1

Unifying Themes in Finite Model Theory

Scott Weinstein

One of the fundamental insights of mathematical logic is that our understanding of mathematical phenomena is enriched by elevating the languages we use to describe mathematical structures to objects of explicit study. If mathematics is the science of pattern, then the media through which we discern patterns, as well as the structures in which we discern them, command our attention. It is this aspect of logic which is most prominent in model theory, "the branch of mathematical logic which deals with the relation between a formal language and its interpretations" [21]. No wonder, then, that mathematical logic, in general, and finite model theory, the specialization of model theory to finite structures, in particular, should find manifold applications in computer science: from specifying programs to querying databases, computer science is rife with phenomena whose understanding requires close attention to the interaction between language and structure.

As with most branches of mathematics, the growth of mathematical logic may be seen as fueled by its applications. The very birth of set theory was occasioned by Cantor's investigations in real analysis, on subjects themselves motivated by developments in nineteenth-century physics; and the study of subsets of the real line has remained the source of some of the deepest results of contemporary set theory. At the same time, model theory has matured through the development of ever deeper applications to algebra. The interplay between language and structure, characteristic of logic, may be discerned in all these developments. From the focus on definability hierarchies in descriptive set theory, to the classification of structures up to elementary equivalence in classical model theory, logic seeks order in the universe of mathematics through the medium of formal languages.

As noted, finite model theory too has grown with its applications, in this instance not to analysis or algebra, but to combinatorics and computer science. Beginning with connections to automata theory, finite model theory has developed through a broader and broader range of applications to problems in graph theory, complexity theory, database theory, computer-aided verification, and artificial intelligence. And though its applications have demanded the development of new techniques, which have given the subject a distinctive character as compared to classical model theory, the fundamental focus on organizing and understanding phenomena through attention to the relation between language and structure remains prominent. Indeed, the detailed investigation of definability hierarchies and classifications of finite structures up to equivalence relations coarser than elementary equivalence, which are defined in terms of a wide variety of fragments of first-order, second-order, fixed-point, and infinitary logics, is a hallmark of finite model theory. The remaining sections of this chapter will highlight common themes among the chapters to follow.

1.1 Definability Theory

The volume begins with a chapter by Phokion Kolaitis, "On the expressive power of logics on finite models", which surveys major topics in the theory of definability in the context of finite structures. "The theory of definability is the branch of logic which studies the complexity of concepts by looking at the grammatical complexity of their definitions." [3]. This characterization indicates that the theory of definability has two main aspects:

- to establish a classification of concepts in terms of definitional complexity
- to establish that such classification is in some way informative about the intrinsic or intuitive "complexity" of the concepts thus classified.

Chapter 2 provides an extended treatment of both these aspects of definability theory, which reappear throughout the volume as important themes in finite model theory and its applications.

1.1.1 Classification of Concepts in Terms of Definitional Complexity

In the context of finite model theory, the "concepts" with which we are concerned are queries on classes of finite relational structures. Chapter 2 provides precise definitions of these notions; for the purposes of introduction, let us focus on Boolean queries on a particular set of finite undirected graphs as follows. Let G_n be the collection of undirected graphs with vertex set $[n]$ $(= \{1, \ldots, n\})$, and let $\mathsf{G} = \bigcup_n \mathsf{G}_n$. Thus, each $G \in \mathsf{G}$ has a vertex set $V^G = [n]$, for some n, and an irreflexive and symmetric edge relation $E^G \subseteq [n] \times [n]$. A Boolean query Q on G is just an isomorphism-closed subset of G, that is, $Q \subseteq \mathsf{G}$ is a Boolean query if and only if, for all $G, H \in \mathsf{G}$,

$$G \cong H \Longrightarrow (G \in Q \Leftrightarrow H \in Q).$$

Logical languages provide a natural means for classifying Boolean queries. A logical language L consists of a set of L-sentences, S_L, and an L-satisfaction relation $\models_L$. In the current setting, we may understand $\models_L$ as a relation between graphs $G \in \mathsf{G}$ and sentences $\varphi \in S_L$: $G \models_L \varphi$, if and only if G satisfies the condition expressed by φ. A fundamental notion is the Boolean query, $\varphi[\mathsf{G}]$, defined by an L-sentence, φ:

$$\varphi[\mathsf{G}] = \{G \in \mathsf{G} \mid G \models_L \varphi\}.$$

A Boolean query Q on G is L-definable if and only if there is an L-sentence φ with $Q = \varphi[\mathsf{G}]$.

Let us look at some examples. Consider the following Boolean queries:

Size_n the set of graphs of size n;
Diam_n the set of graphs of diameter $\leq n$;
Color_k the set of k-colorable graphs;
Conn the set of connected graphs;
Card_X the set of graphs of size n for some $n \in X \subseteq \mathsf{N}$.

The first two queries are defined by first-order sentences σ_n and δ_n, respectively, for each n; for example, the query Size_2 is defined by the first-order sentence σ_2,

$$\exists x \exists y (x \neq y) \wedge \neg \exists x \exists y \exists z (x \neq y \wedge x \neq z \wedge y \neq z),$$

and the query Diam_2 is defined by the first-order sentence δ_2,

$$\forall x \forall y (x = y \vee Exy \vee \exists z (Exz \wedge Ezy)).$$

For each k, the third query is defined by a sentence χ_k of existential monadic second-order logic, that is, the fragment of second-order logic consisting of sentences all of whose second-order quantifiers are existential, bind monadic predicate symbols, and do not occur within the scope of any first-order quantifier or truth-functional connective; for example, Color_2 is defined by the sentence χ_2,

$$\exists Z \forall x \forall y (Exy \rightarrow (Zx \leftrightarrow \neg Zy)).$$

The next query is defined by a sentence γ of $L_{\omega_1 \omega}$, the infinitary logic obtained by adding the operations of countable conjunction and countable disjunction to first-order logic, as follows:

$$\bigvee_{n \in \mathsf{N}} \delta_n.$$

Note that in general, γ expresses the condition that a graph has bounded diameter – over G, this condition coincides with connectedness. Finally, for each $X \subset \mathsf{N}$, the query Card_X is defined by a sentence κ_X of $L_{\omega_1 \omega}$ as follows:

$$\bigvee_{n \in X} \sigma_n.$$

Now, broadly speaking, definability theory provides techniques for determining whether or not given queries, or collections of queries, are definable in a specified logic L, and attempts to extract useful information about queries from the fact that they are L-definable. For example, Chap. 2 develops tools to show that neither Color_k nor Conn is first-order definable, and thus stronger logics are needed to express such basic combinatorial properties.

1.1.2 What More Do We Know When We Know a Concept Is *L*-Definable?

This, of course, depends on L. One striking feature of finite model theory has been that it has drawn attention to the fact that a great deal of interesting information about Boolean queries *can* be extracted from the fact that they are definable in familiar logical languages, and, perhaps even more striking, it has highlighted the importance of some natural, though hitherto neglected, fragments of well-studied languages, such as the finite variable fragments of first-order logic and infinitary logic discussed below.

Before we proceed to explore this aspect of definability theory in the context of finite model theory, let us reflect for a moment on a paradigmatic example of extracting information from the fact that a set is definable in a certain way: the celebrated result of Cantor concerning the cardinality of closed sets of real numbers. Recall that a closed set can be defined as the complement of a countable union of open intervals with rational endpoints (which implies, in modern parlance, that a closed set is $\mathbf{\Pi}^0_1$). Note that we may infer from this definability characterization that there are only $2^{\aleph_0}$ closed sets of reals, while there are $2^{(2^{\aleph_0})}$ sets of reals altogether. Cantor showed that closed sets satisfy a very strong dichotomy with respect to their cardinalities: every infinite closed set is either countable or of cardinality $2^{\aleph_0}$, that is, there is no closed set witnessing a cardinality strictly between $\aleph_0$ and $2^{\aleph_0}$. On the basis of his success with closed sets, Cantor was motivated to formulate the Continuum Hypothesis (CH): the conjecture that all infinite sets of reals satisfy this strong cardinality dichotomy. In 1963, Cohen established that if Zermelo–Fraenkel set theory with the Axiom of Choice (ZFC) is consistent, then it is consistent with the statement that there is an infinite set of reals whose cardinality is neither $\aleph_0$ nor $2^{\aleph_0}$, that is, $\mathrm{ZFC} + \neg\mathrm{CH}$ is consistent relative to ZFC. Thus, Cantor's result shows how it is possible to gain significant structural information about a concept from the knowledge that it admits a "simple" enough definition. In what Moshovakis describes as "one of the first important results of descriptive set theory" [52], Suslin generalized Cantor's solution of the continuum problem from closed sets to analytic sets, that is, projections of closed sets ($\mathbf{\Sigma}^1_1$ sets). Indeed, he showed that every uncountable analytic set contains a nonempty perfect set, as Cantor had established for closed sets. Further generalization of this property to sets whose definitional complexity is greater, even to co-analytic sets, is not possible on the basis of ZFC.

