$.

4. The gate $b[1,i,j]$ computes the Boolean expression $a_Y[j] \wedge a_X[1,i,j]$.

The $a_X[*,*,*]$ inputs to the circuit have the role of describing the circuit C_x. The $a_Y[*]$ inputs represent the (nondeterministic) inputs to C_x. The gates on even-indexed levels L_{2s} provide a PLA-type template on which to simulate the circuit C_x. Note that these are large gates of the same logical character as the gates on level s of C_x. The gates on odd-indexed levels L_{2s-1} are small gates whose function is to interpret the description of C_x so that C_x can be simulated.

The $a_X[*,*,*]$ inputs describe C_x in the following way. Set $a_X[s,i,j] = 1$ if and only if in C_x the gate $g[s,i]$ takes input from $g[s-1,j]$. Let $\chi(C_x)$ denote the length $t(n'')^2$ 0-1 vector that describes C_x in this way. The following claim establishes that the the circuit $C_{m,k'}$ works correctly.

Claim 1. For all $y \in \Sigma^{n'}$ of weight k', $C_{m,k}(\chi(C_x) \cdot y) = 1$ if and only $C_x(y) = 1$.

Claim 1 is easily proved by induction on the levels of the circuit simulation.

An essentially identical argument handles t odd, $t \geq 3$. The case of $t = 1$ presents additional difficulties and must be handled as a special case. (The simulation above would would result in universal circuits of weft 2.)

It suffices to show a "universal" family of circuits for the $W[1]$-complete problem INDEPENDENT SET. What we want is a weft 1 circuit that takes as input the concatenation of two strings x and y where x describes a graph of order n, and y represents the candidate k-element independent set. We can accomplish this by having the first part of the input $x = (x[1,2], x[1,3], ..., x[n-1,n])$ represent the adjacencies of G as a 0-1 string of length $\binom{n}{2}$, and letting $y = (y[1], ..., y[n])$ (the nondeterministic part of the input) have length n and weight specification k. The circuit can simply represent the Boolean expression

$$C = \prod_{1 \leq i < j \leq n} (\neg x[i,j] \vee \neg y[i] \vee \neg y[j]).$$

The above arguments show that $W[t] \subseteq N[t]$. To see that this inclusion reverses, suppose L is a parameterized language in $N[t]$. Then $(x,k) \in L$, $|x| = n$, if and only if $\exists y \in \Sigma^{n'}$ of weight k', such that a nicely produced weft t circuit $C_{n'',k''}$ accepts xy. To exhibit a reduction from L to WEIGHTED CIRCUIT SATISFIABILITY for weft t, we may just take the image of the reduction to be

$C_{n'',k''}$ with the first $n'' - n'$ inputs "removed" by being fixed to the value of x. From $N[t] \subseteq W[t]$ we obtain $< N[t] > \subseteq W[t]$ by the trasitivity of parameterized reducibility. $\square$

In a similar way we can prove the following characterization of $W[poly]$.

Theorem 3.2 $W[poly] = N[poly]$.

Proof. The proof is essentially the same as for Theorem 3.1. Note that the standard argument proving Proposition 2.2 can be used to show that $\langle N[poly] \rangle = N[poly]$, by "folding the reductions into the circuits." $\square$

4 Separation Result

Let Σ_d^{poly} stand for the class of languages recognized by depth-d unbounded fan-in Boolean circuits of polynomial size having a single OR gate at the output, as described in the survey by Boppana and Sipser [BS90]. Let Π_d^{poly} stand for the complements of these languages, which are recognized by depth-d circuits with an AND gate at the output. Sipser [Sip83] showed that for all $d \geq 1$, $\Sigma_d^{poly} \neq \Pi_d^{poly}$. It is not surprising that this carries over to the parameterized setting to show that the $G[t]$ hierarchy is proper, but it is noteworthy that it extends to our nondeterministic classes:

Theorem 4.1 *For all $t \geq 1$, $N[t] \subset N[t+1]$.*

Proof. Suppose $N[t] = N[t{+}1]$, and let A_0 be a language in Π_{t+1}^{poly}. Define a simple parameterized language A by $A = \{\, (x, k) : x \in A_0 \,\}$. Then $A \in G[t{+}1] \subseteq N[t{+}1]$. By our supposition, $A \in N[t]$. By the definition of $N[t]$ there exists a parameterized language $B \in G[t]$ accepted by a bi-indexed family of circuits $\mathcal{C} = \{\, C_{n,k} \,\}$ such that we have for all x and any fixed integer k_0:

$$x \in A_0 \iff (x, k_0) \in A \iff (\exists y \in \{\, 0,1 \,\}^{n'})[wt(y) = k_0' \ \wedge \ (xy, k_0'') \in B]$$

$$\iff (\exists y \in \{\, 0,1 \,\}^{n'})[wt(y) = k_0' \ \wedge \ C_{n'',k_0''}(xy) = 1].$$

Here again the priming indicates that n and the fixed k_0 are part of nice parametric connections, with $n'' = |xy| = n + n'$.

Using $C_{n'',k_0''}$ as a building block, we can create a circuit $\tilde{C}_{n'',k_0''}$ that evaluates $C_{n'',k_0''}(xy)$ for *all possible* y, with an output $\vee$ gate on all these possibilities. There are $\binom{n'}{k_0'}$ possible y, but this is permitted since k_0' is a constant. The family of circuits constructed from $\mathcal{C}$ in this way over all n show that $A_0 \in \Sigma_{t+1}^{poly}$, contradicting the fact that Π_d^{poly} is not contained in Σ_d^{poly}, for all d. $\square$

The above theorem does not prove, of course, that the $W[t]$ hierarchy is proper. If we could prove that, then we would have P $\neq$ NP. What it does show is that any "normal" approach of the kind often employed in the study of the W classes, namely the use of additional (bounded-weight) nondeterminism, will necessarily fail. For example, to show that $W[t+1]$ collapses to $W[t]$ we might hope to design some sort of gadgetry whose operation can be described by a weft t circuit C', that would correctly verify that a circuit C of weft $t+1$ accepts a particular weight k input vector x on the basis of some additional $k' \log n$ bits of nondeterministic information. Collapse would then follow by using C' to process *two* guesses: the input x to C and the "proof" that $C(x) = 1$. Since x has bounded weight and the size of C' can involve a blowup in size of $f(k)n^{g(t)}$ for $|C| = n$ and arbitrary functions f and g, we might well believe that there is some hope for this project. However, if this program were to succeed then we would in fact have shown that $G[t+1] \subseteq N[t]$. By the following easy but important proposition, in which the transitivity of parametric connections enables us to "coalesce" two like quantifiers into one, we would then have $N[t+1] \subseteq N[t]$, contradicting Theorem 4.1.

Proposition 4.2 *Let $\mathcal{C}$ be any class of parameterized languages. Then $\exists \cdot \exists \cdot \mathcal{C} = \exists \cdot \mathcal{C}$.* □

Although the parameterization of A_0 in the proof of Theorem 4.1 is trivial by itself, the manner in which the parameter interacts with the definition of $\exists \cdot$ Uniform-$G[t]$ and with the switch between Σ_d and Π_d circuits is noteworthy, and overall the information in the theorem seems surprisingly good. It lends support to the conjecture that the $W[t]$ hierarchy is proper.

5 The Hierarchy $H[t]$

The classes $N[t]$ are defined by a single bounded-weight existential quantification. It is natural to consider corresponding classes defined by universal and by alternating bounded weight quantification.

