Martin Otto

Bounded Variable Logics and Counting

A Study in Finite Models

November 6, 1996

Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo
Hong Kong Barcelona
Budapest

Preface

Viewed as a branch of model theory, *finite model theory* is concerned with finite structures and their properties under logical, combinatorial, algorithmic and complexity theoretic aspects. The connection of classical concerns of logic and model theory with issues in complexity theory has contributed very much to the development of finite model theory into a field with its own specific flavour.

I like to think of this monograph as a study which — with a particular theme of its own — exemplifies and reflects some central ideas and lines of research in finite model theory. The particular theme is that of bounded variable infinitary logics, with and without counting quantifiers, related fixed-point logics, and corresponding fragments of PTIME. The relations with PTIME exhibit that fruitful exchange between ideas from logic and from complexity theory that is characteristic of finite model theory and, more specifically, of the research programme of descriptive complexity.

Among the main particular topics and techniques I would emphasize:

- the importance of games as a fundamental tool from classical logic; their use in the analysis of finite structures also with respect to algorithmic and complexity theoretic concerns is amply illustrated.
- the rôle of cardinality phenomena, which clearly are amongst the most fundamental guidelines in the analysis of finite structures.
- the importance of combinatorial techniques, and of dealing with concrete combinatorial problems over finite domains. Examples here range from applications of the stable colouring technique in the formation of structural invariants to certain colourings of squares that come up in canonization for logics with two variables.

In order that this study may be useful also as an introduction to some of the important concepts in the field, I have tried to treat the particular theme in a detailed and mostly self-contained manner. On the other hand this treatment leads up to specific results of a more technical nature, and I welcome the opportunity to present some contributions in a broader context.

With respect to the work presented here, I personally owe much to two sources that I would like to mention. One is the Freiburg logic group where I had the opportunity to participate in an intensified engagement in finite model theory after finishing my doctorate with Professor Ebbinghaus. Professor Flum and Professor Ebbinghaus created an encouraging atmosphere for taking up this new field actively. The second important source is the collaboration with Professor Grädel which has had much impact on the research presented here. I am grateful to Erich Grädel for this collaboration and also for his advice and support.

The text itself is a revised version of my Habilitationsschrift, presented at the RWTH Aachen in 1995. I am grateful to Professor Ebbinghaus for numerous comments on the earlier version of this text.

Aachen, October 1996 Martin Otto

Table of Contents

0. Introduction

0.1 Finite Models, Logic and Complexity

Finite model theory deals with the model theory of finite structures. As a branch of model theory it is concerned with the analysis of structural properties in terms of logics. The attention to finite structures is not so much a restriction in scope as a shift in perspective. The main parts of classical model theory (the model theory related to first-order logic) as well as of abstract model theory (the comparative model theory of other logics) almost exclusively concern infinite structures; finite models are disregarded as trivial in some respects and as intractable in others. In fact, the most successful tools of classical model theory fail badly in restriction to finite structures. The compactness theorem in particular, which is one of the corner stones of classical model theory, does not hold in the realm of finite structures. Several examples of other important theorems from classical model theory that are no longer true in the finite case are discussed in [Gur84].

There are on the other hand specific new issues to be considered in the finite. These issues mainly account for the growing interest in finite model theory and promote its development into a theory in its own right. One of the specific issues in a model theory of finite structures is *complexity*. Properties and transformations of finite structures can be considered under *algebraic* and *combinatorial aspects*, under the aspect of *logical definability*, and also under the aspect of *computational complexity*. Issues of computational complexity form one of the main links also between finite model theory and theoretical computer science.

In this introduction I merely intend to indicate selectively some main ideas and lines of research that motivate the present investigations. There are a number of surveys that also cover various other aspects of finite model theory — see for instance [Fag90, Gur84, Gur88, Imm87a, Imm89]. A general reference is the new textbook on finite model theory by Ebbinghaus and Flum [EF95].

0.1.1 Logics for Complexity Classes

The study of the relationship between logical definability and computational complexity of structural properties is an essential branch of finite model theory. This topic is most pronounced in the search for semantic matches between levels of computational complexity and logics. The search for *logics for complexity classes*, also suggestively described as *capturing complexity classes* [Imm87b] or *tailoring logic for complexity* [Gur84], has lead to an active research programme in finite model theory.

Consider any of the standard classes in computational complexity as a class of problems for finite structures, say for finite graphs. The class of all properties of finite graphs that can be recognized by PTIME algorithms is a typical example. A logical characterization of PTIME on finite graphs would have to provide a *logic for* PTIME on graphs in the sense that exactly all PTIME properties of finite graphs are definable by sentences of this logic. For the present purposes we work with a slightly informal notion of a logic for PTIME. The exact definition underlying our treatment is due to Gurevich [Gur88]. A more detailed discussion will be provided with Definition 1.7 in the next chapter. It is also shown in [Gur88] that the restriction to graphs rather than finite structures of arbitrary type is inessential for the present issue.

Definition 0.1 (Sketch). *A logic $\mathcal{L}$ is* a logic for PTIME *if exactly those properties of finite graphs that are* PTIME *recognizable are $\mathcal{L}$-definable.*
Minimal requirements on candidate logics to be imposed are the following: $\mathcal{L}$ has recursive syntax and recursive semantics that associates with each $\mathcal{L}$-sentence a PTIME *algorithm for checking its truth in finite models.*

It is a central open problem in the field whether there is a logic for PTIME.

Relationships between computational complexity classes and definability in logical systems are interesting for a number of reasons:

(a) The potential for theoretical transfer between different fields. Techniques from complexity theory may be brought to bear on logical and model theoretic issues and vice versa. To give an example, several of the outstanding open problems of complexity theory like the PTIME = NPTIME? or PTIME = PSPACE? questions have found appealing non-trivial model theoretic reformulations in terms of semantic equivalences of particular logical systems over finite structures (compare [AV91, DLW95, Daw95b]). Short of solving the original problems this offers new perspectives, and investigations of related logical issues may at least lead to a better understanding of these problems.

(b) Logical analysis of the required kind may yield deeper insights into the fundamental notion of complexity. Definability in logical systems can be viewed as a kind of complexity in itself. Whereas computational complexity controls the computational resources required in the solution of a problem, definability considerations control the logical or descriptive resources required in the specification of the problem: hence the term *descriptive complexity* as used in [Imm89]. The relationship between the structure imposed by these completely different resources thus becomes part of a broader view of complexity theory.

(c) Exact matches between computational complexity classes and logics provide an appealing notion of *semantic completeness* for model theoretic considerations. A logic for PTIME say would be a logic that is complete for the world of PTIME computability over structures — or for computationally feasible problems, if PTIME computability is identified with efficient solvability or feasibility. The classical classes of computational complexity have emerged as natural levels of computational power, certified by robustness criteria and the existence of natural complete problems. Matching logics constitute naturally distinguished levels of expressiveness.

(d) In this connection there is also a strong theoretical interest from computer science. Problems related to structures like graphs (and more generally arbitrary relational structures corresponding to instantiations of relational databases) are ubiquitous in computer science applications and in particular in the theory of databases. A natural logic for PTIME would be a theoretically ideal database language for exactly all feasible queries: anything that can be specified in this language is guaranteed to have an efficient algorithmic solution; by semantic completeness for PTIME such a logic constitutes a universal language for all efficient tasks. And indeed this context is one of the original sources for the problem of capturing PTIME, as formulated by Chandra and Harel in [CH82].

The following are some of the well known results concerning complete matches between distinguished levels of computational complexity and logical systems.

Regular languages and monadic second-order logic: words over any finite alphabet can in a canonical way be identified with linearly ordered structures over an otherwise monadic vocabulary (one unary predicate for each letter to mark its occurrences in the word). Monadic second-order logic $\mathcal{L}^{II}_{\mathrm{mon}}$ for the resulting *word models* defines exactly the regular languages, i.e. those languages that are recognized by finite automata. This is a classical result of Büchi [Büc60], Elgot [Elg61] and Trakhtenbrot [Tra61] that fits into the present framework as a precursor to the recent development of finite model theory (compare the treatment in [EF95]).

NPTIME and existential second-order logic: Fagin's theorem [Fag74] is the first result of this branch of finite model theory proper. It equates nondeterministic polynomial time recognizability with definability in existential second-order logic Σ^1_1.

PTIME and fixed-point logic with order: in restriction to linearly ordered finite structures PTIME has been characterized logically by Immerman [Imm86] and Vardi [Var82] through the very natural extension of first-order logic to fixed-point logic FP by means of an operator for monotone relational induction.

So there are the following semantic equivalences:

$$\begin{aligned} \text{Finite Automata} &\equiv \mathcal{L}^{II}_{\mathrm{mon}} && \text{(for word models)} \\ \text{NPTIME} &\equiv \Sigma^1_1 && \\ \text{PTIME} &\equiv \text{FP} && \text{(in the presence of linear order)} \end{aligned}$$

It is remarkable that all major complexity classes, in particular LOGSPACE, NLOGSPACE, PTIME and PSPACE, are captured by very natural extensions of first-order logic in the presence of order. The fundamental question whether similar matches can be found in the general case of not necessarily ordered structures is open. In particular the question whether there is a logic for PTIME, as raised by Chandra and Harel in [CH82], is a notorious open problem in finite model theory. In fact there is no capturing result at all for any standard complexity class below NPTIME that applies to the general case. Fagin's theorem NPTIME $\equiv \Sigma^1_1$ essentially remains the only general result on a strict match between a complexity class and a logic on finite structures. This phenomenon will be further discussed below.

0.1.2 Semantically Defined Classes

Consider the class of all PTIME recognizable graph properties — for the moment denote it graph-PTIME. It serves as a typical example of a *complexity class on finite structures*.

Why is it difficult to find a logic for graph-PTIME?

Recall that ordinary PTIME is the class of all problems that can be solved by polynomially time bounded Turing machines. A priori Turing machines work with words or strings as inputs. As far as recognition (i.e. decision) problems are concerned a *problem* is a *set of words* over some alphabet. Words over this alphabet are rejected or accepted, according to membership in the set, in time polynomial in their length.

In particular a Turing machine does not work with abstract graphs as inputs but rather with *encodings* of these. The standard encoding scheme for finite graphs uses adjacency matrices for the input representation. The adjacency matrix of a graph whose vertices are labelled $v_1, \ldots, v_n$ is the $n \times n$ boolean matrix with entries $a_{ij} = 0$ or 1 according to whether (v_i, v_j) is an edge. But obviously different adjacency matrices may encode the same, more precisely isomorphic, graphs. Any rearrangement of the vertices in a different order induces an equivalent representation that is different from the given one unless the rearrangement happens to be an automorphism of the abstract graph. Any *graph algorithm*, i.e. any algorithm that recognizes a graph property, must therefore satisfy a non-trivial *semantic invariance condition*: a graph algorithm must produce the same result on any two inputs that represent isomorphic abstract graphs. In other words it may not reject one graph and accept an isomorphic copy of that same graph. We adopt the terminology of complexity theory as presented in [Pap94] to distinguish *semantic presentations* and *syntactic presentations* of complexity classes, or semantic and syntactic classes according to their presentation. Semantic presentations are given in terms of semantic constraints on algorithms. Owing to the invariance condition, graph-PTIME is clearly a semantically presented class. A syntactic presentation of a complexity class in contrast consists of a recursive or at least recursively enumerable set of algorithms of the required complexity, that contains at least one realization for every problem in the given class.[1] We shall mostly speak of *recursive presentations* in this sense.

For an example of a class that is not a priori syntactically defined but nevertheless admits a simple recursive presentation consider PTIME in the ordinary sense as a class of problems for words over finite alphabets. It is clearly presentable by the set of algorithms that limit their computation time by means of a step counter that is initialized in each computation to a polynomial in the input size. This presentation is suggestively referred to as through *polynomially clocked machines*.

Semantic invariance conditions like the one for graph-PTIME are non-recursive conditions on algorithms. In fact the set of *all* (syntactic descriptions of) graph algorithms is not even recursively enumerable (as an index set). It can furthermore be shown that the same applies to any of its intersections with standard complexity classes. In particular the ad-hoc presentation of graph-PTIME through the set of all PTIME graph algorithms does not provide a recursively enumerable presentation.

A logic $\mathcal{L}$ for PTIME in the sense of Definition 0.1 above, however, would induce the following recursive presentation for graph-PTIME. Let S be the recursive semantic mapping that associates a PTIME algorithm with each sentence of $\mathcal{L}$ (in the language of graphs). Obviously image(S) consists of PTIME graph algorithms. By semantic completeness of $\mathcal{L}$ for PTIME on graphs, any PTIME graph property is realized by some member of image(S). The recursively enumerable subset image(S) $\subseteq \{\mathcal{A} \mid \mathcal{A}$ a PTIME graph algorithm $\}$ therefore provides a *recursive presentation* for graph-PTIME. In the terminology of complexity theory, image(S) is a *syntactic presentation* of the *semantically defined* class graph-PTIME. In fact it can be shown that there is a logic for PTIME (in the sufficiently general sense of Definition 0.1) if and only if graph-PTIME admits a syntactic, i.e. recursive or recursively enumerable, presentation.

For properties of linearly ordered structures — properties of linearly ordered graphs say — these problems do not arise because there are canonical encodings for ordered structures. For ordered graphs we may use the adjacency matrix based on the natural labelling of the vertices as $v_1, \ldots, v_n$ in increasing order. This observation is easily turned into a recursive presentation for the class of all PTIME properties of ordered finite graphs.

[1] The difference between recursive and recursively enumerable syntax is not important in this kind of question. If $\mathcal{A}_1$, $\mathcal{A}_2$, ... is a recursive enumeration of syntactic descriptions of algorithms, then the syntax $(\mathcal{A}_1, 1)$, $(\mathcal{A}_2, 2)$, ... is recursive.

This crucial difference between the ordered and the unordered case is at the root of the apparent mismatch with respect to capturing complexity classes in the case of ordered structures or in the general case of not necessarily ordered structures. For the standard complexity classes it is almost trivial to see that the induced classes over ordered structures are presentable as syntactic classes and therefore can be captured by logics. The point of the corresponding capturing results indeed rather is that moreover they are captured by very natural logical systems.

As mentioned above, no complexity class below NPtime has been captured or shown to be recursively presentable in the general case. NPtime here marks a threshold because in NPtime and above, the invariance problem can be side-stepped as follows. Consider the class graph-NPtime of all NPtime graph properties. From a graph property $Q \in$ graph-NPtime we may pass to its ordered version $Q_<$, the class of all ordered graphs that possess the given property:

$$Q_< = \big\{(G,<) \bigm| G \in Q, < \text{ a linear ordering of the vertices } \big\}.$$

$Q_<$ is NPtime recognizable, essentially through the algorithm for Q itself. But a plain graph G belongs to Q if and only if any expansion $(G,<)$ by a linear ordering of its vertices belongs to $Q_<$ (and also if and only if all such expansions belong to $Q_<$). It follows that graph-NPtime is presentable through the class of all NPtime algorithms that first guess a linear ordering and then evaluate an NPtime property of ordered graphs on the result. From this observation we obtain a recursive presentation for graph-NPtime in a standard manner. It is worth noting that this trick is also directly used in Fagin's proof that graph-NPtime coincides with the class of all graph properties that are definable in existential second-order logic. The existential quantification over linear orderings $<$ that is implicit in the passage from $Q_<$ to Q is explicitly available in existential second-order logic.

Note that capturing results for complexity classes in the general case of not necessarily ordered structures are not merely of theoretical interest. The challenge is well motivated by the potential applications in database theory. Natural abstract databases often are not ordered. Their realizations at the machine level may involve an implicit linear ordering for representational purposes (naively: a numbering of memory cells). Even though an ordering is present then, it is not considered part of the intended data. A sound database query in this case corresponds to a property of unordered relational structures. In the query specification it is desirable to hide this ordering. Logically, one would have to have a query language corresponding to a logic for Ptime on unordered structures in order to achieve semantic completeness within Ptime and simultaneously to guarantee soundness — soundness in the sense of independence of a linear ordering that is an artifact of the realization.

0.1.3 Which Logics Are Natural?

Consider possible solutions, positive or negative, to the problem whether there is logic for Ptime. The above definition with its very liberal conditions on candidate logics is theoretically appealing because of its connection with recursive presentability. A negative solution in the sense of this definition would be a strong result to the effect that no reasonable logic at all can possibly capture Ptime. A positive result, however, might still leave much to be desired owing to the liberal notion of a logic. In other words, a recursive presentation of graph-Ptime might intuitively be far from constituting a natural logical system. As with the known positive results in the ordered case or for NPtime much may depend on the style of the logic obtained.

The logics to be considered are extensions of first-order logic, as first-order sentences can be evaluated in Logspace. The systematic study of extensions of first-order logic belongs to the domain of abstract model theory. It is worth to pursue this systematic study with particular focus on logics for finite structures. A systematic study of this kind is a possible approach to problems like that concerning the existence of a logic for Ptime. In particular if one conjectures that the problem of a logic for Ptime has a negative solution, then results that state the impossibility of capturing Ptime by logics that satisfy certain stronger criteria can be interesting approximations.

To some extent the formal framework available in abstract model theory is not necessarily well adapted to the finite case. Complexity considerations and considerations that concern logics under a procedural aspect are not a priori accommodated. The standard formalism in abstract model theory is that of *Lindström extensions* or of extensions by generalized quantifiers (Lindström quantifiers); compare the overview in [Ebb85]. Roughly, each such quantifier incorporates one single new structural property and the resulting extension is a minimal one to make this new property available under some natural closure conditions. While this formalism is universally applicable for many purposes — any extension of first-order logic that satisfies some corresponding closure properties is equivalent with a Lindström extension — it may be argued that it is not always optimally adapted to the demands of finite model theory. It seems that a framework for extensions of logics for finite structures that is sufficiently fine grained to reflect algorithmic constraints is still lacking. This issue is connected with the above-mentioned lack of criteria for the 'naturalness' of a logic for finite structures.

0.2 Natural Levels of Expressiveness

First-order logic is not well adapted to the programme of logics for complexity classes. While any individual finite structure is characterized up to isomorphism by a single sentence of first-order logic, natural properties that are of very low complexity are not first-order definable. For instance neither connectedness nor regularity are first-order properties of finite graphs. In fact these examples are typical of the two most apparent defects in the expressive power of first-order logic: first-order logic does not provide expressive means to capture any relational process that requires true recursion (like the generation of the transitive closure of the edge predicate required for connectedness), and first-order logic has no means to express non-trivial cardinality properties (like the equality of the numbers of direct neighbours required for regularity). In short, first-order logic lacks *recursion* and *counting*.

0.2.1 Fixed-Point Logics and Their Counting Extensions

The first defect is taken care of in the extension to fixed-point logics. The adjunction of fixed-point operators leads to logics that capture certain levels of relational recursion. Least or inductive fixed-point logic FP in particular is a very natural logic that has been studied extensively. *Inductively defined* and *increasing* relational processes are captured by FP. The generation of the transitive closure is a simple but typical example for the expressive power of FP above that of first-order logic. An important point is that the increasing nature of these relational processes guarantees termination in a stationary value within polynomially many steps. A further extension in terms of relational recursion for arbitrary rather than increasing processes (that therefore may or may not terminate in a stationary value) is partial fixed-point logic PFP. The interest in FP is justified because by the theorem of Immerman and Vardi it captures PTIME for ordered structures. Similarly, PFP captures PSPACE in the presence of order [Var82, AV89]. In particular, in the presence of order, FP and PFP automatically remedy the second shortcoming of first-order logic: on ordered structures FP and PFP also capture all counting and PTIME, respectively PSPACE, cardinality properties of definable predicates. In the absence of order, however, this is not at all true. In the extreme case of pure sets (graphs without edges) it is easy to see that relational recursion and all of FP and PFP collapse to first-order. Simple cardinality properties of the size of sets like evenness of the number of vertices are not definable in FP or PFP. Moreover, all the simple examples of properties that are not FP-definable but may be recognized in PTIME involve such cardinality properties.

One of the themes underlying our present investigations is the attempt to treat these two most apparent shortcomings of first-order logic over finite structures — recursion and counting — on an equal footing and to consider the extensions FP and PFP to a framework that incorporates counting.

We thus obtain *fixed-point logic with counting* FP+C and *partial fixed-point logic with counting* PFP+C. Roughly speaking we deal with two-sorted variants of the given finite structures, augmented by a second ordered arithmetical sort. A link between the sorts is induced by counting terms that associate cardinality values with formulae that define sets. The usual fixed-point operations can now be applied in this framework to combine relational recursion with the processing of cardinalities.

The conception of fixed-point logic with counting is due to Immerman [Imm87a]. It has not been studied in its own right or even rigorously formalized in the work of Immerman though. The fact that counting is the most obvious defect in FP as compared with PTIME had led Immerman to conjecture that an appropriate extension of FP to FP+C should even be a logic for PTIME in the general case. This conjecture was disproved in a strong sense by Cai, Fürer and Immerman [CFI89]. The sophisticated nature of their example for the separation of FP+C from PTIME indicates on the other hand that FP+C may still be regarded as an interesting level of expressiveness within PTIME that captures many PTIME properties that naturally arise for instance in graph theory. This view has since been corroborated by model theoretic as well as complexity oriented results in [GO93, Ott96a] and we shall see much of this in the sequel. The claim for the naturalness of FP+C mainly rests on the following:

- The expressive power of FP+C can be understood very well in terms of certain FP+C-definable *structural invariants*. An analogous phenomenon was first discovered and exploited in the analysis of FP itself in the work of Abiteboul and Vianu [AV91] and lead to their beautiful result that FP collapses to PFP if and only if PSPACE = PTIME. This approach could successfully be extended to FP+C and PFP+C. In some respects the link between the expressive power of FP+C and PFP+C and the associated invariants is even neater than for FP and PFP themselves. The resulting characterization of the expressive power of FP+C and its relation to PFP+C show that even though FP+C falls short of PTIME it extends to the general case some of the computational and model theoretic features that apply to FP only in the ordered case.
- FP+C and PFP+C are very robust with respect to the actual formalization of the counting extension. There are a number of equivalent characterizations of the expressive power of FP+C, both in terms of logical systems that turn out to be equivalent with FP+C and in computational terms.

Intuitively these show that FP+C constitutes a natural level of expressiveness that at the same time corresponds to some natural level in complexity — even though it is clear that this level is strictly contained in standard PTIME.

0.2.2 The Framework of Infinitary Logic

A different but related approach to the investigation of logics with respect to complexity classes focuses on the a priori logical framework given by certain fragments of infinitary logic. Consider firstly full-fledged infinitary logic $L_{\infty\omega}$, the logic generated by the usual first-order rules for the formation of formulae together with infinite disjunctions and conjunctions over arbitrary sets of formulae. Now any finite graph is characterized up to isomorphism by a first-order sentence. It follows that *every* property of finite graphs is definable in $L_{\infty\omega}$ by a countable disjunction over first-order sentences that characterize all positive instances of this property. $L_{\infty\omega}$ is a universal logic for finite structures — and overshoots all sensible levels of expressiveness.

Particular fragments of infinitary logic, however, have emerged as very useful tools in finite model theory. These are defined in terms of restrictions on the number of variables that may occur (bound or free). These restrictions are well adapted to the study of relational recursion since the processes considered in relational recursion always involve a fixed bound on the maximal arity of auxiliary relations. Thus pure relational recursion is fully contained within $L^{\omega}_{\infty\omega}$, the fragment of $L_{\infty\omega}$ that consists of all formulae that use a finite number of variable symbols each. In particular fixed-point logic FP and partial fixed-point logic PFP are properly embedded in $L^{\omega}_{\infty\omega}$. It is important to note that the completely non-uniform constructors of infinite disjunctions and conjunctions allow to define non-recursive properties of finite structures as well. $L^{\omega}_{\infty\omega}$, too, is completely at odds with complexity on finite structures. Here this may be seen as an advantage. If we consider problems related to logics for complexity in restriction to the framework of $L^{\omega}_{\infty\omega}$ then it is important that this restriction in itself does not trivialize the issues. Since $L^{\omega}_{\infty\omega}$ allows to define properties of arbitrary complexity, the class of all those PTIME properties of finite graphs, that at the same time are $L^{\omega}_{\infty\omega}$-definable, is a non-trivial subclass of graph-PTIME for our purposes. Furthermore the restriction to bounded arity auxiliary relations can be considered as a natural restriction also in terms of the computational complexity of relational problems.

The points made about the inclusion of counting in connection with FP versus FP+C also apply to the framework of $L^{\omega}_{\infty\omega}$. Evenness of the number of vertices or regularity of graphs are not $L^{\omega}_{\infty\omega}$-definable. The reason is that although each individual expression of the from $\exists^{=m}x\varphi(x)$ asserting the existence of exactly m elements that satisfy φ is in first-order logic, the number of variables required in its formalization grows unboundedly with m. $L^{\omega}_{\infty\omega}$ compensates completely all defects of first-order that concern relational recursion but fails for the defects related to counting. It is natural therefore to study also the fragment $C^{\omega}_{\infty\omega}$ of infinitary logic with only finitely many variables in each formula but allowing all *counting quantifiers* $\exists^{=m}$ instead of the usual existential quantifier. These were also first considered in the work of Immerman on the counting extension of FP. FP+C and PFP+C are comprised in $C^{\omega}_{\infty\omega}$ just as FP and PFP are in $L^{\omega}_{\infty\omega}$. We denote the constituent sublogics with a fixed finite bound k on the number of variables $C^{k}_{\infty\omega}$ and $L^{k}_{\infty\omega}$ so that $C^{\omega}_{\infty\omega} = \bigcup_k C^{k}_{\infty\omega}$ and $L^{\omega}_{\infty\omega} = \bigcup_k L^{k}_{\infty\omega}$. The infinitary logics $C^{\omega}_{\infty\omega}$ and $L^{\omega}_{\infty\omega}$ and their constituents $C^{k}_{\infty\omega}$ and $L^{k}_{\infty\omega}$ will be used extensively as frameworks in our exposition. On the one hand they are used in the analysis of mostly still open restricted problems on capturing complexity classes. On the other hand they provide the setting for the comparative analysis of the expressive powers of FP+C, PFP+C, FP and PFP.

The main asset of the fragments $C^k_{\infty\omega}$ and $L^k_{\infty\omega}$ is in fact a methodological point. Both possess very elegant and tractable Ehrenfeucht-Fraïssé style characterizations in terms of games. Such games that capture the expressive power of a logic are an important tool in classical model theory. The classical Ehrenfeucht-Fraïssé theorem relates first-order equivalence of two structures to the existence of a strategy in a game played on these structures. See [EFT94] for a textbook treatment of this technique in the classical context. Variants of these games have been found and employed for various other logics besides first-order. It is remarkable that such games are among the few tools from classical or abstract model theory that are fully available in restriction to finite structures without any alteration. See also [EF95].

A large part of the present work is devoted to the detailed analysis of corresponding games for the $C^k_{\infty\omega}$ and $L^k_{\infty\omega}$. The games are due to Barwise [Bar77] and Immerman [Imm82], and Immerman and Lander [IL90] respectively. The analysis of the games leads to the abstraction of concise PTIME computable structural invariants that characterize finite relational structures exactly up to equivalence in $C^k_{\infty\omega}$ or $L^k_{\infty\omega}$. As mentioned above such invariants were first considered by Abiteboul and Vianu in [AV91] in the context of a computational model for relational recursion and applied to the study of fixed-point logics. A formalization in terms of the underlying fragments of infinitary logic has been presented in Dawar's dissertation [Daw93] and in [DLW95] for the $L^k_{\infty\omega}$ and in [GO93, Ott96a] for the $C^k_{\infty\omega}$. The relationship between the expressive power of FP+C and the invariants for the $C^k_{\infty\omega}$ is also one of the main topics here. We shall investigate this relationship in comparison with FP and the invariants for the $L^k_{\infty\omega}$ as well.

0.2.3 The Rôle of Order and Canonization

Consider once more the problem of logics for complexity classes. As outlined above the essential difficulty in capturing classes below NPTIME in the absence of order can be attributed to the ambiguity in the input representation — it is this ambiguity that imposes the problematic semantic invariance condition on graph algorithms.

Canonization addresses the problem of providing well defined and unique representatives up to a given equivalence relation. Consider canonization up to isomorphism for finite graphs. Suppose there were a PTIME functor defined on the class of all finite graphs that maps each graph of size n to an isomorphic representative over the standard universe $\{0, \dots, n-1\}$ in such a way that any two isomorphic graphs get mapped to the same representative. Such a mapping would constitute what is called PTIME canonization up to isomorphism or PTIME *normalization*. It is not known whether finite graphs admit PTIME normalization. It is clear that PTIME normalization would induce a PTIME algorithm for graph isomorphism; whether the graph isomorphism problem itself is in PTIME is not known. Note that the entire problem of capturing PTIME is solved trivially if there should be PTIME normalization. Any algorithm applied to the standard encoding of a normalized version of the input graph becomes a graph algorithm. PTIME normalization would in fact reduce the capturing of PTIME in the general case to the ordered case.

Some of our investigations concern the variant of this approach in restriction to the framework of the $C^k_{\infty\omega}$ and $L^k_{\infty\omega}$, respectively. We consider canonization up to equivalence in these logics rather than up to isomorphism. There is a direct connection between PTIME canonization for these rougher equivalences and the capturing of the PTIME fragments of these fragments of infinitary logic. Linking these considerations with the above-mentioned PTIME invariants for the $C^k_{\infty\omega}$ we find appealing sufficient criteria that FP+C indeed captures PTIME in restriction to all of $C^\omega_{\infty\omega}$. It remains a challenging open problem whether these conditions are fulfilled. A reduction procedure furthermore shows that the general cases hinge on the three-dimensional cases, i.e. on canonization up to equivalence in the three-variable fragments of $L_{\infty\omega}$.

A main result that will be treated in full detail in the last chapter concerns the two-variable case [Ott95a, Ott95b]. For $L^2_{\infty\omega}$ and $C^2_{\infty\omega}$ we exhibit a strong form of PTIME canonization and thus obtain non-trivial capturing results. The classes of all PTIME properties that are $C^2_{\infty\omega}$- or $L^2_{\infty\omega}$-definable are indeed recursively presentable and may be captured naturally in terms of the complete invariants for $C^2_{\infty\omega}$ or $L^2_{\infty\omega}$, respectively.

0.3 Guide to the Exposition

We summarize the investigations and results that are presented here in order to provide an outlook that may also help to make the overall organization of the material transparent.

- Chapter 1 reviews and introduces basic terminology, summarizes some facts and simple results related in particular to the fragments of infinitary logic and fixed-point logics. Typical examples illustrate the expressive power of these basic logics.

- Chapter 2 provides an introduction to the games for the bounded variable fragments of infinitary logic with and without counting quantifiers. Proofs of the corresponding Ehrenfeucht-Fraïssé type theorems are given. The analysis of these games is carried further to support the definition of the associated invariants in Chapter 3.

- Chapter 3 is devoted to the definition and discussion of the invariants associated with the games. We review those known applications to fixed-point logic without counting that will later be paralleled by, and contrasted with, the corresponding picture for fixed-point logic with counting.

• Chapter 4 is about fixed-point logic with counting. The formal definitions of FP+C and PFP+C are provided here. Some material is collected to corroborate the view of fixed-point logic with counting as a distinguished level of expressiveness within PTIME. The central results rest on applications of the invariants for the $C^k_{\infty\omega}$.

• Chapter 5 considers the formalism of Lindström quantifiers and extensions of fixed-point logic in restriction to cardinality properties. A structural padding technique is developed which among other applications shows that the extension of fixed-point logic by all cardinality Lindström quantifiers is still to weak to comprise the full power of proper fixed-point logic with counting.

• Chapter 6 provides the general treatment of the connection between canonization up to equivalence in the bounded variable fragments of infinitary logic and recursive presentations of the related fragments of PTIME.

• Chapter 7 finally is concerned with the positive results related to the two-variable fragments. In particular there are detailed proofs that the PTIME fragments of both $L^2_{\infty\omega}$ and $C^2_{\infty\omega}$ can be captured.

Throughout the entire text I have attempted to give an almost self-contained exposition. In the first four chapters in particular numerous examples are given and comments and background material provided towards a thorough introduction to the leading concepts, along with the technical development. The last three chapters on the other hand are more specifically devoted to individual results and correspondingly are of a more technical nature.

The main ideas that concern fixed-point logics and the bounded variable fragments of infinitary logic without counting quantifiers are reviewed and developed along with the corresponding notions for the counting case. This seems justified because the results in the case without counting may be obtained through obvious specializations. And also because the comparison between the two scenarios is a major source of motivation for our investigations.

This two-tiered treatment is also intended to make the main results concerning either individual case individually accessible as far as possible. This is true in particular of those new results that concern the case without counting, mainly in Chapters 6 and 7. The only chapters that are devoted to the counting case proper are Chapters 4 and, to some extent, 5.

The main dependencies between chapters are the following. Chapters 2–4 each build on their predecessors. Chapters 5, 6 and 7 are to a large extent independent of each other. For those developments of Chapters 6 and 7, that concern the case without counting, all major prerequisites can be found in Chapters 1–3.

I should also point out that several sections of Chapter 1 (in particular 1.5–1.7) become essential only for the understanding of specific further developments. Chapter 1 may therefore be read selectively and called upon again where necessary.

In the beginning of each chapter there is a brief summary of its contents. Where appropriate I have appended to the individual chapters short sections that discuss summarily the main sources of ideas and results reported or clarify connections with other work.

1. Definitions and Preliminaries

A major part of this chapter serves to review and fix notation and terminology. The material is standard. Readers familiar with the notions addressed might therefore only want to refer back to particular definitions at later points. The main issues of the individual sections are the following:

• Section 1.1 sums up the basics about structures, global relations, logics and types that are relevant for our purposes.

• In Section 1.2 we consider algorithms that deal with structures as inputs and fix some corresponding conventions. Recognizability of classes of finite structures and computability of global relations are discussed.

• The bounded variable fragments of infinitary logic, and the fixed-point logics, are presented in Section 1.3. We also provide some typical examples for the expressive power of these logics.

• Section 1.4 contains some preliminary material about types and definability in the relevant fragments of infinitary logic.

• Section 1.5 deals with interpretations, a concept that plays an important rôle in many definability considerations.

• In Section 1.6 we review the notions of generalized quantifiers and Lindström extensions. In particular we define the class of cardinality Lindström quantifiers.

• Section 1.7 fixes some terminology with respect to the notion of canonization and of complete invariants for arbitrary equivalence relations. We also sketch some technicalities and conventions concerning orderings and pre-orderings.

1.1 Structures and Types

1.1.1 Structures

We deal with *finite structures* exclusively. $\mathrm{fin}[\tau]$ is the class of all finite τ-structures. Unless explicitly stated otherwise, τ stands for some finite and purely relational vocabulary. A structure in $\mathrm{fin}[\tau]$ consists of its universe together with interpretations for the symbols in τ. If $\tau = \{R_1, \ldots, R_s\}$, where R_i is a relation or predicate symbol of arity r_i, we write $\mathfrak{A} = (A, R_1^{\mathfrak{A}}, \ldots, R_s^{\mathfrak{A}})$ for a τ-structure. Thus $R_i^{\mathfrak{A}} \subseteq A^{r_i}$. The superscripts $^{\mathfrak{A}}$ are mostly omitted when there is no danger of confusion.

It is sometimes convenient in special circumstances to admit some variations and extensions of the basic concept of structures:

(i) In order to deal with fixed tuples of *parameters* over some structure $\mathfrak{A}$, one may think of those parameters as interpretations for a corresponding tuple of extra constant symbols. We here prefer to stick with an entirely relational vocabulary and treat parameters as interpretations for variable symbols. The distinction between parameters and variables becomes purely intentional. The class of all τ-structures with fixed tuples of r parameters is denoted

$$\mathrm{fin}[\tau; r] = \Big\{(\mathfrak{A}, \overline{a}) \;\Big|\; \mathfrak{A} \in \mathrm{fin}[\tau], \overline{a} \in A^r\Big\}.$$

(ii) In our formalization of fixed-point logic with counting in Chapter 4 we deal with *two-sorted structures*. These are structures over two disjoint universes, one for each sort. Each relation symbol comes with a specification telling which components range over the first sort and which over the second. Similarly terms and in particular variables have a designated status with respect to the sorts. There is a standard way to represent two-sorted structures by ordinary one-sorted structures that have two additional unary predicates to distinguish the sorts. A structure of the form

$$(A, U_1, U_2, \ldots) \quad \text{with } A = U_1 \,\dot{\cup}\, U_2$$

can thus naturally encode a two-sorted structure with universes U_i for the two sorts. A binary relation R for instance whose i-th component ranges over the i-th sort for $i = 1, 2$ then gets interpreted as a binary relation over A that satisfies $\forall x \forall y (Rxy \to U_1 x \wedge U_2 y)$.

(iii) At some places we consider *weighted structures*. These are structures together with some functions from their domains to some external standard domain, mostly and without loss of generality to the set ω of the natural numbers. A standard example is that of graphs (V, E) with weights put on the edges, formalized by a weight function $\nu \colon V^2 \to \omega$.

Linearly ordered structures play a special rôle. Assume that τ contains a designated binary relation symbol $<$ for a linear ordering. Then the class of all finite τ-structures which are linearly ordered is denoted

$$\mathrm{ord}[\tau] = \Big\{\mathfrak{A} \in \mathrm{fin}[\tau] \;\Big|\; <^{\mathfrak{A}} \text{ a linear ordering of } A\Big\}.$$

When talking of *classes of finite structures* it is generally understood that these are closed with respect to isomorphism. The only exception in our treatment being that in places we restrict attention to structures over *standard* *domains*, meaning structures with an initial segment of the natural numbers for their universe. We denote by stan$[\tau]$ the set of all finite τ-structures over standard domains $n = \{0, \dots, n-1\}$ [1]:

$$\mathrm{stan}[\tau] = \Big\{\mathfrak{A} \in \mathrm{fin}[\tau] \;\Big|\; A = n, n \geqslant 1\Big\}.$$

There is a direct correspondence between linearly ordered structures and structures over standard domains. Each structure $(\mathfrak{A}, <^{\mathfrak{A}})$ in ord$[\tau \dot\cup \{<\}]$ has a unique representative $\langle\mathfrak{A}\rangle$ in stan$[\tau]$ determined by the requirement that $\mathfrak{A} \simeq \langle\mathfrak{A}\rangle$ and that the linear ordering $<^{\mathfrak{A}}$ translates into the natural ordering on the standard domain of $\langle\mathfrak{A}\rangle$ under the isomorphism. Obviously the mapping $(\mathfrak{A}, <^{\mathfrak{A}}) \mapsto \langle\mathfrak{A}\rangle$ induces a bijective correspondence between isomorphism classes of linearly ordered structures and structures over standard domains

$$\langle\ \rangle\colon\ \mathrm{ord}[\tau \dot\cup \{<\}] \Big/\!\simeq\ \longrightarrow\ \mathrm{stan}[\tau].$$

1.1.2 Queries and Global Relations

A class Q of finite τ-structures may be identified with a boolean valued functor χ on fin$[\tau]$ that maps structures to 1 or 0 according to membership in Q: $Q = \big\{\mathfrak{A} \in \mathrm{fin}[\tau] \;\big|\; \chi(\mathfrak{A}) = 1\big\}$. The term *boolean queries* for classes of structures stresses this functorial view. Since classes of structures are tacitly assumed to be closed under isomorphisms, their characteristic functions χ are *invariant under isomorphisms*.

Consider similarly an isomorphism-invariant boolean valued function on fin$[\tau; r]$, $\chi\colon \mathrm{fin}[\tau; r] \to \{0, 1\}$. Such a functor constitutes an *r-ary query on* fin$[\tau]$. An alternative view is that of a mapping from $\mathfrak{A} \in \mathrm{fin}[\tau]$ to a new r-ary predicate $R^{\mathfrak{A}}$ over $\mathfrak{A}$:

$$R^{\mathfrak{A}} = \Big\{\overline{a} \in A^r \;\Big|\; \chi(\mathfrak{A}, \overline{a}) = 1\Big\}.$$

The mapping $R\colon \mathfrak{A} \mapsto R^{\mathfrak{A}}$ is a *global relation of arity r*. At the level of this mapping isomorphism invariance of χ turns into *equivariance under isomorphisms*: if $\pi\colon \mathfrak{A} \to \mathfrak{B}$ is an isomorphism, then $\pi(R^{\mathfrak{A}}) = R^{\mathfrak{B}}$. This is in fact the standard defining condition on global relations or queries as introduced in [CH80]. We note in particular that the value of a global relation over $\mathfrak{A}$ must be invariant under all automorphisms of $\mathfrak{A}$.

Definition 1.1. *A* global relation or query R of arity r *over* fin$[\tau]$ *is a mapping sending each structure* $\mathfrak{A} \in \mathrm{fin}[\tau]$ *to an r-ary predicate* $R^{\mathfrak{A}} \subseteq A^r$ *in $\simeq$-compatible fashion. Whenever $\pi : \mathfrak{A} \to \mathfrak{B}$ is an isomorphism, then π also preserves R: $\pi(R^{\mathfrak{A}}) = R^{\mathfrak{B}}$. The characteristic functor χ_R of R is the boolean valued mapping on* fin$[\tau; r]$ *that sends $(\mathfrak{A}, \overline{a})$ to 1 if $\overline{a} \in R^{\mathfrak{A}}$. Compatibility*

[1] We apply the usual convention to identify the natural number $n \in \omega$ with the set of its predecessors $\{0, \dots, n-1\} \subseteq \omega$.

of R with isomorphisms is equivalent with invariance of χ_R under isomorphisms.

It is often convenient to regard boolean queries (boolean global relations) as special, namely 0-ary cases of r-ary queries. To accommodate this view formally, we may naturally identify 0-ary predicates with boolean values and $\mathrm{fin}[\tau; 0]$ with $\mathrm{fin}[\tau]$.

Some remark on our usage of the term *functor* is in order. We generally apply it to a mapping f whose domain is a class of structures (or of structures with parameters and the like) if f is required to be invariant under isomorphisms: $\mathfrak{A} \simeq \mathfrak{B} \Rightarrow f(\mathfrak{A}) = f(\mathfrak{B})$. If also the range of f consists of structures, for instance $f\colon \mathrm{fin}[\tau] \to \mathrm{fin}[\sigma]$ then the appropriate form of invariance is $\mathfrak{A} \simeq \mathfrak{B} \Rightarrow f(\mathfrak{A}) \simeq f(\mathfrak{B})$.

1.1.3 Logics

Let $\mathcal{L}$ be a logic. We do not require any formal general notion of *a logic*; the apparent generality here only serves to collect some notions, that we later apply to a few individual concrete logics, into common statements. $\mathcal{L}[\tau]$ denotes the class of all formulae of $\mathcal{L}$ in vocabulary τ. A formula $\varphi \in \mathcal{L}[\tau]$ without free variables (one that semantically evaluates to a boolean value over each τ-structure) is a *sentence*. Sentences define classes of structures, concentrating on finite structures we put

$$\mathrm{fmod}(\varphi) := \Big\{ \mathfrak{A} \in \mathrm{fin}[\tau] \,\Big|\, \mathfrak{A} \models \varphi \Big\}.$$

We mostly use letters $\varphi, \psi, \chi, \ldots$ to denote formulae. Let $\varphi \in \mathcal{L}[\tau]$. Variables displayed in brackets like the x_i in $\varphi(x_1, \ldots, x_r)$ indicate that semantically we consider φ as defining a global relation of arity r on $\mathrm{fin}[\tau]$. Over $\mathfrak{A}$, $\varphi(x_1, \ldots, x_r)$ evaluates to the predicate

$$\varphi[\mathfrak{A}] := \Big\{ \overline{a} \in A^r \,\Big|\, \mathfrak{A} \models \varphi[\overline{a}] \Big\}.$$

$\mathfrak{A} \models \varphi[\overline{a}]$ says that φ is satisfied over $\mathfrak{A}$ when the free variables are interpreted as indicated. In this usage the notation $\varphi(x_1, \ldots, x_k)$ does not imply that the displayed x_i must all be syntactically free in φ, but that the *free variables* of φ are among those displayed. We speak of a formula *in free variables* $x_1, \ldots, x_k$ with this meaning: $\mathrm{free}(\varphi) \subseteq \{x_1, \ldots, x_k\}$. For instance, we allow to regard the formula $x_1 = x_2$ also as a formula in free variables x_1, x_2, x_3, and write $\varphi(x_1, x_2, x_3) = x_1 = x_2$ if this view is intended.

Similar conventions apply to second-order variables (predicate variables) where such occur. In particular notation like $\varphi(\overline{X}, \overline{x}) \in \mathcal{L}[\tau]$ indicates that given a τ-structure plus additional interpretations for the second-order variables $\overline{X}$ by extra predicates and for the $\overline{x}$ by elements, φ evaluates to a boolean value. $\mathfrak{A} \models \varphi[\overline{P}, \overline{a}]$ expresses that φ is satisfied in $\mathfrak{A}$ with the indicated interpretations for $\overline{X}$ and $\overline{x}$.

It is sometimes convenient to consider interpretations for some free first- or second-order variables as momentarily fixed. The notation $(\mathfrak{A}, \Gamma)$ for some partial interpretation of free variables through Γ indicates this meaning.

Definition 1.2. *(i) The sentence* $\varphi \in \mathcal{L}[\tau]$ defines *the boolean query* $Q \subseteq \text{fin}[\tau]$ *if* $Q = \text{fmod}(\varphi)$.
(ii) The formula $\varphi(x_1, \ldots, x_r) \in \mathcal{L}[\tau]$ defines *the global relation* R *on* $\text{fin}[\tau]$ *if* $R^{\mathfrak{A}} = \varphi[\mathfrak{A}]$ *for all* $\mathfrak{A} \in \text{fin}[\tau]$.

The expressive power of a logic is determined in terms of those global relations that are definable in this logic.

Definition 1.3. *Two logics are* semantically equivalent *if they define exactly the same global relations on finite structures.*

$$\mathcal{L}_1 \equiv \mathcal{L}_2$$

denotes this semantic equivalence over finite relational structures. The possible weakening of this requirement, that the two logics define the same classes of finite structures is explicitly indicated. We write "$\mathcal{L}_1 \equiv \mathcal{L}_2$ for sentences*" or "*$\mathcal{L}_1 \equiv \mathcal{L}_2$ for boolean queries*".*

Observe that $\mathcal{L}_1 \equiv \mathcal{L}_2$ says that for every formula of $\mathcal{L}_1$ there is a formula of $\mathcal{L}_2$ that is equivalent over finite structures, and vice versa. The weaker notion of equivalence expresses the same requirement in restriction to sentences. The distinction between the two notions of equivalence is of a purely formal nature for our considerations. Most natural logics admit a faithful reduction from definable global relations to definable boolean global relations so that their expressive power is fully determined by their strength in defining classes, i.e. by their sentences.

The notation $\equiv$ for semantic equivalence extends with analogous meaning to classes of queries that are not specified by logics. For instance if $\mathcal{C}$ is a class of queries and $\mathcal{L}$ a logic then $\mathcal{C} \equiv \mathcal{L}$ says that every query in $\mathcal{C}$ is definable by a formula of $\mathcal{L}$ and that conversely all $\mathcal{L}$-definable queries are in $\mathcal{C}$.

1.1.4 Types

We are interested in $\mathcal{L}$-definable properties of element tuples. The $\mathcal{L}$-type of a tuple $\overline{a} = (a_1, \ldots, a_k)$ of elements of a τ-structure $\mathfrak{A}$ is the class of all $\mathcal{L}$-formulae in free variables $\overline{x} = (x_1, \ldots, x_k)$ that are satisfied by $\overline{a}$ in $\mathfrak{A}$:

$$\text{tp}^{\mathcal{L}}_{\mathfrak{A}}(\overline{a}) = \Big\{\varphi(\overline{x}) \in \mathcal{L}[\tau] \;\Big|\; \mathfrak{A} \models \varphi[\overline{a}]\Big\}.$$

$\text{Tp}^{\mathcal{L}}(\tau; k)$ is the class of all $\mathcal{L}[\tau]$-types in variables $x_1, \ldots, x_k$:

$$\text{Tp}^{\mathcal{L}}(\tau; k) = \Big\{\text{tp}^{\mathcal{L}}_{\mathfrak{A}}(\overline{a}) \;\Big|\; \mathfrak{A} \in \text{fin}[\tau], \overline{a} \in A^k\Big\}.$$

$\text{Tp}^{\mathcal{L}}(\mathfrak{A}; k)$ denotes the set of all $\mathcal{L}[\tau]$-types of k-tuples over a particular $\mathfrak{A}$:

$$\mathrm{Tp}^{\mathcal{L}}(\mathfrak{A};k) = \Big\{ \mathrm{tp}^{\mathcal{L}}_{\mathfrak{A}}\big(\overline{a}\big) \;\Big|\; \overline{a} \in A^k \Big\}.$$

We use Greek letters $\alpha, \beta, \ldots$ to denote types. If $\alpha \in \mathrm{Tp}^{\mathcal{L}}(\tau;k)$ then $\alpha \models \varphi$ means that $\mathfrak{A} \models \varphi[\overline{a}]$ whenever $\mathfrak{A}$ and $\overline{a}$ are such that $\mathrm{tp}^{\mathcal{L}}_{\mathfrak{A}}\big(\overline{a}\big) = \alpha$. In case φ is also in $\mathcal{L}[\tau]$ this is just to say that $\varphi \in \alpha$.