Finite model theory provides a rich collection of phenomena which illustrate this paradigm of wresting structural information about concepts from definability conditions. Let us begin with an example from asymptotic combinatorics which touches on topics dealt with in detail in Chaps. 2 and 4. Let Q be a Boolean query on G. Recall that $\mathrm{card}(\mathsf{G}_n) = 2^{\binom{n}{2}}$. The density μ_n of Q at G_n is defined as follows:

$$\mu_n(Q) = \mathrm{card}(Q \cap \mathsf{G}_n) \cdot 2^{-\binom{n}{2}}.$$

The limit density $\mu(Q) = \lim_{n \longrightarrow \infty} \mu_n(Q)$ may or may not exist, depending on the query Q. For example, if $X \subseteq \mathsf{N}$ is finite or cofinite, then $\mu(\mathrm{Card}_X)$ is 0 or 1, respectively, whereas $\mu(\mathrm{Card}_X)$ is undefined if X is infinite and coinfinite. Thus, definability in $L_{\omega_1\omega}$ does not guarantee that a query has a limit density. Indeed, for every graph $G \in \mathsf{G}$, the query

Isom_G the set of graphs isomorphic to G

is definable by a single first-order sentence ι_G; for example, the graph G with $V^G = \{1, 2\}$ and $E^G = \{\langle 1, 2\rangle, \langle 2, 1\rangle\}$ is defined, up to isomorphism, by the first-order sentence

$$\kappa_2 \wedge \forall x \forall y (Exy \leftrightarrow x \neq y).$$

It follows that for each query Q, the $L_{\omega_1\omega}$ sentence

$$\bigvee_{G \in Q} \iota_G$$

defines Q. Thus, no information flows from the fact that a query is $L_{\omega_1\omega}$-definable, in particular, no information about the limit density of Conn is forthcoming from its definability in $L_{\omega_1\omega}$. (Note that the expressive power of $L_{\omega_1\omega}$ *is* limited on the collection of all finite and infinite structures; indeed, from cardinality considerations, there is an ordinal α such that the isomorphism type of $\langle \alpha, < \rangle$ cannot be characterized by a sentence of $L_{\omega_1\omega}$.) Perhaps we can find another source for such information.

Let us consider the query Diam_2. How can we compute its density at G_n? It will be useful to think of this in probabilistic terms. The density of a query Q at G_n is just the probability of the event $Q \cap \mathsf{G}_n$ with respect to the uniform measure on G_n, that is, the measure u with $\mathsf{u}(\{G\}) = 2^{-\binom{n}{2}}$, for each $G \in \mathsf{G}_n$. The measure u may be thought of as follows: for each pair of vertices $1 \leq i < j \leq n$, we flip a fair coin to determine whether or not there is an edge between i and j. This point of view facilitates the computation of a useful approximation to the density of Diam_2. For a fixed pair of distinct vertices i and j, the probability that a distinct vertex k is a neighbor of both i and j is $1/4$.Therefore, the probability that none of the $n-2$ vertices distinct from i and j is a neighbor of them both is $(3/4)^{(n-2)}$. It is now easy to see that the probability that some pair of vertices lacks a common neighbor is bounded by $\binom{n}{2} \cdot (3/4)^{(n-2)}$. It follows at once that

$$\mu_n(\mathrm{Diam}_2) \geq 1 - \binom{n}{2} \cdot \left(\frac{3}{4}\right)^{(n-2)}.$$

But,

$$\lim_{n\to\infty} \binom{n}{2} \cdot \left(\frac{3}{4}\right)^{(n-2)} = 0.$$

Therefore, $\mu(\mathrm{Diam}_2) = 1$. Note that $\mathrm{Diam}_2 \subseteq \mathrm{Conn}$ (cast logically, δ_2 implies γ), and thus, $\mu(\mathrm{Conn}) = 1$. We shall see that this is no isolated phenomenon, but rather one instance of a beautiful dichotomy revealed by definability theory.

As observed above, there are continuum-many queries whose limit density is undefined; moreover, it is not hard to see that for every real number $r \in [0, 1]$, there is a query with limit density r. A noteworthy dichotomy is enshrined in the following definition. A logic L satisfies the 0–1 law with respect to the uniform measure on G if and only if, for all L-definable queries Q,

$$\mu(Q) = 0 \text{ or } \mu(Q) = 1.$$

A 0–1 law codifies important structural information about L-definable queries and provides a useful tool for establishing that specific queries are not L-definable; for example, none of the queries Card_X, for X infinite and coinfinite, is L-definable if L satisfies the 0–1 law. It is remarkable that some natural logics satisfy the 0–1 law. The first such result is due to Glebskii et al. [28] and, independently, to Fagin [26], who established that first-order logic satisfies the 0–1 law with respect to the uniform measure. A brief look at an argument for this result will be instructive.

The query Diam_2 is an *extension property* – it requires that every pair of vertices share a common neighbor. A generalization of this is the (m, n)-extension property: this requires that for every pair of disjoint sets of vertices $\{x_1, \ldots, x_m\}$ and $\{y_1, \ldots, y_n\}$, there is a vertex z which is a neighbor of all the x_i and none of the y_j. It is easy to see that this condition is expressible by a first-order sentence $\eta_{m,n}$ (with $m+n+1$ variables), and that, just as the limit density of Diam_2 is 1, so too $\mu(\eta_{m,n}[\mathsf{G}]) = 1$, for all m, n with $m+n > 0$. Let η_k be the conjunction $\eta_{m,n}$ with $k = m + n + 1$. Each η_k is a first-order sentence with k variables expressing a query with limit density 1; moreover, for all $l \leq k$, η_k implies η_l. Therefore, by the Compactness Theorem for first-order logic, the set of sentences $\Gamma = \{\eta_k \mid k > 1\}$ is consistent. To complete the argument, it suffices to show that for every first-order sentence φ, there is a k such that η_k implies φ, or η_k implies $\neg\varphi$; indeed, if η_k implies φ, then $\mu(\varphi[\mathsf{G}]) = 1$, and if η_k implies $\neg\varphi$, then $\mu(\varphi[\mathsf{G}]) = 0$. Now, Γ has no finite models, and is $\aleph_0$-categorical, that is, any two countable models of Γ are isomorphic (the back-and-forth argument, used by Cantor to prove that the rational numbers are, up to isomorphism, the unique countable dense linear order without endpoints, may be deployed here; compare Chap. 4). It follows at once, via the Löwenheim–Skolem Theorem, that Γ axiomatizes a complete first-order theory. From this, another application of the Compactness Theorem for first-order logic yields the conclusion that for every first-order sentence φ, there is a k such that η_k implies φ, or η_k implies $\neg\varphi$. Can we say, for a given first-order sentence φ, how large a k is required? Kolaitis and Vardi [48] showed that the answer to this question leads to a significant extension of the 0–1 law to a rich fragment of infinitary logic.

1.1.3 Logics with Finitely Many Variables

For each $k \geq 1$, FO^k is the fragment of first-order logic consisting of exactly those formulas all of whose variables, both free and bound, are among $x_1, \ldots, x_k$. To understand the effect of this restriction, it is useful to observe that variables may be reused within such sentences, so that, for example, the queries Diam_k are all FO^3-definable. Here is a sentence of FO^3 that defines Diam_3:

$$\begin{aligned}\forall x_1 \forall x_2 (\ & x_1 = x_2 \vee E x_1 x_2 \vee \\ & \exists x_3 (E x_1 x_3 \wedge E x_3 x_2) \vee \\ & \exists x_3 (E x_1 x_3 \wedge \exists x_1 (E x_3 x_1 \wedge E x_1 x_2))).\end{aligned}$$

We have already noted that the logic $L_{\omega_1\omega}$ is too powerful to be of interest in the context of finite model theory, since every query is definable in this logic. The logic $L^k_{\omega_1\omega}$ is the fragment of $L_{\omega_1\omega}$ consisting of exactly those formulas all of whose variables, both free and bound, are among $x_1, \ldots, x_k$; $L^\omega_{\omega_1\omega} = \bigcup_k L^k_{\omega_1\omega}$. In light of the FO^3-definability of Diam_k, observe that Conn is $L^3_{\omega_1\omega}$−definable. Indeed, as discussed in Chap. 2, all queries definable in the fixed-point logics **LFP**, **IFP**, and **PFP**, which provide means for definition of relations by recursion, for example the transitive closure of the edge relation, are $L^\omega_{\omega_1\omega}$-definable (note that, in general, these inclusions fail on collections of finite and infinite structures; for example, the notion of well-foundedness is **LFP**-definable on the class of all directed graphs, but is not even definable in the powerful infinitary logic $L_{\infty\omega}$ discussed below).