Definition 5.1. For each $t \geq 1$, define $\Sigma_1[t] = W[t] = \langle \exists \cdot$ Uniform $G[t]\rangle$. Correspondingly define $\Pi_1[t] = \langle \forall \cdot$ Uniform $G[t]\rangle$. For $i \geq 2$ define $\Sigma_i[t] = \langle \exists \cdot \Pi_{i-1}[t]\rangle$ and $\Pi_i[t] = \langle \forall \cdot \Sigma_{i-1}[t]\rangle$. Define $\Sigma_0[t] = \Pi_0[t] = \langle G[t]\rangle = \mathrm{FPT}$. Finally, for each t define $H[t]$ to be the union of these classes, viz.

$$H[t] = \bigcup_{i=0}^{\infty} \Sigma_i[t] \cup \Pi_i[t].$$

As one would expect, the $\Pi_i[t]$ classes consist of the complements of parameterized languages in the $\Sigma_i[t]$ classes. Moreover, by the methods of [DF95a] and induction on i, it follows that the Σ_i-quantified analogue of Weighted t-Normalized Boolean Expression Satisfiability is complete for $\Sigma_i[t]$.

The next theorem shows that in contrast to the proper inclusions of the $N[t]$ hierarchy, the $H[t]$ hierarchy collapses to $H[1]$.

Theorem 5.1 *For all $t \geq 1$, $H[t] = H[1]$*

Proof. By induction, it suffices to show that $H[t] \subseteq H[t-2]$, for t odd. Let $L \in \Sigma_s[t]$. We argue that $L \in \Sigma_{s+2}[t-2]$. By the above remarks, L is FPT-reducible to the Σ_s-quantified version of $WBES(t)$. Accordingly, let E be a Boolean expression over a set of variables V that is partitioned into sets $V = V_1 \cup \cdots \cup V_s$ with $V_i \cap V_j = \emptyset$ for $1 \leq i < j \leq s$, such that E has the form

$$E = \prod_{i=1}^{m} \sum_{j=1}^{m_i} \prod_{k=1}^{m_{ij}} E[i,j,k],$$

where $E[i,j,k]$ is a literal if $t = 3$, and is otherwise a weft $t-3$ expression that is a large $\vee$ of weft $t-4$ subexpressions. Let $(k_1, ..., k_s)$ be a sequence of positive integers. The quantified satisfiability question for E is whether

$\exists$ a weight k_1 truth assignment to the variables of V_1, such that

$\forall$ weight k_2 assignments to the variables of V_2,

$\ldots,$

such that E is satisfied. Now we describe an expression E' over a set of variables

$$\begin{aligned} V' &= V \cup V_\forall \cup V_\exists, &&\text{where} \\ V_\forall &= \{a[i] : 1 \le i \le m\} &&\text{and} \\ V_\exists &= \{e[i,j] : 1 \le i \le m,\ 1 \le j \le m_i\}, \end{aligned}$$

such that the answer to the quantified satisfiability question for E is "yes" if and only if the answer to the quantified satisfiability question for E' is "yes." The latter is defined to hold iff

$\exists$ a weight k_1 assignment to V_1, such that
$\forall$ weight k_2 assignments to V_2,
$\ldots,$
$\forall$ weight 1 assignments to $V_\forall$,
$\exists$ a weight 1 assignment to $V_\exists$,

such that E' is satisfied.

The expression E' is described by $E' = E'_1 \cdot E'_2$, where the two factors are

$$\begin{aligned} E'_1 &= \prod_{i=1}^{m} \prod_{j=1}^{m_i} (e[i,j] \to a[i]), \qquad \text{and} \\ E'_2 &= \prod_{i=1}^{m} \prod_{j=1}^{m_i} \prod_{k=1}^{m_{ij}} (E[i,j,k] \vee \neg e[i,j]). \end{aligned}$$

For $t > 3$, since $E[i,j,k]$ is a large logical sum of subexpressions and has weft $t-3$, the same is true for $(E[i,j,k] \vee \neg e[i,j])$, and therefore E' has weft $t-2$. If $t = 3$ then E' is a product of sums of size 2, and thus has weft 1. The verification that the construction works correctly is straightforward and is left to the reader. $\square$

This proof does not tell us whether $\Sigma_s[t]$ is equal to $\Sigma_{s+2}[t-2]$, and in general we do not know exactly how the hierarchies $H[t]$ intercalate for different t.

