# Structure and Specification as Sources of Complexity

**Anuj Dawar**
University of Cambridge Computer Laboratory
anuj.dawar@cl.cam.ac.uk

## 1 Introduction

Computational complexity is often described as the study of what makes certain computational problems inherently difficult to solve. Of course, it has proved to be extremely difficult to establish unconditional lower bounds, but the theory has provided us with important tools for identifying intractable problems. If one were to pick out the most important contribution that complexity theory has made to the theory and practice of computing, it is arguably in introducing the notion of NP-completeness. The ability to identify NP-complete problems and to construct reductions are skills that are taught to virtually all students of computer science. However, while thousands of problems have been identified as NP-complete, and we have a strong, if informal, understanding of what makes a problem hard, this does not amount to a *theory* of complexity. We understand that an exponential, unstructured search space leads to difficulty, but we do not have an account of what kind of structure in the search space allows for tractable solutions. This is a distinct problem from our inability to prove lower bounds, i.e. to explain why NP-complete problems are truly intractable. It is the problem of explaining what makes certain problems NP-complete in the first place. It may even be argued that, at this point, we do not know what such a theory of difficulty might look like.

In this talk, I review results from descriptive complexity that relate to this issue. The best known results of descriptive complexity are about the characterisations of complexity classes in terms of logical definability. I would argue that one important contribution of these results is the separation they provide between the *specification* of a decision problem and the *structure* against which this specification is checked. The first is usually formalised as a sentence in some suitable formal logic, while the latter is usually a relational structure of some kind. This separation allows some insight into sources of complexity. One can measure the richness of the language in which specifications are written and one can measure the density of the structures considered. These are two aspects of work in descriptive complexity that I will consider. In these notes to accompany the talk, I briefly present some definitions and the main results. Many of these are historical, but I take them up to recent work and provide pointers to the literature. After presenting some background and definitions I briefly consider the complexity of specification languages in Section 2 and of structures at some length in Section 3. The former leads to some recent work on the question of characterisations of P, while the latter leads to connections with paramterized complexity.


Foundations of Software Technology and Theoretical Computer Science (Kanpur) 2009.
Editors: Ravi Kannan and K. Narayan Kumar; pp 407–416
Leibniz International Proceedings in Informatics (LIPIcs), Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany.
Digital Object Identifier: 10.4230/LIPIcs.FSTTCS.2009.2336

## 2 Complexity of Specification

The general situation we consider is of a problem where an instance is a structure (such as a graph) and the problem to be decided is given by a formula in some logic (typically an extension of first-order logic). Indeed, in the examples I consider in this paper, I confine myself to decision problems on graphs. Consider, for example, some classical NP-complete problems on graphs: INDEPENDENT SET, DOMINATING SET, 3-COLOURABILITY and HAMILTONICITY. In the first two cases, the input is a graph together with an integer parameter, while in the second case it consists of a graph alone. As we shall see, it matters whether we consider the integer parameter to be part of the specification of the problem, or the instance.

Suppose then that we are given a graph $G$ and a formula $\varphi$ of first-order logic in the language with one binary relation. How hard is it to decide whether $G \models \varphi$? There are essentially two versions of this question that interest us here (called the *data complexity* and the *combined complexity* of first-order logic, respectively by Vardi in [34]).

In the first, we ask how complex can be the set of graphs that satisfy a fixed first-order sentence. The answer is that it is always decidable in logarithmic space by a straightforward algorithm (and, indeed the set is in fact in $\mathsf{AC}^0$ [1]). Moreover, there are problems in $\mathsf{L}$ which one can easily show are not definable by any first-order sentence. In particular, there is no sentence that defines the graphs with an even number of vertices or the connected graphs (see [15, 26] for proofs). It is also not difficult to show that the Hamiltonian graphs, or the 3-colourable graphs are *provably* not first-order definable. The conclusion one can draw from this is that the expressive power of first-order logic is rather weak. This is one reason that research in finite model theory has focused on extensions of the logic.