We often think of $\mathrm{Tp}^{\mathcal{L}}(\tau;r)$ for $1 \leqslant r < k$ as embedded into $\mathrm{Tp}^{\mathcal{L}}(\tau;k)$ via

$$\mathrm{tp}^{\mathcal{L}}_{\mathfrak{A}}\big(a_1,\ldots,a_r\big) \longmapsto \mathrm{tp}^{\mathcal{L}}_{\mathfrak{A}}\big(\underbrace{a_1,\ldots,a_1}_{k-r}, a_1,\ldots,a_r\big).$$

Some of the logics that play a central rôle in the following possess only a bounded supply of variable symbols. If $\mathcal{L}$ only has variables $x_1,\ldots,x_k$ we agree to apply the notion of $\mathcal{L}$-type only to tuples $\overline{a}$ of length at most k. We adopt the convention that $\mathrm{Tp}^{\mathcal{L}}(\tau;r) = \emptyset$ for $r > k$ in this context.

The most basic types considered are the *atomic* or *quantifier free* types. They are obtained in the above formalism if $\mathcal{L}$ is chosen to be the quantifier free fragment of first-order logic. We write $\mathrm{atp}_{\mathfrak{A}}\big(\overline{a}\big)$ for the collection of all quantifier free formulae that hold true of $\overline{a}$ in $\mathfrak{A}$. Note that each such type can be fully represented by the set of atomic formulae contained in it. In this sense, and for finite relational vocabularies, each atomic type is finite and we may identify an atomic type θ with a single quantifier free formula: the conjunction over all atomic formulae contained in θ together with the negations of all those not contained in θ. Some such syntactic normal form is tacitly assumed when we deal with sets of atomic types. The set of all atomic τ-types in variables $x_1,\ldots,x_k$ is denoted by $\mathrm{Atp}(\tau;k)$:

$$\mathrm{Atp}(\tau;k) = \Big\{ \mathrm{atp}_{\mathfrak{A}}\big(\overline{a}\big) \;\Big|\; \mathfrak{A} \in \mathrm{fin}[\tau], \overline{a} \in A^k \Big\}.$$

Clearly only structures of size up to k need be considered since τ is purely relational. For finite τ therefore, $\mathrm{Atp}(\tau;k)$ is obviously finite. In fact a finite representation of $\mathrm{Atp}(\tau;k)$ in terms of the above syntactic normalization is immediately obtained.

Atomic types in vocabulary $\emptyset$ — in the language of pure sets, where only equality is available — are here called *equality types*. We write $\mathrm{eq}\big(\overline{a}\big)$ for the equality type of $\overline{a}$ and $\mathrm{Eq}(k)$ for the finite set of all equality types in variables $x_1,\ldots,x_k$.

For indistinguishability of structures or structures with parameters in a logic we use the following notation.

Definition 1.4. *For the logic $\mathcal{L}$ we denote by $\equiv^{\mathcal{L}}$ the equivalence relation of indistinguishability in $\mathcal{L}$ or* $\mathcal{L}$-equivalence *both of structures and of structures with parameters:*

(i) $\mathfrak{A} \equiv^{\mathcal{L}} \mathfrak{A}'$ *if* $\mathfrak{A}$ *and* $\mathfrak{A}'$ *satisfy exactly the same $\mathcal{L}$-sentences.*
(ii) $(\mathfrak{A}, \overline{a}) \equiv^{\mathcal{L}} (\mathfrak{A}', \overline{a}')$ *if* $\overline{a}$ *and* $\overline{a}'$ *satisfy exactly the same $\mathcal{L}$-formulae over* $\mathfrak{A}$ *and* $\mathfrak{A}'$ *respectively — equivalently if* $\mathrm{tp}^{\mathcal{L}}_{\mathfrak{A}}\big(\overline{a}\big) = \mathrm{tp}^{\mathcal{L}}_{\mathfrak{A}'}\big(\overline{a}'\big)$.

Note that $\mathrm{Tp}^{\mathcal{L}}(\tau;k)$ may be identified with $\mathrm{fin}[\tau;k] \,/{\equiv^{\mathcal{L}}}$.

1.2 Algorithms on Structures

The informal notion of *algorithms on structures* simply refers to algorithms that are intended to take finite structures as inputs. Algorithms do not deal with abstract structures directly but with *presentations* or *encodings* of these. In standard models of computation — we mainly think of Turing machines — algorithms directly deal with words, ordered strings of symbols over some fixed finite alphabet. Straightforward encoding schemes that faithfully map structures to words are available for structures over standard domains. The implicit ordering of the standard domain allows to enumerate all instantiations of atoms lexicographically. The entire structural information can thus be coded in a binary string that lists the boolean values of all instantiations of atoms in this ordering. Having fixed any such convention for the direct encoding of standard structures we can identify standard structures with their encodings as bit-strings. Without loss of generality we may thus pretend that algorithms for computations on finite τ-structures directly accept elements of $\mathrm{stan}[\tau]$ as inputs. We think of such an algorithm $\mathcal{A}$ as realizing a mapping

$$\begin{aligned} \mathcal{A}\colon \mathrm{stan}[\tau] &\longrightarrow \mathrm{range}(\mathcal{A}) \\ \mathfrak{A} &\longmapsto \mathcal{A}(\mathfrak{A}). \end{aligned}$$

The same considerations apply to algorithms that take structures with parameters as inputs; we replace $\mathrm{stan}[\tau]$ by $\mathrm{stan}[\tau; r]$ for some r. With respect to the range of $\mathcal{A}$ we may distinguish two cases (the distinction is purely intentional): either we regard $\mathrm{range}(\mathcal{A})$ simply as a set of words, or we similarly identify output words with standard objects they encode. In particular we adopt the latter view if we want $\mathcal{A}$ to realize a mapping from structures to structures. As for the input domain, we identify the output domain with some $\mathrm{stan}[\sigma]$ and pretend for instance that $\mathcal{A}$ directly realizes a mapping $\mathcal{A}\colon \mathrm{stan}[\tau] \rightarrow \mathrm{stan}[\sigma]$. Similar conventions can be employed to algorithms which are to output natural numbers: $\mathcal{A}\colon \mathrm{stan}[\tau] \rightarrow \omega$.

Algorithms are unproblematic as far as they realize mappings between certain domains of standard objects (objects with standard encodings). The picture becomes fundamentally different if we want to realize *functors* on structures. Consider the algorithmic evaluation of a boolean query on $\mathrm{fin}[\tau]$. No matter whether we restrict the domain to $\mathrm{stan}[\tau]$ or not, there remains the crucial *invariance condition* that $\mathcal{A}(\mathfrak{A}) = \mathcal{A}(\mathfrak{A}')$ whenever $\mathfrak{A}$ and $\mathfrak{A}'$ are isomorphic. Note that this condition really arises from two sources:

(a) when encoding even a single abstract $\mathfrak{A} \in \mathrm{fin}[\tau]$ through an element of $\mathrm{stan}[\tau]$ then a priori the representative in $\mathrm{stan}[\tau]$ is determined only up to isomorphism.
(b) since a boolean query by definition corresponds to an isomorphism invariant functor, its restriction to $\mathrm{stan}[\tau]$ still has to be invariant under isomorphisms.

Of course there are situations in which uniquely determined representatives of abstract input structures by elements of stan[τ] are available. Most notably this applies to linearly ordered structures. As pointed out above we may identify ord[$\tau \dot\cup \{<\}$]/$\simeq$ with stan[τ]. Linearly ordered structures, even when viewed only up to isomorphism, therefore are themselves objects with standard encodings — whence their notorious special status in considerations concerning logics for complexity classes arises. Somewhat more generally such exceptions occur wherever there is some adequate normalization or canonization procedure available. Variations on this issue will concern us in later chapters.

Entirely similar considerations apply of course to the evaluation of r-ary queries that we choose to realize in the boolean format

$$\mathcal{A}\colon \mathrm{stan}[\tau; r] \longrightarrow \{0,1\}$$

which is also subject to invariance under isomorphism. Finally, functors from structures to structures are realized as

$$\mathcal{A}\colon \mathrm{stan}[\tau] \longrightarrow \mathrm{stan}[\sigma]$$

with invariance condition $\mathfrak{A} \simeq \mathfrak{B} \Rightarrow \mathcal{A}(\mathfrak{A}) \simeq \mathcal{A}(\mathfrak{B})$.

Definition 1.5. *(i) The algorithm* $\mathcal{A}\colon \mathrm{stan}[\tau] \to \{0,1\}$ computes the boolean query $Q \subseteq \mathrm{fin}[\tau]$ *if for all* $\mathfrak{A} \in \mathrm{stan}[\tau]$*:* $\mathcal{A}(\mathfrak{A}) = 1$ *if and only if* $\mathfrak{A} \in Q$*. We also say that* $\mathcal{A}$ recognizes *the class* Q.

(ii) An algorithm $\mathcal{A}\colon \mathrm{stan}[\tau; r] \to \{0,1\}$ computes the r-ary query R *on* fin[τ] *if for all* $(\mathfrak{A}, \overline{a}) \in \mathrm{stan}[\tau; r]$*:* $\mathcal{A}(\mathfrak{A}, \overline{a}) = 1$ *if and only if* $\overline{a} \in R^{\mathfrak{A}}$.

(iii) An algorithm $\mathcal{A}\colon \mathrm{stan}[\tau] \to \mathrm{stan}[\sigma]$ computes the functor $F\colon \mathrm{fin}[\tau] \to \mathrm{fin}[\sigma]$ *if for all* $\mathfrak{A} \in \mathrm{stan}[\tau]$*:* $\mathcal{A}(\mathfrak{A}) \simeq F(\mathfrak{A})$.

(iv) An algorithm $\mathcal{A}\colon \mathrm{stan}[\tau] \to S$ computes the functor $F\colon \mathrm{fin}[\tau] \to S$ *whose range is some domain of standard objects* S *if for all* $\mathfrak{A} \in \mathrm{stan}[\tau]$*:* $\mathcal{A}(\mathfrak{A}) = F(\mathfrak{A})$.

1.2.1 Complexity Classes and Presentations

We are interested in the complexity of problems that concern structures. Consider a structural problem, any of the several kinds of computational problems considered in Definition 1.5. The complexity of such problems is the complexity in the standard sense of its algorithmic realizations $\mathcal{A}$. When dealing with relational input structures we identify the *input size* with the size of the universe of the input structure. Although this parameter may differ from the length of an actual encoding of the input structure, the difference does not matter for our purposes because the complexity classes considered — mainly PTIME and PSPACE, but the same applies to all standard classes from LOGSPACE upward — are robust under polynomially bounded re-scalings of the input size.

A boolean query $Q \subseteq \text{fin}[\tau]$, for instance, is in PTIME if there is a PTIME algorithm $\mathcal{A}$ that computes Q in the sense of Definition 1.5 (i). More precisely, if there is an algorithm $\mathcal{A}$ for Q that terminates its computation on all $\mathfrak{A} = (n, \ldots) \in \text{stan}[\tau]$ in time polynomial in n. It is customary to denote the *class of all* PTIME *queries* again by PTIME, and similar conventions apply to all the usual complexity classes. It should always be clear from context whether we think of for instance PTIME either as the class of all polynomial time computable functions (on the natural numbers, or on some other domain of objects with standard encodings), or as the class of all polynomial time computable queries on finite relational structures. In order to emphasize the latter interpretation we shall sometimes speak of PTIME or other complexity classes *as complexity classes of queries*, a notion introduced by Chandra and Harel [CH80].

As pointed out in the introduction the issue of logics for complexity classes is closely related with the abstract notion of *recursive presentations* for complexity classes of queries.

Definition 1.6. *Let $\mathcal{C}$ be a complexity class of queries. $\mathcal{C}$ is* recursively presented *by a recursive or recursively enumerable set $\mathcal{M}$ of algorithms if each $\mathcal{A} \in \mathcal{M}$ is an algorithmic realization of a query in complexity $\mathcal{C}$, and if $\mathcal{M}$ is semantically complete for $\mathcal{C}$: $\mathcal{C} = \{Q \mid Q$ realized by some $\mathcal{A} \in \mathcal{M}\}$.*

We write $\mathcal{C} \equiv \mathcal{M}$ to stress the underlying semantic equivalence and speak of $\mathcal{M}$ as a recursive presentation *for $\mathcal{C}$. For short we also just call $\mathcal{C}$* recursively enumerable *if it admits a recursive presentation.*

This notion of a recursive presentation similarly applies to any class of problems $\mathcal{C}$ that is specified by algorithmic criteria. Recall from the introduction that ordinary PTIME, as the class of all polynomial time computable problems on natural numbers say, is recursively presentable through the subclass of polynomially clocked PTIME machines (compare Section 0.1.2). PTIME as a class of queries is a paradigmatic semantic class. As a subclass of ordinary PTIME, PTIME as a class of queries is characterized by the *semantic condition of invariance under isomorphism.* The problem whether there are logics for complexity classes of queries essentially is the problem of finding recursive presentations of these semantically defined classes.

1.2.2 Logics for Complexity Classes

This following notion was first presented in precise terms in [Gur88].

Definition 1.7. *Let $\mathcal{C}$ be a complexity class of queries. Assume that $\mathcal{L}$ is a logic with recursive syntax and semantics: for finite τ the set $\mathcal{L}[\tau]$ of τ-formulae of $\mathcal{L}$ is recursive*[2] *and there is a recursive mapping from $\mathcal{L}[\tau]$ to*

[2] Note that just as for recursive presentations it does not really matter whether we require recursive or recursively enumerable syntax. If $(\varphi_i)_{i \geqslant 1}$ is a recursive enumeration then the recursive set $\{(\varphi_i, i) | i \geqslant 1\}$ can replace the original syntax if necessary.

algorithms, $\varphi \mapsto \mathcal{A}_\varphi$ *such that* $\mathcal{A}_\varphi$ *evaluates (the query defined by)* φ *over* $\text{fin}[\tau]$.

$\mathcal{L}$ is a logic for $\mathcal{C}$ *or* captures $\mathcal{C}$ *if* $\mathcal{C}$ *coincides with the class of queries that are definable in* $\mathcal{L}$, $\mathcal{C} \equiv \mathcal{L}$, *and if the recursive semantics* $\varphi \mapsto \mathcal{A}_\varphi$ *maps* $\mathcal{L}[\tau]$ *to algorithmic realizations within* $\mathcal{C}$.

Of course the same notion applies to other classes of queries that are defined in terms of algorithmic criteria, in particular to subclasses of complexity classes of queries. The well known theorem of Immerman and Vardi for instance says that the class of all PTIME queries on ordered structures is captured in exactly this sense by fixed-point logic.

Sometimes it is useful to strengthen these requirements so that other important data also become recursive in terms of the formulae of $\mathcal{L}$, compare [Gur88, EF95]. For instance one may require that data describing complexity bounds on $\mathcal{A}_\varphi$ be recursive in φ. While such strengthenings are crucial for certain arguments we can here stick to the basic notion.

It is worth to note the essential equivalence between the notions of capturing by some logic and that of a recursive presentation. It is clear that a logical representation as in the last definition provides a recursive presentation through $\{\mathcal{A}_\varphi \mid \varphi \in \mathcal{L}\}$. Conversely, any recursive set of algorithms may be regarded as a logic with recursive syntax in the abstract sense; for the semantics choose the obvious one embodied in the algorithms. In this way any recursive presentation of $\mathcal{C}$ can essentially be regarded as a logic that captures $\mathcal{C}$. There are some fine points to be considered if as usual we want abstract logics to satisfy some appropriate regularity criteria as outlined in [Ebb85]. For this it is obviously necessary that $\mathcal{C}$ itself as a set of queries satisfies corresponding closure criteria. At least for classes $\mathcal{C}$ that are natural in such respects it follows that indeed the two notions are equivalent. We do not here enlarge on this issue, in fact an informal concept of 'logics for complexity classes' will be quite sufficient for our purposes.

1.3 Some Particular Logics

1.3.1 First-Order Logic and Infinitary Logic

We write $L_{\omega\omega}$ for first-order logic. The expressive power of first-order logic over finite structures is very unsatisfactory in terms of computational complexity. While all $L_{\omega\omega}$-definable queries are LOGSPACE computable, first-order logic fails to define fundamental structural properties in LOGSPACE or even below. This was briefly discussed in the introduction. First-order equivalence $\equiv^{L_{\omega\omega}}$, however, turns out to be too strong a notion of equivalence over finite structures. Two finite structures are first-order equivalent if and only if they are isomorphic: a first-order sentence, that uses enough variables to enumerate all elements of a given structure and specify all basic relations between them, characterizes that structure up to isomorphism.

Full infinitary logic is the logic $L_{\infty\omega}$ that has the usual first-order rules for the formation of formulae and in addition is closed under *infinitary conjunctions* and *disjunctions*: if Ψ is any set (!) of formulae of $L_{\infty\omega}$ then $\bigwedge\Psi$, the conjunction over Ψ, and $\bigvee\Psi$, the disjunction over Ψ, are also formulae of $L_{\infty\omega}$. Their semantics is the obvious one: $\bigwedge\Psi$ evaluates to true if all formulae in Ψ evaluate to true and $\bigvee\Psi$ evaluates to true if at least one formula in Ψ does. Note that one often has to deal with families of formulae $(\psi_i)_{i\in I}$ and then writes for instance $\bigwedge_{i\in I}\psi_i$ instead of $\bigwedge\{\psi_i \mid i \in I\}$.

As mentioned in the introduction *any* query on finite structures is definable in $L_{\infty\omega}$. This follows from the observation that any finite structure $\mathfrak{A}$ is characterized up to isomorphism by some first-order sentence $\varphi_{\mathfrak{A}}$. If Q is a boolean query, for instance, then the infinite disjunction $\psi = \bigvee\varphi_{\mathfrak{A}}$ over all $\varphi_{\mathfrak{A}}$ for $\mathfrak{A} \in Q \cap \mathrm{stan}[\tau]$ clearly defines Q. Recall, however, that $\varphi_{\mathfrak{A}}$ typically requires $n+1$ variables if the size of $\mathfrak{A}$ is n. This motivates the introduction of the finite variable fragments of $L_{\infty\omega}$.

1.3.2 Fragments of Infinitary Logic

Definition 1.8. *$L^k_{\infty\omega}$ is the fragment of $L_{\infty\omega}$ that consists of formulae using only variable symbols from $\{x_1,\dots,x_k\}$. The union of the $L^k_{\infty\omega}$ is denoted $L^\omega_{\infty\omega}$. It consists of all formulae of $L_{\infty\omega}$ that use finitely many variable symbols (from the standard supply $\{x_i | i \geqslant 1\}$).*

We also consider the corresponding bounded variable fragments of $L_{\omega\omega}$: let $L^k_{\omega\omega}$ denote first-order logic with variable symbols $\{x_1,\dots,x_k\}$.

The union of the $L^k_{\omega\omega}$ is full first-order logic $L_{\omega\omega}$. In actual formalizations we often use variable symbols x, y, z, ... instead of the standardized x_i for the sake of easier readability. The official restriction to standard sets of variables is convenient, however, to have syntactic closure under conjunctions and disjunctions for each $L^k_{\infty\omega}$. We give some examples for the expressive power of the $L^k_{\infty\omega}$. Formalizations with few variable symbols typically require clever re-use of already quantified variables. Examples 1.9 and 1.11 are from [KVa92a], Example 1.16 plays an important rôle in [DLW95] in a context that will also concern us here later.

Example 1.9. Over linear orderings $(A,<)$ two different variable symbols suffice to produce first-order formulae $\varphi_i(x)$, for $i \geqslant 0$, which express that x is the i-th element with respect to $<$. Equivalently, for the standard linear orderings $(n,<)$:

$$(n,<) \models \varphi_i[m] \quad \text{exactly for } m = i.$$

To obtain these formulae put $\varphi_0(x) := \neg\exists y\, y<x$ to define the bottom element in any linear ordering $(A,<)$. Inductively let

$$\varphi_{i+1}(x) := \bigwedge_{j\leqslant i} \neg\varphi_j(x) \wedge \forall y\Big(y<x \rightarrow \bigvee_{j\leqslant i}\varphi_j(y)\Big),$$

where $\varphi_j(y)$ is the result of exchanging x and y throughout the formula $\varphi_j(x)$.

Example 1.10. The class of acyclic directed graphs (A, E) is definable by a sentence of $L^2_{\infty\omega}$. Observe that a finite graph is acyclic if it has no infinite E-paths, or equivalently if there is some finite bound on the length of E-paths. Put $\xi_0(x) := \neg\exists y Eyx$ to characterize those vertices that have no E-predecessors. Inductively let

$$\xi_{i+1}(x) := \forall y\big(Eyx \rightarrow \xi_i(y)\big).$$

Then $(A, E) \models \xi_i[v]$ if and only if there is no E-path of length greater than i reaching v. It follows that

$$\xi := \bigvee_{i\in\omega} \forall x \xi_i(x)$$

characterizes acyclic directed finite graphs as desired.

The sequence of formulae ξ_i from Example 1.10 can be extended to ordinal indices to form formulae $\xi_\alpha(x)$ asserting (over arbitrary structures) that the *E-rank* of x is at most α: inductively $\xi_\alpha(x) = \forall y\big(Eyx \rightarrow \bigvee_{\beta<\alpha} \xi_\beta(y)\big)$. E is *well-founded* if there are no infinite descending E-paths, which is equivalent with the existence of some λ such that $(A, E) \models \bigvee_{\alpha<\lambda} \forall x \xi_\alpha(x)$. It follows that the class of well-founded relations of rank less than λ is $L^2_{\infty\omega}$-definable over arbitrary structures, for each λ. We shall return to two variables, linear orderings, and well-foundedness in Example 1.12 and Corollaries 1.13 and 1.14 below.

Example 1.11. The reflexive transitive closure of a binary relation E is definable in $L^3_{\infty\omega}$. The formula $\psi_1(x, y) := x = y \vee Exy$ describes the pairs of E-distance at most 1. Inductively, $\psi_{i+1}(x, y) := \psi_i(x, y) \vee \exists z\big(\psi_i(x, z) \wedge Ezy\big)$ defines those pairs (x, y), whose E-distance is at most $i + 1$. Thus $\xi(x, y) := \bigvee_{i\geqslant 1} \psi_i(x, y)$ defines the reflexive transitive closure of E.

It is a well known fact (that will also be illustrated in Example 2.6 with a typical game argument) that two variables do not suffice to define transitive closures or to assert transitivity of a given binary relation. The following observation, which is also a direct consequence of the very first exercise in Poizat's [Poi82], is therefore quite intriguing. The way it is proved here is inspired by an argument from [GOR96a].

Example 1.12. The class of finite linear orderings is $L^2_{\infty\omega}$-definable (even over not necessarily finite structures). Let ξ' be an $L^2_{\infty\omega}[<]$-sentence asserting that $<$ is acyclic (obtained from ξ in Example 1.10 through replacing E by $<$). We claim that

$$\begin{aligned} \varphi_{\text{ord}} \quad &:= \quad \xi' \wedge \varphi_0\,, \\ \text{where } \varphi_0 \quad &:= \quad \forall x \forall y\big(x = y \vee x < y \vee y < x\big), \end{aligned}$$

defines the class of all finite linear orderings. It remains to argue that φ_{ord} enforces

(i) irreflexivity — $\forall x \neg x < x$: it does since ξ' forbids loops.
(ii) antisymmetry — $\forall x \forall y \neg\big(x < y \wedge y < x\big)$: it does since ξ' also forbids cycles of length two.
(iii) transitivity — $\forall x \forall y \forall z \big(x < y \wedge y < z \to x < z\big)$: if $x < y$ and $y < z$ then $x \neq z$ (forbidden cycle of length two) and $z \not< x$ (forbidden cycle of length three), whence φ_0 forces $x < z$.

In fact, over not necessarily finite structures and for any ordinal λ the class of all well-orderings of order type less than λ is $L^2_{\infty\omega}$-definable. It follows that for instance the class of all countable well-orderings is $L^2_{\infty\omega}$-definable over arbitrary structures. This claim is justified just as in the last example, if we replace ξ' by a sentence that expresses that $<$ is a well-founded relation of rank less than λ (compare the remark following Example 1.10). ξ' in particular forbids loops and cycles, so that irreflexivity, antisymmetry and transitivity follow as above.

Returning to finite structures, we have the following:

Corollary 1.13. *Let* $< \in \tau$ *and let* τ *have no relation symbols of arity greater than* 2. *Then any query of linearly ordered finite* τ*-structures* $Q \subseteq \text{ord}[\tau]$ *is* $L^2_{\infty\omega}$*-definable.*

Sketch of Proof. Consider without loss $\tau = \{<, E\}$ with one extra binary relation E besides $<$. Let $\mathfrak{A} = (n, <, E)$ have standard domain with the natural interpretation of $<$. Put, for formulae φ_i as in Example 1.9 that characterize the i-th element of the ordering,

$$\begin{aligned}\varphi_{\mathfrak{A}} := \varphi_{\text{ord}} \wedge \forall x \bigvee_{i<n} \varphi_i(x) \wedge &\bigwedge_{(i,j)\in E} \exists x \exists y \big(\varphi_i(x) \wedge \varphi_j(y) \wedge Exy\big) \\ \wedge &\bigwedge_{(i,j)\notin E} \exists x \exists y \big(\varphi_i(x) \wedge \varphi_j(y) \wedge \neg Exy\big).\end{aligned}$$

Then $\varphi_{\mathfrak{A}}$ characterizes $\mathfrak{A}$ up to isomorphism. The disjunction $\bigvee \varphi_{\mathfrak{A}}$ over those of these $\mathfrak{A}$ that are in Q defines Q. □

In particular, for any set $W \subseteq \omega \setminus \{0\}$, the class of those linear orderings whose size is in W is definable in $L^2_{\infty\omega}$. This immediately gives the following:

Corollary 1.14. $L^2_{\infty\omega}$ *is sufficiently expressive to define arbitrarily complex and even non-recursive queries.*

Example 1.15. The class of all finite trees is definable in $L^3_{\infty\omega}$. A structure (A, E) is a *tree* if E is acyclic and if

(i) there is exactly one element without E-predecessors (the root).
(ii) each element has at most one E-predecessor.

Note that connectedness is implied by (i) and (ii) if there cannot be cycles. Now cycles can be forbidden in $L^2_{\infty\omega}$ according to Example 1.10. (i) can even be formalized in $L^2_{\omega\omega}$ through

$$\exists x \forall y \neg Eyx \wedge \forall x \forall y \Big(\big(\forall y \neg Eyx \wedge \forall x \neg Exy\big) \longrightarrow x = y \Big).$$

(ii) actually needs a third variable for mere counting, as for instance in the formalization

$$\forall x \forall y \forall z \big(Eyx \wedge Ezx \to y = z \big).$$

A *binary tree* is a tree in which all nodes have out-degree 0 or 2. A *full binary tree* is a binary tree in which all leaves (nodes of out-degree 0) are at the same distance from the root (at the same height).

Example 1.16. The class of all full binary trees is definable in $L^3_{\infty\omega}$. The condition on equal height of all leaves is formalized in three variables using auxiliary formulae φ_i that define the set of vertices at height i. These are constructed inductively similar to the formulae used in Example 1.9. $\varphi_0(x) := \neg\exists y Eyx$ defines the root. Inductively $\varphi_{i+1}(x) := \exists y(\varphi_i(y) \wedge Eyx)$ is as desired. The $L^2_{\infty\omega}$-sentence $\bigvee_i \forall x\big(\neg\exists y Exy \to \varphi_i(x)\big)$ forces all leaves to be at the same height.

It might look surprising that the condition on out-degree 2 should also be expressible in just three variables. It is however, and quite simply, in the context of trees. The sentence

$$\neg\exists x_1 \exists x_2 \exists x_3 \Big(\bigwedge_{i \neq j} x_i \neq x_j \wedge \bigwedge_{i=1,2,3} \Big(\exists x_i \bigwedge_{j \neq i} Ex_i x_j \Big) \Big)$$

expresses that there are no three vertices with common direct predecessors for any two of them. If the in-degree can at most be 1 then this is equivalent with a bound of 2 on the out-degree. The condition that the out-degree of all vertices apart from the leaves must be at least 2 is trivially first-order expressible in three variables.

In Example 1.15 some variables had to be spent for mere counting in the conditions on the degree of vertices. In general the explicit formalization of $\exists^{\geqslant m} x \varphi(x)$ requires at least m variables as for instance in

$$\exists x_1 \ldots \exists x_m \Big(\bigwedge_{1 \leqslant i < j \leqslant m} x_i \neq x_j \wedge \bigwedge_{1 \leqslant i \leqslant m} \varphi(x_i) \Big).$$

Indeed, over the empty vocabulary it follows easily from game arguments that are treated in Chapter 2, that no sentence of $L^{m-1}_{\infty\omega}$ can express the existence of at least m distinct elements. It is therefore natural to consider fragments of first-order and infinitary logic with a bounded supply of variables that admit, however, arbitrary *counting quantifiers* $\exists^{\geqslant m}$.

Definition 1.17. $C^k_{\infty\omega}$ *is the fragment of* $L_{\infty\omega}$ *that consists of formulae using only variable symbols from* $\{x_1, \dots, x_k\}$ *but allows arbitrary counting quantifiers* $\exists^{\geqslant m}$, $m \geqslant 1$, *instead of the standard existential quantifier* $\exists$. *The union of the* $C^k_{\infty\omega}$ *is denoted* $C^\omega_{\infty\omega}$.

Some simplifications in actual formalizations are achieved with the following agreements. We write $\exists^{=m}x\varphi(x)$ as an abbreviation for $\exists^{\geqslant m}x\varphi(x) \wedge \neg\exists^{\geqslant m+1}x\varphi(x)$. This notation may be extended to allow $\exists^{=0}x\varphi(x)$ as shorthand for $\neg\exists x\varphi(x)$. Quantifiers $\exists^{>m}$, $\exists^{\leqslant m}$ and $\exists^{<m}$ are similarly introduced.

We give a few simple examples for the expressive power of $C^2_{\infty\omega}$. A much more central example, the expressibility of the stable colouring of graphs in $C^2_{\infty\omega}$, is presented in detail in Chapter 2.

Example 1.18. The class of regular graphs is definable in $C^2_{\infty\omega}$. Apart from the standard axioms for graphs, which are in two variables, take

$$\chi := \forall x \forall y \bigwedge_{m \in \omega} \big(\exists^{\geqslant m} y Exy \leftrightarrow \exists^{\geqslant m} x Eyx\big)$$

to express regularity.

Example 1.19. Even $C^2_{\omega\omega}$ does not have the *finite model property*, i.e. there are sentences of $C^2_{\omega\omega}$ without finite models that are satisfiable in infinite structures. For instance it is expressible in $C^2_{\omega\omega}$ that a binary relation E is the graph of an injective total function that is not surjective:

$$\forall x \exists^{=1} y Exy \wedge \forall y \exists^{\leqslant 1} x Exy \wedge \exists y \forall x \neg Exy.$$

First-order logic with two variables $L^2_{\omega\omega}$ on the other hand is known to have the finite property by a result of Mortimer's, see Theorem 7.35.

Example 1.20. Any finite equivalence relation is characterized up to isomorphism among all finite equivalence relations by its $C^2_{\infty\omega}$-theory. The axiomatization of equivalence relations, however, needs at least three variables. Let $\mathfrak{A} = (A, \sim)$ where $\sim$ is an equivalence relation over A. A complete invariant that characterizes any finite equivalence relation up to isomorphism is its spectrum, the set of pairs $\big(i, \nu_i\big)$ such that there are exactly ν_i classes of size i in $(A, \sim)$. This information is expressed in two variables by

$$\bigwedge_i \exists^{=i\nu_i} x\, \exists^{=i} y\; x \sim y.$$

That two variables even with counting quantifiers do not suffice to express transitivity of a binary relation E is shown easily with games discussed in Chapter 2, see Example 2.6.

Example 1.21. The classes of all trees and of all full binary trees are definable in $C^2_{\infty\omega}$. In fact we observed in Example 1.15 and 1.16 that a third variable was only needed for counting purposes. For instance (ii) of Example 1.15 now simply becomes $\forall x \exists^{\leqslant 1} y Eyx$.

1.3.3 Fixed-Point Logics

Fixed-point logics provide extensions of first-order logic that capture some *recursion on relations*. Consider a formula $\varphi(X, \overline{x})$ in free variables X and $\overline{x}$ of matching arities, X a second-order variable for a predicate of arity r and $\overline{x} = (x_1, \ldots, x_r)$. Over structures $\mathfrak{A}$ that interpret φ up to X and the $\overline{x}$ the formula φ induces a mapping on r-ary predicates:

$$\begin{array}{rcl} F_\varphi^{\mathfrak{A}}\colon \mathcal{P}(A^r) & \longrightarrow & \mathcal{P}(A^r) \\ P & \longmapsto & \big\{\overline{a} \in A^r \bigm| \mathfrak{A} \models \varphi[P, \overline{a}]\big\}. \end{array}$$

$\mathcal{P}(A^r)$ denotes the power set of A^r. For any mapping $F\colon \mathcal{P}(A^r) \to \mathcal{P}(A^r)$ consider the following two recursive processes.

The partial fixed point of F. Inductively, a sequence of r-ary predicates is generated through iterated application of F to the empty predicate:

$$\begin{array}{l} X_0 := \emptyset \\ X_{i+1} := F(X_i). \end{array}$$

With this sequence the *partial fixed point* of F is defined to be either the stationary value of this sequence if such exists or the empty set otherwise:

$$\mathrm{PFP}[F] := \begin{cases} X_i & \text{if } X_{i+1} = X_i \\ \emptyset & \text{if } X_{i+1} \neq X_i \text{ for all } i. \end{cases}$$

The inductive or inflationary fixed point associated with F. Inductively, the following *increasing* sequence of r-ary predicates is generated:

$$\begin{array}{l} X_0 := \emptyset \\ X_{i+1} := X_i \cup F(X_i). \end{array}$$

Note that this sequence may alternatively be obtained through iterated application of a modified operation F_{infl} to the empty predicate. F_{infl} is the *inflationary* variant of F, defined by $F_{\text{infl}}(P) := P \cup F(P)$; generally an operation G on sets is called inflationary if always $G(P) \supseteq P$.

Since the domain A is finite, the sequence of the X_i must become stationary within polynomially many steps of the iteration. The limit reached is the *inductive or inflationary fixed point*:

$$\mathrm{IFP}[F] := X_{i_0} \quad \text{where } i_0 = \min\{i | X_{i+1} = X_i\}.$$

$$\begin{array}{rl} \text{Equivalently:} & \mathrm{IFP}[F] = \bigcup_i X_i, \\ \text{or} & \mathrm{IFP}[F] = \mathrm{PFP}[F_{\text{infl}}]. \end{array}$$

The X_i in these definitions are referred to as the *stages* in the generation of the respective fixed points.

Fixed-point logics provide constructors for the definition of those predicates that are obtained as fixed points of operators $F = F_\varphi$ as above. For the inductive definition of their syntax and semantics it is important to consider formulae φ that may have other free first- and second-order variables than X and $\overline{x}$. To make clear which variables are those involved in the fixed-point process, we write $\mathrm{PFP}_{X,\overline{x}}\varphi$ for the partial fixed point associated with the operator F_φ, where free variables other than X and $\overline{x}$ in φ are kept fixed as parameters with respect to the operation F_φ. The same applies to the inductive fixed point.

Syntax and semantics of partial fixed-point logic PFP are defined as follows. Syntactically, PFP is the closure of atomic formulae (where second-order variables are admitted) under the usual first-order constructions and the partial fixed-point constructor. The latter allows to construct a new formula $\psi = \big[\mathrm{PFP}_{X,\overline{x}}\varphi\big]\overline{x}$ from any formula $\varphi \in \mathrm{PFP}$, where X is a second-order variable of some arity r and $\overline{x}$ an r-tuple of distinct first-order variables. For the free occurrence of variables put $\mathrm{free}(\psi) := \big(\mathrm{free}(\varphi) \cup \{\overline{x}\}\big) \setminus \{X\}$.

The semantics for the PFP-constructor is that induced by the partial fixed point of the corresponding operator. If $(\mathfrak{A}, \varGamma)$ is a structure of the appropriate vocabulary together with interpretations for all variables in $\mathrm{free}(\varphi) \setminus \{X, \overline{x}\}$, then

$$(\mathfrak{A}, \varGamma) \models \psi[\overline{a}] \quad \text{if} \quad \overline{a} \in \mathrm{PFP}[F_\varphi^{\mathfrak{A},\varGamma}],$$

where $F_\varphi^{\mathfrak{A},\varGamma} : \mathcal{P}(A^r) \to \mathcal{P}(A^r)$ is the operation induced over $(\mathfrak{A}, \varGamma)$ by φ.

Definition 1.22. *Partial fixed-point logic* PFP *is the smallest logic closed under the usual first-order constructors and the* PFP*-constructor.*

Clearly PFP $\subseteq$ PSPACE. Recall that first-order queries are in LOGSPACE. Over structures of size n PFP-processes in arity r terminate within at most $2^m + 1$ steps, where $m = n^r$. A corresponding step counter can be run in PSPACE and only two stages of the fixed-point generation need be kept simultaneously.

For *fixed-point logic* FP one considers the inductive or inflationary fixed-point operator instead of the partial fixed-point operator. The syntax is exactly the same as for PFP, only that we write FP instead of PFP in fixed-point formulae and choose the semantics based on the inductive fixed point of the underlying operation: for φ and $(\mathfrak{A}, \varGamma)$ as above, and $\psi = \big[\mathrm{FP}_{X,\overline{x}}\varphi\big]\overline{x}$ put

$$(\mathfrak{A}, \varGamma) \models \psi[\overline{a}] \quad \text{if} \quad \overline{a} \in \mathrm{IFP}[F_\varphi^{\mathfrak{A},\varGamma}].$$

Definition 1.23. *Fixed-point logic* FP *is the smallest logic closed under first-order constructors and the* FP*-constructor.*

It is obvious that FP $\subseteq$ PTIME as IFP-processes terminate within a polynomial number of iterations.

We regard FP as a sublogic of PFP in the sense that $\big[\mathrm{FP}_{X,\overline{x}}\varphi\big]\overline{x}$ may be identified with $\big[\mathrm{PFP}_{X,\overline{x}}(X\overline{x} \vee \varphi)\big]\overline{x}$, since the inflationary variant of the operation F_φ is the same as $F_{\varphi'}$ for $\varphi' = X\overline{x} \vee \varphi$.

Theorem 1.24 (Immerman, Vardi). *Over the class of linearly ordered finite structures* FP *captures* PTIME*: a class of linearly ordered structures is recognized by a* PTIME *algorithm if and only if it is the class of finite models of some sentence of* FP*; a global relation on ordered structures is* PTIME *computable if and only if it is definable in* FP*.*

Theorem 1.25 (Abiteboul, Vardi, Vianu). *Over linearly ordered finite structures* PFP *captures* PSPACE*: a class of linearly ordered structures is recognized by a* PSPACE *algorithm if and only if it is the class of finite models of some sentence of* PFP*; a global relation on ordered structures is* PSPACE *computable if and only if it is definable in* PFP*.*

Theorem 1.24 was obtained in [Imm86] and [Var82]. Theorem 1.25 combines results from [Var82] and [AV89] (via equivalences with relational While-queries).

Let us briefly indicate how the fixed-point processes available in FP and PFP can be used to code PTIME, respectively PSPACE, computations over linearly ordered structures — an observation that is the key to the preceding theorems.

Example 1.26. Polynomially space bounded configurations of a Turing machine can be encoded by predicates over the ordered domains of the input structures. The arity of the predicate depends on the degree of the space bound. A definable indexing of the tape cells is provided by the lexicographic ordering on an appropriate power of the universe. In terms of these encodings by predicates it is easily seen that the successor step from one configuration to the next becomes first-order definable.

For a PSPACE machine that computes a boolean query say, consider the partial fixed-point process based on this first-order formalization of the computational successor (with a corresponding definition of the initial configuration). Provided that we adapt the transition function of the machine in such a way that an eventual halting configuration is formally repeated indefinitely, the limit value of this partial fixed-point process is an encoding of the halting configuration if the machine halts on the given input structure.

For a PTIME machine we may pass from the encoding of individual configurations to a cumulative encoding of initial segments of the computation path. We use a fixed-point variable with additional entries for a lexicographic indexing also for a polynomial number of time steps. This leads to an inductive fixed-point process whose i-th level describes the computation path up to step i. The limit reached in this process is an encoding of the entire computation path on the given input structure.

In either case the actual result of the computation (acceptance or rejection in the case of a boolean query) is first-order definable in terms of the final configuration.

The original and intuitively also very appealing definition of fixed-point logic FP in terms of a *least fixed-point operator* LFP rather than the inductive or inflationary operator IFP is equivalent in expressive power. This equivalence between the least fixed-point extension of first-order logic and the formalization using IFP is shown by Gurevich and Shelah [GS86], see also [Lei90]. The formalization using inductive fixed points will be more convenient in connection with the counting extensions to be introduced in Chapter 4.

As a further indication of the versatility of fixed-point constructs we briefly consider *systems of simultaneous fixed points.*

Example 1.27. Let $\overline{\varphi} = (\varphi_i(X_1, \ldots, X_m, \overline{x}^{(i)}))_{i=1,\ldots,m}$ be a tuple of formulae in which the arities r_i of the $\overline{x}^{(i)}$ match those of the X_i. With $\overline{\varphi}$ one may associate an operation $F_{\overline{\varphi}}$ on $\mathcal{P}(A^{r_1}) \times \cdots \times \mathcal{P}(A^{r_m})$, over structures that interpret the φ_i up to the X_i and $\overline{x}^{(i)}$, through:

$$\overline{P} \longmapsto \Big(\big\{\overline{a}^{(i)} \bigm| (\mathfrak{A}, \overline{P}) \models \varphi_i[\overline{a}^{(i)}]\big\}\Big)_{i=1,\ldots,m}.$$

Iteration in the spirit of IFP or PFP yields inflationary or partial fixed points for such systems of interrelated relational transformations. Standard techniques for the encoding of tuples of relations through a single relation of higher arity can be employed to show that fixed-point logic FP and partial fixed-point logic PFP are closed under the formation of respective fixed points for systems.

1.3.4 Fixed-Point Logics and the $L^k_{\infty\omega}$

The fixed-point logics PFP and FP both are sublogics of $L^\omega_{\infty\omega}$. Since FP itself is a sublogic of PFP it suffices to review the argument that yields PFP $\subseteq L^\omega_{\infty\omega}$. Since we have already seen that $L^\omega_{\infty\omega}$ expresses queries of arbitrary complexity, whereas PFP-definable queries obviously are in PSPACE, it follows that PFP $\subsetneq L^\omega_{\infty\omega}$.

For a convenient statement of the following arguments it is useful to eliminate first-order parameters in fixed-point processes. This is easily done at the expense of an increase in the arity of the second-order fixed-point variable.

Lemma 1.28. *Any formula in* PFP *is equivalent with one in which fixed-point operators are applied only in the form* $\mathrm{PFP}_{X,\overline{x}}\varphi$*, where all free first-order variables of* φ *are among those in* $\overline{x}$*. The same holds of* FP.

Sketch of Proof. Consider φ with free first-order variables $\overline{x}$, $\overline{z}$. Assume that $\overline{x}$ and $\overline{z}$ together consist of pairwise distinct variables and that no variable in $\overline{z}$ is bound in φ. Let φ' be the result of replacing any atom $X\overline{u}$ in φ by the atom $Z\overline{z}\,\overline{u}$, where Z is a new second-order variable whose arity is that of X plus arity of $\overline{z}$. It is easily seen that $\big\{\overline{z}\,\overline{x} \bigm| [\mathrm{PFP}_{X,\overline{x}}\varphi]\overline{x}\big\} = \big\{\overline{z}\,\overline{x} \bigm| [\mathrm{PFP}_{Z,\overline{z}\,\overline{x}}\varphi']\overline{z}\,\overline{x}\big\}$. □

Lemma 1.29. *Let* $\varphi \in L^k_{\infty\omega}$ *be a formula, possibly with free second-order variables. Let* X *be a second-order variable of arity* r*, and* $\overline{x}$ *an* r*-tuple of* *pairwise distinct variables from $\{x_1, \dots, x_k\}$. Assume that all free first-order variables of φ are among those in $\overline{x}$. Then $\psi := \big[\mathrm{PFP}_{X,\overline{x}}\varphi\big]\overline{x}$ is equivalent with a formula in $L^k_{\infty\omega}$.*

Proof. Let us write $\varphi(X, \overline{x})$ for the given φ, second-order variables apart from X are irrelevant in the argument. Without loss of generality we assume that all X-atoms in φ are of the form $X\overline{y}$ for an r-tuple of mutually distinct variables $\overline{y}$. An atom $Xx_ix_i\overline{y}'$ for instance can be replaced by the formula $\exists x_j(x_j = x_i \wedge Xx_jx_i\overline{y}')$ for a variable x_j different from x_i and the $\overline{y}'$.

It is shown inductively that the stages X_i in the generation of the fixed point $\mathrm{PFP}_{X,\overline{x}}\varphi(X, \overline{x})$ are definable by $L^k_{\infty\omega}$-formulae $\varphi_i(\overline{x})$. $\varphi_0(\overline{x})$ is the result of replacing each X-atom in $\varphi(\overline{x})$ by some universally false expression like $\neg x = x$ in the same variables. Inductively assume that $\varphi_i(\overline{x})$ is as desired. Semantically $\varphi_{i+1}(\overline{x})$ has to be the result of substituting $\big\{\overline{x} \bigm| \varphi_i(\overline{x})\big\}$ for X in $\varphi(\overline{x})$. Consider a single atom $X\overline{y}$ in $\varphi(\overline{x})$, $\overline{y}$ a tuple of mutually distinct variables. Choose a permutation of the set of variables $\{x_1, \dots, x_k\}$ that maps $\overline{x}$ to $\overline{y}$. An application of this permutation to all variables in $\varphi_i(\overline{x})$ yields a formula $\varphi_i(\overline{y})$ with $X_i = \big\{\overline{y} \bigm| \varphi_i(\overline{y})\big\}$ that can be substituted in place of $X\overline{y}$ in $\varphi(\overline{x})$ without any clashes with bound variables. $\varphi_{i+1}(\overline{x})$ is the result of corresponding substitutions for all X-atoms in $\varphi(\overline{x})$. It follows that $\big[\mathrm{PFP}_{X,\overline{x}}\varphi(X, \overline{x})\big]\overline{x}$ is equivalent with $\bigvee_{i \geqslant 0}\Big(\forall\overline{x}(\varphi_i(\overline{x}) \leftrightarrow \varphi_{i+1}(\overline{x})) \wedge \varphi_i(\overline{x})\Big)$. □

Corollary 1.30. $\mathrm{FP} \subseteq \mathrm{PFP} \subsetneq L^\omega_{\infty\omega}$.

The first semantic inclusion is strict if and only if PTIME $\subsetneq$ PSPACE by a theorem of Abiteboul and Vianu. We shall come back to this in Chapter 3.

Note that the $L^k_{\infty\omega}$ are indeed not closed with respect to PFP or FP, since fixed points may involve first-order parameters. The elimination of these according to Lemma 1.28 may need extra variables. An easy example to this effect is the following one in the language of a single binary relation E.

Example 1.31. The formula $\varphi(x, y) := \big[\mathrm{FP}_{X,x}(x{=}y \vee \exists y(Xy \wedge Eyx))\big]x$ defines the reflexive transitive closure of E. If φ were equivalent with a formula in $L^2_{\infty\omega}$ it would follow that transitivity of a symmetric reflexive relation E is expressible in $L^2_{\infty\omega}$ since it is expressed by $\forall x\forall y\varphi(x, y)$. As we shall see in Example 2.6 below, transitivity is not even definable in $C^2_{\infty\omega}$.

Another corollary to Lemma 1.29 concerns a frequently used collapsing argument over structures that realize few $L^k_{\infty\omega}$-types.