6 Randomized Reduction of $W[t]$ to Unique $W[t]$

It would be interesting to know quite a bit more than we presently do about the calculus of the operators $\exists\cdot$, $\forall\cdot$, $BP\cdot$, $RP\cdot$ and $\bigoplus\cdot$ over the $G[t]$ classes. For example, do the following analogs of the theorems (respectively) of Valiant and Vazirani [VV86] and Toda [Tod91] hold?
(1) $N[t] \subseteq BP \cdot \bigoplus \cdot G[t]$
(2) $H[t] \subseteq BP \cdot \bigoplus \cdot G[t]$

Analogs in parameterized complexity (if they exist) of familiar structural theorems generally present significant and novel difficulties and are in most cases not presently known. A parameterized analog of Ladner's density theorem remains elusive, although substantial partial results have been obtained [DF93]. A parameterized analog of Mahaney's theorem on the complexity of sparse sets is proved in [CF96]. In this section we prove an analog of the Valiant-Vazirani theorem that nevertheless falls short of (1). Our proof is modeled on (and will make use of) the proof of the

Valiant-Vazirani result in §1.4.1 of [KST93]. The main difficulty is in fitting that argument into weft 1 constructions. This can be accomplished by employing additional nondeterminism that is *uniquely determined.*

Definition 6.1. A *randomized (fpt, many-one) reduction* from a parameterized language L to a parameterized language L' is a randomized procedure that transforms (x, k) into (x', k') subject to the following conditions:
(1) The running time of the procedure is bounded by $f(k)|x|^c$ for some constant c and arbitrary function f (i.e. the procedure is fixed-parameter tractable).
(2) There is a function f' and a constant c' such that for all (x, k),

$$\begin{aligned} (x,k) \in L \quad &\Rightarrow \quad \text{Prob}[(x',k') \in L'] \geq 1/f'(k)|x|^{c'} \\ (x,k) \notin L \quad &\Rightarrow \quad \text{Prob}[(x',k') \in L'] = 0 \end{aligned}$$

In §2 we gave the usual definition of the $W[t]$ hierarchy in terms of the WEIGHTED CIRCUIT SATISFIABILITY problem. We consider here the following unique-solution variant.

UNIQUE WCS(t,h)
INSTANCE: A circuit C of weft t and overall depth $t + h$.
PARAMETER: k.
QUESTION: Is there a unique input of Hamming weight k that is accepted by C?

Definition 6.2. For all $t \geq 1$, Unique $W[t]$ is the class of parameterized languages L such that for some h, L is fpt many-one reducible to UNIQUE WCS(t,h).

Our proof will make use of a technical but generally useful lemma showing that a restricted form of WEIGHTED t-NORMALIZED SATISFIABILITY is complete for $W[t]$. This lemma is essentially implicit in earlier work. The variant is defined as follows.

SEPARATED t-NORMALIZED SATISFIABILITY
INSTANCE: A t-normalized Boolean expression E over a set of variables V that is partitioned into k disjoint sets $V_1, ..., V_k$ of equal size, $V_i = \{v_{i,1}, ..., v_{i,n}\}$ for $i = 1, ..., k$.
PARAMETER: k.
QUESTION: Is there a truth assignment of weight k making exactly one variable in each of the V_i *true* and all others *false* and that furthermore satisfies the condition that if $v_{i,j}$ is *true*, then for all $i' > i$ and $j' \leq j$, $v_{i',j'}$ is *false*.

Lemma 6.1 SEPARATED t-NORMALIZED SATISFIABILITY *is complete for* $W[t]$ *for all* $t \geq 1$.