On the other hand, it is easy to write, for each $k$ a first-order sentence that defines the graphs that contain an independent set of $k$ vertices, or a dominating set with $k$ vertices. Thus, if one considers the combined complexity of first-order logic, i.e. the following decision problem: given a graph $G$ and a first-order formula $\varphi$, determine whether $G \models \varphi$, then it is clearly hard. In fact, the problem is PSpace-complete. In terms of parameterized complexity, taking the length of $\varphi$ as parameter, the problem is AW[*]-complete. Moreover, restricting the first-order sentences to a fixed-number of quantifier alternations yields complete problems at every level of the $W$-hierarchy and thus the problem of evaluating first-order sentences in graphs is central to parameterized complexity. I return to connections with parameterized complexity in the next section. Futher details may also be found in the excellent text [19].

Searching for a specification language more expressive than first-order logic, the logician may turn first to second-order logic. Here, it is known since the work of Fagin [17] that the existential fragment is rich enough to express all (and only) the problems in NP. It follows that second-order logic expresses all decision problems in the polynomial hierarchy [32]. From the complexity-theoretic point of view, the interesting logics are intermediate in expressive power between first and second-order logic. In particular, it remains an open question whether there is a logic that expresses exactly the polynomial-time decidable properties of graphs.

Immerman [25] and Vardi [34] showed that LFP, the extension of first-order logic with inductive definitions expesses exactly the polynomial-time properies of *ordered* graphs but this is too weak in general. An extension of LFP with a mechanism for counting was proposed by Immerman, but shown to be too weak in [3]. Since then, a number of further logic have been proposed that all properly extend the expressive power of LFP with counting and for which it remains an open question whether they can express all polynomial-time properties. They include the language of *choiceless polynomial-time with counting* of Blass, Gurevich and Shelah [2] and the language of *specified symmetric choice* of Gire and Hoang [22, 12]. A significant recent development in this direction is the proposal to extend LFP with linear algabraic operators [8]. The mutual interrelationship between these various extensions also remains to be explored (see [13] for related results). A useful recent survey on the problem of characterising P is given by Grohe in [24].

## 3 Resticted Graph Classes

We now turn our attention to the combined complexity of first-order logic and to the question about how constraints on the *structure* can limit the search space and make hard problems tractable. As mentioned above, the problem of deciding, given a graph $G$ and a first-order sentence $\varphi$ whether $G \models \varphi$ is PSpace-complete, while for any fixed $\varphi$, the class of graphs that satisfy it is in L. To be more precise, if $\varphi$ has length $l$ and $m$ distinct variables and $G$ is a graph on $n$ vertices, then $G \models \varphi$ can be decided in time $\mathcal{O}(ln^m)$ and space $\mathcal{O}(m \log n)$. In [33], Stolboushkin and Taitslin asked whether there is a constant $c$ such that every first-order sentence defines a problem decidable in time $\mathcal{O}(n^c)$. They conjectured that this was not the case and noted that a proof of the conjecture would imply a separation of P from PSpace. A more uniform version of their question would ask for a computable function that maps $\varphi$ to a $\mathcal{O}(n^c)$ clocked algorithm for deciding the models of $\varphi$. The existence of such a function would imply that the problem of deciding whether $G \models \varphi$ was fixed-parameter tractable. Since this problem is AW[*]-complete (see [19] for details) this would imply the collapse of the edifice of parameterized complexity.

Indeed, many natural problems that are hard from the point of view of parameterized complexity can be naturally formulated in first-order logic. As an example, consider two problems mentioned above: INDEPENDENT SET and DOMINATING SET. They are complete for $W[1]$ and $W[2]$ respectively and, as noted above, naturally expressed by a (parameter-dependent) first-order formula.

A subject of intensive investigation in recent years has been the fixed-parameter tractability of otherwise hard problems, when the class of input graphs is restricted. A typical example is the fixed-paramter tractability of DOMINATING SET when restricted to planar graphs. Indeed, for many interesting restrictions on graphs, one can show that first-order satisfaction is itself fixed-parameter tractable and as a result the tractability of a whole host of other individual problems follows. In the rest of this section, we briefly survey results that establish the fixed-parameter tractability of first-order satisfaction on a number of such classes. The classes we examine are all classes of *sparse* graphs. That is, though the classes may be defined in other terms, they have the property that the number of edges in a graph in the class as a function of the number of vertices does not grow very fast. It should be remarked that there are other classes of graphs (such as those of bounded cliquewidth) which are not sparse in this sense, but where it is known that first-order logic (and, indeed, even monadic second-order logic) admit fixed-parameter tractable algorithms for the satisfaction problem (see [6, 5]).