Corollary 1.32. *Let $\mathcal{K} \subseteq \mathrm{fin}[\tau]$ be a class such that for some d each $\mathfrak{A} \in \mathcal{K}$ realizes at most d different $L^k_{\infty\omega}$-types. Then any fixed point over an $L^k_{\infty\omega}$-formula without first-order parameters as considered in Lemma 1.29 is reached after at most $2^d + 1$ iterations. It follows in particular that $L^k_{\omega\omega}$ is closed with respect to fixed points without first-order parameters over $\mathcal{K}$.*

Sketch of Proof. Under the assumptions and for $\varphi(X,\overline{x})$ as in Lemma 1.29, there are no more than 2^d different $L^k_{\infty\omega}$-definable k-ary relations over any $\mathfrak{A} \in \mathcal{K}$ that can occur as stages in the fixed-point iteration. This is because each $L^k_{\infty\omega}$-definable relation over $\mathfrak{A}$ corresponds to a union of $L^k_{\infty\omega}$-types over A^k. Therefore, in the notation of the proof of Lemma 1.29, $[\mathrm{PFP}_{X,\overline{x}}\varphi]\overline{x}$ is equivalent over $\mathcal{K}$ with

$$\psi(\overline{x}) := \forall\overline{x}\big(\varphi_m(\overline{x}) \leftrightarrow \varphi_{m+1}(\overline{x})\big) \wedge \varphi_m(\overline{x}),$$

for $m = 2^d$. If φ is in $L^k_{\omega\omega}$, then so is ψ. □

1.4 Types and Definability in the $L^k_{\infty\omega}$ and $C^k_{\infty\omega}$

The following lemma applies to the fragments of infinitary logic considered above. Since they provide conjunctions and disjunctions over arbitrary sets of formulae, they are in particular *closed with respect to countable conjunctions and disjunctions*. Generally, a logic $\mathcal{L}$ is closed with respect to countable conjunctions and disjunctions if for any family $(\varphi_i)_{i\in\omega}$ of formulae in $\mathcal{L}$ there are $\mathcal{L}$-formulae with semantics corresponding to the conjunction and disjunction over the φ_i, respectively. As usual we write $\bigwedge_{i\in\omega}\varphi_i$ and $\bigvee_{i\in\omega}\varphi_i$ for these. *Closure under negation* is similarly defined. The lemma is a consequence of the fact that, for fixed finite vocabulary τ, $\mathrm{fin}[\tau]$ and $\mathrm{fin}[\tau;k]$ are countable up to isomorphism.

Lemma 1.33. *Let $\mathcal{L}$ be closed under negation and with respect to countable conjunctions and disjunctions, τ a finite vocabulary. Then*

(i) each $\mathfrak{A} \in \mathrm{fin}[\tau]$ is characterized up to $\equiv^{\mathcal{L}}$ by some sentence $\varphi_{\mathfrak{A}} \in \mathcal{L}[\tau]$, i.e. for all $\mathfrak{A}' \in \mathrm{fin}[\tau]$: $\mathfrak{A}' \equiv^{\mathcal{L}} \mathfrak{A} \iff \mathfrak{A}' \models \varphi_{\mathfrak{A}}$.

(ii) each $\mathcal{L}$-type over fin[τ] *is* isolated *by a formula of $\mathcal{L}$, i.e. for all $\alpha \in \mathrm{Tp}^{\mathcal{L}}(\tau;r)$ there is a formula $\varphi_\alpha(x_1,\ldots,x_r) \in \mathcal{L}[\tau]$ such that for all $(\mathfrak{A},\overline{a}) \in \mathrm{fin}[\tau;r]$: $\mathrm{tp}^{\mathcal{L}}_{\mathfrak{A}}(\overline{a}) = \alpha \iff \mathfrak{A} \models \varphi[\overline{a}]$.*

(iii) a global relation on fin[τ] *is* definable *in $\mathcal{L}$ if and only if it is closed with respect to $\equiv^{\mathcal{L}}$.*
In the boolean case, $Q \subseteq \mathrm{fin}[\tau]$ is $\mathcal{L}$-definable if for all $\mathfrak{A}$, $\mathfrak{A}' \in \mathrm{fin}[\tau]$: $\mathfrak{A} \equiv^{\mathcal{L}} \mathfrak{A}' \Longrightarrow \big(\mathfrak{A} \in Q \Leftrightarrow \mathfrak{A}' \in Q\big)$.
For an r-ary global relation R on fin[τ]*, R is $\mathcal{L}$-definable if for all $(\mathfrak{A},\overline{a})$, $(\mathfrak{A}',\overline{a}') \in \mathrm{fin}[\tau;r]$: $(\mathfrak{A},\overline{a}) \equiv^{\mathcal{L}} (\mathfrak{A}',\overline{a}') \Longrightarrow \big(\overline{a} \in R^{\mathfrak{A}} \Leftrightarrow \overline{a}' \in R^{\mathfrak{A}'}\big)$.*

Recall that by convention for logics like $L^k_{\infty\omega}$ and $C^k_{\infty\omega}$, which only have variables $x_1,\ldots,x_k$, we do not consider types in more than k variables: $\mathrm{Tp}^{\mathcal{L}}(\tau;r) = \emptyset$ for $r > k$. The statement in (iii) has to to be restricted to arities $r \leqslant k$ accordingly.

Proof. (i) Since $\mathcal{L}$, as any logic, cannot distinguish between isomorphic structures, and since $\mathrm{fin}[\tau]/\simeq$ is countable, it follows that also $\mathrm{fin}[\tau]/\equiv^{\mathcal{L}}$ is countable. Let $(\mathfrak{A}_i)_{i\in I}$, for I finite or $I=\omega$, be a system of representatives for $\mathrm{fin}[\tau]/\equiv^{\mathcal{L}}$. For $i,j\in I, i\neq j$ let $\varphi_{ij}\in\mathcal{L}[\tau]$ be such that $\mathfrak{A}_i\models\varphi_{ij}$ and $\mathfrak{A}_j\models\neg\varphi_{ij}$. It follows that each $\mathfrak{A}_i$ is characterized up to $\equiv^{\mathcal{L}}$ by the $\mathcal{L}$-sentence $\varphi_i := \bigwedge_{j\neq i}\varphi_{ij}$. For the claim about boolean global relations in (iii) assume that Q is $\equiv^{\mathcal{L}}$-closed. Then the disjunction $\bigvee\varphi_i$ over those i, for which $\mathfrak{A}_i\in Q$, defines Q.
(ii) and that part of (iii) that concerns r-ary queries are proved in exactly the same way using a system of representatives for $\mathrm{Tp}^{\mathcal{L}}(\tau;r)=\mathrm{fin}[\tau;r]\,/{\equiv^{\mathcal{L}}}$ instead of $\mathrm{fin}[\tau]\,/{\equiv^{\mathcal{L}}}$. □

Corollary 1.34. *For $\mathcal{L}=L^k_{\infty\omega}$ or $C^k_{\infty\omega}$:*

$$\mathfrak{A}\equiv^{\mathcal{L}}\mathfrak{A}' \quad \textit{if and only if} \quad \mathrm{Tp}^{\mathcal{L}}(\mathfrak{A};k)=\mathrm{Tp}^{\mathcal{L}}(\mathfrak{A}';k).$$

Sketch of Proof. Let $\varphi_{\mathfrak{A}}$ characterize $\mathfrak{A}$ up to $\equiv^{\mathcal{L}}$ as in (i) of Lemma 1.33. Then $\mathfrak{A}\equiv^{\mathcal{L}}\mathfrak{A}'$ iff $\alpha\models\varphi_{\mathfrak{A}}$ for some (any) type $\alpha\in\mathrm{Tp}^{\mathcal{L}}(\mathfrak{A}';k)$. Therefore $\mathrm{Tp}^{\mathcal{L}}(\mathfrak{A};k)=\mathrm{Tp}^{\mathcal{L}}(\mathfrak{A}';k)$ implies $\mathfrak{A}\equiv^{\mathcal{L}}\mathfrak{A}'$.

Conversely assume that there is some $\alpha\in\mathrm{Tp}^{\mathcal{L}}(\mathfrak{A};k)\setminus\mathrm{Tp}^{\mathcal{L}}(\mathfrak{A}';k)$. Let $\varphi_\alpha(\overline{x})$ be as in (ii) of Lemma 1.33. Then $\mathfrak{A}\models\exists\overline{x}\varphi_\alpha(\overline{x})$ whereas $\mathfrak{A}'\models\neg\exists\overline{x}\varphi_\alpha(\overline{x})$. □

We apply the observations of Lemma 1.33 to the $L^k_{\infty\omega}$, $C^k_{\infty\omega}$, and to their *fragments of bounded quantifier rank*. Note that these are all closed with respect to infinitary conjunctions and disjunctions (as well as under negation) while $L^\omega_{\infty\omega}$ and $C^\omega_{\infty\omega}$ are not!

Definition 1.35. *The* quantifier rank *of a formula in $L_{\infty\omega}$ is given by an ordinal-valued function* qr *that is inductively defined by:*

$$\begin{array}{ll}\mathrm{qr}(\varphi)=0 \quad \textit{for atomic } \varphi, & \mathrm{qr}(\bigwedge\Psi)=\mathrm{qr}(\bigvee\Psi)=\sup\{\mathrm{qr}(\psi)\mid\psi\in\Psi\},\\ \mathrm{qr}(\neg\varphi)=\mathrm{qr}(\varphi), & \mathrm{qr}(\exists x\varphi)=\mathrm{qr}(\forall x\varphi)=\mathrm{qr}(\varphi)+1.\end{array}$$

The quantifier rank of formulae in $C^\omega_{\infty\omega}$ is defined similarly, with the last clause replaced by $\mathrm{qr}(\exists^{\geqslant m}x\varphi)=\mathrm{qr}(\varphi)+1$ *for all m.*

Definition 1.36. *For $\mathcal{L}=L^k_{\infty\omega},L^k_{\omega\omega},C^k_{\infty\omega}$ or $C^k_{\omega\omega}$ and $m\in\omega$ let $\mathcal{L}_{;m}$ denote the fragment defined through restriction to those formulae that have quantifier rank at most m.*

In the absence of counting quantifiers, infinitary logic of finitely bounded quantifier rank collapses to the corresponding fragment of first-order logic as follows.

Lemma 1.37. *Let τ be finite and relational, $m\in\omega$. Any $L^k_{\infty\omega;m}[\tau]$-formula is equivalent with a formula in $L^k_{\omega\omega;m}[\tau]$. There are only finitely many formulae in $L^k_{\omega\omega;m}[\tau]$ up to logical equivalence. It follows that for $\mathcal{L}:=L^k_{\infty\omega;m}$*

also $\mathrm{fin}[\tau] \;/\!\equiv^{\mathcal{L}}$ *and* $\mathrm{fin}[\tau;k] \;/\!\equiv^{\mathcal{L}}$ *are finite. Hence each structure in* $\mathrm{fin}[\tau]$ *is characterized up to* $\equiv^{\mathcal{L}}$ *even by a sentence of* $L^k_{\omega\omega;m}$ *and all* $\mathcal{L}$*-types are isolated by formulae in* $L^k_{\omega\omega;m}$.

Sketch of Proof. The proof is by induction on m. The case of quantifier free formulae is clear. For the induction let $\Psi_m \subseteq L^k_{\omega\omega;m}[\tau]$ be a finite system of representatives for all of $L^k_{\infty\omega;m}[\tau]$. Then up to logical equivalence all formulae of quantifier rank at most $m+1$ are boolean combinations over the finite set $\Psi \cup \big\{\exists x_i\psi \;\big|\; \psi \in \Psi, 1 \leqslant i \leqslant k\big\}$. □

The same claim cannot be made for the $C^k_{\infty\omega;m}$, because there are infinitely many counting quantifiers. Over structures of fixed finite size n, however, only quantifiers $\exists^{\geqslant s}$ for $s \leqslant n$ are non-trivial. This fact is used in the following lemma.

Lemma 1.38. *Let* $\mathcal{L} := C^k_{\infty\omega;m}$, $m \in \omega$, τ *finite and relational. Then each* $\alpha \in \mathrm{Tp}^{\mathcal{L}}(\tau;k)$ *is isolated by a formula* $\varphi_\alpha(x_1,\dots,x_k) \in C^k_{\omega\omega;m}$ *and each* $\mathfrak{A} \in \mathrm{fin}[\tau]$ *is characterized up to* $\equiv^{\mathcal{L}}$ *by a sentence of* $C^k_{\omega\omega;m}$.

Proof. Consider the claim for types. The claim is trivial for $m = 0$; so assume $m \geqslant 1$. Let $\alpha = \mathrm{tp}^{\mathcal{L}}_{\mathfrak{A}}(\overline{a})$, $(\mathfrak{A}, \overline{a}) \in \mathrm{fin}[\tau;k]$ with $|A| = n$. Let $\mathrm{fin}_n[\tau]$ and $\mathrm{fin}_n[\tau;k]$ denote the restrictions of $\mathrm{fin}[\tau]$ and $\mathrm{fin}[\tau;k]$ to structures of size n. It is obvious that in restriction to $\mathrm{fin}_n[\tau]$, each formula in $C^k_{\infty\omega;m}[\tau]$ is equivalent with one that only uses quantifiers $\exists^{\geqslant s}$ for $s \leqslant n$, since any $\exists^{\geqslant s} x_i\varphi$ with $s > n$ is universally false on $\mathrm{fin}_n[\tau]$. An adaptation of the argument in the proof of Lemma 1.37 shows that up to equivalence there are only finitely many formulae in $C^k_{\infty\omega;m}[\tau]$ of this kind, and that these can all be represented up to equivalence by formulae in $C^k_{\omega\omega;m}[\tau]$. Therefore again, $\mathrm{fin}_n[\tau;k] \;/\!\equiv^{\mathcal{L}}$ is finite, and each such type is isolated by a formula of $C^k_{\omega\omega;m}[\tau]$ within $\mathrm{fin}_n[\tau;k] \;/\!\equiv^{\mathcal{L}}$. Let $\psi_\alpha \in C^k_{\omega\omega;m}[\tau]$ isolate α in $\mathrm{fin}_n[\tau;k] \;/\!\equiv^{\mathcal{L}}$. Then $\varphi_\alpha := \exists^{=n} x\, x = x \wedge \psi_\alpha$ is in $C^k_{\omega\omega;m}[\tau]$ and isolates α in $\mathrm{Tp}^{\mathcal{L}}(\tau;k) = \mathrm{fin}[\tau;k] \;/\!\equiv^{\mathcal{L}}$. □

We sum up these observations as follows.

Lemma 1.39. *Let* τ *be a finite relational vocabulary.*

(i) For $\mathcal{L} = C^k_{\infty\omega}$ *or* $L^k_{\infty\omega}$, *each type in* $\mathrm{Tp}^{\mathcal{L}}(\tau;k)$ *is isolated by some formula of* $\mathcal{L}[\tau]$.

(ii) If $\mathcal{L} = C^k_{\infty\omega;m}$ *or* $\mathcal{L} = L^k_{\infty\omega;m}$, $m \in \omega$, *then furthermore each type in* $\mathrm{Tp}^{\mathcal{L}}(\tau;k)$ *is isolated by a formula of* $C^k_{\omega\omega;m}[\tau]$ *or* $L^k_{\omega\omega;m}[\tau]$, *respectively.*

As an immediate consequence of (ii) in the lemma we get the following.

Corollary 1.40. *Let* τ *be finite and relational,* $m \in \omega$. *Then over* $\mathrm{fin}[\tau]$ *and over all* $\mathrm{fin}[\tau;r]$ *for* $r \leqslant k$, $C^k_{\infty\omega;m}$*-equivalence coincides with* $C^k_{\omega\omega;m}$*-equivalence and* $L^k_{\infty\omega;m}$*-equivalence coincides with* $L^k_{\omega\omega;m}$*-equivalence.*

The analysis of the games for $C^k_{\infty\omega}$ and $L^k_{\infty\omega}$ in the next chapter will extend this observation from the bounded quantifier fragments to the logics $C^k_{\infty\omega}$, $C^k_{\omega\omega}$ and $L^k_{\infty\omega}$, $L^k_{\omega\omega}$ themselves. See Corollaries 2.3 and 2.4.

1.5 Interpretations

Interpretations concern the definition of one structure within another. The basic form is that of a *direct interpretation.* An $\mathcal{L}[\tau]$-formula $\varphi(x_1,\dots,x_r)$ in free variables $x_1,\dots,x_r$ defines an r-ary global relation on $\mathrm{fin}[\tau]$: the value of this relation over $\mathfrak{A} \in \mathrm{fin}[\tau]$ is

$$\varphi[\mathfrak{A}] = \Big\{\overline{a} \in A^r \;\Big|\; \mathfrak{A} \models \varphi[\overline{a}]\Big\}.$$

Alternatively, φ may be viewed as defining a structure of vocabulary $\{R\}$, R an r-ary relation symbol, over each $\mathfrak{A} \in \mathrm{fin}[\tau]$, namely the structure $\big(A, \varphi[\mathfrak{A}]\big)$. To obtain defined structures in an arbitrary finite relational vocabulary σ we use tuples of formulae, one for each relation symbol in σ.

Definition 1.41. *Let $\sigma = \big\{R_1,\dots,R_l\big\}$, R_i of arity r_i. An $\mathcal{L}$-definable* (σ,τ)-interpretation *is given by a tuple $\overline{\varphi} = \big(\varphi_i(\overline{x}^{(i)})\big)_{1\leqslant i\leqslant l}$ of formulae in $\mathcal{L}[\tau]$, with $\overline{x}^{(i)} = (x_1,\dots,x_{r_i})$. The σ-structure interpreted by $\overline{\varphi}$ over $\mathfrak{A} \in \mathrm{fin}[\tau]$ is*

$$\big(A, \overline{\varphi}[\mathfrak{A}]\big) := \Big(A, \varphi_1[\mathfrak{A}],\dots,\varphi_l[\mathfrak{A}]\Big).$$

1.5.1 Variants of Interpretations

One may first of all allow other free variables in the φ_i to obtain interpretations with parameters (first- and second-order). This is also true of all the variants considered in the following.

Relativized interpretations. Relativizations serve to restrict the universe of the interpreted structure to a definable subset of the parent structure. Let σ and $\overline{\varphi}$ be as above, $\varphi_0(x)$ an extra τ-formula in one free variable. The σ-structure interpreted by $\overline{\varphi}$ over φ_0 on $\mathfrak{A}$ is defined if $\varphi_0[\mathfrak{A}] \neq \emptyset$. It is the restriction of the above to $\varphi_0[\mathfrak{A}]$:

$$\Big(A, \overline{\varphi}[\mathfrak{A}]\Big) \restriction \varphi_0[\mathfrak{A}] \;=\; \Big(\varphi_0[\mathfrak{A}], \varphi_1[\mathfrak{A}] \cap \big(\varphi_0[\mathfrak{A}]\big)^{r_1},\dots,\varphi_l[\mathfrak{A}] \cap \big(\varphi_0[\mathfrak{A}]\big)^{r_l}\Big).$$

Interpretations in powers. It is often natural to regard not the given universe but some power of it as the domain for the interpreted structure. This leads to the concept of an *interpretation over some power of the universe.* Let σ be as above. Let $\overline{\varphi} = \big(\varphi_i(\overline{x}^{(i)})\big)_{1\leqslant i\leqslant l}$ be a tuple of formulae in $\mathcal{L}[\tau]$, where now φ_i has to be in sr_i distinct free variables. We index these so as to indicate a natural identification of the sr_i-tuple with an r_i-tuple of s-tuples:

$$\overline{x}^{(i)} = \big(x_{(1,1)}, \ldots, x_{(1,s)}, \ldots, x_{(r_i,1)}, \ldots, x_{(r_i,s)}\big).$$

Definition 1.42. *The σ-structure* interpreted by $\overline{\varphi}$ over the s-th power of $\mathfrak{A} \in \text{fin}[\tau]$ *is*

$$\Big(A^s, \overline{\varphi}[\mathfrak{A}]\Big) = \Big(A^s, \varphi_1[\mathfrak{A}], \ldots, \varphi_l[\mathfrak{A}]\Big),$$

where $\varphi_i[\mathfrak{A}] \subseteq A^{sr_i}$ is regarded as an r_i-ary predicate over A^s.

Interpretations in quotients. Another variant that occurs in many natural applications further admits that the universe of the interpreted structure is represented as the quotient with respect to a definable equivalence relation. In order to yield a well defined σ-structure, the given equivalence relation must be compatible with the defined σ-predicates. An equivalence relation $\sim$ is a *congruence* for some r-ary predicate R if the following is satisfied for all $\overline{x} = (x_1, \ldots, x_r)$ and $\overline{x}' = (x'_1, \ldots x'_r)$: $\bigwedge_i x_i \sim x'_i \longrightarrow \big(R\overline{x} \leftrightarrow R\overline{x}'\big)$.

Let σ be as above, $\overline{\varphi}$ in a format appropriate for the direct interpretation of a σ-structure. Let in addition $\psi(x, x')$ be in the format for the interpretation of a binary relation $\sim$.

Definition 1.43. *The σ-structure* interpreted as a quotient with respect to ψ by $\overline{\varphi}$ over $\mathfrak{A} \in \text{fin}[\tau]$ *is defined if the binary relation $\psi[\mathfrak{A}]$ is a congruence with respect to the predicates $\overline{\varphi}[\mathfrak{A}]$. In this case it is the quotient structure* $\big(A, \overline{\varphi}[\mathfrak{A}]\big) \;/\; \psi[\mathfrak{A}]$.

It is instructive to think of the congruence defined by ψ as a definition of the equality relation for the interpreted structure.

Note that these variants are not mutually exclusive. Quite to the contrary all combinations are possible and in fact occur naturally, see Example 1.45 below. One may speak for instance of an interpretation as a quotient over the s-th power of the universe, meaning that what is interpreted in the s-th power is itself the interpretation of a σ-structure as a quotient. The most general notion of interpretation we want to consider is that of a relativized interpretation as a quotient in some power. It subsumes the others as special cases.

Definition 1.44. *A* generalized (σ, τ)-interpretation *is an interpretation of σ-structures as relativizations in a quotient over some s-th power of τ-structures.*

For σ as above such an interpretation is specified by formulae φ_0, $\overline{\varphi}$ and ψ where the φ_i in $\overline{\varphi}$ are of arities sr_i, φ_0 and ψ of arities s and $2s$, respectively. Let $\mathfrak{i} = (\varphi_0; \overline{\varphi}; \psi)$ denote this interpretation itself and $\mathfrak{i}(\mathfrak{A})$ the interpreted structure over $\mathfrak{A}$:

$$\mathfrak{i}(\mathfrak{A}) = \Big(\big(A^s, \overline{\varphi}[\mathfrak{A}]\big) \restriction \varphi_0[\mathfrak{A}] \Big) \Big/ \psi[\mathfrak{A}].$$

$\mathfrak{i}(\mathfrak{A})$ is defined if $\varphi_0[\mathfrak{A}]$ is non-empty and if $\psi[\mathfrak{A}]$ interprets a congruence with respect to the $\overline{\varphi}[\mathfrak{A}] \restriction \varphi_0[\mathfrak{A}]$. It is useful to note that as a (partially defined) mapping

$$\begin{array}{rcl} \mathfrak{i}\colon \mathrm{fin}[\tau] & \longrightarrow & \mathrm{fin}[\sigma] \\ \mathfrak{A} & \longmapsto & \mathfrak{i}(\mathfrak{A}) \end{array}$$

an interpretation $\mathfrak{i}$ is a functor that preserves isomorphism. In particular the interpreted structure must be invariant under all automorphisms of the parent structure.

1.5.2 Examples

Example 1.45. The dual of a symmetric graph is interpretable in first-order logic as the relativization of a quotient in the second power. Let $\mathfrak{G} = (V, E)$ be a symmetric graph. Its dual is the graph $\overline{\mathfrak{G}}$, whose vertices are the edges of $\mathfrak{G}$, with an edge connecting two different ones of these if they share a common vertex of $\mathfrak{G}$. An edge of $\mathfrak{G}$ is an element of $E/\sim$ where two different pairs (v_1, v_2) and (v_1', v_2') in E represent the same edge of $\mathfrak{G}$, if $\{v_1, v_2\} = \{v_1', v_2'\}$. Two different edges represented by (v_1, v_2) and (v_1', v_2') share a common vertex if $\{v_1, v_2\} \cap \{v_1', v_2'\} \neq \emptyset$. Thus the dual of $\mathfrak{G}$ is the quotient

$$\Big(\big(V^2, \varphi_1[\mathfrak{G}]\big) \restriction \varphi_0[\mathfrak{G}] \Big) \Big/ \psi[\mathfrak{G}], \text{ where}$$

$$\begin{aligned} &\varphi_0(v_1, v_2) = Ev_1v_2\,, \\ &\varphi_1(v_1, v_2, v_1', v_2') = \text{“}\{v_1, v_2\} \neq \{v_1', v_2'\}\text{”} \wedge \text{“}\{v_1, v_2\} \cap \{v_1', v_2'\} \neq \emptyset\text{”}, \\ &\psi(v_1, v_2, v_1', v_2') = \text{“}\{v_1, v_2\} = \{v_1', v_2'\}\text{”}. \end{aligned}$$

With $\mathfrak{i} = \big(\varphi_0; \varphi_1; \psi\big)$ this is an interpretation of the dual. Explicit first-order formulae for the expressions in quotes are immediately supplied.

Example 1.46. A pre-ordering is a binary relation $\preccurlyeq$ that is reflexive, transitive, and connex, see Definition 1.62 below. It is easily checked, that $(A, \preccurlyeq)$ satisfies these axioms if and only if (i) – (iii):

(i) $\psi(x, y) = x \preccurlyeq y \wedge y \preccurlyeq x$ interprets an equivalence relation $\sim$ on A.
(ii) $\sim$ is a congruence with respect to $\preccurlyeq$.
(iii) $\varphi(x, y) = x \preccurlyeq y$ interprets a linear ordering in the sense of $\leqslant$ in the quotient of A with respect to $\sim$.

The following example restates the theorem of Immerman and Vardi, Theorem 1.24, in the terminology of interpretations.

Example 1.47. Let σ_1 and σ_2 both contain a binary predicate $<$ for a linear ordering. Recall that $\mathrm{ord}[\sigma]$ stands for the class of all finite σ-structures that are linearly ordered by $<$. Let f be a PTIME functor

$$f\colon \mathrm{ord}[\sigma_1] \longrightarrow \mathrm{ord}[\sigma_2].$$

Then there is a FP-definable (σ_2, σ_1)-interpretation $\mathfrak{i}$, more precisely a relativized interpretation in some power, such that for all sufficiently large $\mathfrak{A} \in \mathrm{ord}[\sigma_1]$:

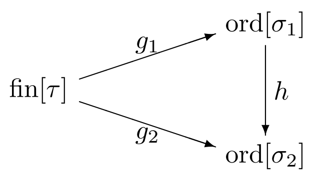
$$f(\mathfrak{A}) \simeq \mathfrak{i}(\mathfrak{A}).$$

The proof is a standard application of the Immerman-Vardi Theorem.

In a situation where g_1 and g_2 are two functors with the same domain and with classes of ordered structures for their ranges

$$g_i\colon \mathrm{fin}[\tau] \longrightarrow \mathrm{ord}[\sigma_i]$$

we shall say that g_2 is FP-*interpretable in terms of* g_1, or that $g_2(\mathfrak{A})$ is *uniformly* FP-*interpretable* over $g_1(\mathfrak{A})$, if there is a (σ_2, σ_1)-interpretation $\mathfrak{i}$ in FP such that $\mathfrak{i}\big(g_1(\mathfrak{A})\big) = g_2(\mathfrak{A})$ for all $\mathfrak{A} \in \mathrm{fin}[\tau]$. By the above this is the case if and only if there is a PTIME computable functor h which makes the following diagram commute:



All these considerations apply analogously to PSPACE computable functors and PFP-interpretability, by the theorem of Abiteboul, Vianu and Vardi, Theorem 1.25.

1.5.3 Interpretations and Definability

Natural logics are often semantically closed with respect to definable properties of definably interpreted structures. For an example think of first-order properties of the dual of a graph. These are first-order definable on the graph itself, since the dual is interpretable over the original graph by first-order means.

Closure under direct interpretations and under relativized interpretations are standard regularity requirements on logics in abstract model theory. They correspond to the *substitution property* and *relativization property*, compare [Ebb85]. Closure properties related to interpretations in quotients are also sometimes considered as an abstract criterion under the name of *congruence closure*.

Let $\mathfrak{i}$ be a (σ, τ)-interpretation, R a global relation on $\text{fin}[\sigma]$. With R we may associate a global relation $\mathfrak{i}(R)$ defined on all those structures $\mathfrak{A} \in \text{fin}[\tau]$ for which $\mathfrak{i}(\mathfrak{A})$ is defined. The value of $\mathfrak{i}(R)$ on $\mathfrak{A}$ is the interpretation over $\mathfrak{A}$ of the value of R on $\mathfrak{i}(\mathfrak{A})$.

Formally, for a generalized interpretation $\mathfrak{i} = (\varphi_0; \overline{\varphi}; \psi)$ in the sense of Definition 1.44 and for $\mathfrak{A}$ such that $\mathfrak{i}(\mathfrak{A})$ is defined, let $\pi : \varphi_0[\mathfrak{A}] \to \varphi_0[\mathfrak{A}]/\psi[\mathfrak{A}]$ be the natural projection with respect to the equivalence relation interpreted by ψ. Then $\mathfrak{i}(R)^{\mathfrak{A}} := \pi^{-1}\big(R^{\mathfrak{i}(\mathfrak{A})}\big)$. We may put $\mathfrak{i}(R)^{\mathfrak{A}} = \emptyset$ for those $\mathfrak{A}$ for which $\mathfrak{i}(\mathfrak{A})$ is not defined. That $\mathfrak{i}(R)^{\mathfrak{A}}$ is well defined as a value for a global relation follows from the fact that $R^{\mathfrak{i}(\mathfrak{A})}$ is closed under isomorphisms of $\mathfrak{i}(\mathfrak{A})$.

Definition 1.48. *Let σ and τ be finite relational. We say that $\mathcal{L}$ is* closed under (σ, τ)-interpretations *(of a certain kind) if the following is satisfied.*

If $\mathfrak{i}$ is an $\mathcal{L}$-definable (σ, τ)-interpretation (of the respective kind) and R is an $\mathcal{L}$-definable global relation on $\text{fin}[\sigma]$*, then* $\mathfrak{i}(R)$ *is $\mathcal{L}$-definable as a global relation over* $\text{fin}[\tau]$.

Consider the boolean case: for any $\mathcal{L}$-definable class Q of σ-structures, the class of those τ-structures for which the interpreted σ-structure is in Q is $\mathcal{L}$-definable itself.

Lemma 1.49. *First-order logic $L_{\omega\omega}$, the infinitary logics $L_{\infty\omega}$,$L^{\omega}_{\infty\omega}$,$C^{\omega}_{\infty\omega}$, and the fixed-point logics* FP, PFP *are each closed under generalized interpretations.*

The proofs by syntactic induction are technically tedious though not difficult at all. In the case of interpretations in some power s one replaces all first-order variables by s-tuples of variables. For atomic expressions involving predicates from σ the corresponding defining formulae are substituted. Equality is replaced by the defined equivalence relation. Quantification translates to higher arity quantification (relativized where necessary) in an obvious manner.

Consider for instance a relativized first-order interpretation, of structures in the vocabulary E of graphs, as a quotient in power 2. Let $\mathfrak{i} = \big(\varphi_0(v_1, v_2); \varphi_1(v_1, v_2, v_1', v_2'); \psi(v_1, v_2, v_1', v_2')\big)$. The graph axiom $\forall x \forall y(Exy \to \neg x = y)$ then translates into

$$\forall v_1 \forall v_2 \forall v_1' \forall v_2' \left(\begin{array}{l} \big(\varphi_0(v_1, v_2) \wedge \varphi_0(v_1', v_2') \wedge \varphi_1(v_1, v_2, v_1', v_2')\big) \\ \to \neg\psi(v_1, v_2, v_1', v_2') \end{array} \right).$$

Second-order variables of arity r are accordingly replaced by second-order variables of arity sr over interpretations in power s. Fixed-point processes are also modelled in correspondingly higher arity in a natural way.

It is obvious that the $L^k_{\infty\omega}$ and $C^k_{\infty\omega}$ on the other hand cannot be robust with respect to interpretations in powers, owing to the bounded supply of variables. This is the only restriction, however, so that the straightforward arguments give the following.

Lemma 1.50. *Let $\mathcal{L}^k = L^k_{\infty\omega}, C^k_{\infty\omega}, L^k_{\omega\omega}$ or $C^k_{\omega\omega}$, respectively. Let accordingly $\mathcal{L}^{sk}$ stand for the respective logic with sk variables instead of k. For any $\mathcal{L}^{sk}$-definable generalized (σ,τ)-interpretation $\mathfrak{i}$ in the s-th power and any $\mathcal{L}^k$-definable global relation R on $\mathrm{fin}[\sigma]$, the global relation $\mathfrak{i}(R)$ on $\mathrm{fin}[\tau]$ — whose value on $\mathfrak{A}$ is the interpretation of $R^{\mathfrak{i}(\mathfrak{A})}$ over $\mathfrak{A}$ — is $\mathcal{L}^{sk}$-definable in restriction to all those $\mathfrak{A} \in \mathrm{fin}[\tau]$ for which $\mathfrak{i}(\mathfrak{A})$ is defined.*[3]

1.6 Lindström Quantifiers and Extensions

Lindström quantifiers provide the means to assert structural properties of definably interpreted structures. Let Q be any isomorphism-closed class of structures of type $\sigma = \{R_1, \ldots, R_l\}$, R_i of arity r_i. With Q we associate a Lindström quantifier of type σ, for which we also write Q. The quantifier Q binds a tuple $\overline{\varphi}$ of formulae apt for a direct interpretation of σ-structures, possibly with parameters. For a logic $\mathcal{L}$ the syntax is extended to allow the construction of a formula

$$\psi := Q\Big(\overline{x}^{(i)}; \varphi_i(\overline{x}^{(i)})\Big)_{i=1\ldots l}$$

from formulae φ_i. Put $\mathrm{free}(\psi) = \bigcup_i (\mathrm{free}(\varphi_i) \setminus \{\overline{x}^{(i)}\})$. The semantics is defined such that

$$\mathfrak{A} \models \psi \quad \text{if} \quad (A, \overline{\varphi}[\mathfrak{A}]) \in Q.$$

Here we have suppressed parameters and assumed that $\mathfrak{A}$ itself interprets the φ_i up to the free variables $\overline{x}^{(i)}$. The "closure" of $\mathcal{L}$ under this new rule of formula formation is denoted $\mathcal{L}(Q)$. As we are only interested in such extensions of first-order and fixed-point logics it suffices to give the following precise definition.

Definition 1.51. *If $\mathcal{Q}$ is a class of Lindström quantifiers, we denote by* $\mathrm{FP}(\mathcal{Q})$ *and* $\mathrm{PFP}(\mathcal{Q})$ *the logics obtained as the simultaneous closure of first-order logic under the respective fixed-point constructor, the usual first-order constructors, and Q-quantification for all quantifiers $Q \in \mathcal{Q}$. $L_{\omega\omega}(\mathcal{Q})$ similarly is obtained from first-order constructors together with Q-quantification for $Q \in \mathcal{Q}$.*

1.6.1 Cardinality Lindström Quantifiers

The class of cardinality Lindström quantifiers is an example of a semantically defined class of quantifiers with natural closure properties. Cardinality Lindström quantifiers express purely numerical relations about the cardinalities of definable predicates.

[3] In non-trivial quotient interpretations more variables may be necessary to express that the interpreted structure is well defined.

Definition 1.52. *Let $S \subseteq \omega^{l+1}$ be a numerical predicate, $\overline{r} = (r_1, \dots, r_l)$ a tuple of arities. Let $\sigma = \{R_1, \dots, R_l\}$, R_i of arity r_i. With S and $\overline{r}$ associate a cardinality Lindström quantifier $Q_{S,\overline{r}}$ of type σ whose defining class is*

$$Q_{S,\overline{r}} = \Big\{(B, R_1, \dots, R_l) \in \mathrm{fin}[\sigma] \;\Big|\; (|B|, |R_1|, \dots, |R_l|) \in S\Big\}.$$

Let $\mathcal{Q}_{\mathrm{card}}$ be the family of all cardinality Lindström quantifiers.

Note that there is no restriction with respect to the number of formulae bound, their arities, or even recursiveness of the underlying numerical relation. Two important cardinality properties which are naturally rendered as Lindström quantifiers are those that express equality of two cardinalities and comparison in the sense of $<$, respectively. For the original sources see [Här65] and [Res62], respectively.

Definition 1.53. *Q_{H} and Q_{R} are the Lindström quantifiers of type $\sigma = \{U_1, U_2\}$, U_1 and U_2 unary, with the following defining classes.*

(i) For the Härtig quantifier*: $Q_{\mathrm{H}} = \Big\{(A, U_1, U_2) \;\Big|\; |U_1| = |U_2|\Big\}$.*
(ii) For the Rescher quantifier*: $Q_{\mathrm{R}} = \Big\{(A, U_1, U_2) \;\Big|\; |U_1| < |U_2|\Big\}$.*

It is natural to extend these quantifiers to higher arities and to introduce for instance a variant of the Härtig quantifier that expresses equicardinality for two definable predicates of arity k. All these natural variants are cardinality quantifiers themselves. A further extension that goes beyond the power of ordinary cardinality Lindström quantifiers replaces the counting of tuples in a relation by the counting of equivalence classes within a relation, relative to a given congruence. Let us call the quantifiers thus obtained *quotient cardinality Lindström quantifiers*. We give an ad-hoc definition here and indicate a more systematic treatment as an aside below.

Definition 1.54. *Let $S \subseteq \omega^{l+1}$, $\overline{r} = (r_1, \dots, r_l)$ a tuple of arities. Let σ consist of r_i-ary relation symbols R_i and $2r_i$-ary relation symbols $\sim_i$ for $i = 1, \dots, l$. With S and $\overline{r}$ associate a cardinality Lindström quantifier $Q^{\sim}_{S,\overline{r}}$ of type σ whose defining class is*

$$\left\{(B, R_1, \dots, R_l, \sim_1, \dots, \sim_l) \;\middle|\; \begin{array}{c} \sim_i \text{ a congruence of } (B^{r_i}, R_i) \text{ and} \\ (|B|, |R_1/\sim_1|, \dots, |R_l/\sim_l|) \in S \end{array}\right\}.$$

$\mathcal{Q}^{\sim}_{\mathrm{card}}$ is the family of all quotient cardinality Lindström quantifiers.

1.6.2 Aside on Uniform Families of Quantifiers

The material presented in this aside will not be used explicitly in the sequel. According to the definitions a Lindström quantifier Q can express the structural property of belonging to the class Q of structures that are *directly* interpreted over the structure at hand. It is often reasonable to make this same property available in application to structures interpreted according to one of the natural variants of interpretations considered above. Formally this can be achieved with derived quantifiers. Suppose for instance that Q is of type $\sigma = \{R_1, \dots, R_l\}$ and that we want to capture the property of belonging to Q for relativized interpreted σ-structures. The derived quantifier that does exactly this is one of type $\sigma \,\dot\cup\, \{U\}$ for a new unary relation symbol U and with defining class

$$Q^{\mathrm{rel}} = \Big\{ (\mathfrak{B}, U) \;\Big|\; \mathfrak{B} \restriction U \in Q \Big\}.$$

The other variants of interpretations are treated similarly. Thus, for example a quantifier that corresponds to Q in interpretations in the s-th power is one of type $\{R_1^{(s)}, \dots, R_l^{(s)}\}$, where $R_i^{(s)}$ is of arity sr_i if r_i is the arity of R_i. Its defining class is

$$Q^{(s)} = \Big\{ (B, R_1^{(s)}, \dots, R_l^{(s)}) \;\Big|\; (B^s, R_1^{(s)}, \dots, R_l^{(s)}) \in Q \Big\}.$$

$Q^{(s)}$ is called the s-th power of Q. The countable set of all quantifiers $Q^{(s)}$ for $s \geqslant 1$ is called the *uniform sequence* generated by Q in [Daw95a]. Let Q^ω stand for this uniform sequence generated by Q and $\mathcal{Q}^\omega$ for the union of the Q^ω for $Q \in \mathcal{Q}$.

To deal with interpretations as quotients we can further pass to type $\sigma \,\dot\cup\, \{\sim\}$ for a new binary relation symbol $\sim$ and consider the quantifier with defining class

$$Q^{\sim} = \big\{ (\mathfrak{B}, \sim) \;\big|\; \sim \text{ a congruence of } \mathfrak{B} \text{ and } \mathfrak{B}/\!\sim\, \in Q \big\}.$$

If Q is any quantifier, let $\overline{Q}$ stand for the class of all quantifiers obtained by translating Q to generalized interpretations so that $\overline{Q}$ consists of all powers of $\big(Q^{\mathrm{rel}}\big)^{\sim}$. Similarly, let for a class $\mathcal{Q}$ of quantifiers $\overline{\mathcal{Q}}$ denote all quantifiers obtained in this manner from quantifiers $Q \in \mathcal{Q}$.

A very weak and fundamental notion of reducibility between quantifiers is that of *quantifier free reducibility.* Q is said to be quantifier free reducible to Q' if Q is quantifier free definable from Q' in the sense that

$$Q = \big\{ \mathfrak{B} \;\big|\; (B, \overline{\varphi}[\mathfrak{B}]) \in Q' \big\}$$

for quantifier free formulae $\overline{\varphi}$ in the vocabulary of Q. Write $Q \lhd Q'$ for this reducibility, and $\mathcal{Q} \lhd \mathcal{Q}'$ if each $Q \in \mathcal{Q}$ is quantifier free reducible to some $Q' \in \mathcal{Q}'$. It is instructive to check that $\lhd$ is preserved in the passage to $\overline{\mathcal{Q}}$: $\mathcal{Q} \lhd \mathcal{Q}'$ implies $\overline{\mathcal{Q}} \lhd \overline{\mathcal{Q}}'$.

The class of all cardinality Lindström quantifiers has nice closure properties. If $Q \in \mathcal{Q}_{\mathrm{card}}$, then Q^{rel} and all $Q^{(s)}$ are quantifier free reducible to $\mathcal{Q}_{\mathrm{card}}$. Consider the relativization $Q^{\mathrm{rel}}_{S,\overline{r}}$ of a cardinality quantifier $Q_{S,\overline{r}}$ to find that

$$Q^{\mathrm{rel}}_{S,\overline{r}} = \Big\{(B, R_1, \dots, R_l, U) \,\Big|\, (B, U, R_1 \cap U, \dots, R_l \cap U) \in Q_{S',\overline{r}'}\Big\}$$

with $\overline{r}' = (1, \overline{r})$ and $S' = \big\{(n, m_0, \overline{m}) \mid (m_0, \overline{m}) \in S\big\}$.

Let $\mathcal{Q}_{\mathrm{mon}}$ be the class of Lindström quantifiers of monadic type . $Q \in \mathcal{Q}_{\mathrm{mon}}$ asserts some property of a tuple of unary predicates. $\mathcal{Q}_{\mathrm{mon}}$ and $\mathcal{Q}_{\mathrm{card}}$ are closely related.

Lemma 1.55. *Any quantifier in $\mathcal{Q}_{\mathrm{mon}}$ is quantifier free reducible to a cardinality quantifier, in fact to one in $\mathcal{Q}_{\mathrm{card}} \cap \mathcal{Q}_{\mathrm{mon}}$. Conversely, any quantifier in $\mathcal{Q}_{\mathrm{card}}$ is quantifier free reducible to some power of a monadic quantifier:*

$$\mathcal{Q}_{\mathrm{mon}} \lhd \mathcal{Q}_{\mathrm{card}} \lhd \mathcal{Q}^{\omega}_{\mathrm{mon}}.$$

Sketch of Proof. Both claims are obvious. For the first claim consider Q of type $\{U_1, \dots, U_l\}$, all U_i unary. Any monadic structure $(B, U_1, \dots, U_l)$ is characterized up to isomorphism by the cardinalities of B and all boolean combinations of the U_i. The corresponding cardinality quantifier is in a type that has one unary predicate for each boolean combination over the U_i. In its numerical predicate collect all tuples of characteristic cardinalities of structures in Q.

For the second claim observe that first of all a cardinality quantifier $Q_{S,\overline{r}}$ is quantifier free reducible to one of homogeneous type $(r, \dots, r)$, r the maximum of the arities in $\overline{r}$. For instance if $\overline{r} = (1, 2)$, then $(B, R_1, R_2) \in Q_{S,(1,2)}$ if $\big(B, \{(x, y) | x = y \wedge R_1 x\}, R_2\big) \in Q_{S,(2,2)}$. Further $Q_{S,(r,\dots,r)} = Q^{(r)}_{S',(1,\dots,1)}$ for $S' = \{(n^r, \overline{m}) | (n, \overline{m}) \in S\}$. □

It is easy to show that the quotient variant of a cardinality quantifier or of a unary quantifier is not in general reducible to a cardinality quantifier.

The above definition of quotient cardinality quantifiers, Definition 1.54, may seem to be more general even than the extension of $\mathcal{Q}_{\mathrm{card}}$ to generalized interpretations, $\overline{\mathcal{Q}}_{\mathrm{card}}$. Up to quantifier free reducibility, however, these classes coincide. We only give a brief sketch of the argument. Note first, that by arguments as in the proof of Lemma 1.55, any quotient cardinality quantifier reduces to one of homogeneous type. In a further reduction process one may also achieve reduction to applications to disjoint predicates R_i. For this the arities are further increased, and some components are used to attach labels consisting of different equality types to the individual predicates. An example would be the passage from R_1 and R_2 to $R_1' = \big\{(x_1, x_2, \overline{x}) \mid x_1 = x_2 \wedge R_1 \overline{x}\big\}$ and $R_2' = \big\{(x_1, x_2, \overline{x}) \mid x_1 \neq x_2 \wedge R_2 \overline{x}\big\}$. In this situation the several $\sim_i$ may be reduced to a single $\sim$ by piecewise definition over the individual R_i'. The combined reduction leads to a quantifier in $\overline{\mathcal{Q}}_{\mathrm{card}}$.

Remark 1.56. *Up to quantifier free reducibility the following classes of quantifiers coincide: $\overline{\mathcal{Q}}_{\mathrm{card}}$, $\mathcal{Q}^{\sim}_{\mathrm{card}}$, and $\overline{\mathcal{Q}}_{\mathrm{mon}}$.*

1.7 Miscellaneous

1.7.1 Canonization and Invariants

Generally the *canonization problem* for an equivalence relation $\sim$ is the problem of assigning unique representatives to each $\sim$-class.

Definition 1.57. *A function $H: X \longrightarrow X$ is called a* canonization *with respect to the equivalence relation $\sim$ over X if it satisfies the following two conditions:*

$$\forall x \quad H(x) \sim x,$$
$$\forall x \, \forall x' \quad x \sim x' \rightarrow H(x) = H(x').$$

Note that the converse implication in the second condition is implied by the first condition, so that H satisfies $x \sim x' \Leftrightarrow H(x) = H(x')$. In particular therefore, a canonization function H for $\sim$ classifies objects in X exactly up to $\sim$. In the sense of the following definition it is a complete invariant for $\sim$ as well.

Definition 1.58. *A function $I: X \longrightarrow S$ is a* complete invariant for $\sim$ *if it satisfies*

$$\forall x \, \forall x' \quad x \sim x' \leftrightarrow I(x) = I(x').$$

The difference between canonizations and complete invariants is that canonizations must map elements to representatives of their class.

Definition 1.59. *Let $I: X \longrightarrow S$ be a complete invariant for $\sim$. A mapping $F: \mathrm{image}(I) \longrightarrow X$ is regarded as an* inverse to I *if $I \circ F$ is the identity on* $\mathrm{image}(I)$, *equivalently if*

$$\forall x \quad F\big(I(x)\big) \sim x.$$

Assume that $I: X \rightarrow S$ and $I': X \rightarrow S'$ are both complete invariants for $\sim$ and that both mappings are surjective. It follows that there is a bijection $\sigma: S \rightarrow S'$ between the ranges such that $I' = \sigma \circ I$. Let now H be a canonization and I any complete invariant for $\sim$. Since H also is a complete invariant, there is a bijection from $\mathrm{image}(I)$ to $\mathrm{image}(H)$ that relates the two. It follows directly from the definition that this bijection is an inverse to I. Conversely, given any complete invariant I and any inverse F to this invariant it is immediate that their composition $F \circ I$ is a canonization. We thus have the following little lemma.

Lemma 1.60. *Given any canonization H and any complete invariant I for $\sim$, there is a uniquely determined inverse F of I such that $H = F \circ I$. If F is an inverse to any complete invariant I for $\sim$ then $H := F \circ I$ is a canonization function with respect to $\sim$.*

It is important to note that under complexity considerations some invariants might be easier to invert than others. It is obvious that the problem of computing some complete invariant reduces to that of computing a canonization function, and that the decision problem for $\sim$ reduces to the computation of any invariant. That the converse reductions are not to be expected for instance as regards PTIME computability and polynomial time Turing reductions in general is shown in [BG84]. General PTIME equivalence of the canonization problem and the problem of the computation of a complete invariant for instance is equivalent with a "shrinking principle" for NP sets that is introduced in [BG84]. Blass and Gurevich also construct an equivalence relation and an oracle relative to which the canonization problem and the problem of computing a complete invariant are not PTIME equivalent. These general results do not have any immediate implications, however, in the case of particular individual equivalence relations.

We shall consider the notions of complete invariants and of canonization for classes of finite relational structures with respect to equivalence relations $\equiv^{\mathcal{L}}$ on these. As any $\equiv^{\mathcal{L}}$ is compatible with isomorphisms, it is clear that canonizations as well as invariants have to be functors that are compatible with isomorphisms.

For computable invariants we also require these to take values in some domain of canonically encoded objects. With respect to computability it is moreover natural to require canonization functors on $\mathrm{fin}[\tau]$ to take canonically encoded structures as their values: the point of canonization is that we get *unique representatives* and not just representatives up to isomorphism.