Kolaitis and Vardi established that the 0–1 law holds for $L^\omega_{\omega_1\omega}$ with respect to the uniform measures on G_n. In particular, they showed that for every $k > 1$, η_k axiomatizes a complete $L^k_{\omega_1\omega}$ theory. Thus, even though $L^\omega_{\omega_1\omega}$ has expressive power sufficient to encompass various forms of recursion, it retains some of the structural simplicity of first-order logic; indeed, every $L^\omega_{\omega_1\omega}$-definable query or its complement is implied by a first-order definable query of limit density 1 (the analogy with Suslin's generalization of the theorem of Cantor mentioned above is irresistible). This result gave a coherent explanation for earlier work on 0–1 laws for fixed-point logics (see [14, 47]), and thereby highlighted the important role that finite-variable logics can play in definability theory over finite structures. Hella, Kolaitis, and Luosto [41] further illuminated the situation by showing that FO and $L^\omega_{\omega_1\omega}$ are *almost everywhere equivalent* with respect to the uniform measure, that is, there is a set $\mathcal{C} \subseteq \mathsf{G}$ of limit density one such that FO and $L^\omega_{\omega_1\omega}$ define exactly the same collection of queries over $\mathcal{C}$ (even including non-Boolean queries). Dawar [22], Grohe [33], and Otto [54] are valuable sources of information about the finite model theory of finite-variable logics. The following chapters offer many other compelling illustrations of the use of definability theory to yield insight into a wide range of mathematical and computational phenomena. Before exploring some of these examples, let us look at some other important notions from definability theory which receive extended treatment in Chap. 2.

1.1.4 Distinguishing Structures: *L*-Equivalence and Comparison Games

One approach to the question whether a query Q is definable in a logic L is to ask whether Q distinguishes between graphs which are indistinguishable from the point of view of L. Two graphs G and H are L-equivalent ($G \equiv_L H$), that is, indistinguishable from the point of view of L, if and only if, for every L-definable query Q,

$$G \in Q \iff H \in Q.$$

Clearly, a query Q must be closed under L-equivalence if it is L-definable.

When L is first-order logic, L-equivalence is the notion of elementary equivalence familiar from classical model theory. The classification of infinite structures up to elementary equivalence plays a central role in classical model theory and in its applications to algebra and analysis. On the other hand, as observed above, elementary equivalence coincides with isomorphism on G (and on the class of finite structures in general), so the foregoing necessary condition is deprived of direct application to definability over G with respect to any logic extending FO. This suggests that analysis of L-equivalence for logics L weaker than, or orthogonal to, first-order logic may be of paramount importance in the context of finite model theory. Indeed, this is the case. Let us approach the matter from the point of view of combinatorial comparison games between graphs.

Suppose we want to compare (finite or countably infinite) graphs G and H with the object of determining whether or not they are isomorphic. One way of doing so (inspired by the celebrated Cantor "back-and-forth" argument mentioned above) would be to play the following game. The game has two players, Spoiler and Duplicator; the equipment for the game consists of "boards" corresponding to the graphs G and H and pebbles $a_1, a_2, \dots$ and $b_1, b_2, \dots$. The game is organized into rounds $r_1, r_2, \dots$. At each round r_i the Spoiler plays first and picks one of the pair of pebbles a_i or b_i to play onto a vertex of G or H, respectively; the Duplicator then plays the remaining pebble of the pair onto a vertex of the structure into which the Spoiler did not play. This completes the round. Let v_i (and w_i) be the vertex of G (and of H, respectively) pebbled at round i, let G_i and H_i be the subgraphs of G and H induced by $\{v_1, \dots, v_i\}$ and $\{w_1, \dots, w_i\}$, respectively, and let $R_i = \{\langle v_j, w_j\rangle \mid 1 \leq j \leq i\}$. The Duplicator loses the game at round r_i if the relation R_i fails to be the graph of an isomorphism from G_i onto H_i. The Duplicator wins the game if she does not lose at any round. The Duplicator has a winning strategy for the game if she has a method of play which results in a win for her no matter how the Spoiler plays. In this case, we say that G is partially isomorphic to H ($G \cong_p H$).

It is easy to see that the Duplicator has a winning strategy for this game played on finite or countably infinite graphs G and H if and only if G is isomorphic to H. Indeed, if I is an isomorphism from G onto H, and the Spoiler pebbles the vertex v in G at some round, then the Duplicator will guarantee a win by pebbling $I(v)$ in H (and similarly, if the Spoiler plays onto w in H, then the Duplicator answers by playing onto $I^{-1}(w)$ in G). On the other hand, suppose the Duplicator has a winning strategy for the game played on G and H. Then, she can win against the following strategy of Spoiler. The Spoiler can enumerate the vertices of G as $s_0, s_1, \ldots$ and the vertices of H as $t_0, t_1, \ldots$. Now the Spoiler plays according to the following strategy. For $i \geq 0$, at round r_{2i+1} he places the pebble a_{2i+1} on s_i and at round r_{2i+2} he places the pebble b_{2i+2} on t_i. The Duplicator now answers the Spoiler's moves according to her winning strategy. It follows at once that the relation $R = \bigcup_{i \in \mathsf{N}} R_i$ is the graph of an isomorphism from G onto H. So, if G and H are countable, and $G \cong_p H$, then $G \cong H$. Carol Karp [46] established an interesting connection between partial isomorphism and logical definability: arbitrary graphs G and H are partially isomorphic if and only if they are $L_{\infty\omega}$-equivalent ($L_{\infty\omega}$ strengthens $L_{\omega_1\omega}$ by allowing conjunctions over arbitrary, not necessarily countable, sets of formulas).

Various modifications of this game, which deprive the players of some of their access to resources, or alter the winning condition, or add rules that restrict legitimate play, lead to useful characterizations of equivalence for much weaker languages. Let us consider some examples of these.

First, we might restrict the number of pebble pairs that are available for the game, and allow players to replay pebbles that they have played earlier in the game. If the Duplicator has a winning strategy for the foregoing game played on G and H when the equipment consists of only k pairs of pebbles, we say that G is k-partially isomorphic to H ($G \cong_p^k H$). This variant is discussed at length in Chap. 2 where a proof sketch of Barwise's result [9] that for all G and H, $G \cong_p^k H$ if and only if G is $L^k_{\infty\omega}$-equivalent to H is presented. We have already seen one application of this result to definability over G: it is easy to see that for all $G, H \in \mathsf{G}$, if $G \models \eta_k$ and $H \models \eta_k$, then G is k-partially isomorphic to H; it follows at once from Barwise's result that for every $L^k_{\infty\omega}$ sentence φ, η_k implies φ, or η_k implies $\neg\varphi$, which is the key step in Kolaitis and Vardi's proof of the 0–1 law for $L^k_{\infty\omega}$.

Second, we might restrict the length of play, so the Duplicator need only successfully respond to the Spoiler's moves through some fixed finite number n of rounds in order to win. This is called the n-round Ehrenfeucht–Fraïssé (E–F) game. As discussed in Chap. 2, these games give a characterization of definability in a hierarchy of fragments of first-order logic; in particular, the Duplicator has a winning strategy for the n-round E–F game played on G and H if and only if G and H are FO_n-equivalent, where FO_n is the fragment of first-order logic consisting of all sentences of quantifier rank bounded by n. This is the key to using logical indistinguishability to establish that queries are not first-order definable over G despite the fact that first-order indistinguishability coincides with isomorphism over G. In order to show that a query Q is not first-order definable, it suffices to show that for every n there are FO_n-equivalent G and H with $G \in Q$ and $H \notin Q$. Chapter 2 includes several examples of this technique, among them the queries Conn and Color_k mentioned earlier.

Third, we might require that beyond the first round, the Spoiler play onto a vertex that is adjacent to some vertex which has been pebbled at an earlier round. The single-pebble variant of the game thus restricted characterizes the relation of bisimilarity between vertex-colored directed graphs. Johan van Benthem first introduced this relation and recognized its significance in connection with the study of Kripke models for modal logic [11, 12]; the notion was rediscovered in the context of analyzing the "behavioral equivalence" of transition systems [42, 57]. Chapter 7 elucidates the fundamental importance of bisimilarity invariance in explaining various nice features of modal logic.