**Sparse Classes** The relationships between various classes of sparse graphs that have been studied are depicted in Figure 1.

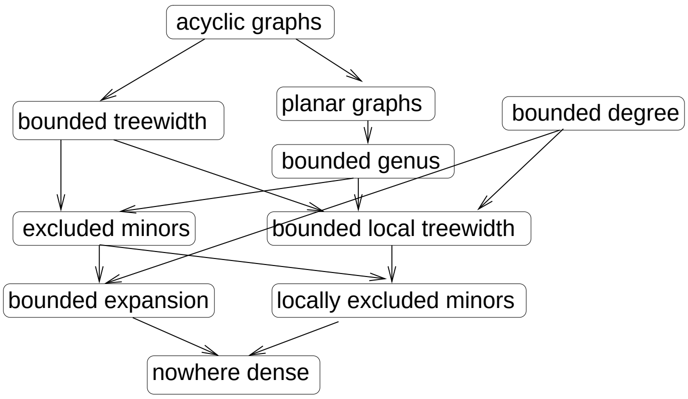


Figure 1: Relationships between sparse graph classes.

Among the restrictions given in Figure 1, those of acyclicity and planarity are of a different character to the others in that they apply to single graphs. We can say of graph $G$ that it is acyclic or planar. When we apply this restriction to a class $\mathcal{C}$, we mean that all structures in the class satisfy it. The other conditions in the figure only make sense in relation to classes of graphs. Thus, it makes little sense to say of a single finite graph that it is of bounded degree (it is necessarily so). When we say of a class $\mathcal{C}$ that it is of bounded degree, we mean that there is a uniform bound on the degree of all graphs in $\mathcal{C}$.

The arrows in Figure 1 should be read as implications. Thus, any graph that is acyclic is necessarily planar. Similarly, any class of acyclic graphs has bounded treewidth. The arrows given in the figure are *complete* in the sense that when two restrictions are not connected by an arrow (or sequence of arrows) then the first does not imply the second and separating examples are known in all such cases.

The restrictions of acyclicity, planarity and bounded degree are self-explanatory. We say that a class of graphs $\mathcal{C}$ has bounded genus if there is a fixed orientable surface $S$ such that all graphs in $\mathcal{C}$ can be embedded in $S$ (see [27]). In particular, as planar graphs are embeddable in a sphere, any class of planar graphs has bounded genus. The treewidth of a graph is a measure of how tree-like it is (see [14]). In particular, trees have treewidth 1, and so any class of acyclic graphs has treewidth bounded by 1. The measure plays a crucial role in the graph structure theory developed by Robertson and Seymour in their proof of the graph minor theorem. We say that a graph $G$ is a minor of $H$ (written $G \prec H$) if $G$ can be obtained from a subgraph of $H$ by a series of edge contractions (see [14] for details). We say that a class of graphs $\mathcal{C}$ excludes a minor if there is some $G$ such that for all $H \in \mathcal{C}$ we have $G \not\prec H$. In particular, this includes all classes $\mathcal{C}$ which are closed under taking minors and which do not include all graphs. If $G$ is embeddable in a surface $S$ then so are all its minors. Since, for any fixed integer $k$, there are graphs that are not of genus $k$, it follows that any class of bounded genus excludes some minor.

The notion of bounded local treewidth was introduced as a common generalisation of classes of bounded treewidth and bounded genus. A variant, called the diameter width property was introduced in [16] while bounded local treewidth is from [20]. Recall that the $r$-neighbourhood of a vertex $v$ in a graph $G$, denoted $N_G^r(v)$, is the subgraph of $G$ induced by the set of vertices at distance at most $r$ from $v$. We say that a class of graphs $\mathcal{C}$ has bounded local treewidth if there is a nondecreasing function $t : \mathbb{N} \to \mathbb{N}$ such that for any graph $G \in \mathcal{C}$, any vertex $v$ in $G$ and any $r$, the treewidth of $N_G^r(v)$ is at most $t(r)$. It is clear that any class of graphs of bounded treewidth has bounded local treewidth (indeed, bounded by a constant function $t$). Similarly, any class of graphs of degree bounded by $d$ has local treewidth bounded by the function $d^r$, since the number of elements in $N_G^r(v)$ is at most $d^r$. The fact that classes of bounded genus also have bounded local treewidth follows from a result of Eppstein [16].