A complete invariant for $\equiv^{\mathcal{L}}$ on $\mathrm{stan}[\tau]$ or a canonization with respect to $\equiv^{\mathcal{L}}$ on $\mathrm{stan}[\tau]$ immediately extend to corresponding functors on all of $\mathrm{fin}[\tau]$. Algorithmically a computable invariant on $\mathrm{stan}[\tau]$ *is* an invariant on $\mathrm{fin}[\tau]$ and the same is true of computable canonization. This is simply because algorithmic realizations of functors on $\mathrm{fin}[\tau]$ take (encodings of) structures in $\mathrm{stan}[\tau]$ as inputs anyway.

Definition 1.61. *Let $\sim$ be an equivalence relation on* $\mathrm{fin}[\tau]$ *that is invariant under isomorphism.*

(i) A computable complete invariant *for $\sim$ is a computable function I from* $\mathrm{stan}[\tau]$ *to some domain S of standard objects such that*

$$\mathfrak{A} \sim \mathfrak{A}' \iff I(\mathfrak{A}) = I(\mathfrak{A}').$$

(ii) A computable canonization functor *with respect to $\sim$ is a computable function H from* $\mathrm{stan}[\tau]$ *to* $\mathrm{stan}[\tau]$ *such that*

$$H(\mathfrak{A}) \sim \mathfrak{A} \quad \textit{and} \quad \mathfrak{A} \sim \mathfrak{A}' \Rightarrow H(\mathfrak{A}) = H(\mathfrak{A}').$$

Extensions to domains $\mathrm{fin}[\tau; k]$ *are straightforward.*

1.7.2 Orderings and Pre-Orderings

Usually we reserve the binary symbol $<$ for linear orderings. $\leqslant$ then stands for the corresponding weak ordering $x \leqslant y \leftrightarrow x < y \vee x = y$, which we call a *linear ordering in the sense of* $\leqslant$.

Definition 1.62. *A* pre-ordering $\preccurlyeq$ *is a binary relation that is transitive, reflexive and connex:*

$$\forall xyz(x \preccurlyeq y \wedge y \preccurlyeq z \rightarrow x \preccurlyeq z) \;\wedge\; \forall x(x \preccurlyeq x) \;\wedge\; \forall xy(x \preccurlyeq y \vee y \preccurlyeq x).$$

We always write $\prec$ *for the associated strict pre-ordering and* $\sim$ *for the induced equivalence relation:*

$$\begin{aligned} x \prec y \quad &:\leftrightarrow \quad x \preccurlyeq y \wedge \neg y \preccurlyeq x, \\ x \sim y \quad &:\leftrightarrow \quad x \preccurlyeq y \wedge y \preccurlyeq x. \end{aligned}$$

It is readily checked that the axioms for $\preccurlyeq$ are equivalent with the statement "$\sim$ is an equivalence relation and $\preccurlyeq\!\big/\!\sim$ is an ordering in the sense of $\leqslant$". Therefore, pre-orderings exactly are the interpretations of linear orderings as quotients, cf. Example 1.46 above.

Obviously $\prec$ and $\preccurlyeq$ are quantifier free definable in terms of each other. $\sim$ is quantifier free definable form both, but contains strictly less information than $\prec$ and $\preccurlyeq$ (unless $\sim$, $\prec$ and $\preccurlyeq$ are trivial).

1.7.3 Lexicographic Orderings

The standard way to construct new orderings in products from given ones in the factors is by *lexicographic orderings*. It is useful to fix one definite convention regarding these. Consider first the case of the product set $D_1 \times D_2$, where D_i is linearly ordered by $<_i$. We write $<_{\text{lex}}$ for the lexicographic ordering on $D_1 \times D_2$ with *dominant first component*: $(d_1, d_2) <_{\text{lex}} ({d_1}', {d_2}')$ if $d_1 <_1 {d_1}'$ or if $d_1 = d_2$ and $d_2 <_2 {d_2}'$. We employ similar conventions to products with any number of components: entries further to the left always dominate those to the right. The lexicographic ordering is always understood if we are dealing with multiply indexed objects. For instance matrices $(d_{ij}) = (d_{ij})_{1 \leqslant i \leqslant t, 1 \leqslant j \leqslant s}$ are interpreted as tuples where the ordering of the components is the lexicographic one on $\{1, \ldots, t\} \times \{1, \ldots, s\}$. If the entries d_{ij} themselves are from an ordered domain $(D, <)$ we further obtain the lexicographic ordering on $\big\{(d_{ij}) \;\big|\; d_{ij} \in D \text{ for } 1 \leqslant i \leqslant t, 1 \leqslant j \leqslant s\big\}$ according to $(d_{ij}) <_{\text{lex}} ({d_{ij}}')$ if $d_{ij} < {d_{ij}}'$ for the least index pair (i, j) such that $d_{ij} \neq {d_{ij}}'$.

Note that lexicographic orderings are always first-order definable from the constituent orderings in the components and the ordering of the components.

2. The Games and Their Analysis

This chapter serves to review the Ehrenfeucht-Fraïssé style analysis of the logics $L^k_{\infty\omega}$ and $C^k_{\infty\omega}$ by means of the corresponding *pebble games*. Emphasis is on the games and their algebraic analysis rather than on the more syntactic descriptions in terms of Hintikka formulae and Scott sentences. The main result of this algebraic analysis is a definable *ordering with respect to types*. We obtain ordered representations of the quotients $\mathrm{Tp}^{\mathcal{L}}(\mathfrak{A};k) = A^k/\equiv^{\mathcal{L}}$ for $\mathcal{L} = L^k_{\infty\omega}$ or $C^k_{\infty\omega}$ on finite relational structures $\mathfrak{A}$.

- Section 2.1 contains the definition of the games and the statement and proofs of the corresponding Ehrenfeucht-Fraïssé theorems which here are due to Barwise [Bar77], Immerman [Imm82], and Immerman and Lander [IL90], respectively. We present some typical examples that apply the game characterizations to derive non-expressibility results. Most notably a construction due to Cai, Fürer and Immerman proves that the logics $C^k_{\infty\omega}$ form a strict hierarchy with respect to k.

  A refined analysis of the games shows that $\equiv^{C^k_{\infty\omega}}$ and $\equiv^{C^k_{\omega\omega}}$, and similarly $\equiv^{L^k_{\infty\omega}}$ and $\equiv^{L^k_{\omega\omega}}$, coincide in restriction to finite structures.

- In Section 2.2 we review the colour refinement technique for graphs and discuss some variants and their definability properties.

- Ideas related to the colour refinement are employed in Section 2.3 to introduce the ordered quotients with respect to $C^k_{\infty\omega}$- or $L^k_{\infty\omega}$-types through a fixed-point process for the classification of game positions.

2.1 The Pebble Games for $L^k_{\infty\omega}$ and $C^k_{\infty\omega}$

The setting for the games is the usual one for comparison games. There are two players denoted **I** and **II** for *first* and *second player*. The game is played on a pair of finite structures $\mathfrak{A}$ and $\mathfrak{A}'$ of the same finite relational vocabulary τ. In the *k-pebble game* there are k marked pebbles for each of the two structures. Let both sets of pebbles be numbered $1, \ldots, k$. A *stage* of the game, or an instantaneous description of a game situation, is determined by a placement of the pebbles on elements of the corresponding structures.

Formally a stage is given by a tuple $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$, with $\overline{a} \in A^k$ and $\overline{a}' \in A'^k$ denoting the current positions of the pebbles. A *position* describes a pebble placement over one of the structures. The position over $\mathfrak{A}$ for instance in stage $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$ is $(\mathfrak{A}, \overline{a})$. Formally a position is an element of $\mathrm{fin}[\tau; k]$: a structure with a designated k-tuple of elements. A stage in the game is a pair of positions, or an element of $\mathrm{fin}[\tau; k] \times \mathrm{fin}[\tau; k]$.

In each *round* of the game exactly one pair of corresponding pebbles is repositioned in the respective structures. This repositioning is governed by an exchange of moves between the two players. The game for L^k and that for C^k differ with respect to the rules for this exchange.

The single round in the L^k-game.

I chooses a pebble index $j \in \{1, \ldots, k\}$ and moves the corresponding pebble in one of the structures to an arbitrary element of that structure, for instance to $b \in A$.

II responds by moving the corresponding pebble over the opposite structure to an arbitrary element of that structure, here to some $b' \in A'$.

If this exchange is carried out in stage $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$ then the resulting stage after this round is $(\mathfrak{A}, \overline{a}\frac{b}{j}; \mathfrak{A}', \overline{a}'\frac{b'}{j})$. We write $\overline{a}\frac{b}{j}$ for the tuple $\overline{a}$ with j-th component replaced by b.

The single round in the C^k-game.

I chooses a pebble index $j \in \{1, \ldots, k\}$ and a subset of the universe of one of the structures, say $B \subseteq A$.

II must choose a subset of exactly the same size in the opposite structure, here some $B' \subseteq A'$ with $|B'| = |B|$.

I now places the j-th pebble within the subset designated by **II**, here on some $b' \in B'$.

II responds by moving the corresponding pebble over the opposite structure to any element within the subset designated by **I**, here to some $b \in B$.

If this exchange is carried out in stage $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$ then the resulting stage is $(\mathfrak{A}, \overline{a}\frac{b}{j}; \mathfrak{A}', \overline{a}'\frac{b'}{j})$.

In both cases the game may continue as long as player **II** can maintain the following condition:

(W) The mapping associating the pebbled elements in $\mathfrak{A}$ with those in $\mathfrak{A}'$ must be a partial isomorphism, i.e. $\mathrm{atp}_{\mathfrak{A}}(\overline{a}) = \mathrm{atp}_{\mathfrak{A}'}(\overline{a}')$ for the current positions $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{A}', \overline{a}')$.

I wins the game as soon as **II** violates this condition, and also if **II** cannot move according to the rules as may happen in the C^k-game owing to different sizes of the two structures.

Player **II** has a *winning strategy in the infinite game on* $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$ if **II** has a strategy to maintain condition (W) indefinitely in the game starting from stage $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$. Similarly we say that **II** has a *winning strategy* for i rounds in the game on $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$ if (W) can be maintained by **II** for at least i rounds starting from $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$. More formal characterizations are developed in an inductive fashion below.

Intuitively the ability of player **II** to respond to challenges of **I** is a measure for the similarity of the underlying positions. In each individual round **II** must preserve atomic indistinguishability of the resulting positions (W), otherwise the game is lost. The ability to maintain (W) for longer sequences of rounds and in response to any manoeuvres of **I** requires a higher degree of similarity of the initial positions. The point of the above rules for single rounds is that they make the games adequate for $L^k_{\infty\omega}$ and $C^k_{\infty\omega}$, respectively. The following two important theorems state that the degree of indistinguishability corresponding to the existence of a strategy precisely is equality of types in the respective logic.

Theorem 2.1 (Barwise, Immerman). *Let $\mathfrak{A}$ and $\mathfrak{A}'$ be finite structures of the same finite relational vocabulary. Player* **II** *has a winning strategy in the infinite L^k-game on $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$ if and only if the the positions $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{A}', \overline{a}')$ cannot be distinguished in $L^k_{\infty\omega}$, i.e. if $(\mathfrak{A}, \overline{a}) \equiv^{L^k_{\infty\omega}} (\mathfrak{A}', \overline{a}')$.*

Theorem 2.2 (Immerman, Lander). *Let $\mathfrak{A}$ and $\mathfrak{A}'$ be finite structures of the same finite relational vocabulary. Player* **II** *has a winning strategy in the infinite C^k-game on $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$ if and only if the the positions $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{A}', \overline{a}')$ cannot be distinguished in $C^k_{\infty\omega}$, i.e. if $(\mathfrak{A}, \overline{a}) \equiv^{C^k_{\infty\omega}} (\mathfrak{A}', \overline{a}')$.*

From the analysis of the games it will further follow that the conditions in Theorems 2.1 and 2.2 are also equivalent with indistinguishability in the *finitary* logics $L^k_{\omega\omega}$ and $C^k_{\omega\omega}$.

Corollary 2.3. *Let τ be finite and relational. The following are equivalent for all $(\mathfrak{A}, \overline{a}), (\mathfrak{A}', \overline{a}') \in \mathrm{fin}[\tau; k]$:*

(i) Player **II** *has a strategy in the infinite L^k-game on $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$.*
(ii) $(\mathfrak{A}, \overline{a}) \equiv^{L^k_{\infty\omega}} (\mathfrak{A}', \overline{a}')$, i.e. $\mathrm{tp}^{L^k_{\infty\omega}}_{\mathfrak{A}}(\overline{a}) = \mathrm{tp}^{L^k_{\infty\omega}}_{\mathfrak{A}'}(\overline{a}')$.
(iii) $(\mathfrak{A}, \overline{a}) \equiv^{L^k_{\omega\omega}} (\mathfrak{A}', \overline{a}')$, i.e. $\mathrm{tp}^{L^k_{\omega\omega}}_{\mathfrak{A}}(\overline{a}) = \mathrm{tp}^{L^k_{\omega\omega}}_{\mathfrak{A}'}(\overline{a}')$.

In particular any $L^k_{\infty\omega}$-type over $\mathrm{fin}[\tau]$ is fully determined by its $L^k_{\omega\omega}$-part.

Corollary 2.4. *Let τ be finite and relational. The following are equivalent for all $(\mathfrak{A}, \overline{a}), (\mathfrak{A}', \overline{a}') \in \mathrm{fin}[\tau; k]$:*

(i) Player **II** *has a strategy in the infinite C^k-game on $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$.*
(ii) $(\mathfrak{A}, \overline{a}) \equiv^{C^k_{\infty\omega}} (\mathfrak{A}', \overline{a}')$, i.e. $\mathrm{tp}^{C^k_{\infty\omega}}_{\mathfrak{A}}(\overline{a}) = \mathrm{tp}^{C^k_{\infty\omega}}_{\mathfrak{A}'}(\overline{a}')$.
(iii) $(\mathfrak{A}, \overline{a}) \equiv^{C^k_{\omega\omega}} (\mathfrak{A}', \overline{a}')$, i.e. $\mathrm{tp}^{C^k_{\omega\omega}}_{\mathfrak{A}}(\overline{a}) = \mathrm{tp}^{C^k_{\omega\omega}}_{\mathfrak{A}'}(\overline{a}')$.

Each $C^k_{\infty\omega}$-type is fully determined by its $C^k_{\omega\omega}$-part over $\mathrm{fin}[\tau]$.

With these equivalences proved, we shall simply speak of the L^k-type and C^k-type, and write for instance $\mathrm{tp}_{\mathfrak{A}}^{L^k}(\overline{a})$ and $\mathrm{tp}_{\mathfrak{A}}^{C^k}(\overline{a})$ for these; and also $\equiv^{L^k}$ and $\equiv^{C^k}$ for the corresponding notions of L^k- and C^k-equivalence.

The following section is devoted to applications of the game characterizations. In the consecutive sections we shall then present a detailed theoretical treatment for the case of the C^k-game. In Section 2.1.2 a direct and straightforward proof of Theorem 2.2 is presented. Section 2.1.3 presents a deeper analysis of the C^k-game, proving among other things Corollary 2.4. The analogous treatment for L^k is easily obtained along the same lines through obvious simplifications; this is summed up in Section 2.1.4.

2.1.1 Examples and Applications

We present examples that employ Theorems 2.1 and 2.2 to show inexpressibility in $L^k_{\infty\omega}$ or $C^k_{\infty\omega}$.

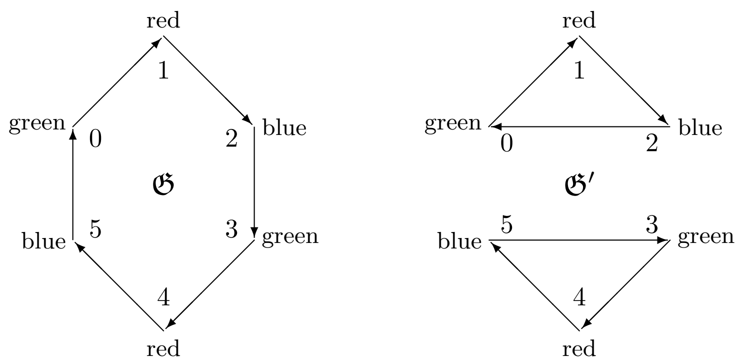
Example 2.5. As a trivial application of the L^k-game we find the following. Any two k-tuples $\overline{a}$ and $\overline{a}'$ over two plain sets A and A' of size at least k are $L^k_{\infty\omega}$-equivalent if and only if they have the same equality type: $\mathrm{eq}(\overline{a}) = \mathrm{eq}(\overline{a}') \Rightarrow \mathrm{tp}_A^{L^k}(\overline{a}) = \mathrm{tp}_{A'}^{L^k}(\overline{a}')$ if $|A|, |A'| \geqslant k$. It follows that $L^k_{\infty\omega}$ cannot distinguish between any two plain sets that have at least k elements. In particular the $L^k_{\infty\omega}$ form a strict hierarchy in expressiveness: $L^1_{\infty\omega} \subsetneq L^2_{\infty\omega} \subsetneq \cdots \subseteq L^\omega_{\infty\omega}$. The same applies to the corresponding fragments of first-order logic: $L^1_{\omega\omega} \subsetneq L^2_{\omega\omega} \subsetneq \cdots \subseteq L_{\omega\omega}$.

The following simple and elegant example is taken from [IL90].

Example 2.6 (Immerman, Lander). Consider the following two coloured directed graphs with six nodes each. $\mathfrak{G} = \big(\{0, \ldots, 5\}, E, U_r, U_b, U_g\big)$. The colours are interpreted $U_g = \{0, 3\}$ for *green*, $U_r = \{1, 4\}$ for *red* and $U_b = \{2, 5\}$ for *blue*. The edge relation E of $\mathfrak{G}$ connects the nodes $0, \ldots, 5$ in cyclic fashion. $\mathfrak{G}'$ is the same as $\mathfrak{G}$ as far as its universe and the colours are concerned. With respect to its edge relation E', however, $\mathfrak{G}'$ splits into two disjoint cycles $0, 1, 2$ and $3, 4, 5$ respectively. Compare the sketches in Figure 2.1. Note that these two graphs realize exactly the same atomic 2-types, $\mathrm{Atp}(\mathfrak{G}; 2) = \mathrm{Atp}(\mathfrak{G}'; 2)$. Furthermore we observe that each of these atomic 2-types is realized exactly twice in each structure.

We claim that $\mathfrak{G}$ and $\mathfrak{G}'$ are indistinguishable in $C^2_{\infty\omega}$. In this special case it can be shown that player **II** actually has a strategy to maintain atomic equivalence of positions. By Theorem 2.2 this implies that (a_1, a_2) from $\mathfrak{G}$ and (a_1', a_2') from $\mathfrak{G}'$ are C^2-equivalent if they satisfy the same atomic type. $\mathfrak{G} \equiv^{C^2} \mathfrak{G}'$ follows by Lemma 1.34 since $\mathrm{Atp}(\mathfrak{G}; 2) = \mathrm{Atp}(\mathfrak{G}'; 2)$ now implies $\mathrm{Tp}^{C^2}(\mathfrak{G}; 2) = \mathrm{Tp}^{C^2}(\mathfrak{G}'; 2)$. Before exhibiting a strategy for maintaining atomic equivalence, let us state the following consequences.

Fig. 2.1



(i) The transitive closure of a binary relation is not definable in $C^2_{\infty\omega}$. If the transitive closure of the binary relation E were definable by some formula $\varphi(x, y)$ of $C^2_{\infty\omega}[E]$ then the $C^2_{\infty\omega}[E]$-sentence $\chi := \forall x \forall y \varphi(x, y)$ would distinguish $\mathfrak{G}$ from $\mathfrak{G}'$.

(ii) Transitivity of a binary relation is not $C^2_{\infty\omega}$-definable and the class of all equivalence relations is not $C^2_{\infty\omega}$-definable. C^2-equivalence of $\mathfrak{G}$ and $\mathfrak{G}'$ directly implies C^2-equivalence also of those structures obtained from $\mathfrak{G}$ and $\mathfrak{G}'$ by removing the colours and replacing the edge relation E by its reflexive and symmetric closure, which is atomically definable from E. From $\mathfrak{G}'$ we thereby obtain an equivalence relation, not from $\mathfrak{G}$. Note that transitivity and the class of equivalence relations are first-order definable with 3 variables.

Let us return to the claim that **II** can maintain atomic equivalence. A strategy for player **II** is extracted from the following observation. Let $a \in \mathfrak{G}$ and $a' \in \mathfrak{G}'$ be of the same colour. Then there is a unique bijection π from $\mathfrak{G}$ to $\mathfrak{G}'$ that maps a to a' and preserves colours as well as edges that are incident with a or a'. This is checked directly; if without loss of generality we consider the case $a = a'$, then the identical mapping on $\{0, \ldots, 5\}$ is as desired.

Suppose now that in the current stage $(\mathfrak{G}, a_1, a_2; \mathfrak{G}', a'_1, a'_2)$ of the game $\text{atp}_{\mathfrak{G}}(a_1, a_2) = \text{atp}_{\mathfrak{G}'}(a'_1, a'_2)$. We want to show that **II** can defend this property against any challenge by player **I**. Assume without loss of generality that player **I** chooses to play with the second pebble. Let π be chosen with respect to a_1 and a'_1 as above. Let then **II** play according to π: if for instance **I** proposes $B \subseteq \{0, \ldots, 5\}$ as a subset of $\mathfrak{G}$ then **II** responds with $B' = \pi(B)$ and upon any choice for $b' \in B'$ by **I** player **II** may answer with $\pi^{-1}(b') \in B$. The defining condition on π guarantees that $\text{atp}_{\mathfrak{G}}(a_1, b) = \text{atp}_{\mathfrak{G}'}(a'_1, b')$.

The next example gives an account of the essential features of the construction by Cai, Fürer and Immerman of non-isomorphic but $C^k_{\infty\omega}$-equivalent finite graphs [CFI89]. We shall later also apply the result of these considerations — Theorem 2.9 below — to show that the counting extension of fixed-point logic does not capture PTIME. See Corollary 4.23 of Chapter 4.

The construction uses certain highly symmetric graphs with a parity-sensitive automorphism group. These "gadgets" were first employed by Immerman in [Imm81] to prove lower bounds on the number of variables needed for expressing certain reachability properties in graphs (without counting quantifiers).

Example 2.7 (Immerman and Cai, Fürer, Immerman). Main building blocks for the construction are the following gadgets. Fix some $m \geqslant 2$. Let $\mathcal{P}(m)$ denote the power set of the set $m = \{0, \dots, m-1\}$. We identify $\mathcal{P}(m)$ with the set of functions $s : m \to \{0,1\}$. Let $\mathfrak{H}$ be the following undirected graph with node set $H = I \dot\cup O$ where $I = \mathcal{P}(m)$, $O = m \times \{0,1\}$. The names I and O stand for *inner* and *outer nodes*, respectively. The edge relation of $\mathfrak{H}$ encodes the rôle of the inner nodes as subsets over m: $s \in I = \mathcal{P}(m)$ is joined exactly with all pairs $(u, s(u)) \in O$ for $u \in m$. For each $u \in m$ we refer to the two nodes $(u,0), (u,1)$ as a *pair of corresponding outer nodes*. The outer nodes of $\mathfrak{H}$ will serve as ports for gluing several copies of $\mathfrak{H}$ together. The crucial properties of the resulting graphs exploit the behaviour under automorphisms of $\mathfrak{H}$ that exchange pairs of corresponding outer nodes. Each $t \subseteq m$ induces an automorphism γ_t of $\mathfrak{H}$ that is determined by its behaviour on outer nodes

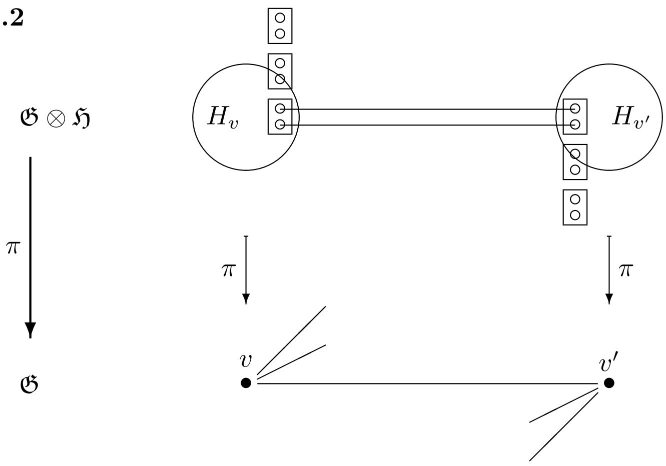
$$\gamma_t : (u, i) \longmapsto \big(u, i \oplus t(u)\big)$$

where $\oplus$ is addition modulo 2. Note that γ_t preserves the set of inner nodes and also each pair of corresponding outer nodes set-wise. On the outer nodes it swaps exactly those pairs of corresponding outer nodes $(u,0), (u,1)$ for which $u \in t$. Inner nodes are mapped according to $s \mapsto s \oplus t$ where $\oplus$ applied to the functions s and t is pointwise addition modulo 2.

We now split the set I of inner nodes into two disjoint subsets $I^i := \big\{ s \subseteq m \;\big|\; |s| \equiv i \bmod 2 \big\}$, for $i = 0,1$. Note that γ_t preserves the subsets I^i if and only if $|t|$ is even. For odd $|t|$ on the other hand γ_t induces a bijection between I^0 and I^1.

Let $\mathfrak{G} = \big(V, E, \leqslant\big)$ be any symmetric connected graph that is regular of degree m and linearly ordered by $\leqslant$. Let $\mathfrak{G} \otimes \mathfrak{H}$ be the result of substituting a copy of $\mathfrak{H}$ for each node of $\mathfrak{G}$ and joining outer nodes by a pair of edges in the natural fashion. In detail let $\mathfrak{G} \otimes \mathfrak{H} = \big(\widehat{V}, \widehat{E}, \preccurlyeq\big)$. $\widehat{V} = V \times H$ and $\preccurlyeq$ is the pre-ordering induced by $\leqslant$ on this product. $\widehat{E}$ consists of all edges from the respective copies of $\mathfrak{H}$ together with the following new links between outer nodes. If $(v, v') \in E$ with v' being the u-th neighbour of v in $\mathfrak{G}$ and v being the u'-th neighbour of v' (with respect to $\leqslant$) we include edges between $\big(v, (u,0)\big)$ and $\big(v', (u',0)\big)$ as well as between $\big(v, (u,1)\big)$ and $\big(v', (u',1)\big)$. We refer to these extra edges as *connecting edges*. Each edge of $\mathfrak{G}$ thus gets replaced by a pair of connecting edges. This is sketched in Figure 2.2. We denote by $\pi\colon \mathfrak{G} \otimes \mathfrak{H} \to \mathfrak{G}$ the natural projection to the first factor. Let $H_v := \pi^{-1}(v)$ denote the subset of nodes of $\mathfrak{G} \otimes \mathfrak{H}$ that belong to that copy of $\mathfrak{H}$ that is substituted for v.

Fig. 2.2



Let $I^i_v \subseteq H_v$ denote the respective subsets of the set of inner nodes within H_v, $i = 0, 1$. Consider automorphisms of $\mathfrak{G}\otimes\mathfrak{H}$ with respect to their behaviour on the sets I^i_v. If $v_0, \ldots, v_l$ is a simple path in $\mathfrak{G}$ then there is an automorphism γ of $\mathfrak{G} \otimes \mathfrak{H}$ with the following properties: γ fixes all H_v for $v \neq v_0, \ldots, v_l$ pointwise, γ preserves the subsets $I^i_{v_j}$ for $j = 1, \ldots, l-1$ and exchanges $I^0_{v_j}$ with $I^1_{v_j}$ for $j = 0, l$. Such γ is pieced together from automorphisms γ_t of the individual embedded $\mathfrak{H}$. For the copy of $\mathfrak{H}$ over v_j choose t to be the subset of m that contains u if the given path connects v_j to its u-th neighbour in $\mathfrak{G}$. Thus $|t|$ is even for all inner nodes of the path and odd for the end points of the path.

For $U \subseteq V$ let $(\mathfrak{G}\otimes\mathfrak{H})_U$ be the subgraph of $\mathfrak{G}\otimes\mathfrak{H}$ that results from deleting all inner nodes in I^0_v for $v \in U$ and those in I^1_v for $v \notin U$. Since $\mathfrak{G}$ is connected, it follows from the above automorphism argument that all the $(\mathfrak{G} \otimes \mathfrak{H})_U$ fall into at most two classes up to isomorphisms. If the symmetric difference between U_1 and U_2 is even, then $(\mathfrak{G} \otimes \mathfrak{H})_{U_1} \simeq (\mathfrak{G} \otimes \mathfrak{H})_{U_2}$. We claim that otherwise indeed $(\mathfrak{G} \otimes \mathfrak{H})_{U_1}$ and $(\mathfrak{G} \otimes \mathfrak{H})_{U_2}$ are non-isomorphic. This can be seen by means of the following numerical invariant on the $(\mathfrak{G}\otimes\mathfrak{H})_U$. Suppose a given graph is isomorphic to some $(\mathfrak{G}\otimes\mathfrak{H})_U$. Note that the projection π to $\mathfrak{G}$ and in particular therefore the node sets $\pi^{-1}(v)$, the sets of inner nodes in $\pi^{-1}(v)$, and the pairs of connecting edges between outer nodes of different copies of $\mathfrak{H}$ are well defined in terms of the given graph. Let $S \subseteq \widehat{E}$ be any set of edges that contains exactly one member from each pair of connecting edges and let N be any set of inner nodes that contains exactly one member from each $\pi^{-1}(v)$. Call a connecting edge incident with an inner node if there is an edge that joins that node with one of the end-points of the given edge. Let i be the result of counting modulo 2 the number of edges in S that are incident with N. We check that i is independent of the choices made. Replacing any edge in S by its partner edge changes the incidence with N in exactly two places. Replacing an inner node of $\pi^{-1}(v)$ by another one changes incidence with S in an even number of places, since either both nodes are in I_v^0 or both are in I_v^1. It is immediate, however, that $i = 0$ on $(\mathfrak{G} \otimes \mathfrak{H})_\emptyset$ and $i = 1$ on $(\mathfrak{G} \otimes \mathfrak{H})_{\{v\}}$ for any single node v.

For definite representatives of the two isomorphism types put $(\mathfrak{G}\otimes\mathfrak{H})^0 := (\mathfrak{G} \otimes \mathfrak{H})_\emptyset$ and $(\mathfrak{G} \otimes \mathfrak{H})^1 := (\mathfrak{G} \otimes \mathfrak{H})_{\{v_0\}}$ where v_0 is the $\leqslant$-least node of $\mathfrak{G}$. We use such representatives in the simple case that $\mathfrak{G}$ is a complete graph to obtain the desired separation result. Let $\mathfrak{K}_{m+1}$ be the ordered complete graph over $m+1$ nodes:

$$\mathfrak{K}_{m+1} = \Big(\{1, \ldots, m+1\}, \{(k,l) | k \neq l\}, \leqslant\Big).$$

Denote the above graph $\mathfrak{H}$ with node set $\mathcal{P}(m) \dot\cup m \times \{0,1\}$ by $\mathfrak{H}_m$ to indicate the dependence on m.

Lemma 2.8. *Let* $\mathfrak{A} = (\mathfrak{K}_{m+1} \otimes \mathfrak{H}_m)^0$ *and* $\mathfrak{A}' = (\mathfrak{K}_{m+1} \otimes \mathfrak{H}_m)^1$. *Then for* $m \geqslant 2$*:*

$$\mathfrak{A} \equiv^{C^m} \mathfrak{A}' \quad \textit{but} \quad \mathfrak{A} \not\equiv^{L^{m+1}} \mathfrak{A}'.$$

Proof. It is instructive to consider first the case $m = 2$. An inspection of the construction in this simple case shows that $\mathfrak{A}$ is the disjoint union of two cycles of length 9, each grouped into three groups of 3 consecutive vertices that belong to the same class of the pre-ordering. $\mathfrak{A}'$ is a single cycle of length 18 with a corresponding grouping into 6 blocks of three vertices each. If we replace the classes of the pre-ordering by three monadic predicates U_r, U_b and U_g for colours red, blue and green as in Example 2.6 then the relation between $\mathfrak{A}$ and $\mathfrak{A}'$ is the same as between the graphs $\mathfrak{G}$ and $\mathfrak{G}'$ in Example 2.6, only each node of the graphs there is replaced by a path of length 3 to obtain the present ones. The claim for $m = 2$ therefore essentially follows from the considerations in Example 2.6.

We turn to the general case. Let the natural projections from $\mathfrak{A}$ and $\mathfrak{A}'$ to $\mathfrak{K}_{m+1}$ be denoted π and π', respectively. Note that membership in $\pi^{-1}(j)$ (respectively $\pi'^{-1}(j)$) is definable in $L^2_{\omega\omega}$, since $\pi^{-1}(j)$ consists of the j-th class with respect to $\preccurlyeq$. Concrete formulae are obtained exactly as in Example 1.9. It follows that in order not to lose, player **II** must necessarily respect π and π' as well as the properties of being an inner node in $\pi^{-1}(j)$ or of being an end point of a connecting edge between $\pi^{-1}(i)$ and $\pi^{-1}(j)$ for any $1 \leqslant i, j \leqslant m+1$. This is true for both the C^k- and the L^k-games.

We first employ the L^{m+1}-game to show that $\mathfrak{A}$ and $\mathfrak{A}'$ are not L^{m+1}-equivalent. By the above considerations, player **I** can force **II** into positions such that the j-th pebbles are placed on inner nodes $a_j \in \pi^{-1}(j)$ and $a'_j \in \pi'^{-1}(j)$ for $1 \leqslant j \leqslant m+1$. For each $j \neq 1$ consider the pair of corresponding outer nodes in $\pi^{-1}(1)$ in $\mathfrak{A}$ that belong to connecting edges between $\pi^{-1}(1)$ and $\pi^{-1}(j)$. Note that exactly one node of this pair has distance 2 from a_j, the other one has distance greater than 2. Let v_j be the one with distance 2. By the construction of $\mathfrak{A}$ it is clear that the number of v_j that are direct neighbours to a_1 is even. Choosing nodes v'_j for $2 \leqslant j \leqslant m+1$ in $\mathfrak{A}'$ in the same manner, we find that the number of v'_j that are direct neighbours to a'_1 must be odd. There is therefore at least one index $j \geqslant 2$ such that v_j is a neighbour of a_1 while v'_j is not a neighbour of a'_1 or vice versa. Assume without loss of generality the former is true of $j = 2$. Let player **I** move pebble 3 in $\mathfrak{A}$ to v_2. **II** must move pebble 3 to a neighbour of a'_1 in $\mathfrak{A}'$ in order not to lose immediately. If **II** places this pebble not on one of the outer nodes in $\pi^{-1}(1)$ belonging to a connecting edge to $\pi^{-1}(2)$ then **II** loses within one more round. Choosing the one of these outer nodes that is a neighbour of a'_1 and therefore different from v'_2 **II** still loses in one more round, since now pebbles 2 and 3 are placed at distance 2 in $\mathfrak{A}$ and at distance greater 2 in $\mathfrak{A}'$.

It remains to exhibit a strategy for player **II** in the C^m-game on $\mathfrak{A}$ and $\mathfrak{A}'$. We show that **II** can maintain the following condition on the stages $\big(\mathfrak{A}, a_1, \ldots, a_m; \mathfrak{A}', a'_1, \ldots, a'_m\big)$:

$$(*) \qquad \begin{array}{l} \pi(\overline{a}) = \pi'(\overline{a}') \quad \text{and} \\ \Big(\mathfrak{A} \restriction \pi^{-1}\big(\pi(\overline{a})\big), \overline{a}\Big) \simeq \Big(\mathfrak{A}' \restriction \pi'^{-1}\big(\pi'(\overline{a}')\big), \overline{a}'\Big). \end{array}$$

We argue that this suffices for $\mathfrak{A} \equiv^{C^m} \mathfrak{A}'$. In any game position $(\mathfrak{A}, \overline{a})$ at least one $\pi^{-1}(j)$ remains unpebbled. Consider a position $\overline{a}$ over $\mathfrak{A}$ in which $\pi^{-1}(1)$ is unpebbled. By construction the identity mapping is an isomorphism between the induced subgraphs of $\mathfrak{A}$ and $\mathfrak{A}'$ on $\pi^{-1}\big(\{2, \ldots, m+1\}$:

$$\mathfrak{A} \restriction \pi^{-1}\big(\{2, \ldots, m+1\}\big) = \mathfrak{A}' \restriction \pi'^{-1}\big(\{2, \ldots, m+1\}\big).$$

Thus $(*)$ is seen to hold of $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a})$ if $\overline{a}$ is disjoint from $\pi^{-1}(1)$. In the general case there still is an isomorphism between $\mathfrak{A} \restriction \big(A \setminus \pi^{-1}(j)\big)$ and $\mathfrak{A}' \restriction \big(A' \setminus \pi'^{-1}(j)\big)$ for any j, because $\mathfrak{A}' = (\mathfrak{K}_{m+1} \otimes \mathfrak{H}_m)^1 = (\mathfrak{K}_{m+1} \otimes \mathfrak{H}_m)_{\{1\}} \simeq (\mathfrak{K}_{m+1} \otimes \mathfrak{H}_m)_{\{j\}}$. Therefore, for all $\overline{a}$ there is some $\overline{a}'$ such that $(*)$ holds of $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{A}', \overline{a}')$, and vice versa. If **II** can maintain $(*)$, this implies that $\mathrm{Tp}^{C^m}(\mathfrak{A}; 2) = \mathrm{Tp}^{C^m}(\mathfrak{A}'; 2)$ and, with Lemma 1.34, that indeed $\mathfrak{A} \equiv^{C^m} \mathfrak{A}'$.

Assume now that $(*)$ is satisfied in the current position. Assume further that **I** chooses pebble 1 to play. Without loss of generality suppose that $\pi(a_2, \ldots, a_m) = \pi'(a'_2, \ldots, a'_m) \subseteq \{3, \ldots, m+1\}$ and that the given isomorphism is the identity mapping in restriction to $\pi^{-1}(\{3, \ldots, m+1\})$:

$$\begin{aligned} & \big(\mathfrak{A} \restriction \pi^{-1}(\{3,\dots,m+1\}), a_2,\dots,a_m\big) \\ = \quad & \big(\mathfrak{A}' \restriction \pi'^{-1}(\{3,\dots,m+1\}), a'_2,\dots,a'_m\big). \end{aligned}$$

Consider any potential target position for pebble 1 over $\mathfrak{A}$ say. If a_1 is placed within $\pi^{-1}(\{3,\dots,m+1\})$ then we want a'_1 to be placed according to the given isomorphism (which happens to be the identity under our assumptions). The interesting case is that a_1 is moved to either $\pi^{-1}(1)$ or $\pi^{-1}(2)$. It follows from the considerations above that for $i = 1, 2$ there are isomorphisms γ_i between $\mathfrak{A} \restriction \pi^{-1}\big(\{1,\dots,m+1\} \setminus \{i\}\big)$ and $\mathfrak{A}' \restriction \pi'^{-1}\big(\{1,\dots,m+1\} \setminus \{i\}\big)$ such that γ_i restricts to the identity mapping over $\pi^{-1}(\{3,\dots,m+1\})$, and thus extends the given isomorphism between $\big(\mathfrak{A} \restriction \pi^{-1}(\{3,\dots,m+1\}), a_2,\dots,a_m\big)$ and $\big(\mathfrak{A}' \restriction \pi'^{-1}(\{3,\dots,m+1\}), a'_2,\dots,a'_m\big)$. Let now γ be the following bijection between $\mathfrak{A}$ and $\mathfrak{A}'$:

$$\gamma(v) := \begin{cases} v & \text{for } v \in \pi^{-1}(\{3,\dots,m+1\}) \\ \gamma_2(v) & \text{for } v \in \pi^{-1}(1) \\ \gamma_1(v) & \text{for } v \in \pi^{-1}(2)\,. \end{cases}$$

Let **II** play according to γ: if **I** proposes $B \subseteq \mathfrak{A}$ say, then **II** answers $B' = \gamma(B)$ and upon a move of pebble 1 in $\mathfrak{A}'$ to $b' \in \gamma(B)$, **II** moves pebble 1 in $\mathfrak{A}$ to $\gamma^{-1}(b')$. $(*)$ is satisfied by construction in the resulting stage — the required isomorphism is provided by the corresponding restriction of γ. ☐

We thus have in particular the following theorem.

Theorem 2.9. *The logics $C^k_{\infty\omega}$ form a strict hierarchy with respect to k even for boolean queries on finite graphs:*

$$C^1_{\infty\omega} \subsetneq C^2_{\infty\omega} \subsetneq \dots \subsetneq C^k_{\infty\omega} \subsetneq C^{k+1}_{\infty\omega} \subsetneq \dots \subseteq C^\omega_{\infty\omega}.$$

It follows that $C^\omega_{\infty\omega} \subsetneq L_{\infty\omega}$ — not every query on finite structures is expressible in $C^\omega_{\infty\omega}$.

The second claim is provable from the first by diagonalization. A concrete graph query which is not in $C^\omega_{\infty\omega}$ is of course $\big\{(\mathfrak{K}_{m+1} \otimes \mathfrak{H}_m)^0 \bigm| m \geqslant 2\big\}$, or rather the closure of this set under isomorphisms.

2.1.2 Proof of Theorem 2.2

The proof is given in two separate lemmas, one for each implication in the theorem.

Lemma 2.10. *If $(\mathfrak{A}, \overline{a}) \not\equiv^{C^k_{\infty\omega}} (\mathfrak{A}', \overline{a}')$ then player* **I** *can force a win in the game on $\big(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}'\big)$.*

Proof. Let $(\mathfrak{A}, \overline{a}) \not\equiv^{C^k_{\infty\omega}} (\mathfrak{A}', \overline{a}')$. There is some formula φ in $C^k_{\infty\omega}$ such that $\mathfrak{A} \models \varphi[\overline{a}]$ but $\mathfrak{A}' \models \neg\varphi[\overline{a}']$. Let ξ be the quantifier rank of φ. $\xi > 0$ unless **I** has already won. We prove that **I** can in one move force resulting positions that can be distinguished by a formula of quantifier rank $\zeta < \xi$. This suffices to give **I** a strategy, since by repeated application of such moves the ordinal valued quantifier rank of the distinguishing formula must reach 0 in finitely many steps — a win for **I**. Assume without loss of generality that φ is of the form $\exists^{\geqslant m} x_j \psi(\overline{x})$. Other cases reduce to this one through the symmetry of the claim and by replacing φ by one of its boolean constituents if necessary. If **I** chooses pebble index j and proposes a set $B := \{b \in A \mid \mathfrak{A} \models \psi[\overline{a}\frac{b}{j}]\}$ of cardinality m, then **II** cannot help but include at least one element b' in the response B' such that $\mathfrak{A}' \models \neg\psi[\overline{a}'\frac{b'}{j}]$. This is simply because by assumption on φ there are less than m positive examples available over $(\mathfrak{A}', \overline{a}')$. **I** need only choose such a b' from B' to force a resulting position in which ψ of quantifier rank less than ξ distinguishes the two tuples. □

Lemma 2.11. *Player* **II** *has a strategy to maintain* $\equiv^{C^k_{\infty\omega}}$*-equivalence of game positions.*

Proof. Assume $(\mathfrak{A}, \overline{a}) \equiv^{C^k_{\infty\omega}} (\mathfrak{A}', \overline{a}')$. It has to be shown that in response to any choices **I** can make during one round **II** can achieve $\equiv^{C^k_{\infty\omega}}$-equivalence in the resulting positions. From Lemma 1.39 we know that each $C^k_{\infty\omega}$-type α is isolated by some formula $\varphi_\alpha(\overline{x}) \in C^k_{\infty\omega}$. For each α and each j, the number

$$\nu^\alpha_j(\mathfrak{A}, \overline{a}) = \left|\{b \in A \mid \mathrm{tp}^{C^k_{\infty\omega}}_{\mathfrak{A}}(\overline{a}\tfrac{b}{j}) = \alpha\}\right| = \left|\{b \in A \mid \mathfrak{A} \models \varphi_\alpha[\overline{a}\tfrac{b}{j}]\}\right|$$

is determined by $\mathrm{tp}^{C^k_{\infty\omega}}_{\mathfrak{A}}(\overline{a})$: $\exists^{=m} x_j \varphi_\alpha(\overline{x})$ is in $\mathrm{tp}^{C^k_{\infty\omega}}_{\mathfrak{A}}(\overline{a})$ exactly for $m = \nu^\alpha_j(\mathfrak{A}, \overline{a})$. $(\mathfrak{A}, \overline{a}) \equiv^{C^k_{\infty\omega}} (\mathfrak{A}', \overline{a}')$ therefore implies that for all α and j the corresponding numbers must be equal for $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{A}', \overline{a}')$: $\nu^\alpha_j(\mathfrak{A}, \overline{a}) = \nu^\alpha_j(\mathfrak{A}', \overline{a}')$. Suppose now that **I** chooses to play in the j-th component and proposes $B \subseteq A$ as a challenge. By the above equality **II** can choose $B' \subseteq A'$ such that for all α:

$$\left|\{b \in B \mid \mathrm{tp}^{C^k_{\infty\omega}}_{\mathfrak{A}}(\overline{a}\tfrac{b}{j}) = \alpha\}\right| = \left|\{b' \in B' \mid \mathrm{tp}^{C^k_{\infty\omega}}_{\mathfrak{A}'}(\overline{a}'\tfrac{b'}{j}) = \alpha\}\right|.$$

But now, no matter which $b' \in B'$ **I** chooses, **II** can make sure to answer with some $b \in B$ such that the resulting tuples, $\overline{a}\frac{b}{j}$ and $\overline{a}'\frac{b'}{j}$ again realize the same $C^k_{\infty\omega}$-type, so that $\equiv^{C^k_{\infty\omega}}$-equivalence is maintained. □

Before pursuing the analysis of the games, let us remark that unlike the standard treatment of the k-pebble games for $L^k_{\infty\omega}$ and $C^k_{\infty\omega}$ we have chosen to consider only positions with all k pebbles placed on their respective structures. The standard treatment allows to start the game with all pebbles outside the structures. Until the point where all pebbles have been placed player **I** may either choose to play a round using one of the pebbles already placed or one of those not yet used. Otherwise everything is unchanged. That choice has the advantage that the main theorems directly apply to naked structures and characterize the equivalence relations $\equiv^{\mathcal{L}}$ over $\mathrm{fin}[\tau]$ rather than over $\mathrm{fin}[\tau;k]$. The disadvantage is that the games are slightly less uniform during the initial phase in which only some of the pebbles have been placed and the formal treatment must make more or less awkward provisions for that. We do not really lose anything in our restriction to full positions, however, because by Lemma 1.34 $\mathfrak{A} \equiv^{\mathcal{L}} \mathfrak{A}'$ if and only if $\mathfrak{A}$ and $\mathfrak{A}'$ realize exactly the same $\mathcal{L}$-types. As we shall mostly study $\equiv^{\mathcal{L}}$ as an equivalence relation on $\mathrm{fin}[\tau;k]$, we prefer to deal with the variant introduced above.

2.1.3 Further Analysis of the C^k-Game

An inductive analysis of strategies. Think of an arbitrary but fixed k throughout the following. The obvious dependence of various introduced notions on the value of k is mostly suppressed in the notation. Recall that $\mathrm{fin}[\tau;k]$ is the class of all finite τ-structures with a k-tuple of designated elements.

Definition 2.12. *Let $\approx_0$ be the relation of atomic equivalence on $\mathrm{fin}[\tau;k]$:*

$$(\mathfrak{A},\overline{a}) \approx_0 (\mathfrak{A}',\overline{a}') \quad \textit{if} \quad \mathrm{atp}_{\mathfrak{A}}(\overline{a}) = \mathrm{atp}_{\mathfrak{A}'}(\overline{a}').$$

Recall that atomic equivalence is what is required in the winning condition for player **II**, (W): player **II** has not yet lost in stage $(\mathfrak{A},\overline{a};\mathfrak{A}',\overline{a}')$ if $(\mathfrak{A},\overline{a}) \approx_0 (\mathfrak{A}',\overline{a}')$. Obviously $\approx_0$ is an equivalence relation on positions. A strategy for **II** must specify possible moves for **II** that allow to stay within $\approx_0$ in response to any moves **I** might make. Inductively this task reduces to the specification of strategies for one additional round. Suppose the relation $\approx_i$ on pairs of positions captures the existence of a strategy for at least i moves. Then the corresponding relation $\approx_{i+1}$ must exactly contain all stages (pairs of positions) in which **II** has a strategy for a single round to enforce a resulting stage in $\approx_i$. What constitutes a strategy for the single round is governed by the rules of the game.