Fourth, we might require that the Spoiler always play onto a vertex of G. In this case, by virtue of the asymmetry of play, a win for Duplicator in the resulting game no longer characterizes an equivalence relation between graphs, but rather a preorder. In particular, the Duplicator has a winning strategy for this variant of the game if and only if every existential sentence of $L_{\infty\omega}$ which is true in G is also true in H. If, in addition, we relax the winning condition to require only that at the end of each round r_i the relation R_i is the graph of a homomorphism from G to H, then the Duplicator has a winning strategy if and only if every positive existential sentence of $L_{\infty\omega}$ that is true in G is also true in H. This last variant, in combination with the resource restriction on the number of pebbles discussed above, characterizes the positive existential fragment of $L^{\omega}_{\infty\omega}$. This fragment is of particular interest from the perspective of database theory, since it suffices to express every Datalog-definable query; several applications of this definability result are discussed in Chaps. 2 and 6.

1.1.5 Random Graphs and 0–1 Laws

Joel Spencer's chapter, "Logic and random structures" (Chap. 4), gives an exposition of a 0–1 law for first-order logic he and Saharon Shelah discovered [62], and related phenomena in the theory of random graphs. From the perspective of this theory, the uniform distribution on finite graphs considered above is an instance of a far more general scenario developed by Erdös and Renyi in [24]. From this perspective, one considers a sequence of finite probability spaces (G_n, μ^p_n), where the measure μ^p_n is determined by an "edge probability" p_n which is a function of n; the uniform distribution is just the special case where $p_n = .5$ for all n. Let us write $\mu^p(Q)$ for the limit probability of the query Q with respect to the sequence of measures μ^p_n, that is,

$$\mu^p(Q) = \lim_{n \longrightarrow \infty} \mu^p_n(Q \cap \mathsf{G}_n).$$

In this context, combinatorists have discovered that threshold phenomena arise, that is, there are queries Q and functions p with the property that for all q, if $q \ll p$, then $\mu^q(Q) = 0$, and if $p \ll q$, then $\mu^q(Q) = 1$. (Here, $p \ll q$ if and only if $\lim_{n \longrightarrow \infty} p_n/q_n = 0$.) One class of cases which arose naturally in the study of threshold phenomena is the edge probabilities $p(\alpha)_n = n^{-\alpha}$, for some real $\alpha \in (0,1)$. Spencer observed that among the many queries analyzed by graph theorists, none possessed a threshold of the form $n^{-\alpha}$ for $\alpha \in (0,1)$ and irrational. Shelah and Spencer discovered a definability result that provided an explanation for these threshold phenomena. They showed that for all $\alpha \in (0,1)$, if α is irrational, then first-order logic satisfies the 0–1 law with respect to $(\mathsf{G}_n, \mu_n^{p(\alpha)})$, that is, for every first-order definable query Q,

$$\mu^{p(\alpha)}(Q) = 0 \text{ or } \mu^{p(\alpha)}(Q) = 1.$$

This is an outstanding example of how definability considerations can provide insight through systematization of apparently disparate combinatorial facts. Further investigations of the complete theories $T^\alpha = \{\varphi \in \mathrm{FO} \mid \mu^{p(\alpha)}(\varphi[\mathsf{G}]) = 1\}$ have revealed interesting connections with classical model theory (see [8, 51]). This aspect of definability theory has also been prominent in computer science, as well as in combinatorics.

1.1.6 Constraint Satisfaction Problems

In Chap. 6 of the volume, "A logical approach to constraint satisfaction", Kolaitis and Vardi survey some applications of definability theory to the study of constraint satisfaction problems, a subject that is important in several areas of computer science, including artificial intelligence, database theory, and operations research. For example, the k-colorability problem for graphs may be formulated as a constraint satisfaction problem. Given a graph (V, E), we may think of its vertices as *variables*. We ask whether there is an assignment of k colors $c_1, \ldots, c_k$, one to each variable, so as to satisfy the constraint that adjacent variables are assigned distinct colors. Feder and Vardi [27] made the following important observation that advanced the understanding of the computational complexity of constraint satisfaction problems: they noted that all such problems may be formulated as homomorphism problems on suitable relational structures (in general, these relational structures will not be graphs). For a simple example using graphs, recall that a homomorphism h from $G = (V, E)$ to $H = (V', E')$ is a map satisfying the condition

$$Eab \Rightarrow E'h(a)h(b), \text{ for all } a, b \in V.$$

A graph G is k-colorable if and only if there is a homomorphism from G into K_k, the complete graph on k vertices (thought of as the colors $c_1, \ldots, c_k$) – the constraint that adjacent "variables" are assigned distinct colors by any homomorphism is enforced by the irreflexivity of the edge relation in K_k.

In general, a constraint satisfaction problem can be formulated as a homomorphism problem: given two classes of relational structures $\mathcal{A}$ and $\mathcal{B}$, the constraint satisfaction problem CSP($\mathcal{A},\mathcal{B}$) asks, for each pair of structures $A \in \mathcal{A}$ and $B \in \mathcal{B}$, whether or not there is a homomorphism from A to B. Insofar as the homomorphism problem in general is NP-complete, the search for "islands of tractability", that is, collections of structures $\mathcal{A}$ and $\mathcal{B}$ such that CSP($\mathcal{A},\mathcal{B}$) can be computed in polynomial time, is of interest.

Two cases which have been studied intensively are nonuniform and uniform constraint satisfaction problems. The case in which $\mathcal{B}$ is a singleton, $\{B\}$, and $\mathcal{A}$ is $\mathcal{U}$, the collection of all finite structures, is called the non-uniform constraint satisfaction problem with template B – CSP(B) for short; on the other hand, the constraint problem CSP($\mathcal{A},\mathcal{U}$) is called the uniform constraint satisfaction problem with input $\mathcal{A}$. To illustrate this terminology by the above example, for each k, the k-colorability problem is the nonuniform constraint satisfaction problem CSP(K_k). This is a suggestive example. Recall that the 2-colorability problem is solvable in polynomial time, while the k-colorability problem is NP-complete, for each $k \geq 3$. Recall too (see [50]) that if P $\neq$ NP, then there are problems in NP which are neither NP-complete nor in P. Could it be that nonuniform constraint satisfaction problems are so special that they would exhibit the following remarkable dichotomy?

F–V For every template B, CSP(B) is either in P or is NP-complete.

This is the well-known Feder–Vardi Dichotomy Conjecture, which was articulated in [27] as a generalization of a theorem of Schaefer [61] that established the dichotomy for the case of templates B with a two-element domain (called *Boolean templates*). Indeed, Schaefer showed that it can be decided in polynomial time whether or not CSP(B) is NP-complete for any Boolean template B. Subsequent investigations have established that the Dichotomy Conjecture holds for other classes of templates. Generalizing the example of k-colorability, Hell and Nešetřil [40] showed that for all templates B which are undirected graphs, if B is bipartite, then CSP(B) is in P, while if B is not bipartite, then CSP(B) is NP-complete. Building on a group-theoretic approach initiated in [27], Bulatov extended Schaefer's dichotomy to CSP(B) for all three element templates B [17]. In their chapter, Kolaitis and Vardi explore definability frameworks for understanding some of the known results concerning the conjectured dichotomy. They also show how definability theory illuminates the study of uniform constraint satisfaction problems.

1.2 Descriptive Complexity

In the preceding section, we have traced the theme of definability as a source of structural information as it arises in several settings throughout the volume. Let us turn our attention to another major theme, the relation between definability and computational complexity. This is the focus of Erich Grädel's chapter on "Finite model theory and descriptive complexity" (Chap. 3).

1.2.1 Satisfaction

Let us look again at the definition of a query Q being definable by a sentence φ of a logic L :

$$\varphi \text{ defines } Q \quad \text{if and only if } \forall G \in \mathsf{G}(G \models_L \varphi \Leftrightarrow G \in Q).$$

If we think of queries as combinatorial problems, it is natural to ask whether we can obtain information about the computational complexity of a problem from the fact that it is definable in one language or another. This question focuses attention on the complexity of the satisfaction relation itself, (also known as the model-checking problem). Vardi [68] formulated three notions of complexity associated with the satisfaction relation for L (relative to a collection of finite structures $\mathcal{C}$). The first, called the *combined complexity* of L is just the complexity of the satisfaction relation itself, viewed as a binary relation on strings encoding structures in $\mathcal{C}$ on the one hand, and sentences of L on the other. The second, called the *data complexity* of L, is the complexity of the decision problems associated with L-definable queries Q over $\mathcal{C}$. The third, called the *expression complexity* of L, is the complexity of the decision problems associated with the L-theories $\mathrm{Th}_L(G)$ of finite structures G in $\mathcal{C}$, where

$$\mathrm{Th}_L(G) = \{\varphi \in L \mid G \models_L \varphi\}.$$

The study of these notions is rooted in the great developments in logic in the 1930s. In the first work which rigorously defined the notion of satisfaction, "On the concept of truth in formalized languages," [65], Tarski famously resolved a basic question concerning expression complexity in the context of infinite structures, and in descriptive terms at that: he showed that the first-order theory of the structure $\mathsf{N} = \langle N, 0, +, \times \rangle$ is not arithmetically definable, that is, there is no first-order formula $\theta(x)$ in the language of arithmetic such that for all $i \in N$,

$$\mathsf{N} \models \varphi(i) \Leftrightarrow \mathsf{N} \models \chi_i,$$

where χ_i is the sentence in the first-order language of arithmetic with code i. Subsequent work by Kleene and Post revealed the intimate connection between arithmetic definability and complexity as measured by Turing degrees, thereby transforming Tarski's undefinability result into a lower bound on recursion-theoretic complexity. Moreover, Tarski's definition of satisfaction itself exhibited that the first-order theory of N could be defined by both an existential and a universal sentence in the second-order language of arithmetic. Again, later work by Kleene yielded a "computational" interpretation of this descriptive result – the first-order theory of N is hyperarithmetical.