We say that a class of graphs $\mathcal{C}$ locally excludes minors if there is a nondecreasing function $t : \mathbb{N} \to \mathbb{N}$ such that for any graph $G \in \mathcal{C}$, any vertex $v$ in $G$ and any $r$, the clique $K_{t(r)}$ is not a minor of the graph $N_G^r(v)$. This notion is introduced in [9] as a natural common generalisation of bounded local treewidth and classes with excluded minors. Classes of graphs with bounded expansion were introduced by Nešetřil and Ossona de Mendez [30] as a common generalisation of classes of bounded degree and proper minor-closed classes. A class of graphs $\mathcal{C}$ has bounded expansion if there is a function $t : \mathbb{N} \to \mathbb{N}$ such that for any graph $G \in \mathcal{C}$, any subgraph $H$ of $G$ and any minor $H'$ of $H$ obtained from $H$ by contracting neighbourhoods of radius at most $r$, the average degree in $H'$ is bounded by $t(r)$. In particular, classes that exclude a minor have bounded expansion witnessed by a constant function $f$.

Finally, we say that a class $\mathcal{C}$ of graphs is *nowhere dense* if there is a function $t : \mathbb{N} \to \mathbb{N}$ such that for each $r$, the graph $K_{t(r)}$ cannot be obtained as a minor of any $G \in \mathcal{C}$ by contracting neighbourhoods of radius at most $r$. This notion is introduced by Nešetřil and Ossona de Mendez in [28, 29]. They present convincing arguments to show that this is the natural upper limit to well-behaved classes of graphs based on sparseness conditions.

**Automata and Locality** The following is a sampling of results on the fixed-parameter tractability of the first-order satisfaction problem on classes of sparse graphs. In each of these, $l$ is the length of the formula $\varphi$, $n$ is the size of the graph $G$ and $f$ is some computable function.

1. If $\mathcal{T}_k$ is the class of graphs of treewidth at most $k$, then $G \models \varphi$ is decidable in time $\mathcal{O}(f(l)n)$. Indeed this is true not just for first-order $\varphi$ but even in monadic second-order logic by [4].
2. If $\mathcal{D}_k$ is the class of graphs of degree at most $k$, then $G \models \varphi$ is decidable in time $\mathcal{O}(f(l)n)$. This is established by Seese in [31].
3. If $\mathsf{LTW}_t$ is the class of graphs of local treewidth bounded by a function $t$, then $G \models \varphi$ is decidable in time $\mathcal{O}(f(l)n^2)$ by a result of Frick and Grohe [20].
4. If $\mathcal{M}_k$ is the class of graphs excluding $K_k$ as a minor, then $G \models \varphi$ is decidable in time $\mathcal{O}(f(l)n^5)$ by results of Flum and Grohe [18].

5. If $\mathsf{LEM}_t$ is the class of graphs with locally excluded minors given by $t$, then $G \models \varphi$ is decidable in time $\mathcal{O}(f(l)n^6)$ by a result of Dawar et al. [9].

These results are established by a combination of two essential methods. One is sometimes called the method of *automata* or the method of *decompositions*. The other is based on the *locality* of first-order logic. These two basic methods are best illustrated by the first two results on the list above.

For two graphs $G$ and $H$ and tuples of vertices $\mathbf{u}$ and $\mathbf{v}$ we write $(G, \mathbf{u}) \equiv_m (H, \mathbf{v})$ to denote that any formula $\varphi(\mathbf{x})$ with quantifier depth at most $m$ is true of $\mathbf{u}$ in $G$ if, and only if, it is true of $\mathbf{v}$ in $H$. Two key facts about this equivalence relation are (1) that, for any fixed $m$ and fixed length of tuple, it has finite index and (2) that it is a congruence with respect to a certain gluing operation. That is, if $\mathbf{v}$ is a tuple of vertices inducing the same subgraph in both $G$ and $H$, let $G \oplus_{\mathbf{v}} H$ denote the graph obtained by taking the disjoint union of $G$ and $H$ while identifying the vertices in $\mathbf{v}$. Then, it can be shown that the $\equiv_m$ equivalence class of $(G \oplus_{\mathbf{v}} H, \mathbf{v})$ is determined by the classes of $(G, \mathbf{v})$ and $(H, \mathbf{v})$ respectively. Since graphs in $\mathcal{T}_k$ can be constructed from a finite collection of graphs (i.e. the graphs with at most $k$ vertices) using this gluing operation (and some vertex renaming operations needed for technical reasons), we can use dynamic programming to determine the $\equiv_m$-class of an arbitrary graph in $\mathcal{T}_k$ in linear time from its tree decomposition.