Lemma 2.13. *Let $\sim$ be an equivalence relation on $\mathrm{fin}[\tau;k]$. Let $\sim'$ be the relation on $\mathrm{fin}[\tau;k]$ that contains $(\mathfrak{A},\overline{a};\mathfrak{A}',\overline{a}')$ if and only if $(\mathfrak{A},\overline{a}) \sim (\mathfrak{A}',\overline{a}')$ and in a single round of the C^k-game on stage $(\mathfrak{A},\overline{a};\mathfrak{A}',\overline{a}')$ player* **II** *can force the resulting stage to be in $\sim$ again. Then $\sim'$ is definable as follows:*

$$\begin{aligned}
(\mathfrak{A},\overline{a}) \sim' (\mathfrak{A}',\overline{a}') \quad & \textit{if} \\
& (\mathfrak{A},\overline{a}) \sim (\mathfrak{A}',\overline{a}') \\
& \textit{and for all } j \in \{1,\ldots,k\} \textit{ and all } \alpha \in \mathrm{fin}[\tau;k]\,/\!\sim \\
& \Big|\big\{b \in A \;\big|\; (\mathfrak{A},\overline{a}\tfrac{b}{j}) \in \alpha\big\}\Big| = \Big|\big\{b' \in A' \;\big|\; (\mathfrak{A}',\overline{a}'\tfrac{b'}{j}) \in \alpha\big\}\Big|.
\end{aligned}$$

In particular $\sim'$ is also an equivalence relation on $\mathrm{fin}[\tau;k]$.

Proof. i) Suppose first that the condition on the right hand side is satisfied by $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$. The proof that **II** can force $\sim$-equivalence in a single round is very similar to the proof of Lemma 2.11 above. Note that both, the rules for a round in the game and the condition in the lemma are symmetric with respect to the constituent positions $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{A}', \overline{a}')$. Let **I** in the first part of the round choose j and $B \subseteq A$. Split B into disjoint subsets B_α for $\alpha \in \text{fin}[\tau; k] /\!\sim$ through: $B_\alpha := \{b \in B \mid (\mathfrak{A}, \overline{a}\frac{b}{j}) \in \alpha\}$. By assumption, there exists for each B_α a subset $B'_\alpha \subseteq \{b' \in A' \mid (\mathfrak{A}', \overline{a}'\frac{b'}{j}) \in \alpha\}$ of exactly the same size as B_α. Note that the sets $\{b' \in A' \mid (\mathfrak{A}', \overline{a}'\frac{b'}{j}) \in \alpha\}$ are disjoint for different α. If **II** responds with $B' := \bigcup_\alpha B'_\alpha$ then, in the second exchange of moves in this round, **II** can force $\sim$-equivalence as desired: **I** chooses $b' \in B'_{\alpha_0}$ for some α_0; **II** need merely choose b from B_{α_0} to ensure $(\mathfrak{A}, \overline{a}\frac{b}{j}) \sim (\mathfrak{A}', \overline{a}'\frac{b'}{j})$ since both positions are in α_0.
ii) Suppose now that the condition on the right hand side is not satisfied. The interesting case is that this is not due to $\sim$-inequivalence. We show how **I** can force a successor stage that is not in $\sim$. By symmetry we may assume that for some j and α, $\left|\{b \in A \mid (\mathfrak{A}, \overline{a}\frac{b}{j}) \in \alpha\}\right| > \left|\{b' \in A' \mid (\mathfrak{A}', \overline{a}'\frac{b'}{j}) \in \alpha\}\right|$. Let **I** choose this j and $B := \{b \in A \mid (\mathfrak{A}, \overline{a}\frac{b}{j}) \in \alpha\}$. Whichever B' of the same size as B player **II** chooses, there has to be some $b' \in B'$ such that $(\mathfrak{A}', \overline{a}'\frac{b'}{j})$ is not in α. If **I** chooses such b' a resulting stage with $\sim$-inequivalent positions is forced. □

Definition 2.14. *Define a family of binary relations $\approx_i$ on $\text{fin}[\tau; k]$ as follows:*

$(\mathfrak{A}, \overline{a}) \approx_i (\mathfrak{A}', \overline{a}')$ *if* *Player* **II** *has a strategy for at least i rounds in the C^k-game on $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$.*

Note that the above definition of $\approx_0$ as equality of atomic types is consistent with this new definition. Lemma 2.13 can be applied to generate inductively equivalence relations $\approx_i$ that capture the existence of a strategy for at least i moves. Obviously $\approx_{i+1}$ is obtained from $\approx_i$ through the refinement step described in Lemma 2.13, $\approx_{i+1} = (\approx_i)'$.

In particular it follows inductively from the condition in Lemma 2.13 that all the $\approx_i$ are equivalence relations on $\text{fin}[\tau; k]$. For future reference we present the inductive description of the $\approx_i$ in detail.

Proposition 2.15. *Let the $\approx_i$ on $\text{fin}[\tau; k]$ be defined through the existence of a strategy for player* **II** *for at least i rounds in the C^k-game. Then these are inductively definable in the following process:*

$(\mathfrak{A}, \overline{a}) \approx_0 (\mathfrak{A}', \overline{a}')$ *iff* $\text{atp}_{\mathfrak{A}}(\overline{a}) = \text{atp}_{\mathfrak{A}'}(\overline{a}')$

$(\mathfrak{A}, \overline{a}) \approx_{i+1} (\mathfrak{A}', \overline{a}')$ *iff*

$(\mathfrak{A}, \overline{a}) \approx_i (\mathfrak{A}', \overline{a}')$

and for all $j \in \{1, \ldots, k\}$ *and all* $\alpha \in \text{fin}[\tau; k] /\!\approx_i$

$$\left|\{b \in A \mid (\mathfrak{A}, \overline{a}\tfrac{b}{j}) \in \alpha\}\right| = \left|\{b' \in A' \mid (\mathfrak{A}', \overline{a}'\tfrac{b'}{j}) \in \alpha\}\right|.$$

As a sequence of successively refined equivalence relations the $\approx_i$ possess a limit or roughest common refinement. Formally this limit $\approx$ is the intersection of all $\approx_i$ for $i \in \omega$:

$$\approx_i \xrightarrow{i \to \infty} \approx = \bigcap_i \approx_i .$$

We show that $\approx$ captures the existence of a strategy in the infinite game.

Lemma 2.16. *Let* $\approx := \bigcap_i \approx_i$. *Then*

$(\mathfrak{A}, \overline{a}) \approx (\mathfrak{A}', \overline{a}')$ *iff* *Player* **II** *has a strategy in the infinite* C^k*-game on* $(\mathfrak{A}, \overline{a}; \mathfrak{A}', \overline{a}')$.

Proof. This is the first place in the analysis of the games where we use the finiteness of the underlying structures. Fix two structures $\mathfrak{A}, \mathfrak{A}'$ and let $\approx^{\mathfrak{A}\mathfrak{A}'}$ and $\approx_i^{\mathfrak{A}\mathfrak{A}'}$ stand for the restrictions of $\approx$ and $\approx_i$ to positions over $\mathfrak{A}$ and $\mathfrak{A}'$. Thus $\approx^{\mathfrak{A}\mathfrak{A}'}$ is the limit of the decreasing sequence of subsets $\approx_i^{\mathfrak{A}\mathfrak{A}'}$ of the finite set $A^k \times A'^k$. It follows that $\approx_{i+1}^{\mathfrak{A}\mathfrak{A}'} = \approx_i^{\mathfrak{A}\mathfrak{A}'} = \approx^{\mathfrak{A}\mathfrak{A}'}$ for some i. But this means that for such i and in games over $\mathfrak{A}$ and $\mathfrak{A}'$ player **II** is guaranteed to have a strategy for at least $i+1$ rounds whenever there is a strategy for at least i rounds. The strategy in the infinite game now simply is to maintain $\approx_i^{\mathfrak{A}\mathfrak{A}'}$-equivalence: $\approx_i^{\mathfrak{A}\mathfrak{A}'}$-equivalence implies $\approx_{i+1}^{\mathfrak{A}\mathfrak{A}'}$-equivalence and this can by definition be used to enforce $\approx_i^{\mathfrak{A}\mathfrak{A}'}$-equivalence in each consecutive round. □

Equivalence of positions and equality of types. We can now show that the $\approx$-classes coincide with the $C^k_{\omega\omega}$-types as well as with the $C^k_{\infty\omega}$-types over fin$[\tau]$. This correspondence in particular yields a proof of Corollary 2.4. Recall form Definition 1.36 that the $C^k_{\infty\omega;i}$ consist of all those formulae of $C^k_{\infty\omega}$ whose quantifier rank is at most i. By what we already have, it suffices to show that $\approx_i$ is equivalence in $C^k_{\infty\omega;i}$ for all $i \in \omega$. For then, the following limit equations prove the claim:

$$\begin{array}{ccc} \equiv^{C^k_{\omega\omega;i}} & \xrightarrow{i \to \infty} & \equiv^{C^k_{\omega\omega}} \\ \| & & \\ \equiv^{C^k_{\infty\omega;i}} & & \\ \| & & \\ \approx_i & \xrightarrow{i \to \infty} & \approx \end{array}$$

The indicated limits are clear: $C^k_{\omega\omega} = \bigcup_i C^k_{\omega\omega;i}$ so that $C^k_{\omega\omega}$-equivalence is the limit of the equivalences with respect to the $C^k_{\omega\omega;i}$. $\approx = \bigcap_i \approx_i$ by the definition of $\approx$.

Coincidence between $C^k_{\omega\omega;i}$-equivalence and $C^k_{\infty\omega;i}$-equivalence follows from our preliminary analysis in Chapter 1, see Corollary 1.40. But from Theorem 2.2 and Lemma 2.16 we already know that $\approx$ is $C^k_{\infty\omega}$-equivalence. It follows that indeed on $\mathrm{fin}[\tau;k]$ all three notions of equivalence

$$\equiv^{C^k_{\omega\omega}}, \equiv^{C^k_{\infty\omega}}, \text{ and } \approx$$

coincide. This is precisely the statement of Corollary 2.4. It remains to prove inductively the coincidence between $\approx_i$ and $C^k_{\infty\omega;i}$-equivalence.

Lemma 2.17. *The equivalence relation $\approx_i$ coincides with $C^k_{\infty\omega;i}$-equivalence on $\mathrm{fin}[\tau;k]$ for all $i \in \omega$.*

Proof. By induction on i. The claim is true for $i = 0$ by definition. Recall from Proposition 2.15 how $\approx_{i+1}$ is characterized in terms of $\approx_i$:

$$\begin{aligned}(\mathfrak{A},\overline{a}) \approx_{i+1} (\mathfrak{A}',\overline{a}') \quad &\text{iff} \\ &(\mathfrak{A},\overline{a}) \approx_i (\mathfrak{A}',\overline{a}') \\ &\text{and for all } j \in \{1,\dots,k\} \text{ and all } \alpha \in \mathrm{fin}[\tau;k] \,/\!\approx_i \\ &\left|\left\{b \in A \mid (\mathfrak{A},\overline{a}\tfrac{b}{j}) \in \alpha\right\}\right| = \left|\left\{b' \in A' \mid (\mathfrak{A}',\overline{a}'\tfrac{b'}{j}) \in \alpha\right\}\right|.\end{aligned}$$

It suffices to prove the following, which says that the $\equiv^{C^k_{\infty\omega;i}}$ are governed by the same rules:

$$\begin{aligned}(\mathfrak{A},\overline{a}) \equiv^{C^k_{\infty\omega;i+1}} (\mathfrak{A}',\overline{a}') \quad &\text{iff} \\ &(\mathfrak{A},\overline{a}) \equiv^{C^k_{\infty\omega;i}} (\mathfrak{A}',\overline{a}') \\ &\text{and for all } j \in \{1,\dots,k\} \text{ and all } \alpha \in \mathrm{fin}[\tau;k] \,/\!\equiv^{C^k_{\infty\omega;i}} \\ &\left|\left\{b \in A \mid (\mathfrak{A},\overline{a}\tfrac{b}{j}) \in \alpha\right\}\right| = \left|\left\{b' \in A' \mid (\mathfrak{A}',\overline{a}'\tfrac{b'}{j}) \in \alpha\right\}\right|.\end{aligned}$$

The "only if"–part is clear, since by Lemma 1.39 each $\equiv^{C^k_{\infty\omega;i}}$-class α is isolated by a formula $\varphi_\alpha(\overline{x}) \in C^k_{\infty\omega;i}$. Therefore, if

$$\nu^\alpha_j(\mathfrak{A},\overline{a}) := \left|\left\{b \in A \mid (\mathfrak{A},\overline{a}\tfrac{b}{j}) \in \alpha\right\}\right|,$$

then $\exists^{=m} x_j \varphi_\alpha(\overline{x})$ is in the $C^k_{\infty\omega;i+1}$-type of $(\mathfrak{A},\overline{a})$ for $m = \nu^\alpha_j(\mathfrak{A},\overline{a})$. For the "if"–part it suffices to show that the numbers $\nu^\alpha_j(\mathfrak{A},\overline{a})$, for all α and j isolate the $C^k_{\infty\omega;i+1}$-type of $(\mathfrak{A},\overline{a})$. This, however, is clear: whether $\mathfrak{A} \models \exists^{\geqslant m} x_j \psi[\overline{a}]$ for $\psi \in C^k_{\infty\omega;i}$ is determined by $\sum \nu^\alpha_j(\mathfrak{A},\overline{a})$ for those α that contain ψ. □

Since we only deal with finite structures we henceforth identify $\equiv^{C^k_{\infty\omega}}$ and $\equiv^{C^k_{\omega\omega}}$ and indistinguishably write $\equiv^{C^k}$. Correspondingly, the distinction between $C^k_{\infty\omega}$- and $C^k_{\omega\omega}$-types is dropped and we may simply speak of C^k-types over finite structures.

Referring back to the inductive generation of the $\approx_i$ as characterized in Proposition 2.15 and combining this with the insight that the limit of the $\approx_i$

is C^k-equivalence, we have the following rather algebraic characterization of $\equiv^{C^k}$ over $\mathrm{fin}[\tau;k]$.

Remark 2.18. *$\equiv^{C^k}$ on $\mathrm{fin}[\tau;k]$ is the roughest equivalence relation $\approx$ on $\mathrm{fin}[\tau;k]$ that is at least as fine as atomic equivalence and satisfies the following fixed-point equation:*

$$\begin{array}{l}(\mathfrak{A},\overline{a}) \approx (\mathfrak{A}',\overline{a}') \\ \Longleftrightarrow \\ \textit{for all } j \in \{1,\dots,k\} \textit{ and all } \alpha \in \mathrm{fin}[\tau;k] \,/\!\approx \\ \Big|\big\{b \in A \mid (\mathfrak{A},\overline{a}\tfrac{b}{j}) \in \alpha\big\}\Big| = \Big|\big\{b' \in A' \mid (\mathfrak{A}',\overline{a}'\tfrac{b'}{j}) \in \alpha\big\}\Big|.\end{array}$$

The fixed-point equation directly corresponds with the equation that governs the refinement step $\approx_i \longmapsto \approx_{i+1}$ in Proposition 2.15.

2.1.4 The Analogous Treatment for L^k

Both, the proof of Theorem 2.1 and the analysis of the L^k-game that leads to Corollary 2.3, are carried out along exactly the same lines as for the C^k-game. The more transparent rules for the single round, however, lead to considerable simplifications. The inductive generation of the corresponding equivalence relations $\approx_i$ on game positions is formally much simpler, though strictly analogous in spirit. Instead of Proposition 2.15 we now find the following.

Proposition 2.19. *With the $\approx_i$ defined through the existence of a strategy for player* **II** *in the L^k-game, these are inductively definable as follows:*

$$\begin{array}{lll}(\mathfrak{A},\overline{a}) \approx_0 (\mathfrak{A}',\overline{a}') & \textit{iff} & \mathrm{atp}_{\mathfrak{A}}(\overline{a}) = \mathrm{atp}_{\mathfrak{A}'}(\overline{a}') \\ (\mathfrak{A},\overline{a}) \approx_{i+1} (\mathfrak{A}',\overline{a}') & \textit{iff} & \\ & \multicolumn{2}{l}{(\mathfrak{A},\overline{a}) \approx_i (\mathfrak{A}',\overline{a}')} \\ & \multicolumn{2}{l}{\textit{and for all } j \in \{1,\dots,k\} \textit{ and all } \alpha \in \mathrm{fin}[\tau;k] \,/\!\approx_i} \\ & \multicolumn{2}{l}{\exists b \in A\Big((\mathfrak{A},\overline{a}\tfrac{b}{j}) \in \alpha\Big) \longleftrightarrow \exists b' \in A'\Big((\mathfrak{A}',\overline{a}'\tfrac{b'}{j}) \in \alpha\Big).}\end{array}$$

The limit $\approx_i \xrightarrow{i\to\infty} \approx$, where the equivalence relations $\approx_i$ now stand for equivalence with respect to the L^k-game, becomes equality of $L^k_{\infty\omega}$-types over finite structures. The $\approx_i$ also correspond to indistinguishability in the bounded quantifier rank fragments $L^k_{\infty\omega;i}$ of $L^k_{\infty\omega}$. $L^k_{\infty\omega;i}$-equivalence is the same as $L^k_{\omega\omega;i}$-equivalence by Corollary 1.40. Thus,

$$\equiv^{L^k_{\omega\omega}}, \equiv^{L^k_{\infty\omega}}, \text{ and } \approx$$

coincide, where $\approx$ now is equivalence in the L^k-game. This is precisely the statement of Corollary 2.3.

It is therefore justified to write $\equiv^{L^k}$ for both, equivalence in $L^k_{\infty\omega}$ or $L^k_{\omega\omega}$. Accordingly we identify $L^k_{\infty\omega}$-types and $L^k_{\omega\omega}$-types over finite structures and address them as L^k-types.

Finally an algebraic characterization of $\equiv^{L^k}$ in the style of Remark 2.18 is obtained: $\equiv^{L^k}$ is the roughest equivalence relation $\approx$ on fin$[\tau; k]$ that is at least as fine as atomic equivalence and satisfies the following fixed-point equation:

$$\begin{array}{l}(\mathfrak{A}, \overline{a}) \approx (\mathfrak{A}', \overline{a}') \\ \Longleftrightarrow \\ \text{for all } j \in \{1, \dots, k\} \text{ and all } \alpha \in \text{fin}[\tau; k] \,/\!\approx \\ \exists b \in A\Big((\mathfrak{A}, \overline{a}\tfrac{b}{j}) \in \alpha\Big) \longleftrightarrow \exists b' \in A'\Big((\mathfrak{A}', \overline{a}'\tfrac{b'}{j}) \in \alpha\Big).\end{array}$$

2.2 Colour Refinement and the Stable Colouring

This section is an intermezzo on our way to obtain definable orderings with respect to C^k- and L^k-types. The basic technique in the underlying inductive processes is intimately related to a similar technique in combinatorial graph theory: the *colour refinement technique* and the *stable colouring*, often also considered under the name of *vertex classification*. We review these notions in some detail and consider variants that are useful in the present development. In particular some definability properties of variants of the stable colouring can later directly be transferred to definability statements for the invariants.

We use the terminology of *pre-orderings* as reviewed in Section 1.7. In particular compare Definition 1.62. We reserve variants of the symbol $\preccurlyeq$ to denote pre-orderings; $\prec$ then denotes the associated strict pre-ordering and $\sim$ the induced equivalence relation. Recall that the quotient $\preccurlyeq/\sim$ is a linear ordering in the sense of $\leqslant$, $\prec/\sim$ the corresponding linear ordering in the sense of $<$. Intuitively $\sim$ describes the *discriminating power* of $\preccurlyeq$. Recall that $\prec$ and $\preccurlyeq$ are quantifier free interdefinable and that $\sim$ is quantifier free definable form either.

2.2.1 The Standard Case: Colourings of Finite Graphs

Let (V, E) be a finite graph. A colouring of (V, E) with finitely many colours $0, \dots, r-1$ is a function $c\colon V \to r$, where $r = \{0, \dots, r-1\}$ as usual. We regard this set of colours as ordered in the natural way. To make the order in the colours explicit, the colouring may be formalized as a pre-ordering on V: $v_1 \preccurlyeq v_2$ if $c(v_1) \leqslant c(v_2)$. The associated $\sim$ is the relation of having the same colour. A particular refinement of c is induced by the following mapping:

$$c'\colon v \longmapsto \Big(c(v), \big|\{w | Evw \wedge c(w) = 0\}\big|, \dots, \big|\{w | Evw \wedge c(w) = r-1\}\big|\Big).$$

Let $\sim'$ be the relation of having the same new colour. Obviously $v_1 \sim' v_2$ if and only if v_1 and v_2 have the same colour under c and the same numbers of direct neighbours in any of the c-colours. We note the similarity of this refinement process with that encountered in the refinement for equivalence of positions in the C^k-game as expressed in Lemma 2.13.

The new colours can be ordered lexicographically so that one may also regard c' as a mapping into some initial subset $r' = \{0, \dots, r' - 1\}$ of natural numbers. With our conventions for lexicographic orderings (see Section 1.7.3) the colours under c' get ordered with dominating c-colour.

The new c' is the *colour refinement* of c. Let $\preccurlyeq'$, $\prec'$ and $\sim'$ be the characteristic descriptions of c' in terms of pre-orderings. The colouring c' is a refinement of c in the sense that $\sim'$ is a refinement of $\sim$ and that for $\prec$ and $\prec'$ we have: $\prec \subseteq \prec'$. The discriminating power of the colouring is possibly enhanced in the passage from c to c', but the new ordering of colours is compatible with the former one.

Since (V, E) is finite, repeated colour refinement must terminate in a stationary colouring after at most $|V|$ steps. In the standard graph theoretic setting this limit process is applied to the trivial monochromatic colouring $c_0 : V \to \{0\}$. Note that this trivial colouring corresponds to the pre-ordering $\preccurlyeq_0 = V \times V$ (with associated strict pre-ordering $\prec_0 = \emptyset$). The limit colouring obtained in this way is called the *stable colouring* of the graph. At the level of the associated strict pre-orderings the stable colouring is the *least fixed point* of the monotone operator corresponding to the single colour refinement step sending $\prec$ to $\prec'$:

$$\prec = \bigcup_i \prec_i \qquad \begin{array}{rl} \text{where} & \prec_0 = \emptyset \\ \text{and} & \prec_{i+1} = (\prec_i)'. \end{array}$$

The first successor level $\prec_1$ is just the pre-ordering according to the degree of vertices. Note that the description of the refinement process is monotone increasing in terms of $\prec$ and monotone decreasing in terms of $\sim$ and $\preccurlyeq$.

2.2.2 Definability of the Stable Colouring

A slight generalization of the setting in which the colour refinement technique is applicable concerns k-graphs with any given initial pre-ordering on the set of vertices. We use the term *k-graph* to denote structures with k binary relations $E_1, \dots, E_k$ instead of the single edge relation in the standard case. Also we here need not require these relations to be irreflexive or symmetric. An additional arbitrary pre-ordering $\preccurlyeq_0$ serves as an initial stage for the colour refinement. In terms of colourings we now pass from a colouring $c: V \to r$ to a refinement c' obtained from a lexicographic ordering of the new colours

$$c' : v \longmapsto \Big(c(v), \big(\nu_j^s(v) \big)_{1 \leqslant j \leqslant k, 0 \leqslant s < r} \Big),$$

where $\nu_j^s(v) = \big| \{ w | E_j vw \wedge c(w) = s \} \big|$.

Recall once more our conventions for the lexicographic ordering: a new colour $\overline{m} = (m, (m_{js}))$ is regarded as a tuple with first component m and consecutive components m_{js} listed according to the lexicographic ordering on the index pairs (j, s). For $\overline{m} = (m, (m_{js}))$ and $\overline{m}' = (m', (m'_{js}))$ we get that $\overline{m} < \overline{m}'$ if $m < m'$ or if $m = m'$ and $m_{js} < m'_{js}$ for the least (j, s) such that $m_{js} \neq m'_{js}$.

For the description in terms of the associated pre-orderings $\preccurlyeq$ and $\preccurlyeq'$ with corresponding strict $\prec$ and $\prec'$ and equivalences $\sim$ and $\sim'$ this becomes:

$$\begin{aligned} v_1 \prec' v_2 \quad & \text{iff} \\ & v_1 \prec v_2 \quad \text{or} \\ & v_1 \sim v_2 \text{ and } \big(\nu_j^s(v_1)\big) <_{\text{lex}} \big(\nu_j^s(v_2)\big), \\ & \text{where } \nu_j^s(v) = \big|\{w | E_j vw \wedge c(w) = s\}\big|. \end{aligned} \tag{2.1}$$

The structural similarity of this refinement process with that in Proposition 2.15 is most apparent for the associated equivalences $\sim$ and $\sim'$:

$$\begin{aligned} v_1 \sim' v_2 \quad & \text{iff} \\ & v_1 \sim v_2 \text{ and for all } j \in \{1, \ldots, k\} \text{ and all } \alpha \in V/\sim \\ & \Big|\big\{u \in V \mid E_j v_1 u \wedge u \in \alpha\big\}\Big| = \Big|\big\{u \in V \mid E_j v_2 u \wedge u \in \alpha\big\}\Big|. \end{aligned} \tag{2.2}$$

Definition 2.20. *The* stable colouring *of a pre-ordered finite k-graph is the limit pre-ordering $\preccurlyeq$ obtained through application of the above refinement operation with the given $\preccurlyeq_0$ as the initial stage:*

$$\preccurlyeq \text{ is the limit } \preccurlyeq_i \xrightarrow{i \to \infty} \preccurlyeq \qquad \text{where inductively } \preccurlyeq_{i+1} := (\preccurlyeq_i)'.$$

We regard the $\preccurlyeq_i$ and $\preccurlyeq$ as global relations on finite pre-ordered k-graphs.

The standard version of the stable colouring of graphs is comprised as a special case for $k = 1$ and for trivial initial pre-ordering $\preccurlyeq_0 = V \times V$. In this form the following result is due to Immerman and Lander, see Theorem 2.23 below.

Lemma 2.21. *The stable colouring $\preccurlyeq$ of finite pre-ordered k-graphs is definable in $C^2_{\infty\omega}$.*

Proof. Let $\prec_i$ and $\prec$ stand for the associated strict pre-orderings, $\sim_i$ and $\sim$ for the induced equivalences. It is sufficient to show that each level $\prec_i$ in the fixed-point process that generates $\prec$ is definable by some $C^2_{\infty\omega}$-formula $\varphi_i(x, y)$. Then the limit of the sequence $\prec_0 \subseteq \prec_1 \subseteq \cdots$ is defined by

$$\varphi(x, y) := \bigvee_{i \in \omega} \varphi_i(x, y).$$

i) Suppose that φ_i defines $\prec_i$. Then for each $s \geqslant 0$ there is a formula $\psi_{i,s}(x)$ of $C^2_{\infty\omega}$ in a single free variable which defines the s-th equivalence class with respect to $\sim_i$ in the sense of the ordering $\prec_i$. We first generate auxiliary $\chi_{i,s}(x)$ that define the union of the classes up to s: $\chi_{i,0}(x) := \neg\exists y(\varphi_i(y, x))$

defines the $\prec_i$-least $\sim_i$-class. As usual, $\varphi_i(y,x)$ is the result of exchanging all occurrences of x and y in $\varphi_i(x,y)$. Inductively let $\chi_{i,s+1}(x) := \forall y(\varphi_i(y,x) \rightarrow \chi_{i,s}(y))$. Finally $\psi_{i,s}(x) := \chi_{i,s}(x) \wedge \neg\chi_{i,s-1}(x)$ is as desired.
ii) Definability of the $\prec_i$ is established by an induction with respect to i. $\varphi_0(x,y) := x \preccurlyeq_0 y \wedge \neg\, y \preccurlyeq_0 x$ defines $\prec_0$ as the strict variant of the given $\preccurlyeq_0$. Recall from the definitions that

$$\begin{aligned} x \prec_{i+1} y \quad &\text{iff} \\ &x \prec_i y \quad \text{or} \\ &x \sim_i y \text{ and } \big(\nu_j^s(x)\big) <_{\text{lex}} \big(\nu_j^s(y)\big). \end{aligned} \tag{2.3}$$

$\nu_j^s(x) = \big|\{z | E_j xz \wedge \psi_{i,s}(z)\}\big|$ is the number of E_j-neighbours to x that are in the s-th class with respect to $\sim_i$.

The crucial lexicographic comparison $\big(\nu_j^s(x)\big) <_{\text{lex}} \big(\nu_j^s(y)\big)$ can be expressed as follows:

$$\bigvee_{(j,s)} \Big(\bigwedge_{(j',s')<(j,s)} \nu_{j'}^{s'}(x) = \nu_{j'}^{s'}(y) \;\;\wedge\;\; \nu_j^s(x) < \nu_j^s(y) \Big).$$

Since $\nu_j^s(x) = \big|\{y \mid E_j xy \wedge \psi_{i,s}(y)\}\big|$ it only remains to dissolve the cardinality equalities and inequalities in the last formula into infinite disjunctions according to the following pattern:

$$\Big|\{u|\chi(x,u)\}\Big| < \Big|\{u|\chi(y,u)\}\Big| \iff \bigvee_{m<n} \Big(\exists^{=m} y\, \chi(x,y) \wedge \exists^{=n} x\, \chi(y,x)\Big).$$

□

Beside infinitary definability in only two variables with counting we also get definability in an extension of fixed-point logic just sufficiently expressive to permit cardinality comparison. Recall the definition of the Rescher quantifier from Definition 1.53.

Lemma 2.22. *The stable colouring $\preccurlyeq$ of finite pre-ordered k-graphs is globally definable in* $\mathrm{FP}(Q_{\mathrm{R}})$*, fixed-point logic with the Rescher quantifier. In particular it is computable in* PTIME.

Proof. Note that equation 2.3 for the inductive refinement is directly adequate for the definition of $\prec$ as an inductive fixed point. Only, in standard fixed-point processes we initialize the fixed-point variable to $\emptyset$, whereas here we want to substitute the given $\prec_0$ for the initial stage. This is possible with the following standard trick. To obtain the inductive fixed-point for the operator given by $\chi(X,\overline{x})$ but with initialization to an X_0 defined by some $\varphi_0(\overline{x})$ one may use the usual inductive fixed-point over $\chi'(X,\overline{x}) = \big(\neg\exists\overline{x}X\overline{x} \wedge \varphi_0(\overline{x})\big) \vee \big(\exists\overline{x}X\overline{x} \wedge \chi(X,\overline{x})\big)$.

It therefore suffices to show that the lexicographic comparison in equation 2.3 is definable with the Rescher quantifier. $\big(\nu_j^s(x)\big) <_{\text{lex}} \big(\nu_j^s(y)\big)$ can now be reformulated as follows:

$$\begin{aligned}
&\big(\nu_j^s(x)\big) <_{\text{lex}} \big(\nu_j^s(y)\big) \\
\Leftrightarrow\quad & \exists (j,s)\Big[\forall (j',s')\big((j',s') < (j,s) \to \nu_{j'}^{s'}(x) = \nu_{j'}^{s'}(y)\big) \ \wedge\ \nu_j^s(x) < \nu_j^s(y)\Big] \\
\Leftrightarrow\quad & \bigvee_{j=1}^{k} \exists s \left[\begin{array}{c} \bigwedge_{j'<j} \forall s'\big(\nu_{j'}^{s'}(x) = \nu_{j'}^{s'}(y)\big) \ \wedge\ \forall s' < s\,\big(\nu_j^{s'}(x) = \nu_j^{s'}(y)\big) \\ \wedge\ \nu_j^s(x) < \nu_j^s(y) \end{array} \right].
\end{aligned}$$

The quantifications over s and s' can be replaced by quantifications over elements z and z' that represent the s-th and s'-th classes with respect to $\sim_i$. If for instance z is in the s-th $\sim_i$-class then $\nu_j^s(x) = \big|\{u \mid E_j x \wedge u \sim_i z\}\big|$. It follows that the cardinality equalities and comparisons in the above formulae can be expressed with applications of Q_{R}. Thus $\big(\nu_j^s(x)\big) <_{\text{lex}} \big(\nu_j^s(y)\big)$ is in first-order logic with the Rescher quantifier in terms of $\prec_i$.

The limit $\prec$, and with it $\preccurlyeq$, therefore are definable in $\text{FP}(Q_{\text{R}})$. □

2.2.3 $C^2_{\infty\omega}$ and the Stable Colouring

For this section we return to the standard case of the stable colouring, with just one edge relation E and initialization to the trivial pre-ordering. Lemma 2.21 was first stated by Immerman and Lander [IL90] in this form:

Theorem 2.23 (Immerman, Lander). *The stable colouring of graphs is $C^2_{\infty\omega}$-definable in the finite: there is a $C^2_{\infty\omega}$-formula $\eta(x, y)$ defining on all finite graphs the pre-ordering associated with the stable colouring.*

The stable colouring receives special attention in graph theory since on generic graphs it provides canonization up to isomorphism. On *almost all* finite graphs the pre-ordering associated with the stable colouring is a linear ordering. This result is due to Babai, Erdös and Selkow [BES80]. The 'almost all' is to say that the proportion of graphs of size n satisfying the statement tends to 1 as n goes to infinity. In [BK80] this result was further used to provide a graph normalization algorithm that operates in average linear time.

Theorem 2.24 (Babai, Erdös, Selkow). *For almost all finite graphs the stable colouring gives different colours to any two distinct vertices. In other words, almost all finite graphs are in fact linearly ordered (in the sense of $\leqslant$) by the pre-ordering $\preccurlyeq$ associated with the stable colouring. It follows that almost all finite graphs are characterized up to isomorphism by their $C^2_{\infty\omega}$-theories, hence also by their $C^2_{\omega\omega}$-theories.*

Immerman and Lander proved that not only is the stable colouring C^2-definable, but it exactly classifies vertices up to C^2-equivalence:

Theorem 2.25 (Immerman, Lander). *The equivalence relation $\sim$ associated with the stable colouring of finite graphs is equality of C^2-types of singletons. The associated pre-ordering $\preccurlyeq$ therefore is a pre-ordering with respect to C^2-types of single vertices.*

Sketch of Proof. Let $\mathfrak{G} = (V, E)$ be a graph. It suffices to show that $u \sim u'$ for $u, u' \in V$ implies that player **II** has a strategy in the infinite game on $(\mathfrak{G}, uu; \mathfrak{G}, u'u')$. Then $\sim$ is at least as fine as equality of C^2-types. It cannot be strictly finer because each $\sim$-class is $C^2_{\infty\omega}$-definable as we have seen in the proof of Lemma 2.21. We show that player **II** can maintain the following condition on game positions $(\mathfrak{G}, uv)$ and $(\mathfrak{G}, u'v')$:

$$(*)\quad u \sim u' \text{ and } v \sim v' \quad\text{ and }\quad \mathrm{atp}_{\mathfrak{G}}\big(u, v\big) = \mathrm{atp}_{\mathfrak{G}}\big(u', v'\big).$$

Let this condition be satisfied in the current stage $(\mathfrak{G}, uv; \mathfrak{G}, u'v')$. Assume without loss of generality that player **I** chooses to play in the second component, $j = 2$, and proposes $B \subseteq V$ as a subset over the first copy of $\mathfrak{G}$. Let the colour classes in $V/\sim$ be enumerated as $\alpha_1, \ldots, \alpha_l$. Split B into colour classes $B_i = B \cap \alpha_i$. Since $u \sim u'$ and since $\sim = \sim'$ is stationary with respect to a further colour refinement step, we have for all α_i:

$$\Big|\big\{w \mid Euw \wedge w \in \alpha_i\big\}\Big| = \Big|\big\{w' \mid Eu'w' \wedge w' \in \alpha_i\big\}\Big|.$$

It follows that also $\big|\{w \mid \neg Euw \wedge w \in \alpha_i\}\big| = \big|\{w' \mid \neg Eu'w' \wedge w' \in \alpha_i\}\big|$. Therefore **II** can choose subsets $B'_i \subseteq \alpha_i$ such that u' has exactly as many E-neighbours and non-neighbours in B'_i as u has in B_i. Let **II** put $B' = \bigcup_i B'_i$. If **I** now chooses for instance a neighbour of u' in B'_i, then **II** can answer with a neighbour of u from B_i. Thus $(*)$ is realized in the resulting stage again. □

2.2.4 A Variant Without Counting

There is also an inductively definable pre-ordering adapted to capture the refinement that corresponds to the moves in the ordinary pebble game for L^k. Its definition does not require cardinality comparison so that it turns out to be FP-definable. In fact, the rôle of cardinality comparisons in the colour refinement is taken by the boolean distinction whether or not there are neighbours (no matter how many) of respective kinds. Consider some colouring $c\colon V \to r$ on a k-graph. For the refinement step pass to a new colouring

$$c'\colon v \longmapsto \Big(c(v), \big(d^s_j(v)\big)_{1 \leqslant j \leqslant k, 0 \leqslant s < r}\Big),$$

$$\text{where}\qquad d^s_j(v) := \begin{cases} 0 & \text{if } \neg\exists w\big(E_j vw \wedge c(w) = s\big) \\ 1 & \text{if } \exists w\big(E_j vw \wedge c(w) = s\big). \end{cases}$$

Note that the entries in all but the first component are boolean values. These take the place of cardinalities in the colour refinement. The new colours are ordered lexicographically just as in the colour refinement. The corresponding refinement in the associated strict pre-orderings can easily be described in a form analogous to condition 2.1 on page 69.

Starting from a pre-ordered k-graph and applying this refinement procedure inductively, a limit pre-ordering is obtained. Let us call this resulting pre-ordering the *Abiteboul-Vianu colouring* of the pre-ordered k-graph.

In complete analogy with the proofs of Lemmas 2.21 and 2.22 above, we find that the Abiteboul-Vianu colouring of pre-ordered k-graphs is globally $L^2_{\infty\omega}$-definable as well as FP-definable.

We shall see in the next sections that the Abiteboul-Vianu colouring serves to construct global pre-orderings with respect to L^k-types just as the stable colouring serves to construct similar pre-orderings with respect to C^k-types. We have seen in Theorem 2.25 a first indication in this direction: the standard stable colouring of graphs provides a global pre-ordering of C^2-types of singletons. It may similarly be shown that the Abiteboul-Vianu colouring is a pre-ordering of L^2-types of singletons.

2.3 Order in the Analysis of the Games

The desired ordering with respect to types is obtained through an ordered classification of positions in the corresponding game. Formally the ordering of the quotients $A^k/\equiv^{\mathcal{L}}$ gets interpreted over each structure $\mathfrak{A}$ through a pre-ordering on the k-th power of the universe. The associated equivalence relation will be equality of types. We have seen a special case of this idea in Theorem 2.25. In the following we present the introduction of the desired pre-orderings in two different approaches, each with its specific advantages.

(a) The first view is an *internal* one in the sense that the pre-ordering is defined as a global relation on the game positions over each individual $\mathfrak{A}$ without reference to positions over other structures. This development is a direct application of the stable colouring to some k-graph associated with each individual $\mathfrak{A}$. From Section 2.2 we infer definability properties of the resulting pre-ordering as a global relation on $\mathrm{fin}[\tau]$.

(b) The other, and indeed more comprehensive, view defines the desired pre-ordering as a pre-ordering on $\mathrm{fin}[\tau; k]$, i.e. as a relation that serves to compare game positions over different structures. In this sense it involves considerations that are *external* to the individual structures. This is in good agreement, however, with the game analysis in terms of the equivalence relations $\approx$. These also primarily are equivalences over $\mathrm{fin}[\tau; k]$. Only their restrictions to the special case that both positions are over the same structure are global relations over $\mathrm{fin}[\tau]$.

Both views are presented in the following. The externally defined pre-ordering agrees with the internally defined one in restriction to each individual structure so that both views contribute to the understanding of the pre-ordering as a global relation. In order not to overburden notation we shall not distinguish between the two notationally. Wherever it matters it will be clear from context which view is intended.

We explicitly treat the case with counting quantifiers first and indicate the analogous treatment for the L^k in the sequel.

2.3.1 The Internal View

We introduce the desired orderings on $\mathrm{Tp}^{C^k}(\mathfrak{A};k) = A^k/\equiv^{C^k}$ as the stable colouring of some k-graph associated with $\mathfrak{A}$.

Let us fix some linear ordering $\leqslant_0$ on the finite set $\mathrm{Atp}(\tau;k)$ of atomic τ-types in k variables. This induces an initial pre-ordering $\preccurlyeq_0$ on the k-th power of the universe of any $\mathfrak{A} \in \mathrm{fin}[\tau]$:

$$\overline{a} \preccurlyeq_0 \overline{a}' \quad \text{if} \quad \mathrm{atp}_{\mathfrak{A}}\big(\overline{a}\big) \leqslant_0 \mathrm{atp}_{\mathfrak{A}}\big(\overline{a}'\big).$$

The associated equivalence relation $\sim_0$ is equality of atomic types, i.e. the above $\approx_0$. With any finite τ-structure $\mathfrak{A}$ we associate a k-graph that encodes the game positions over $\mathfrak{A}$ in the k-pebble game together with the fixed initial pre-ordering with respect to atomic types.

Definition 2.26. *With structures $\mathfrak{A} \in \mathrm{fin}[\tau]$ associate the following structures over universe A^k.*

(i) The game k-graph *of $\mathfrak{A}$, $\mathfrak{A}^{(k)}$. Its vocabulary $\tau^{(k)}$ consists of binary relations E_j, for $j = 1,\ldots,k$, and unary predicates P_θ for each atomic type $\theta \in \mathrm{Atp}(\tau;k)$. These are interpreted on A^k according to $E_j\overline{a}\,\overline{a}'$ if $\bigwedge_{i\neq j} a_i = a'_i$, and $P_\theta\overline{a}$ if $\mathrm{atp}_{\mathfrak{A}}\big(\overline{a}\big) = \theta$.*

$$\mathfrak{A}^{(k)} = \Big(A^k, \big(E_j\big)_{1\leqslant j\leqslant k}, \big(P_\theta\big)_{\theta\in\mathrm{Atp}(\tau;k)}\Big).$$

(ii) For the pre-ordered k-graph *of $\mathfrak{A}$, the identification of the individual atomic types is replaced be the pre-ordering $\preccurlyeq_0$ according to atomic types (as induced by $\leqslant_0$). The pre-ordered k-graph of $\mathfrak{A}$ is*

$$\Big(A^k, \big(E_j\big)_{1\leqslant j\leqslant k}, \preccurlyeq_0\Big).$$

The E_j encode in both structures the accessibility between positions over $\mathfrak{A}$ in a moves that are carried out over the j-th component. It is important to note that both the game k-graph and the pre-ordered k-graph of $\mathfrak{A}$ are *quantifier free interpreted* over the k-th power of $\mathfrak{A}$. Also, the pre-ordering $\preccurlyeq_0$ of the pre-ordered k-graphs is atomically definable over the game k-graphs.

From Section 2.2.2 we obtain a stable colouring $\preccurlyeq$ on the pre-ordered k-graphs.

Proposition 2.27. *The stable colouring of the pre-ordered k-graph of $\mathfrak{A}$ is a pre-ordering with respect to C^k-types: its associated equivalence relation is equality of C^k-types over A^k.*

Proof. Let the $\preccurlyeq_i$ be the stages in the generation of the stable colouring $\preccurlyeq$ on the associated k-graph. Let $\sim$ and the $\sim_i$ be the corresponding equivalence relations on A^k. The proposition is equivalent with the statement that $\sim$ coincides with $\approx$ over A^k. It suffices to show inductively that $\sim_i = \approx_i$ for all i, since we know that

$$\sim_i \xrightarrow{i\to\infty} \sim \qquad \text{and} \qquad \approx_i \xrightarrow{i\to\infty} \approx .$$

Agreement between $\sim_0$ and $\approx_0$ is clear from the definition.

Consider the refinement step in the generation of the stable colouring on the k-graph associated with $\mathfrak{A}$. Recall the inductive definition of the stages for the stable colouring, in particular the formula governing the refinement step for the associated equivalence relation from equation 2.2 on page 69:

$$\begin{array}{lll} \overline{a} \sim_{i+1} \overline{a}' & \text{if} & \overline{a} \sim_i \overline{a}' \text{ and for all } j \in \{1,\dots,k\} \text{ and all } \alpha \in A^k/\sim_i \\ & & \Big|\big\{\overline{b} \mid E_j\overline{a}\,\overline{b} \wedge \overline{b} \in \alpha\big\}\Big| = \Big|\big\{\overline{b} \mid E_j\overline{a}'\,\overline{b} \wedge \overline{b} \in \alpha\big\}\Big|. \end{array}$$

But obviously $\big|\{\overline{b} \in \alpha \mid E_j\overline{a}\,\overline{b} \wedge \overline{b} \in \alpha\}\big| = \big|\{b \in A \mid \overline{a}\tfrac{b}{j} \in \alpha\}\big|$ so that

$$\begin{array}{lll} \overline{a} \sim_{i+1} \overline{a}' & \text{if} & \overline{a} \sim_i \overline{a}' \text{ and for all } j \in \{1,\dots,k\} \text{ and all } \alpha \in A^k/\sim_i \\ & & \Big|\big\{b \in A \mid \overline{a}\tfrac{b}{j} \in \alpha\big\}\Big| = \Big|\big\{b \in A \mid \overline{a}'\tfrac{b}{j} \in \alpha\big\}\Big|. \end{array}$$

Comparing Proposition 2.15 for the inductive refinement step in the $\approx_i$ — and specializing to the case that both positions are over the same structure $\mathfrak{A}$ — it follows that $\sim_i = \approx_i$ implies $\sim_{i+1} = \approx_{i+1}$. This yields an inductive proof of the claim. □

Recall from Lemma 2.21 that the stable colouring of pre-ordered k-graphs is $C^2_{\infty\omega}$-definable. $\preccurlyeq$ is the stable colouring of a pre-ordered k-graph that itself is quantifier free interpreted over the k-th power of $\mathfrak{A}$. It follows with Lemma 1.50 that $\preccurlyeq$ is globally definable as a global relation over $\mathrm{fin}[\tau]$ in $C^{2k}_{\infty\omega}[\tau]$.

By Lemma 2.22 $\preccurlyeq$ is definable in $\mathrm{FP}(Q_{\mathrm{R}})$. Thus we have the following.

Theorem 2.28. *For each k there is a global pre-ordering $\preccurlyeq$ over the k-th power of the universe of structures in* $\mathrm{fin}[\tau]$*, such that*

(i) the associated equivalence relation is equality of C^k-types. Thus $\preccurlyeq$ is the quotient interpretation of a global linear ordering of the $A^k/\equiv^{C^k}$.
(ii) as a global relation over $\mathrm{fin}[\tau]$*, $\preccurlyeq$ is definable in $C^{2k}_{\infty\omega}[\tau]$ as well as in* $\mathrm{FP}(Q_{\mathrm{R}})[\tau]$*, fixed-point logic with the Rescher quantifier.*

2.3.2 The External View

Recall how the equivalence relation $\approx$ was introduced as a binary relation on $\mathrm{fin}[\tau;k]$. Together with its inductive stages $\approx_i$ it serves to analyze the equivalence of k-tuples over different structures. $\approx$ and the $\approx_i$ as global relations on structures in $\mathrm{fin}[\tau]$ merely are the restrictions of these externally defined relations. It is possible to treat $\preccurlyeq$ and its stages $\preccurlyeq_i$ under the same external view as pre-orderings not only on individual structures in $\mathrm{fin}[\tau]$, but on $\mathrm{fin}[\tau;k]$. In this view an inductive definition of the $\prec_i$ can be given as follows. We here choose the strict variants $\prec_i$ because their inductive definition is the formally more transparent one. $<_0$ is the strict variant of the fixed linear ordering $\leqslant_0$ on $\mathrm{Atp}(\tau;k)$.

$$\begin{array}{ll} (\mathfrak{A},\overline{a}) \prec_0 (\mathfrak{A}',\overline{a}') & \text{if} \quad \mathrm{atp}_{\mathfrak{A}}\big(\overline{a}\big) <_0 \mathrm{atp}_{\mathfrak{A}'}\big(\overline{a}'\big) \\ (\mathfrak{A},\overline{a}) \prec_{i+1} (\mathfrak{A}',\overline{a}') & \text{if} \\ \quad (\mathfrak{A},\overline{a}) \prec_i (\mathfrak{A}',\overline{a}') & \text{or} \\ \quad (\mathfrak{A},\overline{a}) \sim_i (\mathfrak{A}',\overline{a}') \text{ and } \big(\nu_j^\alpha(\mathfrak{A},\overline{a})\big) <_{\mathrm{lex}} \big(\nu_j^\alpha(\mathfrak{A}',\overline{a}')\big) & \\ \quad \text{where} \quad \nu_j^\alpha(\mathfrak{A},\overline{a}) = \Big|\big\{ b \in A \;\big|\; (\mathfrak{A},\overline{a}\tfrac{b}{j}) \in \alpha \big\}\Big|. & \end{array} \tag{2.4}$$

The indices (j,α) range over $\{1,\ldots,k\} \times \mathrm{fin}[\tau;k] \,/{\approx_i}$. The ordering of the index sets in the lexicographic comparison is chosen with dominant first component j. Note that the tuples involved in this comparison each only have a finite number of non-zero entries. Only types that are realized over $\mathfrak{A}$ or $\mathfrak{A}'$ enter non-trivially. Comparison with the inductive generation of the $\approx_i$ in Proposition 2.15 shows that the equivalence relations $\sim_i$ associated with the $\prec_i$ defined in this manner are indeed the $\approx_i$. It follows that the limit $\prec$ of the $\prec_i$ is a strict pre-ordering with respect to C^k-types over $\mathrm{fin}[\tau;k]$.