Chapter 3 presents a comprehensive overview of results concerning combined, data, and expression complexity in the context of finite model theory. One theme that runs through the chapter is the role of combinatorial games in analyzing the combined complexity of many logics, among them first-order logic and various fixed-point logics, including the modal μ-calculus, a natural fixed-point extension of propositional modal logic with applications ranging from hardware verification to analysis of hybrid systems. The chapter begins with an incisive analysis of the complexity of first-order logic using the technique of model-checking games $\mathcal{G}(A, \varphi)$ in which a Verifier and a Falsifier compete, and Verifier has a winning strategy just in case $A \models \varphi$. In the case of first-order logic, the model-checking games are positional and have a finite game graph. The strategy problem for such games in general, "does Player I have a winning strategy for the game from position p?", can be solved in linear time. Analysis of an alternating algorithm for the first-order model-checking game yields the following information: the combined complexity of FO is PSPACE-complete, while the combined complexity of FO^k is PTIME-complete – yet another source of interest in the finite-variable fragments. Moreover, PSPACE-completeness follows from the fact that the satisfiability problem for quantified Boolean formulas is easily reduced to the first-order theory of the unary structure $A = \langle\{0, 1\}, \{0\}\rangle$, from which it follows at once that the expression complexity of FO is also PSPACE-complete. On the other hand, the data complexity of FO is in deterministic LOGSPACE. This gap between expression complexity and data complexity obtains for many well-known logics.

When we turn from first-order to second-order logic, the situation is quite different. For example, the data complexity of the monadic existential fragment of second-order logic (mon-ESO) is NP-complete, that is, every mon-ESO-definable query is in NP, and some such queries, for example, 3-colorability, are NP-hard. On the other hand, as discussed in Chap. 2, there are PTIME queries on G, for example, connectivity, which are not mon-ESO-definable. This suggests that definability theory could be used to illuminate differences in complexity which are not easily characterized in terms of computational resources – a good example of this is the result of Ajtai and Fagin that undirected reachability is mon-ESO-definable, while directed reachability is not [4] (recently, Reingold has established that undirected reachability is in DLOGSPACE, whereas directed reachability is a paradigmatic NLOGSPACE-complete problem [58] – separating these two complexity classes remains an outstanding open question). On the other hand, it is also interesting when definability of queries in well-understood logics coincides with resource complexity classes, from at least two points of view: first, the logical language could then be used as a transparent specification language for queries in the class, and second, methods of logic could be deployed in complexity-theoretic investigations.

A logic L *captures* a complexity class K on a collection of structures $\mathcal{C}$ if and only if, for every query Q over $\mathcal{C}$,

$$Q \text{ is } L\text{-definable} \Leftrightarrow Q \in \mathsf{K}.$$

In 1970, Fagin [25] showed that the existential fragment of second-order logic captures the complexity class NP over the class of all finite structures (see

Chap. 3 for discussion and a proof). Fagin's result launched an active search for characterizations of other complexity classes in logical terms. Since the natural specification of many combinatorial problems is given by an existential second-order sentence, Fagin's Theorem provides a convenient tool for recognizing that problems are in NP. From the point of view of specification languages for database queries, it would be most useful to find logics that capture complexity classes below NP. Though Fagin's Theorem extends easily to show that full second-order logic captures the polynomial-time hierarchy, PH, over arbitrary finite structures, thus far no logic has been identified that captures a complexity class presumed to be strictly contained in NP over the collection of all finite structures. On the other hand, much has been learned about logics that capture such complexity classes over particular collections of finite structures. Indeed, the first capturing result was of just this kind. In 1960, Büchi [16] showed that mon-ESO captures the collection of regular languages over the class of string structures, that is, structures of the form $\langle [n], S, \overline{P} \rangle$, where S is the usual successor relation on $[n]$ and $\overline{P}$ is a finite sequence of unary predicates; it is worth noting that over string structures, all of monadic second-order logic is no more expressive than its existential fragment (see [63] and [32] for extended treatments of connections between logic and automata theory). Chapter 3 shows how other fragments of second-order logic yield characterizations of complexity classes over ordered finite structures, that is, structures which interpret a distinguished binary relation as a linear order on the universe. These include Grädel's results that second-order Horn logic (and its existential fragment) captures polynomial time on ordered finite structures and that second-order Krom logic (and its existential fragment) captures nondeterministic logarithmic space on ordered finite structures.

An especially active area of investigation in descriptive complexity theory is the analysis of logics with fixed-point operators that allow for defining queries by induction. The clarification of the nature of inductive definitions was a task undertaken by the pioneers of modern logic. Indeed, among Frege's great contributions in *Die Grundlagen der Arithmetik* was the analysis of one of the simplest fixed-point operators, which allows definition of the ancestral of a relation (now called transitive closure), in the universal fragment of second-order logic: a is an E-ancestor of b ($\mathrm{tc}xy(Exy)ab$) if and only if

$$\forall R((\forall x \forall y(Exy \rightarrow Rxy) \wedge \forall z((Rxy \wedge Ezx) \rightarrow Rzy)) \rightarrow Rab).$$

From the point of view of descriptive complexity, transitive closures appear to be quite weak compared with universal second-order quantification. Immerman [43] showed that the extension of first-order logic with the transitive-closure operator (TC) captures NLOGSPACE over the class of ordered finite structures, while, by Fagin's Theorem, the universal fragment of second-order logic captures co-NP, which has been conjectured to properly include NLOGSPACE. If transitive closure is applied only to single-valued relations, one obtains, as an extension of first-order logic, deterministic transitive-closure logic (DTC), which captures DLOGSPACE over the class of ordered finite structures [43]. In this instance, the descriptive separation, DTC $\neq$ TC, over the class of all finite structures was established by Grädel and McColm [31], whereas the separation on ordered finite structures is equivalent to the unresolved complexity-theoretic question: is DLOGSPACE distinct from NLOGSPACE?

Richer fixed-point logics yield characterizations of PTIME and PSPACE over ordered finite structures. Chapters 2 and 3 contain detailed developments of logical and complexity-theoretic results concerning the least fixed-point (LFP), inflationary fixed-point (IFP), and partial fixed-point (PFP) extensions of first-order logic, including proofs that LFP captures PTIME over ordered finite structures [43, 68], that PFP captures PSPACE over ordered finite structures [1, 68], and that LFP = IFP over arbitrary finite structures [39] (indeed, Kreutzer established that LFP = IFP over arbitrary, not just finite, structures [49]). In contrast to the aforementioned descriptive separation of TC and DTC, and in spite of the fact that LFP and PFP do not capture PTIME and PSPACE over finite graphs without an ordering, Abiteboul and Vianu [2] established that there are PFP-definable queries on finite graphs which are not LFP-definable, if and only if PSPACE is distinct from PTIME, a striking result which solved an open problem posed by Chandra [20].

As noted earlier, the fixed-point logics LFP, IFP, and PFP are all fragments of $L^{\omega}_{\infty\omega}$ with respect to definability over the class of finite structures, and consequently they lack the means to express any nontrivial cardinality queries on finite graphs. The extension of IFP with counting quantifiers (IFP+C) yields a logic that captures PTIME over wider classes of finite structures; for example, Grohe established that IFP+C captures PTIME on the class of planar graphs (in fact, on any class of structures whose Gaifman graphs are of bounded genus) [34, 35] and on any class of structures of bounded tree-width [36]. On the other hand, Cai, Fürer, and Immerman established that IFP+C does not capture PTIME over the class of all finite graphs [18]. It is natural to ask: is there a logic that captures PTIME on the class of all finite graphs?