Abstractly, the method of decompositions can be formulated as follows. Suppose $\mathcal{C}$ is a class of graphs such that there is a finite class $\mathcal{B}$ and a finite collection of operations Op such that:

- $\mathcal{C}$ is contained in the closure of $\mathcal{B}$ under the operations in Op;
- there is a polynomial-time algorithm which constructs, given any $G \in \mathcal{C}$ an Op-decomposition of $G$ over $\mathcal{B}$; and
- for each $m$, the equivalence relation $\equiv_m$ is an *effective congruence* with respect to all the operations $o \in$ Op (by which we mean that the $\equiv_m$ class of $o(G_1, \ldots, G_s)$ can be computed from the classes of $G_1, \ldots, G_s$),

then, satisfaction of first-order formulas for graphs in $\mathcal{C}$ is fixed-parameter tractable.

More generally, instead of requiring $\mathcal{B}$ to be finite, it suffices that first-order satisfaction is itself fixed-parameter tractable on $\mathcal{B}$. Indeed, result (4) above, on classes of graphs that exclude a minor, is obtained by considering a tree-decomposition of graphs in such a class *over* a class of bounded local treewidth and then using the result (3).

Another possible relaxation of the method is to replace $\equiv_m$ by some other sequence $\sim_m$ of congruence relations. The properties required to make this work are that for every first-order formula $\varphi$ there is an $m$ such that $\varphi$ is invariant under $\sim_m$ and that for each $m$, $\sim_m$ is a relation of finite index. In this context, it should be noted that taking $G \sim_m H$ to denote that $G$ and $H$ cannot be distinguished by any formula of *length* at most $m$ does not yield a congruence relation even with respect to disjoint union. Indeed, it was shown in [10] that there is no elementary function $e$ such that $G_1 \sim_{e(m)} H_1$ and $G_2 \sim_{e(m)} H_2$ implies $G_1 \oplus G_2 \sim_m H_1 \oplus H_2$.

In contrast, the proof of result (2) above is based on the *locality* of first-order logic. This property essentially says that the truth of a formula $\varphi$ in a graph $G$ can be determined by examining local neighbourhoods inside $G$. A precise statement is given by Gaifman's locality theorem [21] the statement of which requires some definitions.

For every integer $r \geq 0$, let $\delta(x,y) \leq r$ denote the first-order formula expressing that the distance between $x$ and $y$ in the Gaifman graph is at most $r$. Let $\delta(x,y) > r$ denote the negation of this formula. Note that the quanfier rank of $\delta(x,y) \leq r$ is bounded by $r$. A *basic local sentence* is a sentence of the form

$$(\exists x_1) \cdots (\exists x_n) \left( \bigwedge_{i \neq j} \delta(x_i, x_j) > 2r \wedge \bigwedge_i \psi^{N^r(x_i)}(x_i) \right), \tag{1}$$

where $\psi$ is a first-order formula with one free variable. Here, $\psi^{N^r(x_i)}(x_i)$ stands for the relativization of $\psi$ to $N^r(x_i)$; that is, the subformulas of $\psi$ of the form $(\exists x)(\theta)$ are replaced by $(\exists x)(\delta(x,x_i) \leq r \wedge \theta)$, and the subformulas of the form $(\forall x)(\theta)$ are replaced by $(\forall x)(\delta(x,x_i) \leq r \rightarrow \theta)$.

**THEOREM 1.***[Gaifman Locality] Every first-order sentence is equivalent to a Boolean combination of basic local sentences.*

We call the Boolean combination of basic local sentences that is equivalent to a given first-order sentence $\varphi$ a *Gaifman normal form* of $\varphi$. Since the proof of Theorem 1 (see for instance [15, Thm 2.5.1]) gives an effective construction of the Gaifman normal form from $\varphi$, to prove (2), it suffices to consider how a basic local sentence can be evaluated. Since, in a graph of bounded degree, there is a bound on the size of neighbourhoods, we can easily (in linear time) label elements by whether or not they satisfy the formulas $\psi^{N^r(x_i)}(x_i)$. The problem then reduces to determining in a vertex-coloured graph whether there is a large enough $r$-scattered set of a given colour. This can be done easily enough on graphs of bounded degree. However, Frick and Grohe [20] show that this can be solved in a somewhat more general setting giving an abstract method of locality. See [23, Sec. 4] for a very readable account.