Lemma 2.29. *The pre-orderings $\prec_i$, as inductively defined on* $\mathrm{fin}[\tau;k]$ *according to equations 2.4, and their limit $\prec$ coincide in restriction to each individual $\mathfrak{A} \in$* $\mathrm{fin}[\tau]$ *with those defined through the stable colouring of the k-graph associated with $\mathfrak{A}$.*

Sketch of Proof. One need only specialize equations 2.4 to a single structure $\mathfrak{A} = \mathfrak{A}'$. The obvious equality $\big|\{ b \in A \mid (\mathfrak{A},\overline{a}\tfrac{b}{j}) \in \alpha \}\big| = \big|\{ \overline{b} \in A^k \mid E_j\overline{a}\,\overline{b} \wedge (\mathfrak{A},\overline{b}) \in \alpha \}\big|$ shows the agreement of the lexicographic comparison in 2.4 with that of the colour refinement over the k-graph associated with $\mathfrak{A}$, cf. equation 2.1 on page 69. This proves equality for the inductive stages and implies equality in the limits as well. □

This external view of $\preccurlyeq$ and the $\preccurlyeq_i$ really goes beyond the view of these as global relations on individual structures: it immediately shows that two C^k-types that are both realized in two different structures get ordered the same way in both structures.

Corollary 2.30. *As global relations on* fin$[\tau]$*, the* $\preccurlyeq$ *provide a coherent ordering with respect to* C^k*-types across all structures in* fin$[\tau]$*:*
if $\mathrm{tp}^{C^k}_{\mathfrak{A}}(\overline{a}_1) = \mathrm{tp}^{C^k}_{\mathfrak{A}'}(\overline{a}'_1)$ *and* $\mathrm{tp}^{C^k}_{\mathfrak{A}}(\overline{a}_2) = \mathrm{tp}^{C^k}_{\mathfrak{A}'}(\overline{a}'_2)$*, then* $\overline{a}_1 \preccurlyeq^{\mathfrak{A}} \overline{a}_2$ *if and only if* $\overline{a}'_1 \preccurlyeq^{\mathfrak{A}'} \overline{a}'_2$*.*

This is immediate here from Lemma 2.29. The same coherence claim can also be proved directly on the basis of the global definition of the individual pre-orderings. Note, however, that it does not follow directly from the fact that the associated equivalence relation is equality of C^k-types. Even though it is clear that whenever $\mathfrak{A}$ and $\mathfrak{A}'$ share even a single C^k-type they must be C^k-equivalent, coherent ordering of the types might a priori seem to require C^{2k}-equivalence.

2.3.3 The Analogous Treatment for L^k

We sketch the introduction of a pre-ordering with respect to L^k-types. An inductive characterization of the relation $\equiv^{L^k}$ or equality of L^k-types has been obtained in the analysis of the L^k-game. Recall Proposition 2.19 for the inductive generation of equivalences $\approx_i$ appropriate for the L^k-game. Their limit $\approx$ over fin$[\tau; k]$ is $\equiv^{L^k}$.

The desired pre-ordering, for which we also write $\preccurlyeq$, can once more be defined as a global relation internal to each individual structure, or externally as a pre-ordering on fin$[\tau; k]$ whose restriction to individual structures is the same as the former. As global relations internal to each $\mathfrak{A}$ the pre-ordering $\preccurlyeq$ and its stages $\preccurlyeq_i$ are obtained as the limit and the stages of the Abiteboul-Vianu colouring applied to the pre-ordered k-graphs associated with $\mathfrak{A}$. This immediately gives the analogous definability results as in the case of the C^k, cf. Theorem 2.28.

Theorem 2.31. *For each k there is a global pre-ordering* $\preccurlyeq$ *over the k-th power of the universe of structures in* fin$[\tau]$*, such that its associated equivalence relation is equality of* L^k*-types. This pre-ordering is obtained as the Abiteboul-Vianu colouring of the pre-ordered k-graphs associated with structures in* fin$[\tau]$*. As a global relation over* fin$[\tau]$*,* $\preccurlyeq$ *is definable in* $L^{2k}_{\infty\omega}[\tau]$ *as well as in* FP$[\tau]$*.*

The more general external version of $\preccurlyeq$ over fin$[\tau; k]$ is obtained in an inductive definition analogous to equations 2.4:

$$\begin{array}{ll}
(\mathfrak{A}, \overline{a}) \prec_0 (\mathfrak{A}', \overline{a}') & \text{if} \quad \mathrm{atp}_{\mathfrak{A}}(\overline{a}) <_0 \mathrm{atp}_{\mathfrak{A}'}(\overline{a}') \\
(\mathfrak{A}, \overline{a}) \prec_{i+1} (\mathfrak{A}', \overline{a}') & \text{if} \\
\quad (\mathfrak{A}, \overline{a}) \prec_i (\mathfrak{A}', \overline{a}') & \text{or} \\
\quad (\mathfrak{A}, \overline{a}) \sim_i (\mathfrak{A}', \overline{a}') & \text{and } \big(d^\alpha_j(\mathfrak{A}, \overline{a})\big) <_{\mathrm{lex}} \big(d^\alpha_j(\mathfrak{A}', \overline{a}')\big) \\
\quad \text{where} \quad d^\alpha_j(\mathfrak{A}, \overline{a}) := & \begin{cases} 0 & \text{if } \neg\exists b(\mathfrak{A}, \overline{a}\frac{b}{j}) \in \alpha \\ 1 & \text{if } \exists b(\mathfrak{A}, \overline{a}\frac{b}{j}) \in \alpha\,. \end{cases}
\end{array}$$

The indices (j, α) range over $\{1, \ldots, k\} \times \text{fin}[\tau; k] \,/\!\approx_i$.

Recall that $\approx_i$ is the i-th stage in the generation of $\approx$ — where now $\approx$ is $\equiv^{L^k}$ on $\text{fin}[\tau; k]$. In order to verify that indeed $\approx_i$ also is the equivalence relation associated with $\preccurlyeq_i$ as defined here, compare Proposition 2.19. In analogy with Lemma 2.29 it is shown that in restriction to individual structures this externally defined pre-ordering coincides with the one obtained internally. In particular, as a global relation on $\text{fin}[\tau]$, $\preccurlyeq$ is a coherent pre-ordering with respect to L^k-types.

Lemma 2.32. *As a global relation on* $\text{fin}[\tau]$ *the pre-ordering* $\preccurlyeq$ *obtained as the Abiteboul-Vianu colouring of the k-graphs of structures in* $\text{fin}[\tau]$ *provides a coherent ordering with respect to L^k-types across all structures in* $\text{fin}[\tau]$*: its associated equivalence relation is equality of L^k-types, and if* $\text{tp}^{L^k}_{\mathfrak{A}}(\overline{a}_1) = \text{tp}^{L^k}_{\mathfrak{A}'}(\overline{a}'_1)$ *and* $\text{tp}^{L^k}_{\mathfrak{A}}(\overline{a}_2) = \text{tp}^{L^k}_{\mathfrak{A}'}(\overline{a}'_2)$*, then* $\overline{a}_1 \preccurlyeq^{\mathfrak{A}} \overline{a}_2$ *if and only if* $\overline{a}'_1 \preccurlyeq^{\mathfrak{A}'} \overline{a}'_2$.

Sources and attributions. As pointed out above, the Fraïssé style analysis for finite variable logics in terms of back-and-forth systems is due to Barwise [Bar77], the introduction of the corresponding pebble games and their analysis to Immerman [Imm82]. For some more background on the finite variable fragments of first-order logic see also [Poi82]. The games for finitely many variables and counting quantifiers were introduced by Immerman and Lander in [IL90]. Cai, Fürer and Immerman applied these games in the analysis of their construction of non-isomorphic but C^k-equivalent graphs in [CFI89]. In this construction counting is, however, easily eliminated. A systematic analysis of the C^k-game over graphs is presented in [CFI92] and was independently developed in [GO93, Ott96a]. Cai, Fürer and Immerman attribute the underlying graph theoretic technique connected with the stable colouring in higher dimension to Lehman and Weisfeiler. The approach in [GO93, Ott96a] grew out of the generalization of the Abiteboul-Vianu approach to the case with counting. It should be remarked that the notions of a k-ary stable colouring underlying [GO93, Ott96a] — which is the one used here as well — differs in some technical details from the one attributed to Lehman and Weisfeiler in the work of Cai, Fürer and Immerman. Our k-ary variant is tuned exactly to yield a classification of k-tuples with respect to C^k; the other one rather corresponds to the classification of k-tuples with respect to types in C^{k+1}. Both ways have their merits but the difference has to be kept in mind to avoid confusion when comparing the statements. We find our convention more suitable in connection with definability issues concerning the pre-orderings with respect to types and the invariants to be introduced in the next chapter.

The work of Abiteboul and Vianu [AV91] is the essential source for the introduction of the definable ordered quotients $A^k/\equiv^{L^k}$ that will form the backbone of the invariants. An excellent presentation of the related results for the L^k-game in logical terms is given by Dawar, Lindell and Weinstein in [Daw93, DLW95].

3. The Invariants

We introduce *complete structural invariants* that classify finite relational structures up to C^k- and L^k-equivalence, respectively. These invariants are based on the definable pre-orderings with respect to C^k- and L^k-types obtained in the analysis of the games in the preceding chapter. The invariants are PTIME computable and inherit specific definability properties from the pre-orderings with respect to types. These definability properties and a close relationship with the fixed-point logics make the invariants extremely useful in investigations concerning fixed-point logics and complexity issues. This approach has been initiated and lead to success in the seminal work of Abiteboul and Vianu. They first introduced a kind of ordered invariants with respect to their model of relational computation and with this technique derived important results concerning the relationship between FP and PFP.

• In the introductory Section 3.1 we relate the concept of the proposed invariants to the abstract notion of complete invariants.

• Section 3.2 provides the definition of our C^k-invariants and states their fundamental definability properties.

• Section 3.3 similarly treats the invariants for L^k.

• In Section 3.4 we consider applications of the invariants to the analysis of fixed-point logics. A main point is the discussion of the Abiteboul-Vianu Theorem on the relation between FP and PFP. As far as the C^k-invariants are concerned, the corresponding considerations are of a preliminary nature here. This analysis will be pursued further in Chapter 4 where it becomes possible to link the C^k-invariants directly with fixed-point logic with counting. We include here a comparison between the C^k- and the L^k-invariants.

• In Section 3.5 it is indicated that — up to interpretability in powers — our invariants essentially reduce to the two-dimensional ones, i.e. to those for C^2 and L^2.

3.1 Complete Invariants for L^k and C^k

Recall from Definition 1.58 the notion of a complete invariant for an equivalence relation $\sim$: I is a complete invariant for $\sim$ if I classifies objects exactly up to $\sim$: $x \sim x'$ if and only if $I(x) = I(x')$. We apply this notion to the equivalence relations $\equiv^{\mathcal{L}}$, $\mathcal{L} = C^k_{\infty\omega}$ or $L^k_{\infty\omega}$. These may be regarded as equivalence relations on $\mathrm{fin}[\tau]$ as well as on the $\mathrm{fin}[\tau; r]$ for $r \leqslant k$. Accordingly we actually get two notions of complete invariants in each case.

Definition 3.1. *Let $\mathcal{L}$ be a logic. A functor I on* $\mathrm{fin}[\tau]$ *is a* complete $\mathcal{L}$-invariant *if*

$$\forall \mathfrak{A} \forall \mathfrak{A}' \quad \mathfrak{A} \equiv^{\mathcal{L}} \mathfrak{A}' \iff I(\mathfrak{A}) = I(\mathfrak{A}').$$

Similarly, I is a complete invariant for $\mathcal{L}$ on $\mathrm{fin}[\tau; r]$ *if for $(\mathfrak{A}, \overline{a})$ and $(\mathfrak{A}', \overline{a}')$ in* $\mathrm{fin}[\tau; r]$*: $(\mathfrak{A}, \overline{a}) \equiv^{\mathcal{L}} (\mathfrak{A}', \overline{a}') \Leftrightarrow I(\mathfrak{A}, \overline{a}) = I(\mathfrak{A}', \overline{a}')$.*

A computable complete invariant I is a corresponding mapping from $\mathrm{fin}[\tau]$ or $\mathrm{fin}[\tau; k]$ to some set S of objects with a standard encoding realized by an algorithm $\mathcal{A}\colon \mathrm{stan}[\tau] \to S$ or $\mathcal{A}\colon \mathrm{stan}[\tau; k] \to S$. Compare Definition 1.61. The C^k- and L^k-invariants considered in the following are PTIME computable and take as their values linearly ordered structures (or structures over standard domains n). One of the goals of this chapter is the following theorem.

Theorem 3.2. *There are* PTIME *computable complete $\mathcal{L}$-invariants for $\mathcal{L} = C^k_{\infty\omega}$ and $L^k_{\infty\omega}$.*

The backbones of the invariants are the ordered representations of the $A^k/\equiv^{\mathcal{L}}$ derived in the preceding chapter. On $\mathrm{fin}[\tau]$ we shall have

$$I_{\mathcal{L}}(\mathfrak{A}) = \Big(A^k \big/ {\equiv^{\mathcal{L}}}, \leqslant, \underbrace{\ldots\ldots}_{(*)} \Big),$$

where $\leqslant$ is the linear ordering of the quotient that is interpreted by the corresponding pre-ordering with respect to types over A^k. $(*)$ encodes additional combinatorial information so that exactly the structural information for $\mathcal{L}$-games over $\mathfrak{A}$ is retrievable from $I_{\mathcal{L}}(\mathfrak{A})$. This ensures that $I_{\mathcal{L}}(\mathfrak{A})$ comprises a complete description of the $\mathcal{L}$-theory of $\mathfrak{A}$ as required of a complete $\mathcal{L}$-invariant over $\mathrm{fin}[\tau]$.

By definition the rôle of the invariants over $\mathrm{fin}[\tau]$ is comparable to that of Scott sentences — they provide concise abstractions of the complete theories of structures. While Scott sentences may be regarded as the syntactic correlate of the games, the proposed invariants are structural correlates of the games. This structural aspect of the invariants has particular advantages in the further model theoretic applications: these invariants are adapted to simulate fixed-point processes over the original structures in a natural manner as we shall see in Section 3.4.

3.2 The C^k-Invariants

We introduce and discuss the C^k-invariants on fin[τ]. Complete invariants for C^k-equivalence on fin[$\tau; r$] for $r \leqslant k$ are easily derived as extensions of those on fin[τ]. The invariants are built upon the ordered quotients $(A^k/\equiv^{C^k}, \leqslant)$, where $\leqslant$ is the ordering induced by the pre-ordering $\preccurlyeq$ with respect to C^k-types, compare Theorem 2.28. In order to put the full information about the C^k-game over $\mathfrak{A}$ — or about the complete C^k-theory of $\mathfrak{A}$ — into the invariant, this ordered quotient is expanded with the following:

(i) Atomic components of types:
for each atomic type $\theta \in \mathrm{Atp}(\tau; k)$ the unary predicate P_θ is introduced on $A^k/\equiv^{C^k}$. For $\alpha \in A^k/\equiv^{C^k}$ put $\alpha \in P_\theta$ if $\mathrm{atp}_{\mathfrak{A}}(\overline{a}) = \theta$ for $\overline{a} \in \alpha$.

(ii) Accessibility:
for each j the binary predicate E_j, which encodes accessibility in moves concerning the j-th component, is transferred to $A^k/\equiv^{C^k}$ as follows. For $\alpha_1, \alpha_2 \in A^k/\equiv^{C^k}$ put $(\alpha_1, \alpha_2) \in E_j$ if for $\overline{a} \in \alpha_1$ there is some $b \in A$ with $\overline{a}\frac{b}{j} \in \alpha_2$.

(iii) Symmetries:
for each permutation ρ in the symmetric group S_k acting on $\{1, \ldots, k\}$ a binary predicate S_ρ is defined on $A^k/\equiv^{C^k}$ by:
$(\alpha_1, \alpha_2) \in S_\rho$ if $\rho(\overline{a}) \in \alpha_2$ for $\overline{a} \in \alpha_1$.

(iv) Multiplicities:
for each j a weight function ν_j from $A^k/\equiv^{C^k}$ to natural numbers is introduced which sends α to $\left|\left\{b \in A \mid \overline{a}\frac{b}{j} \in \alpha\right\}\right|$, for $\overline{a} \in \alpha$.

It has to be checked that the given definitions are independent of choices of representatives for the $\equiv^{C^k}$-classes on A^k. Recall that $A^k/\equiv^{C^k} = \mathrm{Tp}^{C^k}(\mathfrak{A}; k)$. Clearly for the P_θ, $\alpha \in P_\theta$ if $\alpha \models \theta$. For the others choose for each C^k-type α a $C^k_{\infty\omega}$-formula $\varphi_\alpha(\overline{x})$ that isolates α (cf. Lemma 1.33). Consider now the E_j. For any two C^k-types $\alpha, \beta \in \mathrm{Tp}^{C^k}(\mathfrak{A}; k)$, either $\alpha \models \exists x_j \varphi_\beta(\overline{x})$ or $\alpha \models \neg\exists x_j \varphi_\beta(\overline{x})$. Accessibility of a position of type β in the j-th component is thus determined by the type α of the given position. Similarly for the multiplicities ν_j. For each α there must be some natural number m such that $\alpha \models \exists^{=m} x_j \varphi_\alpha(\overline{x})$; this m is the value of $\nu_j(\alpha)$. The operation of $\rho \in S_k$ preserves C^k-equivalence so that the representation of ρ as a binary predicate on the quotient is also sound. Alternatively the operations ρ may actually be defined syntactically on C^k-types through an operation on the variable symbols.

Note that the information about atomic types, accessibility and permutations are encoded by relations over $A^k/\equiv^{C^k}$ whereas for the multiplicities we have to resort to external weight functions. The values of the ν_j over $A^k/\equiv^{C^k}$ are bounded by $|A|$.

Definition 3.3. *Let for each k and each fixed finite relational vocabulary τ, the C^k-invariant I_{C^k} be the functor which sends a finite τ-structure $\mathfrak{A}$ to the weighted linearly ordered structure*

$$I_{C^k}(\mathfrak{A}) \;=\; \Big(A^k/{\equiv^{C^k}}, \leqslant, (P_\theta), (E_j), (S_\rho); (\nu_j)\Big),$$

where θ ranges over $\mathrm{Atp}(\tau; k)$*, j over $\{1, \ldots, k\}$, and ρ over S_k. The interpretations of $\leqslant$, the P_θ, E_j, S_ρ and ν_j are as defined above.*

Obviously I_{C^k} is a PTIME computable functor. Formally we regard the *relational part* of $I_{C^k}(\mathfrak{A})$,

$$\big(A^k/{\equiv^{C^k}}, \leqslant, (P_\theta), (E_j), (S_\rho)\big),$$

as a relational structure on the standard universe of size $|A^k/\equiv^{C^k}| \leqslant |A|^k$ with its natural ordering. The weight functions ν_j take values in $\big\{1, \ldots, |A|\big\}$. $I_{C^k}(\mathfrak{A})$ as a whole may therefore in some canonical way be encoded as a relational structure over the standard universe of size $|A|$, if k-tuples are used to encode the elements of $A^k/\equiv^{C^k}$.

Remark 3.4. *We regard I_{C^k} as a* PTIME *functor that takes standard objects — namely canonical relational encodings of the weighted ordered structures $I_{C^k}(\mathfrak{A})$ over standard domains of size $|A|$ — as its values. The* size *of I_{C^k} is taken to be $|A|$.*[1]

The data encoded in $I_{C^k}(\mathfrak{A})$ are redundant in several respects. In the presence of the $\rho \in S_k$ it would in particular suffice to keep only one of the E_j and only one of the ν_j. For instance $\nu_{j_1} = \nu_{j_2} \circ \rho$ where ρ is the permutation exchanging j_1 and j_2. We keep this redundancy because the highly symmetric format is easier to handle in some applications.

It is slightly less obvious that also the S_ρ are PTIME computable (and hence FP-definable) from the complete set of the E_j and ν_j. To see this observe that the quotients $A^k/\approx_i$, that occur in the inductive generation of $\approx$, are all naturally interpreted over $A^k/\equiv^{C^k}$: the equivalence classes of the $\approx_i$ are unions of $\equiv^{C^k}$-classes, as $\equiv^{C^k}$ is a refinement of the $\approx$. At $\approx_0$-level, the classes are just the P_θ over $A^k/\equiv^{C^k}$. The operation of S_k on $\mathrm{Atp}(\mathfrak{A}; k) = A^k/\approx_0$ is trivially definable in this interpretation. $\rho \in S_k$ sends P_θ to $P_{\rho(\theta)}$, where $\rho(\theta)$ is obtained by operating with ρ^{-1} on the variables in θ. Inductively, in each refinement step, the operation of S_k on $A^k/\approx_{i+1}$ is determined by that on $A^k/\approx_i$. The refinement is governed by the values of the functions $\nu_j^\alpha(\overline{a}) = \big|\big\{b \in A \;/\; \overline{a}\frac{b}{j} \in \alpha\big\}\big|$ for $\alpha \in A^k/\approx_i$. And for these we obviously have

[1] The value $|A|$, rather than for instance $|A|^k$ or $\sum_j \sum_\alpha \nu_j(\alpha)$ is a matter of convention. The size of $I_{C^k}(\mathfrak{A})$ is naturally only determined up to polynomial transformations. The point is that the size of $I_{C^k}(\mathfrak{A})$, with weights taken into account, is polynomially related to the size of $\mathfrak{A}$, and not to $|A^k/{\equiv^{C^k}}|$.

$$\nu_j^\alpha(\overline{a}) = \nu_{\rho(j)}^{\rho(\alpha)}\big(\rho(\overline{a})\big) \quad \text{for all } \rho \in S_k.$$

Furthermore, even the ordering $\leqslant$ is PTIME computable form the remaining data on $A^k/\equiv^{C^k}$, since the entire refinement process in the generation of the $\preccurlyeq_i$ can also be simulated over $\big(A^k/\equiv^{C^k}, (P_\theta), (E_j); (\nu_j)\big)$.

These facts are stated for future reference in the following remark.

Remark 3.5. *The ordering $\leqslant$ and the interpretations of the S_ρ in $I_{C^k}(\mathfrak{A})$ are* PTIME *computable from the reduct of the I_{C^k} to vocabulary consisting only of the E_j and the P_θ together with the weight functions ν_j.*

The relational part of the I_{C^k} gets naturally interpreted over the original structures as a quotient over the k-th power. By definition it *is* the quotient of the k-th power of the universe with respect to $\equiv^{C^k}$. More precisely, we get the following. Recall that $\mathrm{FP}(Q_\mathrm{R})$ is fixed-point logic with the Rescher quantifier.

Proposition 3.6. *The relational part of $I_{C^k}(\mathfrak{A})$, i.e. the relational structure $\big(A^k/\equiv^{C^k}, \leqslant, (P_\theta), (E_j), (S_\rho)\big)$ is $\mathrm{FP}(Q_\mathrm{R})$-interpretable as a quotient over the k-th power of $\mathfrak{A}$. Moreover, the weights ν_j are the cardinalities of atomically definable predicates in this interpretation.*

Sketch of Proof. The intended interpretation is straightforward since I_{C^k} is defined as a quotient on the k-th power. $\mathrm{FP}(Q_\mathrm{R})$-definability of the relational part is also obvious. $\mathrm{FP}(Q_\mathrm{R})$ is needed to define the equivalence relation $\equiv^{C^k}$ for the quotient and for the linear ordering $\leqslant$ on this quotient; this is just $\mathrm{FP}(Q_\mathrm{R})$-definability of the pre-ordering $\preccurlyeq$ as stated in Theorem 2.28. The P_θ, E_j and S_ρ are in fact first-order interpretable relative to the interpreted $\big(A^k/\equiv^{C^k}, \leqslant\big)$. The ν_j finally are defined in terms of this interpretation over A^k according to $\nu_j(\alpha) = \big|\{\overline{b} \in \alpha | E_j\overline{a}\overline{b}\}\big|$ for any $\overline{a} \in \alpha$. □

It remains to establish the I_{C^k} as complete invariants for C^k on $\mathrm{fin}[\tau]$.

Theorem 3.7. *The functor I_{C^k} is a complete $C^k_{\infty\omega}$-invariant on* $\mathrm{fin}[\tau]$*. It classifies finite τ-structures exactly up to equivalence in C^k:*

$$\forall\mathfrak{A}\forall\mathfrak{A}' \quad I_{C^k}(\mathfrak{A}) = I_{C^k}(\mathfrak{A}') \iff \mathfrak{A} \equiv^{C^k} \mathfrak{A}'.$$

Proof. By Lemma 1.34 $\mathfrak{A} \equiv^{C^k} \mathfrak{A}'$ if and only if $\mathfrak{A}$ and $\mathfrak{A}'$ realize exactly the same C^k-types. This is used in the proof.
i) Assume first that $\mathfrak{A} \equiv^{C^k} \mathfrak{A}'$, so that $\mathfrak{A}$ and $\mathfrak{A}'$ realize exactly the same C^k-types. The crucial fact for the proof that $I_{C^k}(\mathfrak{A}) = I_{C^k}(\mathfrak{A}')$ is that these types get ordered in exactly the same way by $\preccurlyeq$ over $\mathfrak{A}$ and $\mathfrak{A}'$. This follows from the global view of $\preccurlyeq$ on $\mathrm{fin}[\tau; k]$ as expressed in Lemma 2.29 and Corollary 2.30. It follows that the natural isomorphism between $\big(A^k/\approx, \leqslant\big)$ and $\big(A'^k/\approx, \leqslant\big)$ as ordered structures is the identity function on C^k-types. Thus $I_{C^k}(\mathfrak{A}) = I_{C^k}(\mathfrak{A}')$ follows directly from the definitions, since we have seen

above that all the extra information encoded in the invariants is determined by the constituent types.
ii) For the converse implication it suffices to prove that the C^k-type corresponding to an element $\alpha \in I_{C^k}(\mathfrak{A})$ can be recovered from the invariant. Recall that the universe $A^k/\equiv^{C^k}$ of $I_{C^k}(\mathfrak{A})$ is the set $\mathrm{Tp}^{C^k}(\mathfrak{A};k)$ of C^k-types realized over $\mathfrak{A}$. The claim is clear at the atomic level because of the unary predicates P_θ. Inductively assume that for each formula $\varphi(\overline{x}) \in C^k_{\infty\omega}$ of quantifier rank at most i the subset $\underline{\varphi} := \{\alpha \in A^k/\equiv^{C^k} \mid \varphi \in \alpha\}$ has been determined as a subset of $I_{C^k}(\mathfrak{A})$. Without loss of generality consider a formula $\exists^{\geqslant m} x_j \varphi(\overline{x})$ with φ of quantifier rank at most i for the inductive step. It follows from the definition of the E_j and ν_j that $(\exists^{\geqslant m} x_j \varphi) \in \alpha$ if and only if

$$\sum_{(\alpha,\alpha')\in E_j, \alpha' \in \underline{\varphi}} \nu_j(\alpha') \ \geqslant \ m\,.$$

Therefore $I_{C^k}(\mathfrak{A})$ fully determines the set of C^k-types realized in $\mathfrak{A}$, and thus the complete C^k-theory of $\mathfrak{A}$. □

The proof also shows that classification up to $\equiv^{C^k}$ naturally extends from structures in $\mathrm{fin}[\tau]$ to structures with parameters, in particular to the classification of $\mathrm{fin}[\tau;k]$ up to $\equiv^{C^k}$. This is expressed in the following corollary.

Corollary 3.8. *The following extension of the I_{C^k} to $\mathrm{fin}[\tau;k]$ provides a complete invariant for C^k on $\mathrm{fin}[\tau;k]$:*

$$(\mathfrak{A},\overline{a}) \longmapsto \Big(I_{C^k}(\mathfrak{A}), [\overline{a}]\Big),$$

where $[\overline{a}]$ is that element of $I_{C^k}(\mathfrak{A})$ representing $\mathrm{tp}^{C^k}_{\mathfrak{A}}(\overline{a})$. In terms of the interpretation of the relational part of $I_{C^k}(\mathfrak{A})$ over $\mathfrak{A}$ it is just the equivalence class of $\overline{a}$ as induced by $\preccurlyeq$. The extended invariants share all the above-mentioned definability and interpretability properties with the standard ones.

Embedding A^r into A^k and $\mathrm{Tp}^{\mathcal{L}}(\mathfrak{A};r)$ into $\mathrm{Tp}^{\mathcal{L}}(\mathfrak{A};k)$ for $r \leqslant k$ via $(a_1,\dots,a_r) \mapsto (a_1,\dots,a_1,a_1,\dots,a_r)$, with $k-r$ additional entries of a_1 as usual, we similarly obtain complete invariants for C^k on $\mathrm{fin}[\tau;r]$ for all $r \in \{1,\dots,k\}$.

Corollary 3.9. *C^k-equivalence on $\mathrm{fin}[\tau]$ as well as on $\mathrm{fin}[\tau;k]$ is in* PTIME. *For any finite relational τ there are* PTIME *algorithms that decide for $\mathfrak{A}$ and $\mathfrak{A}'$ in $\mathrm{fin}[\tau]$ whether $\mathfrak{A} \equiv^{C^k} \mathfrak{A}'$ and for $(\mathfrak{A},\overline{a})$ and $(\mathfrak{A}',\overline{a}')$ in $\mathrm{fin}[\tau;k]$ whether $(\mathfrak{A},\overline{a}) \equiv^{C^k} (\mathfrak{A}',\overline{a}')$.*

These algorithms need merely evaluate the corresponding invariants and check for equality. C^k-equivalence of finite relational structures can also be shown to be definable in the extension of fixed-point logic by the Härtig quantifier — when suitably formalized as a query C^k-EQ.

Let to this end τ' consist of τ together with two new unary predicate symbols U_1 and U_2. Pairs of structures in fin$[\tau]$ are naturally encoded as τ'-structures in which the U_i separate the universe into two disjoint subsets for the universes of the individual τ-structures. Correspondingly put

$$C^k\text{-EQ} = \Big\{ \big(\mathfrak{B}, U_1, U_2\big) \in \text{fin}[\tau \dot\cup \{U_1, U_2\}] \;\Big|\; B = U_1 \dot\cup U_2, \mathfrak{B} \restriction U_1 \equiv^{C^k} \mathfrak{B} \restriction U_2 \Big\}.$$

It is obvious that this class is definable in FP(Q_{R}), fixed-point logic with the Rescher quantifier. FP(Q_{R}) interprets the relational parts of the I_{C^k} of the individual structures over the pair structure. Whether two interpreted linearly ordered relational structures are isomorphic is even FP-definable. For agreement also in the weight functions it suffices to check equalities of the cardinalities that define their values, and this can be done in a fixed-point process that uses the Rescher quantifier for the individual cardinality comparisons.

Now this can be strengthened to definability in the weaker extension of FP by the Härtig quantifier, for cardinality equality (compare Definition 1.53). In fact the invariants themselves as ordered structures need not actually be evaluated. We may instead directly consider the inductive generation of the stages $\approx_i$ in the generation of $\approx\, = \,\equiv^{C^k}$ over positions in $\mathfrak{A}$ and $\mathfrak{A}'$.

Checks for cardinality equality suffice for a fixed-point process whose stages are the $\approx_i$ in restriction to $\mathfrak{A}$ and $\mathfrak{A}'$. Compare Proposition 2.15 where the refinement step in the inductive definition of the $\approx_i$ is formalized in terms of cardinality equalities. (Formally one should use the complements of the $\approx_i$ to make the inductive process increasing.)

$\mathfrak{A} \equiv^{C^k} \mathfrak{A}'$ if the restriction of $\approx$ to $A^k \times A'^k$ induces an isomorphism between $\big(A^k/\approx, (P_\theta), (E_j); (\nu_j)\big)$ and $\big(A'^k/\approx, (P_\theta), (E_j); (\nu_j)\big)$. As these quotients and the candidate isomorphism between them are FP(Q_{H})-interpretable, FP(Q_{H}) also suffices to check the isomorphism property: Q_{H} is here used again for equality checks on the weights. We thus get the following.

Proposition 3.10. *C^k-equivalence of structures is definable in* FP(Q_{H}), *fixed-point logic with the Härtig quantifier, in the sense that the above class C^k-EQ of encodings of pairs of C^k-equivalent structures is definable in* FP(Q_{H}).

3.3 The L^k-Invariants

We sketch the introduction of the corresponding complete invariants I_{L^k} for the L^k. These are based on the pre-orderings with respect to L^k-type as characterized in Theorem 2.31 of the preceding chapter. The ordered quotients $\big(A^k/\equiv^{L^k}, \leqslant\big)$ are augmented by exactly the same relational expansions as in the case of I_{C^k}. There are, of course, no numerical weights to be kept here.

Definition 3.11. *For each k and each fixed finite relational vocabulary τ, the L^k-invariant I_{L^k} is the following* PTIME *computable functor which sends a finite τ-structure $\mathfrak{A}$ to the linearly ordered structure*

$$I_{L^k}(\mathfrak{A}) \;=\; \Big(A^k/{\equiv^{L^k}}, \leqslant, (P_\theta), (E_j), (S_\rho)\Big),$$

where θ ranges over $\mathrm{Atp}(\tau; k)$, *j over $\{1, \dots, k\}$, and ρ over S_k.*

$\leqslant$ is the linear ordering induced by the pre-ordering with respect to L^k-types (Theorem 2.31), the P_θ, E_j and S_ρ are defined exactly as for I_{C^k}:

- *P_θ contains those $\alpha \in A^k/\equiv^{L^k} = \mathrm{Tp}^{L^k}(\mathfrak{A}; k)$, for which $\alpha \models \theta$;*
- *$(\alpha_1, \alpha_2) \in E_j$ if for $\overline{a} \in \alpha_1$ there is some $b \in A$ such that $\overline{a}\frac{b}{j} \in \alpha_2$;*
- *$(\alpha_1, \alpha_2) \in S_\rho$ if for $\overline{a} \in \alpha_1$ the permuted tuple $\rho(\overline{a})$ is in α_2.*

Formally $I_{L^k}(\mathfrak{A})$ is regarded as a relational structure on the standard universe of size $\left|A^k/\equiv^{L^k}\right|$ with its natural ordering.

As with the I_{C^k} above it would suffice to keep one of the E_j because the others are definable from any particular one with the help of the S_ρ. And again, in the presence of all the E_j, the S_ρ and the ordering $\leqslant$ are PTIME computable from the remaining data in I_{L^k} (compare Remark 3.5).

In analogy with Proposition 3.6 for the I_{C^k} we here obtain the following.

Proposition 3.12. *$I_{L^k}(\mathfrak{A})$ is* FP*-interpretable as a quotient over the k-th power of $\mathfrak{A}$.*

And of course the I_{L^k} are complete invariants for L^k.

Theorem 3.13. *The functor I_{L^k} is a complete invariant for $L^k_{\infty\omega}$ on* $\mathrm{fin}[\tau]$. *It classifies finite τ-structures exactly up to equivalence in L^k:*

$$\forall\mathfrak{A}\forall\mathfrak{A}' \quad I_{L^k}(\mathfrak{A}) = I_{L^k}(\mathfrak{A}') \iff \mathfrak{A} \equiv^{L^k} \mathfrak{A}'.$$

The proof can be given along exactly the same lines as that for Theorem 3.7 with the obvious simplifications. Extensions of the I_{L^k} to complete invariants on the $\mathrm{fin}[\tau; r]$ for $r \leqslant k$ are obtained as in Corollary 3.8.

Corollary 3.14. *L^k-equivalence over* $\mathrm{fin}[\tau]$ *as well as over* $\mathrm{fin}[\tau; k]$ *can be checked through evaluation of the corresponding invariants and hence is in* PTIME.

We get more, namely FP-definability of L^k-equivalence as a query on pair structures, a result due to Kolaitis and Vardi [KVa92b]. Putting

$$L^k\text{-EQ} \;=\; \Big\{(\mathfrak{B}, U_1, U_2) \in \mathrm{fin}[\tau \dot\cup \{U_1, U_2\}] \;\Big|\; B = U_1 \dot\cup U_2, \mathfrak{B} \restriction U_1 \equiv^{L^k} \mathfrak{B} \restriction U_2\Big\},$$

we obtain the following corollary.

Corollary 3.15 (Kolaitis, Vardi). *L^k-equivalence is* FP*-definable in the sense that the class L^k-EQ is definable in* FP*. In particular L^k-equivalence is in* PTIME*.*

Sketch of Proof. The invariants I_{L^k} are FP-interpretable and isomorphism of embedded linearly ordered structures is obviously in FP. The claim follows with closure of FP under generalized interpretations. □

3.4 Some Applications

3.4.1 Fixed-Points and the Invariants

From Lemma 1.29 we know that the fixed-point logics FP and PFP are sublogics of $L^{\omega}_{\infty\omega}$. For any fixed-point process there is some k such that this fixed-point process and its stages are all $L^k_{\infty\omega}$-definable. But this implies in particular that this fixed-point process does not distinguish between $L^k_{\infty\omega}$-equivalent tuples. In other words, the generation of the fixed point on $\mathfrak{A}$ can faithfully be represented on the quotient $A^k/\equiv^{L^k}$. This observation is the key to important insights into the nature of FP and PFP in relation to computational complexity that are due to Abiteboul and Vianu. In this first part we present the technical basis.

Let $\varphi(Z_1,\dots,Z_l,\overline{x}) \in L^k_{\omega\omega}[\tau]$ be free in the indicated variables. Assume that the arity of the Z_i is at most k and that $\overline{x}$ is a tuple of (at most k) distinct variables. We want show that there is a first-order formula $\underline{\varphi}$ that captures the semantics of φ over the quotients $A^k/\equiv^{L^k}$, more precisely over the $I_{L^k}(\mathfrak{A})$. To make this precise we introduce some ad-hoc conventions. Predicates of arity k are naturally representable over A^k as unary predicates. For predicates of arity $1 \leqslant r < k$ we adopt a representation via the passage from $R \subseteq A^r$ to $R' := \big\{(\underbrace{a_1,\dots,a_1}_{k-r},a_1,\dots,a_r) \;\big|\; Ra_1\dots a_r\big\} \subseteq A^k$.

Since R is first-order definable from R' and vice versa, we may in particular restrict second-order parameters Z_i as in φ above to arity k rather than $r_i \leqslant k$. The same convention is applied to global relations. In our considerations about $L^k_{\infty\omega}$-definable queries we may here restrict attention to global relations of arity k. Boolean queries can be represented by these as well if we identify 0 with $\emptyset \subseteq A^k$ and 1 with the full predicate A^k. This translation, too, is sound up to first-order interdefinability.

For $\mathfrak{A} \in \text{fin}[\tau]$, call $R \subseteq A^k$ *L^k-admissible* if it is a union of $\equiv^{L^k}$-classes over A^k. Thus by Lemma 1.33, $R \subseteq A^k$ is L^k-admissible if R is $L^k_{\infty\omega}$- (and hence also $L^k_{\omega\omega}$-) definable over $\mathfrak{A}$. Note that we are here talking about definability over an individual structure, not about definability of global relations. Any L^k-admissible $R \subseteq A^k$ is faithfully represented over $A^k/\equiv^{L^k}$ by a unary predicate

$$\underline{R} = \{\alpha \in A^k/\equiv^{L^k} \mid \overline{a} \in \alpha \Rightarrow \overline{a} \in R\}.$$

With this translation for admissible interpretations of free second-order variables we obtain a uniform translation from L^k-formulae over fin$[\tau]$ to $L^2_{\omega\omega}$-formulae over the I_{L^k} as follows. Recall that if $\varphi(Z_1,\dots,Z_l,\overline{x})$ is in free variables Z_i and $\overline{x}$ as indicated we write $\varphi[\mathfrak{A},W_1,\dots,W_l]$ for the predicate defined by φ in variables $\overline{x}$ over $\mathfrak{A}$ if the Z_i are interpreted by predicates W_i over $\mathfrak{A}$.

Lemma 3.16. *Let $\varphi(Z_1,\dots,Z_l,x_1,\dots,x_k) \in L^k_{\omega\omega}[\tau]$ with second-order variables Z_i of arity k. Then there is an $L^2_{\omega\omega}$-formula $\underline{\varphi}(\underline{Z}_1,\dots,\underline{Z}_l,x)$ in the language of the I_{L^k} and with unary second-order variables $\underline{Z}_i$ that uniformly captures φ over the I_{L^k} in the following sense. For all $\mathfrak{A} \in$ fin$[\tau]$ and all L^k-admissible interpretations W_i for the Z_i over $\mathfrak{A}$:*

$$\underline{\varphi[(\mathfrak{A},W_1,\dots,W_l]} = \underline{\varphi}\big[I_{L^k}(\mathfrak{A}),\underline{W}_1,\dots,\underline{W}_l\big].$$

Proof. The proof is a straightforward induction over formulae φ. Assume without loss of generality that there is just one second-order variable Z and that each Z-atom in φ is in a tuple of mutually distinct variables (otherwise pass for instance from $Zx_1x_1\dots$ to $\exists x_2(x_2 = x_1 \wedge Zx_1x_2\dots)$).
i) Consider atomic φ. Let φ be a Z-atom of the admitted kind. Then $\varphi = Zx_{\rho(1)}\dots x_{\rho(k)}$ for some $\rho \in S_k$. The formula $\underline{\varphi}(\underline{Z},x) = \exists y\big(S_\rho xy \wedge \underline{Z}y\big)$ is as desired. If φ is an atom not involving Z then it is equivalent with a finite disjunction over atomic τ-types. These translate into a disjunction over formulae $P_\theta x$ for the corresponding $\theta \in \mathrm{Atp}(\tau;k)$.
ii) Boolean operations carry over trivially.
iii) It remains to consider existential quantification. Let $\varphi = \exists x_j\psi$ and assume that $\underline{\psi}\big(\underline{Z},x\big)$ is as desired for ψ. Let $\underline{\psi}\big(\underline{Z},y\big)$ be the result of exchanging x and y throughout $\underline{\psi}$. Then the formula $\underline{\varphi}(\underline{Z},x) = \exists y\big(E_jxy \wedge \underline{\psi}\big(\underline{Z},y\big)\big)$ is an adequate translation of φ. □

It follows immediately that fixed-point processes over fin$[\tau]$ translate into corresponding fixed-point processes over the I_{L^k}.

Lemma 3.17. *Let $\varphi(Z_1,\dots,Z_l,\overline{x}) \in$ PFP$[\tau]$. Then for sufficiently large k there is a* PFP*-formula $\underline{\varphi}\big(\underline{Z}_1,\dots,\underline{Z}_l,x\big)$ in the language of the I_{L^k} and with unary second-order variables $\underline{Z}_i$ that uniformly captures φ over the I_{L^k}. For all $\mathfrak{A} \in$ fin$[\tau]$ and all admissible interpretations W_i for the Z_i over $\mathfrak{A}$:*

$$\underline{\varphi[(\mathfrak{A},W_1,\dots,W_l]} = \underline{\varphi}\big[I_{L^k}(\mathfrak{A}),\underline{W}_1,\dots,\underline{W}_l\big].$$

The same holds of FP *in place of* PFP.

Sketch of Proof. The proof is obvious on the basis of the last lemma. Formally it is by induction on PFP-formulae. The PFP-step is as follows. Assume $\varphi = [\mathrm{PFP}_{X,\overline{x}}\psi]\overline{x}$ and disregard for convenience second-order parameters.

By the inductive hypothesis there is a PFP-formula $\underline{\psi}(\underline{X}, x)$ such that for appropriate k: $\underline{\psi[\mathfrak{A}, X]} = \underline{\psi}\big[I_{L^k}(\mathfrak{A}), \underline{X}\big]$ for all L^k-admissible X.

It follows inductively that the stages X_i in any fixed-point generation based on ψ are L^k-admissible predicates: the empty predicate is L^k-admissible and $\psi[\mathfrak{A}, X]$ is L^k-admissible for L^k-admissible X as it admits a representation over $A^k/\equiv^{L^k}$ through $\underline{\psi}\big[I_{L^k}(\mathfrak{A}), \underline{X}\big]$.

Let the $\underline{X}_i$ be the representations of the X_i. It is obvious that the $\underline{X}_i$ are the stages of a partial fixed-point process over I_{L^k} that is induced by $\underline{\psi}$. It follows that the partial fixed-point of the stages $\underline{X}_i$ over I_{L^k} is the representation of $\mathrm{PFP}_{X,\overline{x}}\psi$: $\underline{[\mathrm{PFP}_{X,\overline{x}}\psi]} = [\mathrm{PFP}_{\underline{X},x}\underline{\psi}]$. ☐

The C^k-invariants behave much like the L^k-invariants. All the information expressed in the L^k-invariant about $A^k/\equiv^{L^k}$ is expressed by the C^k-invariant about the finer representation $A^k/\equiv^{C^k}$. It is immediate therefore that the statement of the last lemma carries over to I_{C^k} in place of I_{L^k}. We state it without (C^k-admissible) second-order parameters, merely for notational convenience.

Corollary 3.18. *Let $\varphi \in \mathrm{PFP}[\tau]$, respectively $\mathrm{FP}[\tau]$. Then for sufficiently large k there is a PFP-formula, respectively FP-formula $\underline{\varphi}(x)$ in the language of the I_{C^k} that uniformly captures φ over the I_{C^k} in the sense that for all $\mathfrak{A} \in \mathrm{fin}[\tau]$: $\underline{\varphi[\mathfrak{A}]} = \underline{\varphi}\big[I_{C^k}(\mathfrak{A})\big]$.*

I_{C^k} contains numerical information encoded in the weight functions ν_j. In an extension of the statement of the last corollary we thus get that for instance the Härtig quantifier can also be captured. This will become useful later. Recall that the Härtig quantifier Q_{H} expresses cardinality equality of two definable unary predicates, cf. Definition 1.53.

Lemma 3.19. *Let $\varphi \in \mathrm{PFP}(Q_{\mathrm{H}})[\tau]$. Then for sufficiently large k there is a PFP-formula $\underline{\varphi}(x)$ in the language of the I_{C^k} that captures φ over the I_{C^k} in the sense that for all $\mathfrak{A} \in \mathrm{fin}[\tau]$: $\underline{\varphi[\mathfrak{A}]} = \underline{\varphi}\big[I_{C^k}(\mathfrak{A})\big]$.*[2]

The same holds of $\mathrm{FP}(Q_{\mathrm{H}})$ and FP in place of $\mathrm{PFP}(Q_{\mathrm{H}})$ and PFP.

Sketch of Proof. Above the proofs of Lemma 3.16 and 3.17 we only need to show that an application of the Härtig quantifier carries over to the representation on I_{C^k}. Let $\varphi = Q_{\mathrm{H}}\big((x_j;\psi_1);(x_{j'};\psi_2)\big)$. Semantically this formula says that $\big|\{x_j|\psi_1\}\big| = \big|\{x_{j'}|\psi_2\}\big|$. Assume that there are PFP-formulae $\underline{\psi_i}(x)$ satisfying the claim of the lemma for appropriate k. Then

$$\big|\{x_j|\psi\}\big| = \textstyle\sum_{x\in\underline{\psi}} \nu_j(x)$$

is a number whose value is PTIME computable from the unary predicate $\underline{\psi}$ over the ordered domain of I_{C^k}. Hence this value is fixed-point definable over I_{C^k}, and so is equality of two such values. ☐

[2] Compare Definition 3.3 and Remark 3.4. It is essential that we consider the relational encoding of the full invariants, the weights ν_j inclusive.

3.4.2 The Abiteboul-Vianu Theorem

For the L^k we can already demonstrate the power of the invariants in the analysis of FP and PFP in relation to computational complexity. This leads to a theorem of Abiteboul and Vianu which is one of the major results in the field.

Definition 3.20. *Let* $\mathrm{PTIME}(I_{L^k})$*, respectively* $\mathrm{PSPACE}(I_{L^k})$*, stand for the class of all queries that are* PTIME*, respectively* PSPACE *computable in terms of the* I_{L^k}*. More precisely for instance for* $\mathrm{PTIME}(I_{L^k})$*:*

- *(i) a boolean query* Q *on* $\mathrm{fin}[\tau]$ *is in* $\mathrm{PTIME}(I_{L^k})$ *if membership of* $\mathfrak{A}$ *in* Q *is a* PTIME *property of* $I_{L^k}(\mathfrak{A})$*.*
- *(ii) an* r*-ary query* R *on* $\mathrm{fin}[\tau]$ *for* $r \leqslant k$ *is in* $\mathrm{PTIME}(I_{L^k})$ *if membership of* $\overline{a}$ *in* $R^{\mathfrak{A}}$ *is a* PTIME *property of* $I_{L^k}(\mathfrak{A}, \overline{a})$*. Here* I_{L^k} *stands for the extension to an invariant on* $\mathrm{fin}[\tau; r]$*.*

Equivalently, a query is in $\mathrm{PTIME}(I_{L^k})$ or $\mathrm{PSPACE}(I_{L^k})$ if it is $L^k_{\infty\omega}$-definable (and therefore its values $R^{\mathfrak{A}}$ will in particular be L^k-admissible over $\mathfrak{A}$) and if the natural representations $\underline{R}^{\mathfrak{A}}$ of $R^{\mathfrak{A}}$ over $A^k/\equiv^{L^k}$ are PTIME or PSPACE computable over the $I_{L^k}(\mathfrak{A})$.