1.2.2 What Is a Logic for PTIME?

In order to sensibly address the preceding question, we need to refine the notion of a logic capturing a complexity class – otherwise, for all we have said about logics in the abstract, we might be tempted to answer that the collection of PTIME queries itself is a logic that captures PTIME. Chandra and Harel [19] introduced the notion of an effectively enumerable query complexity class and posed the question of whether the PTIME-computable queries are effectively enumerable; Gurevich [38] introduced the closely related notion of a logic for PTIME (see also [23] and [53] for further discussion of logics for complexity classes). In order to explain this notion, we need to focus closely on the satisfaction relation. Recall that a logic L is a pair consisting of a set of sentences S_L and a satisfaction relation $\models_L$. We say that L is uniformly contained in PTIME on a collection of finite structures $\mathcal{C}$ if and only if S_L and $\models_L$ are decidable, and there are effectively computable functions m and t such that for every $\varphi \in S_L$, $m(\varphi)$ is a deterministic Turing machine which decides $Q(\varphi) \cap \mathcal{C}$ in time $n^{t(\varphi)}$. Note that SO-Horn, LFP, and IFP are uniformly contained in PTIME on the collection of all finite structures. A logic L effectively captures PTIME on $\mathcal{C}$ if and only if L is uniformly contained in PTIME on $\mathcal{C}$ and every PTIME query on $\mathcal{C}$ is L-definable. In this sense, a logic for PTIME embodies a query language which can be compiled into machine code with explicit bounds on running time, and which expresses every PTIME query. (The notion of "effectively capturing" can easily be extended to other resource complexity classes; for example, in the obvious sense, Fagin's Theorem establishes that ESO effectively captures NP.)

Insofar as we have placed only quite abstract requirements on a logic L effectively capturing PTIME, the question naturally arises whether the collection T_p of n^k-clocked Turing machines, for all $k \in \mathsf{N}$, itself might not be such a logic, where the associated satisfaction relation is just acceptance. The problem with this suggestion is that the "queries" definable in this logic are not necessarily queries, that is, they are not in general isomorphism-invariant. One way of overcoming this obstacle would be to preprocess input graphs so that a fixed representative of each isomorphism type of structure would be presented to a clocked machine. Given an equivalence relation $\sim$ on G, we say $f : \mathsf{G} \longrightarrow \mathsf{G}$ is a $\sim$-canon if and only if, for all $G, H \in \mathsf{G}$, $G \sim f(G)$ and if $G \sim H$, then $f(G) = f(H)$. Given a Turing machine M which computes an isomorphism canon, we could "compose" M with each of the machines $M' \in T_p$ and thereby arrive at a logic which captures PTIME on G; if, moreover, M ran in polynomial time in the length of its input, this would yield a logic that effectively captures PTIME on G. The existence of a polynomial-time-computable isomorphism canon for graphs is a major open problem in complexity theory. It is well known that if P = NP, then there is a polynomial-time-computable isomorphism canon for finite graphs, though it is unknown whether the existence of such a canon would imply that P = NP [5]. It follows at once that if there is no logic that effectively captures PTIME on G, then there is no polynomial-time-computable isomorphism canon for graphs, and hence P $\neq$ NP. (Indeed, if P = NP, then existential second-order logic is a logic for P. This follows from Fagin's Theorem and the "polynomially uniform" completeness of typical NP-complete problems.) On the other hand, if there is a polynomial-time-computable $\sim$-canon for a class of graphs $\mathcal{C}$, then there is a logic L that effectively captures $\sim$-invariant PTIME on $\mathcal{C}$, that is, a logic which is uniformly contained in PTIME and expresses all and only the PTIME-computable queries on $\mathcal{C}$ which are closed under $\sim$. In some cases, such as Grohe's capturing results for IFP+C cited above, there is a "familiar" logic that does the capturing. Another example of this phenomenon is Otto's result [55] that bisimulation-invariant PTIME is uniformly captured by the multidimensional μ-calculus (see Chap. 3 and references there).

1.3 Finite Model Theory and Infinite Structures

The concluding section of Chap. 3 surveys several areas where the perspective of descriptive complexity theory has been extended to the study of certain classes of infinite structures. Such extension requires, at minimum, that the structures in question be finitely presentable and that the satisfaction relation be computable when restricted to the given setting (structures and language). One active research direction here is the study of automatic structures, that is, structures whose universe and relations are regular sets of strings. Automatic structures have nice closure properties from the point of view of definability theory; for example, all first-order-definable relations on such structures are regular, and so the expansion of an automatic structure by first-order-definable relations is itself automatic, a property not shared, for example, by recursively presented structures.

Another research direction where the point of view of descriptive complexity is extended to infinite structures is the study of metafinite structures, which were introduced by Grädel and Gurevich in [30]. A paradigmatic example of such structures is edge-weighted graphs. Here one has a finite graph and a numerical structure, such as the ring Z or the ordered field R, and a function which assigns weights in the numerical structure to edges in the graph. Such two-sorted structures arise naturally in several areas of computer science, including database theory, optimization theory, and complexity theory. A hallmark of metafinite model theory is the simplicity of the languages deployed to describe these hybrid structures. In particular, there is no quantification allowed over the numerical structure, indeed, no variables which admit assignment from the numerical domain. The only access to the numerical structure is via weight terms that assign numerical values to tuples from the nonnumerical sort, and terms which combine these by use of operations on the numerical universe. Following [30], Chap. 3 shows how the notion of a generalized spectrum admits two extensions to the context of metafinite structures (one allowing projection of weight functions, in addition to projection of relations on the finite structure). In the context of arithmetical structures (those whose numerical part consists of the standard model of arithmetic with additional polynomial-time-computable multiset operations) with "small weights", the more restricted notion of a generalized spectrum captures NP, whereas on arithmetical structures in general, the wider notion captures the class of all recursively enumerable relations. Chapter 3 concludes with a proof of the result, due to Grädel and Meer, that in the case of metafinite structures whose numerical part is the real ordered field extended with constants for all real numbers, the wide notion of a metafinite spectrum captures NP_R, the collection of nondeterministic polynomial-time-acceptable relations on the reals in the Blum–Schub–Smale model of computation over the reals [15].

A third area which involves a blend of finite and classical model theory is the study of "Embedded finite models and constraint databases", the subject of Leonid Libkin's chapter (Chap. 5). In the context of geographical information systems, the management of spatio-temporal data, bioinformatics, and numerous other database application areas, it is useful to look at relational data over infinite sets which may themselves be endowed with additional structure. The approach via constraint databases, pioneered by Kanellakis *et al.* [45], where, for example, geographical regions are stored as logical formulas that define them, via coordinatization, over the real ordered field R or the real ordered group, has proven to be fruitful. In this context, new definability questions arise; for example, can one define topological connectivity of (definable) spatial regions? As discussed in Chap. 5, the work of Grumbach and Su [37] revealed that many definability questions of this kind could be reduced to definability questions about *embedded finite structures*, that is, finite structures whose domain is drawn from some ambient infinite structure such as the real ordered field. For example, if G is a finite graph whose vertices are real numbers, then the expansion $A = \langle \mathsf{R}, E^G \rangle$ of R is an embedded finite model with "active domain" the set of nonisolated vertices of G. Now, it can be shown that there is a first-order formula $\varphi(x, y)$ such that the region in R^2 defined by φ in A is topologically connected, if and only if G is a connected graph. Thus, if topological connectivity of definable planar regions were first-order-definable in R, then connectivity of embedded finite graphs would also be definable over $\{\langle \mathsf{R}, E \rangle \mid E \subset_{\text{fin}} \mathsf{R}^2\}$. This is exactly the point at which embedded finite model theory comes into play in offering a variety of techniques to answer definability questions of the latter sort. One of the main thrusts of embedded finite model theory is to establish "collapse results", which reduce questions about definability over embedded finite structures to questions about definability over finite structures. It turns out that general model-theoretic conditions on the ambient infinite structure are of paramount importance in determining the extent to which such collapse results obtain. Chapter 5 provides a detailed account of such phenomena. These phenomena provide considerable evidence that infinite structures which are well-behaved from the point of view of definability theory in the infinite are similarly tame with respect to embedded finite structures. For example, Benedikt *et al.* [10] have shown that if M is an *o*-minimal structure and Q is an order-generic query on finite structures A embedded in M, which is first-order definable over $\langle M, A \rangle$, then Q is first-order definable (with order) over finite structures A; the real ordered field is a paradigmatic *o*-minimal structure, and recent work in model theory has established the *o*-minimality of various of its extensions [66, 70]. Baldwin and Benedikt [7] have shown, more generally, that the same collapse obtains for any M which lacks the *independence property*, a condition familiar from stability theory. Chapter 5 reveals deep connections between the independence property and definability over embedded finite models.