The abstract formulation of the method of locality is as follows. Suppose we have a function, associating an integer parameter $k_G$ with each graph $G$. Suppose further that we have an algorithm which, given a graph $G$ and a formula $\varphi$ decides $G \models \varphi$ in time $g(l, k_G)n^c$ for some computable $g$ and some constant $c$. Finally, let $\mathcal{C}$ be a class of graphs of *bounded local k*. That is, there is a computable function $t$ such that for every $G \in \mathcal{C}$ and every vertex $v$ in $G$, $k_{N^r_G(v)} < t(r)$. Then, there is an algorithm which decides $G \models \varphi$ in time $f(l)n^{c+1}$ for some computable $f$.

It is this general localisation principle that gives us (3) from (1) above. It may seem that (5) follows from (4) by a similar application of the method of locality. However, while the result in [18] gives, for each $k$, a fixed-parameter tractable algorithm for deciding $G \models \varphi$ for classes that exclude $K_k$ as a minor, it is not clear from the proof that the parameter dependence is computable from $k$. The proof relies on Robertson-Seymour decompositions which do not yield computable bounds. Thus, the result in [9] relies on rather different decompositions.

**Nowhere-Dense Classes** As of this writing, it remains an open question whether the fixed-parameter tractability of first-order satisfaction can be pushed beyond the classes of locally excluded minors. In particular, the box at the bottom of Figure 1, containing the nowhere-dense classes, is an interesting case. This property was identified by Nešetřil and Ossona de Mendez in [28, 29]. They show that it is closely related to a property of classes of graphs called *quasi-wideness* in [7]. They give strong evidence that this property is the natural limit for methods which rely on the sparsity of graphs. To be precise, they associate the following parameter with any infinite class $\mathcal{C}$ of graphs.

$$d_{\mathcal{C}} = \lim_{r\to\infty} \limsup_{G\in\mathcal{C}_r} \frac{\log ||G||}{\log |G|},$$

where $\mathcal{C}_r$ denotes the collection of graphs that can be obtained as minors of a graph in $\mathcal{C}$ by contracting neighbourhoods of radius at most $r$. As usual, $||G||$ and $|G|$ denote the number of edges and the number of vertices in $G$ respectively. The remarkable result they then prove is what they call the *trichotomy theorem* [29] which states that $d_{\mathcal{C}}$ only takes values 0, 1 and 2. Moreover, the nowhere-dense classes are exactly the ones where it does not take value 2.

So, could it be that first-order satisfaction is fixed-parameter tractable on all nowhere-dense classes? The connection with quasi-wideness provides some clues. It is easy to establish that problems such as INDEPENDENT SET are fixed-parameter tractable on such classes. A paper in the present volume [11] shows that variations on the DOMINATING SET problem are also fixed-parameter tractable. However, it remains a challenge to extend this to all first-order definable properties. In particular, such a result would generalise the tractability of first-order logic on excluded minor classes, which depends on deep decomposition theorems. In contrast, the results in [11] depend on rather more straightforward combinatorial properties of nowhere-dense classes.

## References

[1] D.M. Barrington, N. Immerman, and H. Straubing. On uniformity within $NC_1$. *Journal of Computer and System Sciences*, 41:274–306, 1990.

[2] A. Blass, Y. Gurevich, and S. Shelah. On polynomial time computation over unordered structures. *Journal of Symbolic Logic*, 67(3):1093–1125, 2002.

[3] J-Y. Cai, M. Fürer, and N. Immerman. An optimal lower bound on the number of variables for graph identification. *Combinatorica*, 12(4):389–410, 1992.

[4] B. Courcelle. Graph rewriting: An algebraic and logic approach. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics (B)*, pages 193–242. Elsevier, 1990.

[5] B. Courcelle, J.A. Makowsky, and U. Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory Comput. Syst.*, 33:125–150, 2000.

[6] B. Courcelle and S. Olariu. Upper bounds to the clique width of graphs. *Discrete Applied Mathematics*, 101:77–114, 2000.