Note that these classes are recursively presentable. The PTIME or PSPACE algorithms featuring in the definition are not subject to any semantic constraints: unlike the original input structures $\mathfrak{A}$, the $I_{L^k}(\mathfrak{A})$ are objects with standard encodings.

$\mathrm{PTIME}(I_{L^k})$ and $\mathrm{PSPACE}(I_{L^k})$ are natural classes under the following view. Consider the case of boolean queries $Q \subseteq \mathrm{fin}[\tau]$. We identify Q with its characteristic functor $\chi\colon \mathrm{fin}[\tau] \to \{0, 1\}$ which is subject to the condition of invariance under isomorphism. Q is $L^k_{\infty\omega}$-definable if and only if χ is in fact $\equiv^{L^k}$-invariant. This is equivalent with the existence of a presentation of χ as $\chi = \chi^* \circ I_{L^k}$ for a boolean valued mapping χ^*. Note that χ^* is defined on a set of objects with standard encodings and is not subject to any additional constraints. The same considerations apply to k-ary queries, which we may identify with isomorphism invariant boolean functors on $\mathrm{fin}[\tau; k]$. $\mathrm{PTIME}(I_{L^k})$ and $\mathrm{PSPACE}(I_{L^k})$ consist exactly of those queries which are presentable by $\chi = \chi^* \circ I_{L^k}$ with PTIME or PSPACE computable functions χ^*.

It follows from the Theorems of Immerman and Vardi (Theorem 1.24) and of Abiteboul, Vardi and Vianu (Theorem 1.25) that these classes are semantically equivalent with logical systems based on FP and PFP:

$$\begin{aligned} \mathrm{PTIME}(I_{L^k}) &\equiv \mathrm{FP}(I_{L^k}), \\ \mathrm{PSPACE}(I_{L^k}) &\equiv \mathrm{PFP}(I_{L^k}). \end{aligned}$$

The logics on the right-hand side consist of those formulae that are obtained as FP- or PFP-formulae applied to the FP-definable interpretations of I_{L^k} as a quotient over the k-th power. Using the fact that I_{L^k} itself is

FP-interpretable and the closure properties of FP and PFP with respect to interpretations, Lemma 1.49, $\mathrm{FP}(I_{L^k})$ and $\mathrm{PFP}(I_{L^k})$ are seen to be fragments of FP and PFP, respectively. They could obviously be characterized in purely syntactic terms if one so wishes.

Lemma 3.21. *The following semantic equivalences hold on the class of all finite relational structures:*

$$\begin{array}{rcccl} \mathrm{FP} & \equiv & \bigcup_k \mathrm{FP}(I_{L^k}) & \equiv & \bigcup_k \mathrm{PTIME}(I_{L^k}), \\ \mathrm{PFP} & \equiv & \bigcup_k \mathrm{PFP}(I_{L^k}) & \equiv & \bigcup_k \mathrm{PSPACE}(I_{L^k}). \end{array}$$

Proof. The inclusions $\mathrm{FP}(I_{L^k}) \subseteq \mathrm{FP}$ and $\mathrm{PFP}(I_{L^k}) \subseteq \mathrm{PFP}$ follow from the closure of FP and PFP under definable interpretations, Lemma 1.49. For FP-interpretability of $I_{L^k}(\mathfrak{A})$ over $\mathfrak{A}$ see Proposition 3.12. The converse inclusions $\mathrm{FP} \subseteq \bigcup_k \mathrm{FP}(I_{L^k})$ and $\mathrm{PFP} \subseteq \bigcup_k \mathrm{PFP}(I_{L^k})$ follow from Lemma 3.17. □

As a corollary to these equivalences we finally obtain the following.

Theorem 3.22 (Abiteboul, Vianu). FP $\equiv$ PFP *on the class of all finite relational structures if and only if* PTIME = PSPACE.

Proof. FP $\equiv$ PFP $\Rightarrow$ PTIME = PSPACE follows by considering the class of ordered structures and applying the theorems of Immerman, Vardi and of Abiteboul, Vardi, Vianu that equate FP with PTIME and PFP with PSPACE over these.

The real content of the theorem is the converse: if PTIME = PSPACE then FP $\equiv$ PFP over the class of *all* finite relational structures. Lemma 3.21 yields the necessary reduction of the general case to the ordered case. If PTIME = PSPACE then $\mathrm{FP} \equiv \bigcup_k \mathrm{PTIME}(I_{L^k}) \equiv \bigcup_k \mathrm{PSPACE}(I_{L^k}) \equiv \mathrm{PFP}$. □

3.4.3 Comparison of I_{C^k} and I_{L^k}

There is an obvious formal difference between the L^k- and the C^k-invariants. $I_{L^k}(\mathfrak{A})$ is interpretable as a purely relational structure over the given structures $\mathfrak{A}$. For $I_{C^k}(\mathfrak{A})$ this applies only to the relational part to which weight functions have to be added to obtain an invariant that characterizes up to $\equiv^{C^k}$. A complete relational representation of $I_{C^k}(\mathfrak{A})$ has size $|A|$, the same as $\mathfrak{A}$ itself. Setting aside our particular encoding convention, its size is at least polynomially related to the size of the original structure. The size of $I_{L^k}(\mathfrak{A})$ on the other hand is $|\, A^k/\equiv^{L^k}|$. Below, an example is reviewed of a theory in $L^3_{\infty\omega}$ which forces the size of $I_{L^k}(\mathfrak{A})$ to be logarithmically small in terms of $|A|$ in all its finite models. We have seen in the case of $L^k_{\infty\omega}$ in the last sections that the size of the invariants is directly related to the expressive power of FP. Lemma 3.21 implies that FP-evaluations over $\mathfrak{A}$ close within polynomially many steps — not in the size of $\mathfrak{A}$ but in the size of $I_{L^k}(\mathfrak{A})$

for some sufficiently large k. A similar dependence of fixed-point logic with counting on the size of the I_{C^k} will be derived in the next chapter. That this distinction is a logical phenomenon (and not just an artifact of the particular formalizations of the invariants) follows even in the very trivial case of pure sets, i.e. for $\tau = \emptyset$. Note that in this case the size of I_{L^k} is bounded by a constant, namely the number of equality types in k variables. Correspondingly, FP and all of $L^{\omega}_{\infty\omega}$ collapse to first-order logic over pure sets, see Corollary 1.32. The $C^k_{\infty\omega}$ on the other hand define arbitrarily complex classes of pure sets, and any reasonable formalization of *fixed-point logic with counting* has to render definable all PTIME arithmetical properties of the size of pure sets.

We review Example 1.16 concerning $L^3_{\infty\omega}$-definability of the class of full binary trees. It serves to show that even three variables suffice to force a logarithmic collapse in the size of I_{L^k}. The example is presented and discussed in this context by Dawar, Lindell and Weinstein in [DLW95]. The formalization in just three variables indicated in Example 1.16 is interesting because we shall see in the last chapter that no $L^2_{\infty\omega}$-theory can force a similar collapse: $k = 3$ in fact delineates the border-line for this phenomenon.

Example 3.23. By Example 1.16 there is a sentence φ in $L^3_{\infty\omega}[E]$ defining the class of full finite binary trees. Obviously the size of full binary trees is exponential in their height. The number of $L^k_{\infty\omega}$-types, however, is bounded by a polynomial in the height, since even the number of isomorphism types of k-tuples is bounded by a polynomial. The isomorphism type of a k-tuple $(v_1, \dots, v_k)$ within a given full binary tree is completely characterized by the heights of the vertices v_{ij}, $1 \leqslant i \leqslant j \leqslant k$, where v_{ij} is that vertex in which the paths from v_i and v_j to the root meet. It follows that the number of $L^k_{\infty\omega}$-types in models $\mathfrak{A} \models \varphi$ and therefore the sizes of all $I_{L^k}(\mathfrak{A})$ are polylogarithmic in the size of these models. For suitable polynomials p_k: $|I_{L^k}(\mathfrak{A})| = \big|\mathrm{Tp}^{L^k}(\mathfrak{A}; k)\big| \leqslant p_k\big(\log(|A|)\big)$ for all $\mathfrak{A} \models \varphi$.

Dawar, Lindell and Weinstein also employ tree structures with this logarithmic collapse in a padding argument to prove the second main result of Abiteboul and Vianu about the relationship between FP and PFP stated below. For the proof we refer to [DLW95]. The statement of this result is important here because we shall find the opposite for the counting extensions — the reason for this fundamental difference is that the I_{L^k} may collapse the size of structures while the I_{C^k} do not.

Let $\mathrm{PFP}|_{\mathrm{poly}}$ stand for the subclass of PFP which admits PFP-applications only where the limit is reached in a polynomially bounded number of steps. In particular $\mathrm{FP} \subseteq \mathrm{PFP}|_{\mathrm{poly}}$. Intuitively FP captures PTIME relational recursion, PFP captures PSPACE relational recursion. It would be tempting therefore to conjecture that $\mathrm{PFP}|_{\mathrm{poly}} = \mathrm{FP}$.

Theorem 3.24 (Abiteboul, Vianu).
If $\mathrm{PFP}|_{\mathrm{poly}} \subseteq \mathrm{FP}$ *then* PTIME = PSPACE.

Note that the converse implication holds as a consequence of the first theorem of Abiteboul-Vianu, Theorem 3.22.

3.5 A Partial Reduction to Two Variables

The invariants I_{C^k} and I_{L^k} have as their backbones pre-orderings defined as the stable colourings of certain graphs interpretable over the k-th power of the given structures. In the standard setting these pre-orderings themselves can be defined in $C^2_{\infty\omega}$ and $L^2_{\infty\omega}$, respectively, as shown in Section 2.2. Pursuing this connection further one can show that I_{C^k} and I_{L^k} are in fact FP-interpretable over the two-variable invariants of the game k-graphs of the underlying structures. These results later play a rôle in potential reductions for canonization problems.

Recall from Definition 2.26 that the game k-graph $\mathfrak{A}^{(k)}$ associated with $\mathfrak{A} \in \text{fin}[\tau]$ is the structure with universe A^k and with binary predicates E_j for the accessibility relations in each component and unary predicates P_θ for the identification of atomic types $\theta \in \text{Atp}(\tau; k)$. The vocabulary of $\mathfrak{A}^{(k)}$ is denoted $\tau^{(k)}$. For the technical notion of interpretability of functors compare the remarks made in connection with Example 1.47.

Proposition 3.25. *$I_{L^k}(\mathfrak{A})$ is uniformly* FP*-interpretable in $I_{L^2}(\mathfrak{A}^{(k)})$.*
$I_{C^k}(\mathfrak{A})$ is uniformly FP*-interpretable in $I_{C^2}(\mathfrak{A}^{(k)})$.*

The mere functional dependencies expressed in these interpretability statements imply in particular that

$$\begin{aligned} \mathfrak{A}^{(k)} \equiv^{C^2} \mathfrak{A}'^{(k)} \quad &\Longrightarrow \quad \mathfrak{A} \equiv^{C^k} \mathfrak{A}', \\ \mathfrak{A}^{(k)} \equiv^{L^2} \mathfrak{A}'^{(k)} \quad &\Longrightarrow \quad \mathfrak{A} \equiv^{L^k} \mathfrak{A}'. \end{aligned}$$

The claim of the proposition goes beyond these implications, since it requires FP-interpretability or PTIME computability of one invariant in terms of the other.

Sketch of Proof. The proof is somewhat technical though not difficult. We indicate the proof for the interpretability of $I_{C^k}(\mathfrak{A})$ in $I_{C^2}(\mathfrak{A}^{(k)})$. Since we are dealing with ordered structures it suffices to show that $I_{C^k}(\mathfrak{A})$ is PTIME computable from $I_{C^2}(\mathfrak{A}^{(k)})$, compare Example 1.47.

Consider the generation of the $\preccurlyeq_i$ with limit $\preccurlyeq$, where $\preccurlyeq$ is the quotient interpretation over A^k of the ordering $(A^k/\equiv^{C^k}, \leqslant)$ underlying $I_{C^k}(\mathfrak{A})$. For each i let $(A^k/\approx_i, \leqslant_i)$ be the ordered quotient induced by $\preccurlyeq_i$.

We first show inductively how $(A^k/\approx_i, \leqslant_i)$ is interpretable in the relational part of $I_{C^2}(\mathfrak{A}^{(k)})$. For this interpretation we use those elements of $I_{C^2}(\mathfrak{A}^{(k)})$ that represent types of singletons over $\mathfrak{A}^{(k)}$, i.e. that have $x_1 = x_2$ in their atomic $\tau^{(k)}$-type. We denote this subset of the universe of $I_{C^2}(\mathfrak{A}^{(k)})$ by Δ and identify it with $\text{Tp}^{C^2}(\mathfrak{A}^{(k)}; 1)$. Recall that singletons over $\mathfrak{A}^{(k)}$ are

k-tuples over $\mathfrak{A}$. The desired interpretation is such that the $\approx_i$-class of $\overline{a} \in A^k$ is represented by the set of all $\mathrm{tp}^{C^2}_{\mathfrak{A}^{(k)}}(\overline{b})$ for $\overline{b} \approx_i \overline{a}$. It can be described by the mapping

$$\begin{array}{rcl} A^k/\approx_i & \longrightarrow & \mathcal{P}(\Delta) \\ \alpha & \longmapsto & \underline{\alpha} := \{\mathrm{tp}^{C^2}_{\mathfrak{A}^{(k)}}(\overline{b}) \mid \overline{b} \in \alpha\}, \end{array}$$

where $\mathcal{P}(\Delta)$ stands for the power set of Δ.

$(A^k/\approx_0, \leqslant_0) = (\mathrm{Atp}(\mathfrak{A}; k), \leqslant_0)$ is quantifier free interpretable over Δ since each atomic type $\theta \in \mathrm{Atp}(\mathfrak{A}; k)$ corresponds to those elements of Δ whose atomic $\tau^{(k)}$-type contains $P_\theta x_1$:

$$\underline{\theta} = \{\beta \in \mathrm{Tp}^{C^2}(\mathfrak{A}^{(k)}; 1) \mid \beta \models P_\theta x_1\}.$$

Now for the inductive step from $(A^k/\approx_i, \leqslant_i)$ to $(A^k/\approx_{i+1}, \leqslant_{i+1})$. Recall that $\preccurlyeq_{i+1}$ is determined in terms of the numbers

$$\nu^\alpha_j(\overline{a}) = \left|\{b \in A \mid \overline{a}\tfrac{b}{j} \in \alpha\}\right|$$

for $\overline{a} \in A^k$ and $\alpha \in A^k/\approx_i$. By the inductive hypothesis any such α is interpreted by a subset $\underline{\alpha}$ of Δ in $I_{C^2}(\mathfrak{A}^{(k)})$. Obviously $\nu^\alpha_j(\overline{a})$ is represented over $\mathfrak{A}^{(k)}$ as

$$\nu^\alpha_j(\overline{a}) = \left|\{\overline{b} \in A^k \mid E_j\overline{a}\,\overline{b} \wedge \overline{b} \in \alpha\}\right|.$$

This shows that these numbers can only depend on $\mathrm{tp}^{C^2}_{\mathfrak{A}^{(k)}}(\overline{a})$ and therefore are directly computable on $I_{C^2}(\mathfrak{A}^{(k)})$ from $\mathrm{tp}^{C^2}_{\mathfrak{A}^{(k)}}(\overline{a})$ and $\underline{\alpha}$. Thus the desired interpretation of $(A^k/\approx_{i+1}, \leqslant_{i+1})$ over $I_{C^2}(\mathfrak{A}^{(k)})$ is PTIME computable from that of $(A^k/\approx_i, \leqslant_i)$.

This refinement process terminates after polynomially many many steps and its limit is the interpretation of the ordered quotient $(A^k/\equiv^{C^k}, \leqslant)$ needed for $I_{C^k}(\mathfrak{A})$. The other data in $I_{C^k}(\mathfrak{A})$ are easily definable and computable in terms of this interpretation as follows.

The P_θ are trivially represented by atomic formulae over $\mathfrak{A}^{(k)}$.

The E_j are also atomically represented in $\mathfrak{A}^{(k)}$ and can be transferred to the interpreted $I_{C^k}(\mathfrak{A})$ as follows: α and α' are E_j-related in $I_{C^k}(\mathfrak{A})$ if they are realized by some $\overline{a}$ and $\overline{a}'$ that are E_j-related in $\mathfrak{A}^{(k)}$. Therefore $(\alpha, \alpha') \in E_j$ in $I_{C^k}(\mathfrak{A})$ if there is some

$$\beta \in \mathrm{Tp}^{C^2}(\mathfrak{A}^{(k)}; 2) \text{ such that } \beta \models E_j x_1 x_2\,,\ \beta|x_1 \in \underline{\alpha} \text{ and } \beta|x_2 \in \underline{\alpha}'.$$

Here $\beta|x_i$ denotes the restriction of the 2-type β to the i-th component, which is an element of $\Delta = \mathrm{Tp}^{C^2}(\mathfrak{A}^{(k)}; 1)$.

The weights ν_j of $I_{C^k}(\mathfrak{A})$ reduce to numerical data that are available on $I_{C^2}(\mathfrak{A}^{(k)})$ in the manner exhibited for the $\nu^\alpha_j(\overline{a})$ above.

The S_ρ are PTIME computable from the remaining data anyway according to Remark 3.5. □

Sources, attributions and remarks. As pointed out above the important concept of an ordered invariant is due to Abiteboul and Vianu [AV91]. Their invariants were abstracted from a model of relational computation and employed in an analysis FP, $\mathrm{PFP}|_{\mathrm{poly}}$ and PFP over arbitrary relational structures in terms of complexities of computations over ordered structures. The major results are the theorems of Abiteboul and Vianu, Theorems 3.22 and 3.24 above. The systematic formalization of this approach in terms of the $L^k_{\infty\omega}$ is due to Dawar [Daw93] and Dawar, Lindell and Weinstein [DLW95]. The extension and logical formulation for the $C^k_{\infty\omega}$ is presented in [GO93, Ott96a]. The corresponding applications to fixed-point logics with counting will form a main topic of the following chapter.

4. Fixed-Point Logic with Counting

This chapter is devoted to the introduction and analysis of the natural extensions of the fixed-point logics FP and PFP that have expressive means for cardinality properties.

• The actual formalization of fixed-point logics with counting, FP+C and PFP+C, in a two-sorted framework is given in Section 4.1.

• In Section 4.2 the relation of FP+C and PFP+C with the $C^k_{\infty\omega}$ and with the C^k-invariants is investigated. In particular we obtain the analogue of the first theorem of Abiteboul and Vianu (Theorem 3.22 above) in the presence of counting. In contrast with the second theorem of Abiteboul and Vianu (Theorem 3.24) we here find that FP+C *is* the polynomial restriction of PFP+C.

• Section 4.3 deals with the separation result FP+C $\subsetneq$ PTIME, which is due to Cai, Fürer and Immerman, in a framework that lends itself to relativization. In restriction to classes with certain closure properties FP+C can only capture PTIME if some I_{C^k} provides a complete invariant *up to isomorphism* (equivalently, if some $C^k_{\infty\omega}$ coincides with $L_{\infty\omega}$) over this class.

• Section 4.4 summarizes some results on equivalent characterizations of the expressive levels of FP+C and PFP+C.

As pointed out in the introduction, first-order logic at first sight suffers from *two* independent shortcomings over finite structures: it completely lacks mechanisms to model recursion — the fixed-point operations provided in FP and PFP answer this requirement; and it also lacks expressive means to assess cardinalities of definable sets. The latter defect is obviously overcome automatically together with the former over ordered structures. By the theorems of Immerman, Vardi and Abiteboul, Vardi, Vianu, FP and PFP capture PTIME and PSPACE over ordered structures. In particular all PTIME, respectively PSPACE, properties of cardinalities are expressible in FP, respectively PFP, over ordered structures. Not so in the case of not necessarily ordered structures: in fact the most obvious examples that FP and PFP do not correspond to standard complexity classes in the general case all involve counting. Over pure sets for instance FP, PFP and even $L^\omega_{\infty\omega}$ collapse to first-order logic and cannot express low complexity cardinality properties like "there is an even number of elements". For some time therefore it had been conjectured, mainly by Immerman, that FP enriched with counting might capture PTIME in the general case. This expectation was later disproved by Cai, Fürer and Immerman, who showed that not even $C^{\omega}_{\infty\omega}$ comprises all of PTIME. There remains good motivation to study the extensions of FP and PFP by expressive means for counting, however.

(a) FP and PFP are successful extensions of first-order logic and capture an interesting notion of *relational recursion* on finite structures even in the absence of order. Without order, however, they do not add to the power of first-order with respect to cardinality properties. Many natural structural properties involve counting in addition to relational recursion.
(b) It is reasonable to treat the two obvious defects of first-order logic on an equal footing and to investigate natural levels of expressiveness that address both defects.
(c) As indicated in the previous chapter, the relationship between FP and PFP on the one hand and $L^{\omega}_{\infty\omega}$ on the other leads to valuable insights into the nature of relational recursion on finite structures. FP and PFP *are*, in some intuitive sense, PTIME *and* PSPACE *in the world of* $L^{\omega}_{\infty\omega}$. $C^{\omega}_{\infty\omega}$ is a natural richer and still well-behaved fragment of $L_{\infty\omega}$. In particular $C^{\omega}_{\infty\omega}$ shares with $L^{\omega}_{\infty\omega}$ the benefit of elegant game characterizations and the existence of PTIME computable invariants. It is natural therefore to expect appropriate counting extensions of FP and PFP to *represent* PTIME *and* PSPACE *in the world of* $C^{\omega}_{\infty\omega}$.
(d) It turns out that fixed-point logics with counting represent robust levels of expressive power in the sense that the semantic strength proves to be independent of several choices in the actual formalization. More importantly they offer a number of interesting equivalent characterizations.
(e) Finally we shall see that some properties of the counting extensions resemble those found for FP and PFP themselves only in the ordered case. Roughly speaking, with counting one is closer to the ordered case.

4.1 Definition of FP+C and PFP+C

The natural modelling for the counting extensions uses two-sorted structures. The given relational structure forms the first sort, an *ordered numerical domain* the second. In this way counting terms that take values in the numerical domain can naturally be introduced. The two-sorted structures can of course in the standard way be encoded in a one-sorted framework with extra unary predicates to denote the different universes. We shall at some points appeal to this possibility. For the basic formalizations, however, the two-sorted picture is easier to handle and intuitively neater. Let τ be finite and relational as usual.

Definition 4.1. *Let* * *be the functor that takes* $\mathfrak{A} \in \mathrm{fin}[\tau]$ *to the two-sorted structure* $\mathfrak{A}^*$ *which is the disjoint union of* $\mathfrak{A}$ *itself for the first sort and the canonical ordered structure of size* $|A|+1$ *for the second sort.*

$$\mathfrak{A}^* := \mathfrak{A} \,\dot{\cup}\, (n+1, <^{n+1}) \quad \textit{where } n = |A|.$$

Recall that we identify $n+1$ *with* $\{0, \dots, n\}$*. Let* $\mathrm{fin}[\tau]^* := \{\mathfrak{A}^* \mid \mathfrak{A} \in \mathrm{fin}[\tau]\}$*.*

We apply the following formalism to the two-sorted structures in $\mathrm{fin}[\tau]^*$. Variable symbols x, y, $z, \dots$ range over the elements of the first sort, variables ν, $\mu, \dots$ range over the second sort. Of a second-order variable X we say that it is of type (r_1, r_2) if it ranges over subsets of $(\text{first sort})^{r_1} \times (\text{second sort})^{r_2}$. All second-order variables come with a definite typing in this sense.

We consider first-order logic and its extensions by FP- and PFP-operators over $\mathrm{fin}[\tau]^*$. The first-order constructors comprise

- the formation of atomic expressions, which have to respect the type of second-order variables in the obvious way,
- boolean connectives and
- quantifications with respect to first-order variables of each type.

For the fixed-point operators we admit the most general kind of fixed-point generations in the two-sorted framework by allowing fixed-point variables X of arbitrary mixed types. Otherwise no changes are necessary to accommodate fixed-point operations over the $\mathfrak{A}^*$. Compare Section 1.3.3 and Definitions 1.22 and 1.23. Let for instance $\varphi(X, \overline{x}, \overline{\nu})$ be in the indicated free variables, where X is of type (r_1, r_2) and $\overline{x}$ and $\overline{\nu}$ are tuples of r_1, respectively r_2, distinct variables for elements of the first, respectively second, sort.

Over each $\mathfrak{A}^* \in \mathrm{fin}[\tau]^*$ the formula φ induces the following mapping $F_\varphi^{\mathfrak{A}^*}$ ($\mathcal{P}$ denotes the power set).

$$\begin{aligned} F_\varphi^{\mathfrak{A}^*} \colon \mathcal{P}\Big(A^{r_1} \times \{0, \dots, |A|\}^{r_2}\Big) &\longrightarrow \mathcal{P}\Big(A^{r_1} \times \{0, \dots, |A|\}^{r_2}\Big) \\ P &\longmapsto \Big\{(\overline{a}, \overline{m}) \;\Big|\; \mathfrak{A}^* \models \varphi[P, \overline{a}, \overline{m}]\Big\}. \end{aligned}$$

The semantics of formulae $[\mathrm{PFP}_{X,\overline{x}\overline{\nu}}\varphi(X, \overline{x}, \overline{\nu})]\overline{x}\,\overline{\nu}$ and $[\mathrm{FP}_{X,\overline{x}\overline{\nu}}\varphi(X, \overline{x}, \overline{\nu})]\overline{x}\,\overline{\nu}$ is defined in terms of the partial, respectively inductive or inflationary, fixed points of F_φ just as in the one-sorted case.

Definition 4.2. *Let* $L^*_{\omega\omega}$ *be two-sorted first-order logic for* **-structures. Similarly* $L^{\omega\,*}_{\infty\omega}$ *is that fragment of infinitary logic for two-sorted* **-structures that consists of formulae using only finitely many first-order variables (of either sort).* FP* *and* PFP* *stand for the two-sorted variants of fixed-point and partial fixed-point logic for these two-sorted structures.*

Note that these logics admit formulae with free first-order variables of both sorts, or, where applicable, also free second-order variables of mixed type. We ultimately only consider formulae that are free over first-order variables of the first sort and define global relations over the original relational structures. This might be regarded as the *standard part* of the semantics for these logics. All considerations about the expressive power of these logics concern these standard parts. A statement like $C^{\omega}_{\infty\omega} \supseteq \mathrm{FP}^*$, for instance, means that any global relation over $\mathrm{fin}[\tau]$ that is FP^*-definable as a global relation over the first sort is $C^{\omega}_{\infty\omega}$-definable. Formulae with other free variables are important, however, for the inductive generation of formulae and accordingly play some rôle in particular in syntactic arguments by induction.

A technical comment is in order with respect to the standard one-sorted modelling of two-sorted structures. In the sequel we shall want to apply results that formally deal with one-sorted structures also in the present two-sorted formalization. Rather than reproving them in a tedious adaptation of the standard arguments one may directly apply them on the basis of the following remark.

Remark 4.3. *For* $\mathcal{L} = L_{\omega\omega}$, $L^{\omega}_{\infty\omega}$, FP, PFP *and with* $\mathcal{L}^* = L^*_{\omega\omega}$, $L^{\omega\,*}_{\infty\omega}$, FP^*, PFP^* *according to Definition 4.2: the expressive power of* $\mathcal{L}$ *on the standard one-sorted encodings of structures in* $\mathrm{fin}[\tau]^*$ *is the same as that of* $\mathcal{L}^*$.

Sketch of Proof. The argument is via mutual simulations between the one-sorted and the two-sorted frameworks.
i) First-order constructors. Consider first the simulation of the two-sorted framework in the one-sorted encodings, where the i-th sort is described by a unary predicate U_i. The distinction between first-order variables of different sorts is faithfully simulated through relativizations to the respective subdomains. Conversely, a formula $\varphi(x_1, \ldots, x_r)$ of the one-sorted framework, whose first-order variables range over the combined domain $U_1 \dot{\cup} U_2$, translates into a tuple of 2^r formulae φ_s, $s \subseteq \{1, \ldots, r\}$, of the two-sorted framework — one for each possible typing. For instance if $s = \{1, 2\}$, then $\varphi_s = \varphi_s(x_1, x_2, \nu_3, \ldots, \nu_r)$ takes care of the case that just x_1 and x_2 get interpreted over U_1. The inductive definition of the φ_s is straightforward. For instance, if $\psi(x_1, \ldots, x_{r-1}) = \exists x_r \varphi(x_1, \ldots, x_r)$, then $\psi_{\{1,2\}} = \exists x_r \varphi_{\{1,2,r\}} \vee \exists \nu_r \varphi_{\{1,2\}}$.
ii) Second-order variables and fixed-point processes. A second-order variable X that is of type (r_1, r_2) over the two-sorted structures is simulated over their one-sorted encodings by a second-order variable of arity $r_1 + r_2$ which can easily be relativized to interpretations of the correct type. Fixed-point processes carry over directly. In the other direction consider an r-ary second-order variable X over the one-sorted encodings. Since its interpretations do not come with a fixed typing, it has to be modelled in general by a tuple of 2^r second-order variables $(X_s)_{s \subseteq \{1,\ldots,r\}}$, one for each possible typing. We think of the original X as the union of the X_s where, for instance, $X_{\{1,2\}}$ is the collection of tuples in X whose first two components come from U_1. Obviously X and the X_s are first-order interdefinable (over the one-sorted encodings). A fixed-point process involving X naturally translates into a simultaneous fixed-point process for a system of formulae. In this system there is one formula $\varphi_s\big((X_t)_{t \subseteq \{1,\ldots,r\}}\big)$ in first-order variables typed according to s,

for each s. The resulting fixed points of systems can be recast into ordinary fixed points using standard techniques as discussed in Example 1.27. $\square$

This observation also implies that the usual semantic inclusions carry over to the two-sorted framework.

Remark 4.4. $L^*_{\omega\omega} \subseteq \mathrm{FP}^* \subseteq \mathrm{PFP}^* \subseteq L^{\omega\,*}_{\infty\omega}$.

The functor $^*\colon \mathrm{fin}[\tau] \longrightarrow \mathrm{fin}[\tau]^*$ is isomorphism preserving: $\mathfrak{A}^* \simeq \mathfrak{A}'^*$ if and only if $\mathfrak{A} \simeq \mathfrak{A}'$. Similarly it preserves the substructure relation. It does not, however, preserve definability of substructures even at the atomic level. As a consequence, FP^* and PFP^* do not have the relativization property. For a simple example consider evenness. Evenness of the universe is obviously definable in FP^*: $|A|$ is even if the ordered second sort of $\mathfrak{A}^*$ has an odd number of elements, and FP^*-recursion over the second sort suffices for checking this. Evenness of a unary predicate $U \in \tau$, however, is not in FP^*. The straightforward adaptation of the standard game argument shows that evenness of $U \subseteq A$ is not even definable in $L^{\omega\,*}_{\infty\omega}$. In a sense, only the cardinality of the universe has yet been made available in the ordered numerical sort. To introduce counting and to remedy the defects just pointed out, it suffices to render the cardinalities of definable subsets over the first sort definable over the second sort. We present below two equivalent ways of doing so. The first approach introduces *counting terms* in a straightforward way. The other one — more elegant maybe from a model theoretic view — uses the extension by the Härtig quantifier.

Counting terms. Counting terms link the two sorts so that the second, numerical sort can be used for talking about the size of definable subsets. It suffices to consider unary subsets of the first sort, for reasons discussed below.

Definition 4.5. *With each formula φ and any variable x of the first sort associate a counting term:*

$$t := \#_x\varphi(x)$$

of the second sort. Put $\mathrm{free}(t) = \mathrm{free}(\varphi) \setminus \{x\}$. *If $(\mathfrak{A}^*, \Gamma)$ interprets all free variables of φ apart from x, then the interpretation of t in $(\mathfrak{A}^*, \Gamma)$ is that element of the second sort that describes the size of the predicate $\varphi[\mathfrak{A}^*, \Gamma]$ defined by φ:*

$$t^{\mathfrak{A}^*,\Gamma} := \Big|\big\{a \in A \mid (\mathfrak{A}^*, \Gamma) \models \varphi[a]\big\}\Big|.$$

To obtain fixed-point logic with counting, we simultaneously close first-order logic $L^*_{\omega\omega}$ for the two-sorted structures under the FP^*-constructor, the formation of counting terms and substitution of these for variables of the second sort. The formal definition of the syntax would be via a combined inductive generation of formulae and terms.

Definition 4.6. *Let* FP+C *be the smallest extension of* FP* *that is closed under formation and substitution of counting terms and the* FP**-constructor.* PFP+C *is the corresponding closure with respect to* PFP**.*

Clearly FP+C $\subseteq$ PTIME and PFP+C $\subseteq$ PSPACE.

Using the Härtig quantifier instead. The Härtig quantifier, cf. Definition 1.53, expresses cardinality equality. Its semantics extends naturally to two-sorted structures. Over the $\mathfrak{A}^*$ it may be used to define counting-terms:

$$\begin{array}{ll} \nu = \#_x \varphi(x) & \text{is equivalent with} \\ Q_{\mathrm{H}}\Big((x;\varphi);(\mu;\psi)\Big) & \text{for } \psi(\mu,\nu) = \ \mu < \nu. \end{array}$$

Denote by FP$(Q_{\mathrm{H}})^*$ and PFP$(Q_{\mathrm{H}})^*$ the logics that result from adjoining the Härtig quantifier in the two-sorted framework. It turns out that these provide equivalent characterizations for FP+C and PFP+C, respectively.

Lemma 4.7. PFP+C $\equiv$ PFP$(Q_{\mathrm{H}})^*$ *and* FP+C $\equiv$ FP$(Q_{\mathrm{H}})^*$*.*

Proof. We point out that the statement of Remark 4.3 extends to the extensions of FP and PFP by the Härtig quantifier. Adjoining the Härtig quantifier over the one-sorted encodings of two-sorted structures, we formally gain cardinality equalities for mixed-sorted unary predicates. These however are dissolved into equalities for the sums of cardinalities for two unary pure-sorted predicates each. Sums over the second sort, however, are definable in FP over the ordered second sort since they are PTIME computable.

For the proof of the lemma note that the inclusions "$\subseteq$" follow directly from the definability of counting terms through the Härtig quantifier. Consider then the converse inclusion for FP. An application of the Härtig quantifier may involve two predicates over the first-sort — this case translates into an equality for the corresponding counting terms directly. It may also involve at least one predicate over the second sort — but over ordered domains, values of counting terms of type $\#_\nu\varphi$ are even FP-definable since they are PTIME computable. □

FP+C and PFP+C turn out to be very robust with respect to the formal details concerning the introduction of counting terms. For example, it is natural to allow counting not only for unary predicates but also in higher arities and over mixed sorts. We have just seen that unary counting over the second sort is for free. The reason for this robustness is that in FP+C we already have the full power of PTIME operations over the second sort. This is at the root of the following model theoretic statement of robustness. It should be noted that a corresponding counting extension of first-order logic does not at all share these properties, see Example 4.13 below. For the notion of generalized interpretations and closure with respect to these compare Definitions 1.44 and 1.48 in Section 1.5.

Proposition 4.8. FP+C *and* PFP+C *are closed with respect to generalized interpretations.*

Proof. Consider for instance FP+C. The statement to be proved is the following. Let $\mathfrak{i}$ be some FP+C-definable generalized (σ, τ)-interpretation, functorially $\mathfrak{i}\colon \mathrm{fin}[\tau] \to \mathrm{fin}[\sigma]$. Let R be some FP+C-definable global relation over $\mathrm{fin}[\sigma]$. Then the global relation $\mathfrak{i}(R)$ over $\mathrm{fin}[\tau]$ whose value over $\mathfrak{A}$ is the interpretation over $\mathfrak{A}$ of $R^{\mathfrak{i}(\mathfrak{A})}$ has to be FP+C-definable as well. Since we know that FP and PFP have the required closure properties, it suffices to prove the following.

(∗) Each definable interpretation of σ-structures over $\mathrm{fin}[\tau]$ induces a definable interpretation of the corresponding two-sorted structures in $\mathrm{fin}[\sigma]^*$ over $\mathrm{fin}[\tau]^*$. This interpretation is such that counting terms for the interpreted $\mathrm{fin}[\sigma]^*$-structures are FP+C-definable over the parent structures in $\mathrm{fin}[\tau]^*$.

Sufficiently large numerical domains are interpretable in powers of the given numerical domain. The set of s-tuples over $n+1$ together with the first-order definable lexicographic ordering provides an interpretation of $\big((n+1)^s, <\big)$ over $(n+1, <)$ as always $(n+1)^s \geqslant n^s + 1$. This numerical domain is sufficiently large to provide the second sort for interpretations over the s-th power. The numerical value represented by an s-tuple $\overline{m}$ in $(n+1)^s$ is the number of lexicographic predecessors of $\overline{m}$: $\big|\{\overline{m}'|\overline{m}' <_{\mathrm{lex}} \overline{m}\}\big|$. Having these numerical domains, (∗) reduces to the following lemma: FP+C suffices to simulate counting terms over interpretations in powers and quotients. □

Lemma 4.9. *The analogues of counting terms for counting in higher arity and for counting modulo definable congruences (counting equivalence classes) are definable over* $\mathrm{fin}[\tau]^*$ *in* FP+C.

Proof. The claim for higher arity counting means that for $\varphi(x_1, \ldots, x_s)$ in FP+C (where other variables are suppressed without loss of generality) there is a formula $\psi(\nu_1, \ldots, \nu_s)$ in FP+C such that

$$\mathfrak{A}^* \models \psi[\overline{m}] \quad \Longleftrightarrow \quad \big|\varphi[\mathfrak{A}^*]\big| = \big|\{\overline{m}'|\overline{m}' <_{\mathrm{lex}} \overline{m}\}\big|.$$

Consider for instance the binary case, a formula $\varphi(x, y)$. For each m, the number of x such that there are exactly m many y satisfying φ with that x is $t(\mu) = \#_x\big(\#_y\varphi(x, y) = \mu\big)$, where μ is the second-sort variable for m. But obviously the desired lexicographic representation of the number $l = \big|\{(x, y)|\varphi\}\big|$ is PTIME computable in terms of the function $m \mapsto t(m)$ through

$$l = \sum_m m\, t(m).$$

The graph of the function $m \mapsto t(m)$ is FP+C-definable over the second sort so that FP+C-definability of l follows immediately.

Counting with respect to a definable congruence, or the lexicographic representation of the number $l = \big|\varphi[\mathfrak{A}^*]/\psi[\mathfrak{A}^*]\big|$, is treated analogously. Without loss of generality let now $\varphi = \varphi(x)$ be unary, $\psi = \psi(x,y)$ binary. Here l is PTIME computable from the function $m \mapsto t(m)$ where $t(\mu) = \#_x\big(\varphi(x) \wedge \#_y(\varphi(y) \wedge \psi(x,y)) = \mu\big)$, such that $t(m)$ is the number of elements whose ψ-class in φ has exactly m elements and

$$l = \sum_m m^{-1} t(m).$$

□

Example 4.10. All (even quotient) cardinality Lindström quantifiers (see Definitions 1.52 and 1.54) that are based on PTIME computable numerical predicates are expressible in FP+C. This is an obvious consequence of the above fact that FP+C has definable counting terms for counting in arbitrary arities and with respect to definable congruences together with the Immerman-Vardi theorem applied to fixed-point definability over the second sort.

Example 4.11. Since in particular the Rescher quantifier (Definition 1.53) is definable in FP+C we obtain from Lemma 2.22 that the stable colouring of graphs is FP+C-definable. It similarly follows from Proposition 3.6 that the relational parts of the the C^k-invariants are FP+C-interpretable as quotients over the k-th power. This is further explored in the next section.

Example 4.12. $\equiv^{C^k}$ is in FP+C, just as $\equiv^{L^k}$ is in FP according to Corollary 3.15. This is easier to see than the stronger claim made in Proposition 3.10 about definability in FP(Q_{H}), since one may here argue directly with interpretability of the relational parts of the I_{C^k} together with availability of counting terms to check equality for the weights.

Aside on first-order logic with unary counting. As pointed out above, first-order logic is far more sensitive to slight changes in the definition of a "counting extension" than FP and PFP are. This is not surprising since the robustness of FP+C and PFP+C is due to their recursive power over the second sort. Let for the considerations of the following example *first-order logic with unary counting* be defined as the closure of $L^*_{\omega\omega}$ with respect to the formation and substitution of counting terms in the sense of Definition 4.5.

Example 4.13. First-order logic with unary counting does not capture binary counting. Consider $\tau = \{U_1, U_2, U_3\}$ consisting of three unary predicates. Let Q be the class of those τ-structures whose universe is partitioned into three disjoint sets by the U_i. Let always m_i stand for the cardinality of U_i, and $n = m_1 + m_2 + m_3$ for the overall size of $\mathfrak{A} \in Q$. The tuple (m_1, m_2, m_3) characterizes $\mathfrak{A}$ up to isomorphism, of course. Let $Q_0 \subseteq Q$ be the subclass defined by the condition $m_2 = m_1^2$. Clearly Q_0 is definable in first-order logic with counting terms for binary predicates: one need merely equate the cardinalities of the first-order definable predicates $\{(x,y)|x=y\wedge U_2x\}$ and $\{(x,y)|U_1x\wedge U_1y\}$.

We claim that Q_0 is not definable in $L^*_{\omega\omega}$ with unary counting terms. Call this logic $\mathcal{L}$ for the purposes of this proof. The proof involves a reduction of definability in $\mathcal{L}$ to ordinary first-order definability over the second, arithmetical sort of the $\mathfrak{A}^*$ expanded with just a fixed finite number of constants for some particular values of counting terms. Standard Ehrenfeucht-Fraïssé arguments for linear orderings then apply to show that Q_0 cannot be separated from $Q\setminus Q_0$ by these first-order means.

A trivial automorphism argument will be used repeatedly. If $\overline{a}$ and $\overline{a}'$ are such that $\mathrm{atp}_{\mathfrak{A}}(\overline{a})=\mathrm{atp}_{\mathfrak{A}}(\overline{a}')$ then there is an automorphism of $\mathfrak{A}^*$ which maps $\overline{a}$ to $\overline{a}'$ and fixes the second sort of $\mathfrak{A}^*$ pointwise. It follows that

(i) for $\varphi(\overline{x},\overline{\nu})\in\mathcal{L}$ and fixed interpretation $\overline{m}$ for $\overline{\nu}$ over $\mathfrak{A}^*$, the predicate $\varphi[\mathfrak{A}^*,\overline{m}]=\{\overline{a}\,|\,\mathfrak{A}^*\models\varphi[\overline{a},\overline{m}]\}$ is a union of sets $\theta[\mathfrak{A}]$ for $\theta\in\mathrm{Atp}(\mathfrak{A};k)$, ($k$ the arity of $\overline{x}$).

(ii) for $\theta\in\mathrm{Atp}(\mathfrak{A};k)$ and $\Theta\subseteq\mathrm{Atp}(\mathfrak{A};k)$ the counting values

$$t(\theta,\Theta)^{\mathfrak{A}}=\big|\{b\in A\;\big|\;\mathrm{atp}\big(\overline{a}\tfrac{b}{1}\big)\in\Theta\}\big|$$

for $\overline{a}\in\theta[\mathfrak{A}]$ only depend on $\mathfrak{A}$, θ and Θ (and not on $\overline{a}\in\theta[\mathfrak{A}]$).

(iii) for θ, Θ as above and for all $\mathfrak{A}\in Q$ with sufficiently large $m_i=|U_i^{\mathfrak{A}}|$, $t(\theta,\Theta)$ is of the form $\sum_{i\in s}m_i\ \pm d$, where $s\subseteq\{1,2,3\}$ and $0\leqslant d\leqslant k$, s and d depending only on θ and Θ.

Consider the second sort of $\mathfrak{A}^*$, for $\mathfrak{A}\in Q$, as equipped with parameters $\overline{t}$ for the values of all $t(\theta,\Theta)^{\mathfrak{A}}$ (for fixed k, as appropriate).

Claim. For each $\varphi(\overline{x},\overline{\nu})\in\mathcal{L}$ and $\theta\in\mathrm{Atp}(\mathfrak{A};k)$ there is a $<$-formula $\underline{\varphi}_{\theta}(\overline{\nu},\overline{\mu})$ in first-order logic (for the the second sort) such that for all $\mathfrak{A}\in Q$ with sufficiently large m_i and for all interpretations $\overline{m}$ for the $\overline{\nu}$:

$$\theta[\mathfrak{A}]\subseteq\varphi[\mathfrak{A}^*,\overline{m}]\quad\Leftrightarrow\quad\mathfrak{A}^*\models\underline{\varphi}_{\theta}[\overline{m},\overline{t}].$$

This claim is justified inductively. The atomic cases and boolean connectives are trivially dealt with.

If $\varphi=\exists x_j\psi(\overline{x},\overline{\nu})$, then $\underline{\varphi}_{\theta}$ is the disjunction over all $\underline{\psi}_{\theta'}$ with $\theta'\in\mathrm{Atp}(\tau;k)$ such that θ' and θ agree on $\{x_1,\dots,x_k\}\setminus\{x_j\}$. For $\varphi=\exists\nu\psi(\overline{x},\overline{\nu})$ one can simply take $\underline{\varphi}_{\theta}=\exists\nu\underline{\psi}_{\theta}$.

Finally let $\varphi=\#_{x_1}\psi(\overline{x},\overline{\nu})=\nu$. Then $\theta[\mathfrak{A}]\subseteq\varphi[\mathfrak{A}^*,\overline{m},m]$ if $t(\theta,\Theta)^{\mathfrak{A}}=m$ for $\Theta=\big\{\theta'\in\mathrm{Atp}(\tau;k)\;\big|\;\theta'[\mathfrak{A}]\subseteq\psi[\mathfrak{A}^*,\overline{m}]\big\}$. But by the inductive hypothesis $\Theta=\big\{\theta'\;\big|\;\mathfrak{A}^*\models\underline{\psi}_{\theta'}[\overline{m},\overline{t}]\big\}$. The equation $t(\theta,\Theta)=m$ can be put into the desired form through a distinction of cases: $t(\theta,\Theta)=\nu$ is equivalent with the disjunction of the following formulae, over all subsets $\Theta'\subseteq\mathrm{Atp}(\tau;k)$:

$$\bigwedge_{\theta'\in\Theta'}\underline{\psi}_{\theta'}[\overline{\nu},\overline{\mu}]\ \wedge\bigwedge_{\theta'\notin\Theta'}\neg\underline{\psi}_{\theta'}[\overline{\nu},\overline{\mu}]\ \wedge\ t(\theta,\Theta')=\nu\,.$$

This proves the claim.

For sentences $\varphi \in \mathcal{L}$ it follows that there is a formula $\underline{\varphi}$ of $L_{\omega\omega}[<]$ such that for all $\mathfrak{A} \in Q$: $\mathfrak{A}^* \models \varphi \Leftrightarrow \big(\{0, \dots, |A|\}, <, \bar{t}^{\mathfrak{A}}\big) \models \underline{\varphi}$.

The standard Ehrenfeucht-Fraïssé analysis of linear orderings shows that no first-order formula of quantifier rank q can distinguish

$$(n+1, <, t_1, \dots, t_l) \text{ from } (n+1, <, t'_1, \dots, t'_l)$$

if $0 = t_1 < t_2 < \cdots < t_l = n$,
$0 = t'_1 < t'_2 < \cdots < t'_l = n$,

and if for all i, j: either $|t_j - t_i| = |t'_j - t'_i|$ or $|t_j - t_i|, |t'_j - t'_i| \geqslant 2^q$.

By (iii) above we see that this degree of similarity is achieved for structures $\big(n+1, <, \bar{t}^{\mathfrak{A}}\big)$ and $\big(n+1, <, \bar{t}^{\mathfrak{A}'}\big)$ whenever 0, m_1, m_2, m_3, and $n+1 = m_1 + m_2 + m_3 + 1$ are spaced sufficiently far apart. Therefore, no first-order formula can separate those $\big(n+1, <, \bar{t}^{\mathfrak{A}}\big)$ for $\mathfrak{A} \in Q_0$ from those for $\mathfrak{A} \notin Q_0$, and Q_0 cannot be definable in $\mathcal{L}$ either.

4.2 FP+C and the C^k-Invariants

We saw in Section 3.4.1 that interpretability of the L^k-invariants in fixed-point logic on the one hand and representability of fixed-point processes over the invariants on the other hand lead to characterizations of the expressive power of FP and PFP in terms the I_{L^k}. An important aspect of this characterization is the reduction to ordered domains. FP and PFP over not necessarily ordered structures can be analyzed in terms of FP and PFP over the linearly ordered invariants. This section is devoted to the corresponding analysis for fixed-point logics with counting.

The first lemma concerns FP+C-interpretability of the I_{C^k}-invariants. Essentially this is a restatement of the definability properties of the I_{C^k} expressed in Proposition 3.6 above — now put in terms of FP+C.