1.4 Tame Fragments and Tame Classes

The book concludes with a concise, modern introduction to modal logic, "Local variations on a loose theme: modal logic and decidability", by Maarten Marx and Yde Venema (a comprehensive treatment in this spirit can be found in [13]). Modal logics have numerous applications to computer science, ranging from specification of hybrid systems to knowledge representation, and these applications rest on the delicate balance between the expressive power of modal languages and their good algorithmic properties. The chapter provides an incisive analysis of this balance (other useful discussions include [29, 69]).

Propositional modal languages can be viewed, via the Kripke modeling, as vehicles for expressing unary queries over labeled transition systems, that is, structures whose universe consists of a collection of states equipped with binary "accessibility" relations and unary labels. When viewed in this way, a propositional modal sentence φ, such as

$$P \to \Box(\neg P \wedge \Diamond P),$$

can be translated into a first-order formula φ° with one free variable,

$$P(x) \to \forall y(Rxy \to (\neg P(y) \wedge \exists x(Ryx \wedge P(x)))),$$

so that, for any Kripke model $M = \langle U^M, R^M, P^M \rangle$ and any $s \in U^M$,

$$M, s \Vdash \varphi \Leftrightarrow M \models \varphi^\circ[s].$$

For example, if M is the structure with

$$U^M = \{1,2,3\}, R^M = \{\langle 1,1\rangle, \langle 1,2\rangle, \langle 2,3\rangle, \langle 3,2\rangle\}, P^M = \{1,3\},$$

then φ° defines the set $\{2,3\}$ in M. It is easy to check that for any basic modal sentence φ, φ° is in the the two-variable fragment of first-order logic, and all quantifiers in φ° are relativized to the collection of states directly accessible from a given state. The collection of translations of modal sentences is called the modal fragment of first-order logic.

Chapter 7 emphasizes that bisimulation invariance is the fundamental property of the modal fragment of first-order logic. As mentioned above, bisimilarity can be characterized in terms of a simple one-pebble comparison game. Kripke structures M and M' with states $s \in M$ and $s' \in M'$ are bisimilar if and only if the Duplicator has a winning strategy in the following game. Initially, pebbles are placed on the distinguished states s and s'. At each round of play, the Spoiler chooses one of the pebbles and moves it to a state accessible from the state on which it lies. The Duplicator must move the other pebble in like fashion, and to a state which is labeled identically to the state onto which Spoiler has moved. The game ends with a win for the Spoiler if the Duplicator cannot thus move at some round. Otherwise, the Duplicator wins the (perhaps infinite) play of the game.

It is easy to check that every formula in the modal fragment is bisimulation invariant; that is, if M, s is bisimilar to M', s' then

$$M \models \varphi^\circ[s] \Leftrightarrow M' \models \varphi^\circ[s'],$$

for every modal sentence φ. The authors show that by "unraveling" a Kripke structure M at a state s one can create a tree model M' (that is, $\langle U', R^{M'} \rangle$ is a directed tree) that is bisimilar to M at s (the unraveling consists of collecting all finite walks in M starting at s and ordering them by immediate extension). Thus, any bisimulation-invariant language has the "tree model" property. The authors refer to this as the looseness property of modal logic, and identify it as one of the sources of the good algorithmic behavior of modal logics. They observe that this is not the entire story, and note that modal logics also exhibit some interesting locality properties that also partly account for the relatively low complexity of their satisfiability and model-checking problems. Indeed, since there are continuum-many bisimulation-invariant queries even on finite labeled transition systems, the tree model property could not be the complete account for the computational tameness of the modal fragment. The authors identify two locality properties that are important in explaining the behavior of modal logic. The first is related to the Hanf and Gaifman locality of first-order logic as discussed in Chap. 2 (note that modal depth equates to quantifier depth in the modal fragment); the second is related to the fact that the modal fragment is contained in FO^2.

The connection between bisimilarity invariance and modal definability is intimate – Johan van Benthem established [11, 12] a preservation theorem for the modal fragment: every bisimulation-invariant first-order formula is equivalent to a formula in the modal fragment. Eric Rosen [59] showed that this preservation theorem persists to the class of finite structures; that is, if a formula of first-order logic is preserved under bisimulation over the collection of finite Kripke structures, then it is equivalent, over finite Kripke structures, to a formula in the modal fragment. This result provides evidence that the modal fragment is tame not only from an algorithmic point of view, but also from the point of view of finite model theory. How so? Several well-known preservation theorems from classical model theory fail when relativized to finite structures. For example, Tait [64] showed that the Łoś–Tarski existential preservation theorem does not persist to the class of finite structures – there is a first-order sentence that is preserved under extensions relative to the collection of finite structures, but is not equivalent over finite structures to an existential sentence. An even more telling example in the current context is the failure of a preservation theorem for the two-variable fragment of first-order logic to persist to the class of finite structures. A query is 2-invariant if and only if it is closed under $L^2_{\infty\omega}$ equivalence. Immerman and Kozen [44] showed that if a query is 2-invariant and first-order definable, then it is expressible by a sentence of FO^2. This result does not persist to the finite case; for example, the collection of finite linear orderings is 2-invariant and FO^3-definable with respect to the collection of finite structures, but is not FO^2-definable over finite structures. So the modal fragment is in some sense tamer than the two-variable fragment with respect to model theory over the class of finite structures. Otto [56] has proved a generalization of Rosen's preservation result which gives yet more evidence that the tameness of modal finite model theory is connected to the relativization of quantification in the modal fragment. He established that any formula of FO^2 that is invariant under *guarded bisimulations* with respect to the class of finite structures is equivalent, over finite structures, to a formula in the guarded fragment of FO^2. Chapter 7 explains how the guarded fragment of first-order logic is a natural extension of the modal fragment and discusses aspects of its good algorithmic behavior. Rossman [60] recently established that the homomorphism preservation theorem persists to finite structures, that is, if a first-order definable query is closed under homomorphisms with respect to the class of finite structures, then it is equivalent over finite structures to a positive existential sentence. So, in the sense to hand, the positive existential fragment of first-order logic is also "tame" for finite model theory. It is worth noting that some fragments that are ill-behaved with respect to the collection of all finite structures may be tame with respect to interesting subclasses. Though the existential preservation theorem fails over the collection of all finite structures, Atserias, Dawar, and Grohe [6] have shown that it holds with respect to classes of finite structures of bounded degree and bounded tree-width. To echo a motto proposed by Hrushovski ("model theory = geography of tame mathematics" [67]), a geography of tame fragments and tame classes may yield some insight into finite model theory.

Acknowledgments

I am grateful to Phokion G. Kolaitis for many corrections and suggestions which substantially improved this chapter. This work was supported in part by NSF grant CCR-9820899. Part of this work was done while I was visiting the Isaac Newton Institute for Mathematical Sciences, during a Special Programme on Logic and Algorithms.

References

1. S. Abiteboul and V. Vianu. Datalog extensions for database queries and updates. *Journal of Computer and System Sciences*, 43:62–124, 1991.
2. S. Abiteboul and V. Vianu. Generic computation and its complexity. In *STOC*, pages 209–219. ACM, 1991.
3. J. W. Addison. Tarski's theory of definability: common themes in descriptive set theory, recursive function theory, classical pure logic, and finite-universe logic. *Annals of Pure and Applied Logic*, 126:77–92, 2004.
4. M. Ajtai and R. Fagin. Reachability is harder for directed than for undirected finite graphs. *Journal of Symbolic Logic*, 55:113–150, 1990.