[7] A. Dawar. Homomorphism preservation on quasi-wide classes. *J. Compute and System Sciences*, 2009. to appear. See arXiv:0811.4497v1 [cs.LO].

[8] A. Dawar, M. Grohe, B. Holm, and B. Laubner. Logics with rank operators. In *Proc. 24th IEEE Symp. on Logic in Computer Science*, pages 113–122, 2009.

[9] A. Dawar, M. Grohe, and S. Kreutzer. Locally excluding a minor. In *Proc. 22nd IEEE Symp. on Logic in Computer Science*, pages 270–279, 2007.

[10] A. Dawar, M. Grohe, S. Kreutzer, and N. Schweikardt. Model theory makes formulas large. In *ICALP'07:Proc. 34th International Colloquium on Automata, Languages and Programming*, LNCS. Springer, 2007.

[11] A. Dawar and S. Kreutzer. Domination problems in nowhere-dense classes of graphs. In *FSTTCS 2009*, 2009.

[12] A. Dawar and D. Richerby. Fixed-point logics with nondeterministic choice. *Journal of Logic and Computation*, 13:503–530, 2003.

[13] A. Dawar, D. Richerby, and B. Rossman. Choiceless polynomial time, counting and the Cai-Fürer-Immerman graphs. *Annals of Pure and Applied Logic*, 152:31–50, 2008.

[14] R. Diestel. *Graph Theory*. Springer, 3rd edition, 2005.

[15] H-D. Ebbinghaus and J. Flum. *Finite Model Theory*. Springer, 2nd edition, 1999.

[16] D. Eppstein. Diameter and treewidth in minor-closed graph families. *Algorithmica*, 27:275–291, 2000.

[17] R. Fagin. Generalized first-order spectra and polynomial-time recognizable sets. In R. M. Karp, editor, *Complexity of Computation, SIAM-AMS Proceedings, Vol 7*, pages 43–73, 1974.

[18] J. Flum and M. Grohe. Fixed-parameter tractability, definability, and model checking. *SIAM Journal on Computing*, 31:113 – 145, 2001.

[19] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, 2006.

[20] M. Frick and M. Grohe. Deciding first-order properties of locally tree-decomposable structures. *Journal of the ACM*, 48:1184–1206, 2001.

[21] H. Gaifman. On local and non-local properties. In J. Stern, editor, *Proceedings of the Herbrand Symposium Logic Colloquium '81*, pages 105–135. North-Holland, 1982.

[22] F. Gire and H. Hoang. An extension of fixpoint logic with a symmetry-based choice construct. *Information and Computation*, 144:40–65, 1998.

[23] M. Grohe. Logic, graphs, and algorithms. In J. Flum, E. Grädel, and T. Wilke, editors, *Logic and Automata: History and Perspectives*, pages 357–422. Amsterdam University Press, 2007.

[24] M. Grohe. The quest for a logic capturing PTIME. In *Proc. 22nd IEEE Symp. on Logic in Computer Science*, pages 267–271, 2008.

[25] N. Immerman. Relational queries computable in polynomial time. *Information and Control*, 68:86–104, 1986.

[26] L. Libkin. *Elements of Finite Model Theory*. Springer, 2004.

[27] B. Mohar and C. Thomassen. *Graphs on Surfaces*. Johns Hopkins University Press, 2001.

[28] J. Nešetřil and P. Ossona de Mendez. First-order properties of nowhere dense structures. *Journal of Symbolic Logic*, 2009. to appear.

[29] J. Nešetřil and P. Ossona de Mendez. On nowhere dense graphs. *European Journal of Combinatorics*, 2009. to appear.

[30] J. Nešetřil and P. Ossona de Mendez. The grad of a graph and classes with bounded expansion. In *International Colloquium on Graph Theory*, pages 101 – 106, 2005.

[31] D. Seese. Linear time computable problems and first-order descriptions. *Math. Struct. in Comp. Science*, 6:505–526, 1996.

[32] L. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3:1–22, 1976.

[33] A. Stolbouskin and M. Taitslin. Is first order contained in an initial segment of PTIME? In *Computer Science Logic 94*, volume 933 of *LNCS*. Springer-Verlag, 1995.

[34] M. Y. Vardi. The complexity of relational query languages. In *Proc. of the 14th ACM Symp. on the Theory of Computing*, pages 137–146, 1982.