Proof. We give separate arguments for t even and t odd. For t even we reduce from MONOTONE t-NORMALIZED SATISFIABILITY and use the construction described in [DF95a]. Suppose the parameter is k and that F is the monotone expression. The reduction is to a normalized expression F' and the parameter $k' = 2k$. The key point is that the variables for F' consist of $2k$ disjoint blocks, and that any weight $2k$ truth assignment for F' must make exactly one variable *true* in each block. The blocks can be padded so that they are of equal size. Including additional enforcement for the condition in the definition of SEPARATED t-NORMALIZED SATISFIABILITY is straightforward. It is possible for this to be done in such a way that monotonicity is preserved.

Fot t odd we similarly employ the reduction described in [DF95b], starting from ANTI-MONOTONE t-NORMALIZED SATISFIABILITY. In this case, antimonotonicity can be preserved.

$\square$

Theorem 6.2 *For all $t \geq 1$ there is an fpt many-one randomized reduction of $W[t]$ to Unique $W[t]$.*

Proof. We reduce from SEPARATED t-NORMALIZED SATISFIABILITY. Let E be the relevant t-normalized Boolean expression over the k blocks of n variables:

$$X_i = \{x[i,1], ..., x[i,n]\} \text{ for } i = 1, ..., k$$

Let X denote the union of the X_i and assume for convenience (with no loss of generality) that n is a power of 2, $n = 2^s$, and that $k-1$ divides s.

We describe how to produce (by a randomized procedure) a weft t expression E' of bounded depth, and an integer k' so that the conditions defining a randomized reduction are met.

The reduction procedure consists of the following steps:

(1) Randomly choose $j \in \{1, ..., k \log n\}$.

(2) Randomly choose j length n 0-1 vectors

$$y_i = (y[i,1], ..., y[i,n]), \quad 1 \leq i \leq j.$$

(3) Randomly choose $m \in \{1, ..., 12\}$.

(4) Output

$$E' = E_1 \wedge E_2 \wedge \cdots \wedge E_9 \text{ and } k'$$

where the constituent subexpressions E_i and the weight parameter k' are as described below.

The set X' of variables for E' is

$$X' = X'_1 \cup X'_2 \cup X'_3$$

where

$$\begin{aligned} X'_1 &= \{u[a,b,c] : 1 \leq a \leq m,\ 1 \leq b \leq k,\ 1 \leq c \leq n\} \\ X'_2 &= \{v[a,b] : 1 \leq a \leq k(k-1),\ 1 \leq b \leq n\} \\ X'_3 &= \{w[a,b] : 1 \leq a \leq m-1,\ 1 \leq b \leq k\} \end{aligned}$$

We next describe the various constituent subexpressions of E'.

The subexpression E_1.

Write $X'_1(i)$ to denote the variables of X'_1 that have first index i, for $i = 1, ..., m$. That is,

$$X'_1(i) = \{u[i,b,c] : 1 \leq b \leq k,\ 1 \leq c \leq n\}$$

Note that the set $X'_1(i)$ can be paired in a natural way with the set of variables X of the expression E by the correspondence:

$$x[b,c] \leftrightarrow u[i,b,c]$$

Let $E_1(i)$ denote the expression obtained from E (essentially, a copy of E) by substituting the variables of $X'_1(i)$ for the variables of X according to this correspondence.

$$E_1 = \prod_{i=1}^{m} E_1(i)$$

The role of E_1 is to hold each of the m copies of the variables of E accountable for satisfying a copy of E.

The subexpression E_2.

$$E_2 = \prod_{a=1}^{m} \prod_{b=1}^{k} \prod_{1 \le c < c' \le n} (\neg u[a,b,c] \vee \neg u[a,b,c'])$$

The role of E_2 is to enforce that at most one variable is set true in each "block" of the variables of X'_1 (there are km blocks, corresponding the m copies X, each copy consisting in a natural way of k blocks).