5. V. Arvind and J. Torán. Isomorphism testing: Perspective and open problems. *Bulletin of the EATCS*, 86:66–84, 2005.
6. A. Atserias, A. Dawar, and M. Grohe. Preservation under extensions on well-behaved finite structures. In L. Caires, G. F. Italiano, L. Monteiro, C. Palamidessi, and M. Yung, editors, *ICALP*, Lecture Notes in Computer Science, volume 3580, pages 1437–1449. Springer, 2005.
7. J. Baldwin and M. Benedikt. Stability theory, permutations of indiscernibles, and embedded finite models. *Transactions of the American Mathematical Society*, 352:4937–4969, 2000.
8. J. Baldwin and S. Shelah. Randomness and semi-genericity. *Transactions of the American Mathematical Society*, 349:1359–1376, 1997.
9. J. Barwise. On Moschovakis closure ordinals. *Journal of Symbolic Logic*, 42: 292–296, 1977.
10. M. Benedikt, G. Dong, L. Libkin, and L. Wong. Relational expressive power of constraint query languages. *Journal of the ACM*, 45(1):1–34, 1998.
11. J. van Benthem. *Modal Correspondence Theory*. PhD thesis, Mathematisch Instituut & Instituut voor Grondslagenonderzoek, University of Amsterdam, 1976.
12. J. van Benthem. *Modal Logic and Classical Logic*. Bibliopolis, 1983.
13. P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2002.
14. A. Blass, Y. Gurevich, and D. Kozen. A zero–one law for logic with a fixed-point operator. *Information and Control*, 67(1-3):70–90, 1985.
15. L. Blum, M. Shub, and S. Smale. On a theory of computation over the real numbers; np completeness, recursive functions and universal machines (extended abstract). In *FOCS*, pages 387–397. IEEE Computer Society, 1988.
16. J. R. Büchi. Weak second-order arithmetic and finite automata. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 6:66–92, 1960.
17. A. A. Bulatov. A dichotomy theorem for constraints on a three-element set. In *FOCS*, pages 649–658. IEEE Computer Society, 2002.
18. J-y. Cai, M. Fürer, and N. Immerman. An optimal lower bound on the number of variables for graph identification. In *FOCS*, pages 612–617. IEEE Computer Society, 1989.
19. A. Chandra and D. Harel. Structure and complexity of relational queries. *Journal of Computer and System Sciences*, 25:99–128, 1982.
20. A. K. Chandra. Theory of database queries. In *PODS*, pages 1–9. ACM, 1988.
21. C. C. Chang and H. J. Keisler. *Model Theory*. North-Holland, 1990.
22. A. Dawar. Finite models and finitely many variables. In D. Niwinski and R. Maron, editors, *Logic, Algebra and Computer Science*, Banach Center Publications, volume 46, pages 93–117. Polish Academy of Sciences, 1999.
23. A. Dawar. Generalized quantifiers and logical reducibilities. *Journal of Logic and Computation*, 5(2):213–226, 1995.
24. P. Erdös and A. Rényi. On the evolution of random graphs. *Public Mathematical Institute of Hungary Academy of Sciences*, 5:17–61, 1960.
25. R. Fagin. Generalized first-order spectra and polynomial-time recognizable sets. In R. M. Karp, editor, *Complexity of Computation, SIAM-AMS Proceedings*, volume 7, pages 43–73, 1974.
26. R. Fagin. Probabilities on finite models. *Journal of Symbolic Logic*, 41(1):50–58, March 1976.

27. T. Feder and M. Y. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: a study through datalog and group theory. *SIAM Journal on Computing*, 28:57–104, 1998.
28. Y. Glebskii, D. Kogan, M. Liogon'kii, and V. Talanov. Range and degree of realizability of formulas in the restricted predicate calculus. *Cybernetics*, 5:142–154, 1969.
29. E. Grädel. Why are modal logics so robustly decidable? *Bulletin of the EATCS*, 68:90–103, 1999.
30. E. Grädel and Y. Gurevich. Metafinite model theory. *Information and Computation*, 140:26–81, 1998.
31. E. Grädel and G. L. McColm. Hierarchies in transitive closure logic, stratified datalog and infinitary logic. *Annals of Pure and Applied Logic*, 77:166–199, 1996.
32. E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research*, Lecture Notes in Computer Science, volume 2500, Springer, 2002.
33. M. Grohe. Finite variable logics in descriptive complexity theory. *Bulletin of Symbolic Logic*, 4:345–398, 1998.
34. M. Grohe. Fixed-point logics on planar graphs. In *LICS*, pages 6–15. IEEE Computer Society, 1998.
35. M. Grohe. Isomorphism testing for embeddable graphs through definability. In *STOC*, pages 63–72. ACM, 2000.
36. M. Grohe and J. Mariño. Definability and descriptive complexity on databases of bounded tree-width. In C. Beeri and P. Buneman, editors, *ICDT*, Lecture Notes in Computer Science, volume 1540, pages 70–82. Springer, 1999.
37. S. Grumbach and J. Su. Queries with arithmetical constraints. *Theoretical Computer Science*, 173(1):151–181, 1997.
38. Y. Gurevich. Logic and the challenge of computer science. In E. Börger, editor, *Current Trends in Theoretical Computer Science*, pages 1–57. Computer Science Press, 1988.
39. Y. Gurevich and S. Shelah. Fixed-point extensions of first-order logic. *Annals of Pure and Applied Logic*, 32:265–180, 1986.
40. P. Hell and J. Nešetřil. On the complexity of H-coloring. *Journal of Combinatorial Theory-Series B*, 48:92–110, 1990.
41. L. Hella, Ph. G. Kolaitis, and K. Luosto. Almost everywhere equivalence of logics in finite model theory. *The Bulletin of Symbolic Logic*, 2(4):422–443, 1996.
42. M. Hennessy and R. Milner. Algebraic laws for nondeterminism and concurrency. *Journal of the ACM*, 32(1):137–161, 1985.
43. N. Immerman. Relational queries computable in polynomial time. *Information and Control*, 68:86–104, 1986.
44. N. Immerman and D. Kozen. Definability with bounded number of bound variables. *Information and Computation*, 83:121–139, 1989.
45. P. C. Kanellakis, G. M. Kuper, and P. Z. Revesz. Constraint query languages. In *PODS*, pages 299–313. ACM, 1990.
46. C. Karp. Finite quantifier equivalence. In J. W. Addison, L. Henkin, and A. Tarski, editors, *The Theory of Models*, pages 407–412. North-Holland, 1965.
47. Ph. G. Kolaitis and M. Y. Vardi. The decision problem for the probabilities of higher-order properties. In *STOC*, pages 425–435. ACM, 1987.

48. Ph. G. Kolaitis and M. Y. Vardi. Infinitary logics and 0–1 laws. *Information and Computation*, 98(2):258–294, 1992.
49. S. Kreutzer. Expressive equivalence of least and inflationary fixed-point logic. *Annals of Pure and Applied Logic*, 130(1–3):61–78, 2004.
50. R. E. Ladner. On the structure of polynomial time reducibility. *Journal of the ACM*, 22(1):155–171, 1975.
51. M. C. Laskowski. A simpler axiomatization of the Shelah–Spencer almost sure theories. *Israel Journal of Mathematics*, to appear.
52. Y. N. Moschovakis. *Descriptive Set Theory.* North-Holland, 1980.
53. A. Nash, J. B. Remmel, and V. Vianu. Ptime queries revisited. In T. Eiter and L. Libkin, editors, *ICDT*, Lecture Notes in Computer Science, volume 3363, pages 274–288. Springer, 2005.
54. M. Otto. *Bounded Variable Logics and Counting.* Springer, 1997.
55. M. Otto. Bisimulation-invariant ptime and higher-dimensional μ-calculus. *Theoretical Computer Science*, 224(1-2):237–265, 1999.
56. M. Otto. Modal and guarded characterisation theorems over finite transition systems. *Annals of Pure and Applied Logic*, 130:173–205, 2004.
57. D. Park. Concurrency and automata on infinite sequences. In P. Deussen, editor, *Theoretical Computer Science*, Lecture Notes in Computer Science, volume 104, pages 167–183. Springer, 1981.
58. O. Reingold. Undirected st-connectivity in log-space. In H. N. Gabow and R. Fagin, editors, *STOC*, pages 376–385. ACM, 2005.
59. E. Rosen. Modal logic over finite structures. *Journal of Logic, Language and Information*, 6(4):427–439, 1997.
60. B. Rossman. Existential positive types and preservation under homomorphisms. In *LICS*, pages 467–476. IEEE Computer Society, 2005.
61. T. J. Schaefer. The complexity of satisfiability problems. In *STOC*, pages 216–226. ACM, 1978.
62. S. Shelah and J. Spencer. Zero–one laws for sparse random graphs. *Journal of the American Mathematical Society*, 1:97–115, 1988.
63. H. Straubing. *Finite Automata, Formal Logic, and Circuit Complexity.* Birkhäuser, 1994.
64. W. Tait. A counterexample to a conjecture of Scott and Suppes. *Journal of Symbolic Logic*, 24(1):15–16, 1959.
65. A. Tarski. The concept of truth in formalized languages. In *Logic, Semantics, Metamathematics*, pages 152–278. Clarendon Press, Oxford, 1956.
66. L. van den Dries. *Tame Topology and o-Minimal Structures.* Cambridge University Press, 1998.
67. L. van den Dries. Classical model theory of fields. In D. Haskell, A. Pillay, and C. Steinhorn, editors, *Model Theory, Algebra, and Geometry*, pages 37–52. Cambridge University Press, 2000.
68. M. Y. Vardi. The complexity of relational query languages. In *STOC*, pages 137–146. ACM, 1982.
69. M. Y. Vardi. Why is modal logic so robustly decidable? In N. Immerman and Ph. G. Kolaitis, editors, *Descriptive Complexity and Finite Models*, volume 31 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 149–184. American Mathematical Society, 1996.
70. A. J. Wilkie. Model completeness results for expansions of the ordered field of real numbers by restricted pfaffian functions and the exponential function. *Journal of the American Mathematical Society*, 9:1051–1094, 1996.