The subexpression E_3.

$$E_3 = \prod_{a=1}^{m} \prod_{b=1}^{k} \prod_{c=1}^{n} \prod_{b'=b+1}^{k} \prod_{c'=1}^{c} (u[a,b,c] \rightarrow \neg u[a,b',c'])$$

The role of E_3 is to enforce the ascending order condition on truth assignments (with respect to the k blocks of variables) that occurs in the definition of SEPARATED t-NORMALIZED SATISFIABILITY. This condition is enforced for each of the m copies of the variables of E.

The subexpression E_4.

We view X'_3 as consisting of $m-1$ blocks:

$$X'_3(a) = \{w[a,b] : 1 \le b \le k\}$$

$$E_4 = \prod_{a=1}^{m-1} \prod_{1 \le b < b' \le k} (\neg w[a,b] \vee \neg w[a,b'])$$

The role of this subexpression is to enforce that at most one variable is set true in each of the blocks of X'_3 in any any satisfying truth assignment for E'.

The subexpression E_5.

$$E_5 = \prod_{a=1}^{k(k-1)} \prod_{1 \le b < b' \le n} (\neg v[a,b] \vee \neg v[a,b'])$$

The role of this subexpression is to enforce that at most one variable is set true in each of the $k(k-1)$ blocks of X'_2.

The subexpressions E_6 and E_7.

$$E_6 = \prod_{a=1}^{m-1} \prod_{b=1}^{k} \prod_{b'=1}^{b-1} \prod_{c \ne c' : 1 \le c,c' \le n} (\neg w[a,b] \vee \neg u[a,b',c] \vee \neg u[a+1,b',c'])$$

$$E_7 = \prod_{a=1}^{m-1} \prod_{b=1}^{k} \prod_{1 \le c' \le c \le n} (\neg w[a,b] \vee \neg u[a,b,c] \vee \neg u[a+1,b,c'])$$

The $m-1$ variables that are set true in the blocks of X'_3 in a satisfying assignment for E' provide evidence that the m "solutions" for E recorded in the m blocks of X'_1 are distinct and recorded in the m blocks in increasing lexicographic order. The nature of this evidence is an indication of the first of the k choice blocks in which two consecutive solutions differ. The subexpressions E_6 and E_7 enforce the increasing lexicographic ordering based on this evidence.

The subexpressions E_8 and E_9.

In order to describe the subexpressions E_8 and E_9 we first must construct an *interpretation* of the variables of X_2'. This consists of the following information:
(1) Each $a \in \{1, ..., k(k-1)\}$ is assigned a subset $J_a \subseteq \{1, ..., j\}$ so that $|J_a| = \log n/(k-1)$ and $\bigcup_{1 \leq a \leq k(k-1)} = \{1, ..., j\}$.
(2) Each even-cardinality subset $S_\alpha \subseteq \{1, ..., k\}$ is assigned a unique 0-1 vector α of length $k-1$. (Note that this is possible, since there are 2^{k-1} such even-cardinality subsets.)
(3) Each variable $v[a, b] \in X_2'$ is interpreted as assigning an even-cardinality subset $S(j', a, b)$ to each $j' \in J_a$. This assignment is made in the following way. The index b can be regarded as a 0-1 vector of length $\log n$. This index vector can be read as a sequence of $|J_a|$ blocks of size $k-1$. If the r^{th} block is α then the r^{th} element of J_a is assigned the even-cardinality subset S_α.

$$E_8 = \prod_{p=1}^{m} \prod_{1 \leq a \leq k(k-1)} \prod_{1 \leq b \leq n} \prod_{j' \in J_a} \prod_{r \in S(j',a,b)} \prod_{q: y[j',q]=0} (\neg v[a, b] \vee \neg u[p, j', q])$$

$$E_9 = \prod_{p=1}^{m} \prod_{1 \leq a \leq k(k-1)} \prod_{1 \leq b \leq n} \prod_{j' \in J_a} \prod_{r \notin S(j',a,b)} \prod_{q: y[j',q]=1} (\neg v[a, b] \vee \neg u[p, j', q])$$

The variables that are set true in X_2' in a satisfying truth assignment for E' are intended to indicate a proof (that can be checked by a weft 1 circuit) that each of the m weight k truth assignments that are solutions for E recorded in the m blocks of X_1' are orthogonal to the randomly chosen length n 0-1 vectors y_i. The proof that is indicated consists of showing that an even subset of the k positions set to true in X_1' have corresponding positions that are 1 in the y_i. A variable $v[a, b]$ indicates part of such a proof, according to the interpretation mechanism described above. The subexpressions E_8 and E_9 provide an enforcement for the interpretation.

The parameter.

The description of the reduction is completed by specifying the parameter that accompanies E'.

$$k' = mk + k(k-1) + (m-1)$$

We now argue for the correctness of the reduction. Half of this is easy. If E is not satisfiable by a weight k truth assignment, then because of E_2 and E_1 there is no weight k' truth assignment that satisfies E' (never mind whether it is unique).

For the other half we must argue that if E has a weight k truth assignment, then with the required probability bound, E' has a unique weight k' truth assignment. Let $X_0 = \{x[1], ..., x[n]\}$. The weight k truth assignments to X that satisfy the additional conditions that define SEPARATED t-NORMALIZED SATISFIABILITY can be put in a natural 1:1 correspondence weight k truth assignments to X_0. The correspondence is that if the r^{th} variable assigned the value 1 in X_0 is $x[s]$ then $x[r, s]$ is assigned 1 in the truth assignment for X. Because of this correspondence we can speak of a weight k truth assignment to X_0 that satisfies E.

It follows from the arguments in [KST93] §1.4.1 that if there is any weight k truth assignment to X_0 that satisfies E (and noting that there are no more than n^k such assignments), then with probability at least $\frac{1}{24k \log n}$ there are exactly m distinct weight k truth assignments that satisfy

E and that are hashed by the function

$$h(x[1], ..., x[n])[s] = \bigoplus_{i=1}^{n} (x[i] \wedge y[s, i])$$

to 0^j.

We argue that in this case, E' is uniquely satisfied by a weight k' truth assignment to X'. The subexpressions E_1, E_2, E_3, E_4, E_6 and E_7 can be satisfied if the m distinct truth assignments are represented in lexicographically increasing ascending order in the blocks of X_1', and if the evidence for the lexicographic ordering is represented in X_3'. It is easy to check that if there are exactly m distinct weight k truth assignments that satisfy E, then there is a unique truth assignment to $X_1' \cup X_3'$ that satisfies these subexpressions, and it must have weight $mk + (m-1)$. The key point for this assertion is that the subexpressions E_6 and E_7 are sufficiently restrictive that not only is increasing lexicographic ordering enforced, but also the evidence for this is uniquely determined.

In the above situation, the subexpressions E_5, E_8 and E_9 can be satisfied by a weight $k(k-1)$ assignment to X_2' that represents the hash function condition. Because this is also uniquely determined, there is a unique weight k' truth assignment for E'.

The subexpressions E_2 through E_9 have weft 1, and therefore the weft of E' is the same as the weft of E. □

There are several obstacles to a proof of the statement (1) discussed at the beginning of this section. Among these is the matter that our proof of Theorem 6.2 uses $kn \log n$ random bits, while the definition of the BP$\cdot$ operator provides only $k \log n$ random bits. Furthermore, a method of probability amplification would be needed (also employing only $k \log n$ random bits). How to achieve this with weft 1 circuits is unclear. The question of whether (1) and (2) hold is quite interesting, since together with Theorem 5.1 they would yield that UNIQUE CLIQUE is as hard any parameterized problem in the $W[t]$ hierarchy.

7 Conclusion

We have placed the W hierarchy on a computationally more useful basis. Indeed, we have proved that the $N[t]$ classes giving $W[t] = \langle N[t] \rangle$ arise naturally from a notion $G[t]$ of "AC^0 circuits for parameterized problems" *and* form a proper hierarchy. Thus in the effort to determine whether the $W[t]$ hierarchy itself is proper, we need to focus attention on FPT-reductions themselves. Put another way, the structure provided by the ideas of circuit weft and bounded Hamming weight is robust by itself. A number of interesting and challenging questions remain open about the structure of the $H[t]$ hierarchy, especially about the calculus of complexity operators on the $G[t]$ classes.

References

[ADF95] K.A. Abrahamson, R.G. Downey and M.R. Fellows. Fixed parameter tractability and completeness IV: on completeness for W[P] and PSPACE analogs. *Annals of Pure and Applied Logic* 73 (1995), 235–276.

[BFH94] H. Bodlaender, M.R. Fellows and M. Hallett. Beyond NP-completeness for problems of bounded width: hardness for the W hierarchy. *Proceedings of the ACM Symposium on the Theory of Computing (STOC)* (1994), 449–458.

[BS90] R. Boppana and M. Sipser. The complexity of finite functions. In J. Van Leeuwen, ed., *Handbook of Theoretical Computer Science*, vol. A, 757–804. Elsevier and MIT Press, 1990.

[CF96] M. Cesati and M. Fellows. Sparse parameterized problems. *Annals of Pure and Applied Logic* 62 (1996).

[CW90] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetical progressions. *J. Symbolic Computation* 9 (1990), 251–280.

[DF93] R.R. Downey and M.R. Fellows. Fixed parameter tractability and completeness III: some structural aspects of the W hierarchy. *Proceedings of the 1992 Dagstuhl Workshop on Structural Complexity*, Complexity Theory: Current Research, K. Ambos-Spies, S. Homer and U. Schöning (eds.), Cambridge Univ. Press (1993), 191–226.

[DF95a] R.G. Downey and M.R. Fellows. Fixed-parameter tractability and completeness I: basic theory. *SIAM Journal of Computing* 24 (1995), 873–921.

[DF95b] R.G. Downey and M.R. Fellows. Fixed-parameter tractability and completeness II: completeness for W[1]. *Theoretical Computer Science A* 141 (1995), 109–131.

[DF95c] R.G. Downey and M.R. Fellows. Parameterized computational feasibility. *Proceedings of the Second Cornell Workshop on Feasible Mathematics*, Feasible Mathematics II, P. Clote and J. Remmel (eds.), Birkhauser Boston (1995), 219–244.

[DFHKW94] R.G. Downey, M.R. Fellows, M.T. Hallett, B.M. Kapron, and H.T. Wareham. The parameterized complexity of some problems in logic and linguistics. *Proceedings Symposium on Logical Foundations of Computer Science (LFCS)*, Springer-Verlag, Lecture Notes in Computer Science vol. 813 (1994), 89–100.

[FK93] M.R. Fellows and N. Koblitz. Fixed-parameter complexity and cryptography. *Proceedings of the Tenth International Symposium on Applied Algebra, Algebraic Algorithms and Error-Correcting Codes (AAECC'93)*, Springer-Verlag, Berlin, Lecture Notes in Computer Science vol. 673 (1993), 121–131.

[GJ79] M. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H. Freeman, 1979.

[KST93] J.Köbler, U. Schöning and J. Torán. *The Graph Isomorphism Problem: Its Structural Complexity.* Birkhäuser, 1993.

[NP85] J. Nesetril and S. Poljak. On the complexity of the subgraph problem. *Commen. Math. Univ. Carol.* 26 (1985), 415–419.

[Reg89] K. Regan. Finitary substructure languages. *Proceedings of the 4th Annual IEEE Conference on Structure in Complexity Theory* (1989), 87–96.

[Sip83] M. Sipser. Borel sets and circuit complexity. In the *Proceedings of the 15th ACM Symposium on the Theory of Computing* (1983), 61–69.

[Tod91] S. Toda. PP is as hard as the polynomial-time hierarchy. *SIAM J. Computing* (1991), 865–877.

[VV86] L. Valiant and V. Vazirani. NP is as easy as detecting unique solutions. *Theoretical Computer Science* 47 (1986), 85–93.