Lemma 4.14. $I_{C^k}(\mathfrak{A})$ *is* FP+C-*interpretable over* $\mathfrak{A}^*$. *More precisely all the following are* FP+C-*interpretable:*

- *(i) the relational part of* I_{C^k} *as a quotient over the* k*-th power over the first sort.*
- *(ii)* I_{C^k} *as a whole (and being a standard structure) over the second sort.*
- *(iii) the natural projection from the quotient interpretation of the relational part of* I_{C^k} *over the first sort to its representation over the second sort.*

Proof. Proposition 3.6 applies to show (i) since $\mathrm{FP}(Q_{\mathrm{R}}) \subseteq$ FP+C. FP* itself suffices to define the natural projection from the pre-ordering in this interpretation of the relational part of I_{C^k} to the ordered quotient structure over the second sort (iii). Definability of the weight functions through simple counting terms as stated in Proposition 3.6 completes the interpretability of the full invariant as expressed in (ii). □

Let $I^*_{C^k}(\mathfrak{A}) := I_{C^k}(\mathfrak{A}^*)$ stand for the C^k-invariant of $\mathfrak{A}^* \in \mathrm{fin}[\tau]^*$, more precisely of the standard one-sorted encoding of $\mathfrak{A}^*$.

Lemma 4.15. *The $I^*_{C^k}$ are* FP*-interpretable over the I_{C^k}.*

Proof. Since we are dealing with ordered structures it suffices to show that there is a PTIME algorithm that computes $I^*_{C^k}(\mathfrak{A})$ from $I_{C^k}(\mathfrak{A})$. But the inductive generation of $I^*_{C^k}(\mathfrak{A})$ is obviously in PTIME and requires no other data than those encoded in $I_{C^k}(\mathfrak{A})$. The initial stage for instance is based on some fixed ordering of the atomic types of k-tuples in $\mathfrak{A}^*$. Since $\mathfrak{A}^*$ is the disjoint union of $\mathfrak{A}$ with the linear ordering $(|A|+1, <)$, these atomic types can be presented by pairs of atomic types, one in vocabulary τ for the components in the first sort and one in vocabulary $<$ for the components in the second sort. Note that all the relevant information about r-tuples over $\mathfrak{A}$ for $r < k$ is also encoded in $I_{C^k}(\mathfrak{A})$ since the C^k-type of a tuple $(x_1, \ldots, x_r)$ is encoded by the C^k-type of the k-tuple $(x_1, \ldots, x_1, x_1, \ldots, x_r)$ with $r - k$ additional entries x_1. In this fashion the inductive steps in the generation of $I^*_{C^k}(\mathfrak{A})$ are easily simulated over $I_{C^k}(\mathfrak{A})$. □

The lemma is in fact a special case of the following more general observation that can be proved along the same lines. The statement admits further generalizations in the style of Feferman-Vaught Theorems for the L^k- and C^k-theories of finite structures.

Remark 4.16. *The L^k- and C^k-invariants are modular with respect to disjoint unions and direct products in the sense that, for example for I_{C^k} and for disjoint unions, there is a* PTIME *function Σ such that for all $\mathfrak{A}, \mathfrak{B} \in \mathrm{fin}[\tau]$:*

$$I_{C^k}\big(\mathfrak{A} \,\dot{\cup}\, \mathfrak{B}\big) = \Sigma\Big(I_{C^k}(\mathfrak{A}), I_{C^k}(\mathfrak{B})\Big).$$

This implies also that $I_{C^k}\big(\mathfrak{A} \,\dot{\cup}\, \mathfrak{B}\big)$ is FP*-interpretable over the disjoint union of $I_{C^k}(\mathfrak{A})$ and $I_{C^k}(\mathfrak{B})$.*

PTIME$\big(I_{C^k}\big)$ and PSPACE$\big(I_{C^k}\big)$ are defined in analogy with Definition 3.20:

Definition 4.17. PTIME$\big(I_{C^k}\big)$ *and* PSPACE$\big(I_{C^k}\big)$ *stand for the classes of all those queries that are* PTIME, *respectively* PSPACE, *computable in terms of the I_{C^k}.*

More precisely, a boolean query Q on $\mathrm{fin}[\tau]$ is in PTIME$\big(I_{C^k}\big)$ *if membership of $\mathfrak{A}$ in Q is a* PTIME *property of $I_{C^k}(\mathfrak{A})$. A similar characterization can be applied to global relations (of arity at most k) using the extensions of the invariants to the $\mathrm{fin}[\tau; r]$.*

As in the corresponding treatment of the I_{L^k} a query is in PTIME$\big(I_{C^k}\big)$ respectively PSPACE$\big(I_{C^k}\big)$ if it is $C^k_{\infty\omega}$-definable and its natural representation over the relational part of the I_{C^k} can be computed in PTIME, respectively PSPACE over the I_{C^k}. Logically these classes can further be identified with classes $\mathrm{FP}\big(I_{C^k}\big)$ and $\mathrm{PFP}\big(I_{C^k}\big)$ since FP and PFP capture PTIME and PSPACE over the ordered I_{C^k}:

$$\begin{aligned}\mathrm{FP}\big(I_{C^k}\big) &\equiv \mathrm{PTIME}\big(I_{C^k}\big),\\ \mathrm{PFP}\big(I_{C^k}\big) &\equiv \mathrm{PSPACE}\big(I_{C^k}\big).\end{aligned}$$

Syntactically the formulae of $\mathrm{FP}\big(I_{C^k}\big)$ or $\mathrm{PFP}\big(I_{C^k}\big)$ are FP^*-formulae, respectively PFP^*-formulae, in terms of the interpreted I_{C^k}. These logics may thus be regarded as fragments of FP+C or PFP+C. See the proof of the following theorem.

Theorem 4.18. *With the* $\mathrm{FP}(I_{C^k})$ *and* $\mathrm{PFP}(I_{C^k})$ *as characterized:*

$$\begin{aligned}\mathrm{FP{+}C} &\equiv \textstyle\bigcup_k \mathrm{FP}\big(I_{C^k}\big) &&\equiv \textstyle\bigcup_k \mathrm{PTIME}\big(I_{C^k}\big)\\ \mathrm{PFP{+}C} &\equiv \textstyle\bigcup_k \mathrm{PFP}\big(I_{C^k}\big) &&\equiv \textstyle\bigcup_k \mathrm{PSPACE}\big(I_{C^k}\big).\end{aligned}$$

Proof. We prove the equivalences between the logical characterizations. The arguments for FP+C and PFP+C are completely analogous. Consider FP+C. By Lemma 3.19 $\mathrm{FP}(Q_{\mathrm{H}}) \subseteq \bigcup_k \mathrm{FP}\big(I_{C^k}\big)$. An application to the one-sorted encodings of structures in $\mathrm{fin}[\tau]^*$ yields

$$\mathrm{FP}(Q_{\mathrm{H}})^* \subseteq \bigcup_k \mathrm{FP}\big(I^*_{C^k}\big).$$

But $\mathrm{FP}(Q_{\mathrm{H}})^*$ is FP+C by Lemma 4.7. On the right-hand side of the above inclusion we apply Lemma 4.15 and the closure of FP with respect to interpretations to see that $\mathrm{FP}\big(I^*_{C^k}\big) \equiv \mathrm{FP}\big(I_{C^k}\big)$. This proves $\mathrm{FP{+}C} \subseteq \bigcup_k \mathrm{FP}\big(I_{C^k}\big)$. The converse inclusion follows directly from closure of FP+C with respect to interpretations (Proposition 4.8) and interpretability of I_{C^k} in FP+C (Lemma 4.14). □

The analogue of the Abiteboul-Vianu Theorem (Theorem 3.22) follows immediately.

Corollary 4.19. FP+C $\equiv$ PFP+C *if and only if* PTIME = PSPACE.

We may now also infer the basic inclusion $\mathrm{PFP{+}C} \subseteq C^{\omega}_{\infty\omega}$ from the characterization of PFP+C in Theorem 4.18 without getting involved in technicalities.

Corollary 4.20. $\mathrm{FP{+}C} \subseteq \mathrm{PFP{+}C} \subsetneq C^{\omega}_{\infty\omega}$

Proof. It suffices to show that every PFP-definable global relation is closed with respect to $\equiv^{C^k}$ for some k, cf. Lemma 1.33. But this is obvious from $\mathrm{PFP{+}C} \equiv \bigcup_k \mathrm{PFP}\big(I_{C^k}\big)$. Strictness of the inclusion $\mathrm{PFP{+}C} \subsetneq C^{\omega}_{\infty\omega}$ is clear since PFP+C is in PSPACE whereas $C^{\omega}_{\infty\omega}$ expresses even non-recursive queries. □

There is of course also a straightforward direct proof of these inclusions parallel to the proof for FP, PFP $\subseteq L^{\omega}_{\infty\omega}$, cf. Lemma 1.29 and Corollary 1.30. Technically these are more tedious, however, since mixed-type predicates over the $\mathfrak{A}^*$ have to be represented in the one-sorted framework of the $\mathfrak{A}$ themselves. A single type $(1,1)$ formula $\varphi(x,\nu)$ of the two-sorted framework for instance can be decomposed into a family of formulae $\varphi_{n,j}(x)$ for $j \leqslant n$ with the intended meaning that for all $\mathfrak{A}$ of size n: $\varphi[\mathfrak{A}^*] = \bigcup_{0 \leqslant j \leqslant n} \big(\varphi_{n,j}[\mathfrak{A}] \times \{j\}\big)$.

In characterizations like FP+C $\equiv$ PTIME$\big(I_{C^k}\big)$ for fixed points with counting, it is important to note that the size of I_{C^k} is of the same order as the size of the original structure. This essential difference between I_{L^k} and the I_{C^k} leads to a picture that is in sharp contrast with the second theorem of Abiteboul and Vianu for FP and PFP without counting (Theorem 3.24 above). Let PFP+C$|_{\text{poly}}$ be the sublogic of PFP+C in which all occurrences of the PFP-constructor must be such that the limit in the partial fixed-point process is always reached within a polynomial number of steps. The following very simple theorem shows FP+C to be better behaved as a logic for PTIME recursion within $C^{\omega}_{\infty\omega}$ than FP is within $L^{\omega}_{\infty\omega}$.

Theorem 4.21. PFP+C$|_{\text{poly}} \equiv$ FP+C.

Sketch of Proof. Let $\text{PFP}_{X,\overline{x\nu}}\varphi$ be such that the fixed-point process is polynomially bounded. This fixed-point process is then represented by a polynomially bounded PFP-process over the I_{C^k} for some k. Over the ordered I_{C^k} it must therefore be equivalent with an FP-process. Inductively we obtain PFP+C$|_{\text{poly}} \subseteq \bigcup_k \text{FP}(I_{C^k}) \equiv$ FP+C. □

4.3 The Separation from PTIME

It is an important result of Cai, Fürer and Immerman [CFI89] that also FP+C is too weak to capture the class of all PTIME queries on not necessarily ordered finite structures. The construction has been reviewed in Example 2.7.

It is worth to note that on the basis of the present analysis we may infer FP+C $\not\supseteq$ PTIME from the fact that none of the $C^k_{\infty\omega}$ defines *all* queries. This argument is of some interest in its own because it relativizes to many subclasses of the class of all finite structures. The only requirement on the subclass $\mathcal{K}$ is that it admits some kind of padding: some simple construction should be available within $\mathcal{K}$ that allows to increase arbitrarily the size of structures. We choose closure under disjoint unions as a corresponding prerequisite on $\mathcal{K}$ in the statement of the following theorem. It will be clear from the proof that a number of other natural closure conditions would serve just as well.

Theorem 4.22. *Let $\mathcal{K} \subseteq \text{fin}[\tau]$ be a class of finite τ-structures that is closed under disjoint unions. Assume that* FP+C *captures* PTIME *on $\mathcal{K}$, in particular that any* PTIME *computable boolean query on $\mathcal{K}$ is definable by a sentence* *of* FP+C *and hence also by a sentence in* $C^{\omega}_{\infty\omega}$. *Then there is some* k *satisfying the following two (equivalent) conditions.*

(i) I_{C^k} *classifies structures in* $\mathcal{K}$ *up to isomorphism:* *for all* $\mathfrak{A}, \mathfrak{A}' \in \mathcal{K}$: $I_{C^k}(\mathfrak{A}) = I_{C^k}(\mathfrak{A}') \iff \mathfrak{A} \simeq \mathfrak{A}'$.

(ii) In restriction to $\mathcal{K}$, $C^k_{\infty\omega} \equiv L_{\infty\omega}$ *for sentences; in other words, any boolean query on* $\mathcal{K}$ *must be definable in* $C^k_{\infty\omega}$.

Applying this to the class of all finite graphs, and using the result of Cai, Fürer and Immerman just to the effect that no $C^k_{\infty\omega}$ coincides with $L_{\infty\omega}$ on the class of all finite graphs (as expressed in Theorem 2.9) we obtain the following.

Corollary 4.23 (Cai, Fürer, Immerman).
FP+C $\subsetneq$ PTIME, *in fact even* PTIME $\not\subseteq C^{\omega}_{\infty\omega}$.

On the basis of Theorem 4.22 this separation is also obtained as a corollary of recent results of Gurevich and Shelah [GS96]. They prove that in a suitable vocabulary τ there are for each k rigid structures in fin$[\tau]$ that do not admit a $C^k_{\infty\omega}$-definable linear ordering. Again it follows that on (expansions) of these structures no $C^k_{\infty\omega}$ coincides with $L_{\infty\omega}$ (for sentences even).

Note that the only separation results between FP+C and PTIME that can be obtained along the lines of Theorem 4.22 — and these are all there are, as yet — are in fact separations of $C^{\omega}_{\infty\omega} \cap$ PTIME from PTIME.

Proof (of Theorem 4.22). Let $\mathcal{K}$ be as required. Choose some sufficiently fast growing monotone function $f\colon \omega \to \omega$ such that $f(n)$ is computable from n in time polynomial in $f(n)$. Assume that $f(n) > n$ for all n. It follows that there is a PTIME algorithm that recognizes numbers of the form $n\big(f(n) + f(n)^2\big)$ and computes n for these: for given m it suffices to compute $n\big(f(n) + f(n)^2\big)$ for all n with $n^3 \leqslant m$ and check for equality with m.

It further follows that m_1 and m_2 can be computed in PTIME from n and $m_1 f(n) + m_2 f(n)^2$ for any $m_1, m_2 \leqslant n$: simply expand the given number $m = m_1 f(n) + m_2 f(n)^2$ in base $f(n)$ to obtain the m_i as its digits.

We claim that for suitable f the following padded variant of the isomorphism query on $\mathcal{K}$ becomes a PTIME query:

$$Q := \Big\{ \mathfrak{C} \;\Big|\; \mathfrak{C} \simeq \underbrace{\mathfrak{A} \,\dot\cup \ldots \dot\cup\, \mathfrak{A}}_{m}, \text{ where } m = f(n) + f(n)^2, n = |A| \Big\}.$$

The intended algorithm first checks whether the size of an input $\mathfrak{C}$ is of the form $n\big(f(n) + f(n)^2\big)$ and computes n in this case. It then checks for all isomorphism types of connected[1] τ-structures $\mathfrak{D}$ of size at most n how many connected components of $\mathfrak{C}$ are isomorphic with $\mathfrak{D}$ (and that $\mathfrak{C}$ has no components of size greater than n). This is done in time polynomial in $|\mathfrak{C}|$

[1] A structure is called connected if it is not the disjoint union of two other structures.

provided $f(n)$ is sufficiently large for n; the precise meaning of 'sufficiently large' has to take into account the arities in τ.

Let $\nu(\mathfrak{D})$ be the corresponding number for each $\mathfrak{D}$. Then $\mathfrak{C} \in Q$ if and only if all $\nu(\mathfrak{D})$ are of the form $\nu(\mathfrak{D}) = \mu(\mathfrak{D})\big(f(n) + f(n)^2\big)$ for appropriate $\mu(\mathfrak{D}) \leqslant n$. Necessity of this condition is clear. For sufficiency observe that, if $\nu(\mathfrak{D}) = \mu(\mathfrak{D})\big(f(n) + f(n)^2\big)$ for all $\mathfrak{D}$, then $\mathfrak{C}$ is of the required form if for $\mathfrak{A}$ one takes the disjoint union of $\mu(\mathfrak{D})$ copies of each $\mathfrak{D}$.

By assumption Q therefore is definable in some $C^k_{\infty\omega}$. But the above characterization of $\mathfrak{C} \in Q$ through the $\nu(\mathfrak{D})$ also implies that for any two $\mathfrak{A}, \mathfrak{B} \in \mathrm{fin}[\tau]$ of the same size n,

$$\mathfrak{C} = \underbrace{\mathfrak{A} \,\dot{\cup} \ldots \dot{\cup}\, \mathfrak{A}}_{f(n)} \;\dot{\cup}\; \underbrace{\mathfrak{B} \,\dot{\cup} \ldots \dot{\cup}\, \mathfrak{B}}_{f(n)^2}$$

is in Q if and only if $\mathfrak{A} \simeq \mathfrak{B}$. It follows from Remark 4.16 on the other hand that $I_{C^k}(\mathfrak{C})$ is a function of $I_{C^k}(\mathfrak{A})$ and $I_{C^k}(\mathfrak{B})$, so that $\mathfrak{A} \simeq \mathfrak{B}$ is determined by $I_{C^k}(\mathfrak{A})$ and $I_{C^k}(\mathfrak{B})$. This implies claim (i) of the theorem, and equivalence of (i) and (ii) is obvious. The argument given here is a structural variant of the so-called *padding technique* that is often useful in complexity considerations. □

The results of this chapter show that FP+C is the right logic for PTIME recursion in the world of $C^\omega_{\infty\omega}$. In this respect its relation to $C^\omega_{\infty\omega}$ resembles that of FP to $L^\omega_{\infty\omega}$. It is known from the result of Cai, Fürer and Immerman that real PTIME is not within $C^\omega_{\infty\omega}$. On the other hand all known separation results for FP+C from PTIME are separations of $C^\omega_{\infty\omega} \cap$ PTIME from PTIME. The question that arises at this point is the following:

> Does FP+C capture PTIME $\cap\, C^\omega_{\infty\omega}$, the class of all those queries that are both PTIME computable and definable in $C^\omega_{\infty\omega}$?

More suggestively:

> Does FP+C capture PTIME in the world of $C^\omega_{\infty\omega}$?

This question is further explored in the last two chapters. Note that the same question with FP and $L^\omega_{\infty\omega}$ in the place of FP+C and $C^\omega_{\infty\omega}$ can be answered negatively unless PTIME = PSPACE. Obviously $\mathrm{PFP}|_{\mathrm{poly}} \subseteq \mathrm{PTIME} \cap L^\omega_{\infty\omega}$, but $\mathrm{PFP}|_{\mathrm{poly}} \subseteq \mathrm{FP}$ only if PTIME = PSPACE by the second theorem of Abiteboul and Vianu. There is a reasonable variant of the issue that remains an open problem for FP and $L^\omega_{\infty\omega}$, too. We shall come back to these issues in Chapter 6. In the last Chapter we find positive solutions to such questions in the very restricted case of just two variables, i.e. for $L^2_{\infty\omega}$ and $C^2_{\infty\omega}$.

4.4 Other Characterizations of FP+C

It may be a further indication of the naturalness of FP+C as a level of expressiveness within PTIME that it admits several different equivalent logical characterizations and also a natural algorithmic characterization. We here only indicate some of these briefly. More detailed accounts can be found in [GO93] and [Ott96a], respectively.

Among the logical variations we mention the following:

(a) FP+C can be obtained as a straightforward extension of Datalog. For our purposes Datalog is the logic of positive Horn-clause programs with the least fixed-point semantics. Its counting extension is based on the two-sorted variants of structures in $\text{fin}[\tau]^*$ and allows the use of counting terms and cardinality comparisons in the sense of $\leqslant$ in clauses. It is not difficult to see that the counting extension leads to closure under negation. It follows that this extension of Datalog comprises the full power of fixed-point logic and thus is semantically equivalent with FP+C.
(b) The approach to extend finite structures with standard sorts, like the arithmetical second sort of the structures in $\text{fin}[\tau]^*$, has been carried much further in the framework of *meta-finite structures* put forward by Grädel and Gurevich in [GG95]. Here finiteness of the second standardized sorts is given up in order to obtain a more uniform modelling for issues on finite structures that essentially involve reference to infinite standard structures (like the natural or the real numbers). In order to obtain an adequate limitation on the access to the infinite standard domains, recursive processes like those in fixed-point are restricted to the finite relational domain. The infinite standard parts are accessed through terms and *multiset operations*. The latter can roughly be described as arithmetical operations that are performed on weight functions from the finite relational domain to the infinite standard part. It turns out that FP+C can be isolated in this framework by taking arithmetic on the natural numbers $(\omega, <, +, \cdot)$ for the infinite standard structure, with exactly the PTIME multiset operations. It is shown in [GG95] that the expressive power of fixed-point logic in this meta-finite frame coincides with FP+C.

We mention two more characterizations of different kinds in slightly greater detail. One is in terms of uniform sequences of formulae, the other by means of a computational model.

P-uniform sequences of formulae. Logical characterizations in terms of sequences of formulae are proposed and investigated in the work of Immerman, see for instance [Imm82]. Let $\text{fin}_n[\tau]$ stand for the restriction of $\text{fin}[\tau]$ to structures of size n. The idea is to associate for instance with a boolean query $Q \subseteq \text{fin}[\tau]$ a *sequence* of sentences $(\varphi_n)_{n\geqslant 1}$ in some logic $\mathcal{L}[\tau]$ such that for all sizes n:

$$Q \cap \text{fin}_n[\tau] = \Big\{\mathfrak{A} \in \text{fin}_n[\tau] \;\Big|\; \mathfrak{A} \models \varphi_n\Big\}.$$

A priori this is a completely non-uniform notion of logical definability. Restrictions on the constituent formulae φ_n in terms of quantifier rank, numbers of variables and size (all regarded as functions in n) or constructibility criteria for the mapping $n \mapsto \varphi_n$ serve to employ this approach as a tool in the logical analysis of complexity. It turns out that FP+C and PFP+C are isolated by very natural uniformity conditions on sequences. Note that in the presence of counting quantifiers and for sequences of formulae $\varphi_n \in C^k_{\omega\omega}$ the semantics given to the sequence is that of $\bigvee_{n\geqslant 1}\big(\exists^{=n}x\, x\!=\!x \wedge \varphi_n\big) \in C^k_{\infty\omega}$.

Definition 4.24. *Call a sequence $(\varphi_n)_{n\geqslant 1}$ of formulae in some $C^k_{\omega\omega}$* PTIME-uniform*, respectively* PSPACE-uniform*, if φ_n is constructible in time, respectively space, polynomial in n. Let* PTIME–$C^\omega_{\infty\omega}$ *and* PSPACE–$C^\omega_{\infty\omega}$ *stand for the sublogics of $C^\omega_{\infty\omega}$ corresponding to all* PTIME-*, respectively* PSPACE-*uniform sequences.*

Clearly PTIME–$C^\omega_{\infty\omega} \subseteq$ PSPACE–$C^\omega_{\infty\omega} \subseteq C^\omega_{\infty\omega}$. The following is proved in [Ott96a].

Proposition 4.25. FP+C $\equiv$ PTIME–$C^\omega_{\infty\omega}$ *and* PFP+C $\equiv$ PSPACE–$C^\omega_{\infty\omega}$.

This is quite unlike the situation for FP and PFP themselves: trivial examples involving pure sets show for instance that FP is properly contained in the correspondingly defined PTIME–$L^\omega_{\infty\omega}$.

A computational characterization. Finally there is a natural computational model whose PTIME and PSPACE restrictions coincide with FP+C and PFP+C, respectively. This model is the obvious generalization of the relational computational model of Abiteboul and Vianu [AV91] that incorporates counting operations in a generic manner. Let us call the machines under consideration *relational machines with counting.* We give a brief sketch. A relational machine with counting consists of two components. First there is a *relational store* with a fixed number of relational registers of fixed arities. These can hold sets of tuples from the domain of the input structure. Among these relational registers there are specified ones that are initialized to represent the given predicates in the input structure. The others are initially empty. In any case, at each stage of the computation, the content of a relational register is a relation over the input domain. The second component of the machine resembles an ordinary Turing machine with a work tape with a read–write head, an extra communication tape with a write–only head, and the usual finite state control. The interaction between the two components is the following.

- Each transition, as laid down in the transition table of the Turing control, may depend not only on the current internal state and symbol read on the work tape but also on the information which of the relational registers are currently empty. These implicit emptiness queries constitute the only flow of information from the relational part of the machine to the Turing component.
- The execution of a transition may involve not only the printing of tape symbols and movements of the heads but may also include one of several

update operations on the relational store. The operations available here are the following:

– copy and move operations between relational registers.
– boolean operations on the current contents of specified relational registers (e.g. union and complementation).
– operations corresponding to the natural action of the permutation groups S_r on the contents of r-ary registers.
– counting projections.

Counting projections take as input a numerical parameter ν whose current value is read from the communication tape. The content of a prescribed relational register R is then replaced by all those tuples for which there are at least ν substitutes for the first component that are currently in R: $R' := \{\overline{a} \mid \exists^{\geqslant \nu} b(\overline{a}\frac{b}{1} \in R)\}$. Only in this operation does the present model extend the one proposed by Abiteboul and Vianu. Their model only allows ordinary existential projections which appear here as a special case for $\nu = 1$.

Computations of these machines are formalized in the natural manner. The result of a computation that is to produce a boolean value can be encoded in the final state reached by the Turing control. For machines that are to compute an r-ary query, the output is the content of one specified r-ary relational register when the machine reaches its halting state.

This model of computation is entirely isomorphism-preserving ('generic' is the term usually applied in the literature). Any isomorphism between input structures naturally extends to all stages of the computation, so that the resulting computations are not only equivalent but really isomorphic themselves.

Complexities for this model are defined in terms of the Turing component. PTIME and PSPACE for the relational machines with counting comprise those queries that are computable by one of these machines within a number of steps, respectively with the use of a number of tape cells (of the Turing component) that is polynomially bounded in the size of the input structure.

Theorem 4.26. *On finite relational structures,* PTIME *and* PSPACE *for the relational machines with counting exactly correspond to* FP+C *and* PFP+C.

In particular FP+C and PFP+C are the polynomial time and space restrictions of a generic model of computation — a situation that in a sense is ruled out by the second theorem of Abiteboul and Vianu, Theorem 3.24 above, for FP and PFP themselves.

Attributions and remarks. FP+C — roughly in our formalization — is implicit in the work of Immerman, in particular see [Imm87a]. The present explicit form was first presented in [GO93]. Most of the material treated in this chapter can also be found in [Ott96a] and [GO93]. The latter source should be consulted in particular for those characterizations of FP+C that are only sketched here.

5. Related Lindström Extensions

In this chapter FP+C is shown to be more expressive than the natural extensions of fixed-point logic by cardinality Lindström quantifiers.

• Section 5.1 introduces a structural padding technique that is suitable for the proof of this separation result. More generally, this technique serves to expose weaknesses of quantifier extensions in the case that these quantifiers do not have the right scaling properties with respect to certain extensions of structures.

• This technique is applied in Section 5.2 to show that $\mathrm{FP}(\mathcal{Q}_{\mathrm{card}})$ cannot express all FP+C-definable boolean queries. The same applies to $\mathrm{FP}(\mathcal{Q}^{\sim}_{\mathrm{card}})$ with quantifiers for all cardinality properties based on the counting of equivalence classes. In fact the separation even establishes that not all of FP^* can be captured by these quantifier extensions.

• In Section 5.3 we apply the padding technique to derive corollaries concerning the weakness of two other quantifier classes. The classes of all properties of rigid structures and that of all properties of sparse structures, respectively, are shown to fall short of FP^* and in particular of PTIME.

In the previous chapter FP+C has been characterized as the natural extension of fixed-point logic that incorporates expressive means for dealing with cardinalities and corresponding arithmetic. Recall that a main feature of the formalization was the introduction of a second, arithmetical sort. This type of a *functorial extension* — based partly on the manipulation of the structures under consideration — is intuitively different from the established formalism for extensions in abstract model theory, namely that of Lindström extensions or extensions through generalized quantifiers. Can this difference in appearance be substantiated in more rigorous terms? There is some sense in which this cannot be achieved: it is a known fact that the Lindström approach to extensions of logics is sufficiently general to describe any reasonable extension of first-order logic, more precisely any extension with the appropriate closure properties. No doubt therefore FP+C is equivalent with a Lindström extension of first-order logic, and also with a Lindström extension of fixed-point logic. As FP+C is a logic with recursive syntax and semantics these Lindström extensions can trivially be chosen to use recursive families of quantifiers. That one is forced to consider extensions by infinite families of quantifiers follows with an argument of Dawar and Hella [DH94] that applies to show that FP+C cannot be equivalent with a finite Lindström extension (see Theorem 5.9 below). The standard modelling of a logic $\mathcal{L} \supseteq$ FP with the right closure properties as a Lindström extension essentially turns each individual $\mathcal{L}$-definable class into a quantifier. Clearly this is unsatisfactory: the resulting presentation of FP+C as a Lindström extension of FP is quite artificial. It is not at all clear, however, which kinds of Lindström extensions should be considered *natural*. Two different types of criteria come to mind.

Syntactic criteria. One may consider certain *uniform sequences* of quantifiers. These are meant to adjoin the same structural property in varying context. Uniform sequences as considered for instance in [Daw95a] consist of all powers of a given quantifier and capture one structural property across all arities, or as applied to interpreted structures in any power. Compare Section 1.6.2. The usual way in which the transitive closure operator is adjoined to first-order logic to get transitive closure logic provides a natural example. Transitive closures are made definable for binary relations interpreted in any power of the universe.

While FP+C cannot be a finite extension of FP it is conceivable that it is obtained as an extension by finitely many uniform sequences of quantifiers. Indeed, it follows from Dawar's work that a class or logic, that is recursively presented (in some sufficiently strong sense; compare remarks in connection with Definition 1.7) and has natural closure properties, is equivalent with an extension of FP and even of first-order logic by just a single uniform series of quantifiers. In the general construction the quantifier giving rise to such a sequence embodies an enumeration of all queries that are to be captured. In special cases, as for instance for FP itself one may also abstract such a quantifier from typical and natural problems that are complete under appropriate logical reductions, cf. [Dah87, Gro95]. Whether such natural problems exist for FP+C, relative either to FP or to first-order, remains open.

Semantic criteria. One may also impose purely semantic conditions on the quantifiers adjoined. The investigations of this chapter are of this kind. In connection with fixed-point with counting there is an obvious issue in this line:

> Can FP+C be obtained as an extension of FP by cardinality Lindström quantifiers, i.e. by quantifiers whose semantics is entirely defined in terms of cardinalities of predicates?

Indeed, FP with the class of all Ptime cardinality Lindström quantifiers is the *natural a priori candidate* to capture a counting extension of FP in the Lindström formalism. Compare Definitions 1.52 and 1.54 for (quotient) cardinality quantifiers.

The main point of this chapter is that even the extension of FP by all cardinality Lindström quantifiers does not comprise all of FP+C, in fact not even all of FP*: $\mathrm{FP}(\mathcal{Q}_{\mathrm{card}}) \not\supseteq \mathrm{FP}^*$. Admitting further all quantifiers that capture cardinality properties in quotient interpretations — for the counting of equivalence classes rather than tuples — does not help either, even $\mathrm{FP}(\mathcal{Q}^{\sim}_{\mathrm{card}}) \not\supseteq \mathrm{FP}^*$.

Theorem 5.1. $\mathrm{FP{+}C} \not\subseteq \mathrm{FP}(\mathcal{Q}_{\mathrm{card}})$. *In particular the extension of* FP *by all* PTIME *cardinality Lindström quantifiers is strictly weaker than* FP+C. *These separations also hold for* $\mathrm{FP}(\mathcal{Q}^{\sim}_{\mathrm{card}})$, *the extension by all quotient cardinality Lindström quantifiers.*

It follows with Lemma 1.55 that the extension $\mathrm{FP}(\mathcal{Q}_{\mathrm{mon}})$ of fixed-point logic by the class of all monadic Lindström quantifiers does not contain PTIME, and that similarly all quantifiers obtained from monadic ones through generalized interpretations cannot suffice. The latter extension is in fact equivalent with $\mathrm{FP}(\mathcal{Q}^{\sim}_{\mathrm{card}})$ by Remark 1.56. We mention in this context the work of Kolaitis and Väänänen [KVä95] on extensions of the $L^k_{\infty\omega}$ by monadic quantifiers that bind single formulae (simple monadic quantifiers). Using sophisticated combinatorial techniques they obtain interesting separation results within the realm of monadic quantifiers, for instance that the Härtig quantifier is not expressible in any extension of $L^{\omega}_{\infty\omega}$ by finitely many simple monadic quantifiers.

The present results are obtained with a technique that resembles so-called *padding arguments* in complexity theory. Intuitively the situation of Theorem 5.1 can be understood through the following. With FP+C the results of counting operations can be processed recursively, and this FP-recursion (over the arithmetical sort) is full PTIME recursion in terms of the size of the universe. The FP-recursion captured by any sentence in an extension of FP by $C^{\omega}_{\infty\omega}$-definable quantifiers, on the other hand, is polynomially bounded in the size of the quotient of the k-th power of the universe with respect to $\equiv^{C^k}$ for some k. The latter is the size of the *relational part* of I_{C^k}. (This situation is reminiscent of that exhibited by FP; there a gap between the size of $\mathfrak{A}$ and of $I_{L^k}(\mathfrak{A})$ accounts for the complexity behaviour described in the second theorem of Abiteboul and Vianu.)

In the case of cardinality Lindström quantifiers this gap can be manifested unconditionally to obtain the desired separation. The structures employed in these arguments are trivial extensions of ordered structures, with an increase in the size without any gain in internal relational structure, just as in padding arguments. FP with cardinality Lindström quantifiers is shown to have not the right *scaling properties* with respect to such extension.

5.1 A Structural Padding Technique

We consider functors that scale finite structures in size without otherwise adding structural complexity. Taking the disjoint sum with a pure set is a typical example. This operation increases the size but as for definable predicates, nothing is gained. We formalize this as follows. Consider a functor

$$\Gamma\colon \mathrm{fin}[\tau] \times \omega \longrightarrow \mathrm{fin}[\tau].$$

The second argument of Γ will serve as a scaling parameter for the desired extensions. The main example below is that of $\Gamma(\mathfrak{A}, n)$ being a trivial product of $\mathfrak{A}$ with the pure set $n = \{0, \dots, n-1\}$. Assume further that for each r there is an encoding scheme that maps r-ary predicates R over $\Gamma(\mathfrak{A}, n)$ that are closed under automorphisms of $\Gamma(\mathfrak{A}, n)$ to tuples of predicates $[R]$ on $\mathfrak{A}$. We want to regard $[R]$ as an encoding or a *pull-back* for the values of global relations over the $\Gamma(\mathfrak{A}, n)$.

Definition 5.2. *A* good encoding scheme for Γ *is a mapping* $[\ ]$ *sending automorphism invariant* R *on* $\Gamma(\mathfrak{A}, n)$ *to tuples* $[R] = \big(R^1, \dots, R^l\big)$ *on* $\mathfrak{A}$*, such that* $(\mathfrak{A}, [R])$ *and* n *determine* $\big(\Gamma(\mathfrak{A}, n), R\big)$ *up to isomorphism, and such that*

(i) $[\ ]$ *is monotone:* $R_1 \subseteq R_2$ *implies* $R_1^i \subseteq R_2^i$ *for* $i = 1, \dots l$.
(ii) $[\ ]$ *is compatible with first-order definability in the following sense: if* R *is first-order definable from some global relations* $R_1, \dots, R_k$ *over the* $\Gamma(\mathfrak{A}, n)$*, then the encoding relations* $[R]$ *for* R *are first-order definable over the* $\mathfrak{A}$ *from the encodings* $[R_i]$ *of the* R_i.

More precisely, (ii) means, that for first-order formula $\varphi(X_1, \dots, X_k, \overline{x})$ there are first-order formulae φ_*^i such that for all sufficiently large n and for all $R_1, \dots, R_k$ that are automorphism closed over $\Gamma(\mathfrak{A}, n)$:

$$\big[\varphi[\Gamma(\mathfrak{A}, n), R_1, \dots, R_k]\big] = \big(\varphi_*^i[\mathfrak{A}, [R_1], \dots, [R_k]]\big)_{i=1,\dots,l}\,.$$

Note the uniformity with respect to n that is expressed in this notion.

We shall below need to extend the notion of good encodings to allow for parameters in the $\Gamma(\mathfrak{A}, n)$, see Definition 5.4.

Consider two examples: the disjoint sum and the trivial product with the pure set n.

$$\begin{aligned} (\mathfrak{A}, n) &\longmapsto \mathfrak{A} \,\dot{\cup}\, n\,, \text{ and} \\ (\mathfrak{A}, n) &\longmapsto \mathfrak{A} \otimes n\,. \end{aligned}$$

• $\mathfrak{A} \,\dot{\cup}\, n$: if $\mathfrak{A} = (A, R_1^{\mathfrak{A}}, \dots, R_s^{\mathfrak{A}})$ then $\mathfrak{A} \,\dot{\cup}\, n = (A \,\dot{\cup}\, n, R_1^{\mathfrak{A}}, \dots, R_s^{\mathfrak{A}})$ is the disjoint union with the set n.
• $\mathfrak{A} \otimes n$: the universe of $\mathfrak{A} \otimes n$ is the product $A \times n$. Let $\pi_1\colon A \times n \to A$ and $\pi_2\colon A \times n \to n$ denote the natural projections to the factors as well as their extensions to higher powers as for instance in $\pi_1\colon (A \times n)^r \to A^r$. Then $\mathfrak{A} \otimes n = (A \times n, R_1^{\mathfrak{A}} \otimes n, \dots, R_s^{\mathfrak{A}} \otimes n)$, where $R_i^{\mathfrak{A}} \otimes n = \pi_1^{-1}\big(R_i^{\mathfrak{A}}\big)$.

Good encoding schemes are available for both functors. Consider the trivial product with n. Clearly a tuple $\overline{b} \in (A \times n)^r$ is described up to automorphisms of $\mathfrak{A} \otimes n$ by the pair $\big(\pi_1(\overline{b}), \mathrm{eq}\big(\pi_2(\overline{b})\big)\big)$ consisting of its projection to

$\mathfrak{A}$ and the equality type of its projection to n. Let R be an r-ary predicate over $\mathfrak{A} \otimes n$ that is closed under automorphisms of $\mathfrak{A} \otimes n$. Then R is faithfully encoded by the tuple

$$[R] = (R^e)_{e \in \mathrm{Eq}(r)}$$
$$\text{where} \quad R^e = \big\{\pi_1(\bar{b}) \mid \bar{b} \in R, \mathrm{eq}\big(\pi_2(\bar{b})\big) = e\big\}.$$

Actually R is easily reconstructed from the R^e as

$$R = \big\{\bar{b} \mid \pi_1(\bar{b}) \in R^e \text{ for } e = \mathrm{eq}\big(\pi_2(\bar{b})\big)\big\}.$$

Monotonicity and compatibility with first-order transformations can be checked immediately. For instance, if $R = \{\overline{x} \mid \exists y R_1 \overline{x} y\}$, then R^e is the union over all sets $\{\overline{x} \mid \exists y(\overline{x} y \in R_1^{e'})\}$ where e' extends e to $r+1$ variables.

For trivial sums with n, a similar decomposition of predicates with respect to equality types of those parts of tuples that lie outside A would be a natural encoding. The universe A of $\mathfrak{A}$ is not definable as a subset of $A \dot{\cup} n$, however, so that the decomposition should be applied with respect to the parts lying outside the field of the $R_i^{\mathfrak{A}}$. We leave out the details, since in the explicit arguments of this chapter we choose to work with trivial products.

Since good encodings uniformly translate first-order manipulations on global relations to first-order manipulations on their encodings we have the following pull-back for fixed-point logic.

Lemma 5.3. *If there is a good encoding scheme for $\Gamma\colon \mathrm{fin}[\tau] \times \omega \to \mathrm{fin}[\tau]$, then* FP *over the $\Gamma(\mathfrak{A}, n)$ is captured by* FP *over the $\mathfrak{A}$ themselves. This means, in the case of boolean queries, that for any sentence $\varphi \in \mathrm{FP}[\tau]$ there is a sentence $\varphi_* \in \mathrm{FP}[\tau]$ such that for all sufficiently large n*

$$\Gamma(\mathfrak{A}, n) \models \varphi \quad \Longleftrightarrow \quad \mathfrak{A} \models \varphi_*.$$

Proof. Inductively it suffices to show that also FP-applications can be simulated at the level of the encodings $[R]$. Consider the formula $\mathrm{FP}_{X,\overline{x}}\varphi(X, \overline{x})$ where we assume that $\overline{x}$ contains all free first-order variables of φ (compare Lemma 1.28). Suppose that $\varphi_*^1, \ldots, \varphi_*^l$ are such that for all automorphism invariant P over $\Gamma(\mathfrak{A}, n)$ (with sufficiently large n)

$$\Big[\varphi[\Gamma(\mathfrak{A}, n), P]\Big] = \Big(\varphi_*^1\big[\mathfrak{A}, [P]\big], \ldots, \varphi_*^l\big[\mathfrak{A}, [P]\big]\Big).$$

Then the encoding tuple $\big[\mathrm{FP}_{X,\overline{x}}\varphi(X, \overline{x})\big]$ for $\mathrm{FP}_{X,\overline{x}}\varphi(X, \overline{x})$ is obtained over $\mathfrak{A}$ as the simultaneous fixed point determined by the system $\varphi_*^1, \ldots, \varphi_*^l$ (when appropriately initialized to $[\emptyset]$). Compare Example 1.27 for fixed-point systems, and the proof of Lemma 2.22 about initialization. $\Box$

If Γ scales the size of the $\Gamma(\mathfrak{A}, n)$ with n then the lemma implies that the power of FP does not correctly scale with the size of the $\Gamma(\mathfrak{A}, n)$, since FP-recursion on $\Gamma(\mathfrak{A}, n)$ collapses to FP-recursion on $\mathfrak{A}$ in a manner independent of n. Our aim is to extend this phenomenon to quantifier extensions of FP.

Consider a Lindström quantifier Q of type $\sigma = \{R_1, \ldots, R_k\}$. Without loss of generality we may assume that applications of Q are in the following normal form:

$$\psi(\overline{z}) = Q\big(\overline{x}^{(i)}; \varphi_i(\overline{z}, \overline{x}^{(i)})\big)_{i=1,\ldots,k} .$$

While the $\varphi_i[\Gamma(\mathfrak{A}, n)]$ are invariant under automorphisms and therefore covered by our encoding scheme, this need not be true of the predicates

$$\varphi_i[\Gamma(\mathfrak{A}, n), \overline{c}] = \big\{\overline{b} \in \Gamma(\mathfrak{A}, n) \;\big|\; \Gamma(\mathfrak{A}, n) \models \varphi_i[\overline{c}, \overline{b}]\big\}$$

for fixed parameters $\overline{c}$. But Q is applied to predicates of this type in the evaluation of ψ over $\Gamma(\mathfrak{A}, n)$. Note that the resulting predicate

$$\psi[\Gamma(\mathfrak{A}, n)] = \big\{\overline{c} \in \Gamma(\mathfrak{A}, n) \;\big|\; \Gamma(\mathfrak{A}, n) \models \psi[\overline{c}]\big\},$$

however, will again be automorphism invariant over $\Gamma(\mathfrak{A}, n)$.

In order to deal with the intermediate predicates $\varphi_i[\Gamma(\mathfrak{A}, n), \overline{c}]$ we consider an extension of our encoding schemes that covers such *fibres* of automorphism closed predicates. For predicates R and parameter tuple $\overline{c}$ let $R|\overline{c}$ denote the *fibre of R over $\overline{c}$*:

$$R|\overline{c} = \big\{\overline{b} \;\big|\; R\overline{c}\overline{b}\big\}.$$

Definition 5.4. *A* good encoding scheme with parameters *for Γ extends a good encoding scheme to a mapping $[\ ]$ that encodes parameter defined fibres of automorphism invariant R over $\Gamma(\mathfrak{A}, n)$ through tuples of predicates $[R]_{\overline{c}} = \big(R^1_{\overline{c}}, \ldots, R^l_{\overline{c}}\big)$, such that*

(i) $(\mathfrak{A}, [R]_{\overline{c}})$ and n determine $\big(\Gamma(\mathfrak{A}, n), R|\overline{c}, \overline{c}\big)$ up to isomorphism.
(ii) the $[R]_{\overline{c}}$ are uniformly first-order interdefinable with $[R]$ over $\mathfrak{A}$: there is an l-tuple of first-order formulae $\overline{\chi}$ such that
  - *(a) $\big\{[R]_{\overline{c}} \;\big|\; \overline{c} \in \Gamma(\mathfrak{A}, n)\big\} = \big\{\overline{\chi}\big[\mathfrak{A}, [R], \overline{a}\big] \;\big|\; \overline{a} \in \mathfrak{A}\big\}$.*
  - *(b) for automorphism closed P over $\Gamma(\mathfrak{A}, n)$, $[P]$ is first-order definable from the set of those $\overline{a}$ for which $\overline{\chi}\big[\mathfrak{A}, [R], \overline{a}\big] \in \big\{[R]_{\overline{c}} \;\big|\; \overline{c} \in P\big\}$.*

For $\Gamma(\mathfrak{A}, n) = \mathfrak{A} \otimes n$ such an extension of the encoding scheme considered above is obtained as follows. For R of arity $t+r$ we used $[R] = (R^e)_{e \in \mathrm{Eq}(t+r)}$. This extends to cover encodings of $R|\overline{c}$ with parameter tuples $\overline{c}$ of arity t, if we choose for $[R]_{\overline{c}}$ the tuple of predicates

$$R^e_{\overline{c}} := \Big\{\pi_1(\overline{c}, \overline{b}) \;\Big|\; \overline{b} \in R|\overline{c}, \mathrm{eq}\big(\pi_2(\overline{c}, \overline{b})\big) = e\Big\},$$

for $e \in \mathrm{Eq}(t+r)$. $R^e_{\overline{c}}$ can be non-empty only for those e that extend $\mathrm{eq}\big(\pi_2(\overline{c})\big)$ to $t+r$ variables. Note for (ii) above that, for such e, each $R^e_{\overline{c}}$ is first-order interdefinable with $\pi_1(\overline{c})$ and the fibre of R^e at $\pi_1(\overline{c})$. Therefore, $[R]_{\overline{c}}$ is first-order definable in terms of $[R]$, $\pi_1(\overline{c})$ and $\mathrm{eq}\big(\pi_2(\overline{c})\big)$.

For quantifier applications $\psi(\overline{z}) = Q(\overline{x}^{(i)}; \varphi_i(\overline{z}, \overline{x}^{(i)}))_{i=1,\dots,k}$ it remains to capture the semantics of Q over $\Gamma(\mathfrak{A}, n)$ in terms of the encodings of the fibres $\varphi_i[\Gamma(\mathfrak{A}, n), \overline{c}] = \varphi_i[\Gamma(\mathfrak{A}, n)]|\overline{c}$ over the base structure $\mathfrak{A}$.

Assume that this is possible. Then one can pass from the encodings of the $\varphi_i[\Gamma(\mathfrak{A}, n)]$ to the encodings of all $\varphi_i[\Gamma(\mathfrak{A}, n), \overline{c}]$, through a first-order variation of the parameters in the $\overline{\chi}$ according to (ii) (a) in Definition 5.4. If it can be determined in terms of these, whether $\big(\Gamma(\mathfrak{A}, n), (\varphi_i[\Gamma(\mathfrak{A}, n), \overline{c}])\big)$ is in Q, then (ii) (b) serves to obtain the encoding of $\psi[\Gamma(\mathfrak{A}, n)]$ from the collection of those choices of parameters for which this is the case.

Quantifiers, and in particular cardinality quantifiers, cannot be expected to display an independence of the scaling parameter n as expressed for FP in Lemma 5.3. But the n-dependence of quantifiers Q can be isolated in a non-uniform way.

We now fix some Γ and a good encoding scheme $R \mapsto [R]$, with parameter extensions $R, \overline{c} \mapsto [R]_{\overline{c}}$ for Γ. Let Q be a Lindström quantifier of type $\sigma = \{R_1, \dots, R_k\}$. Introduce a series of quantifiers Q_n^Γ where

$$Q_n^\Gamma = \left\{ \big(A, [R'_1]_{\overline{c}}, \dots, [R'_k]_{\overline{c}}\big) \;\middle|\; \begin{array}{l} (\Gamma(\mathfrak{A}, n), R'_1|\overline{c}, \dots, R'_k|\overline{c}) \restriction \sigma \in Q, \\ \text{the } R'_i \simeq\text{-closed on } \Gamma(\mathfrak{A}, n) \end{array} \right\}.$$

Here the R'_i are of arity $t + r_i$ if the arity of R_i is r_i and if parameter tuples of arity t are considered. The type of the Q_n^Γ is that obtained from the encoding scheme $R, \overline{c} \mapsto [R]_{\overline{c}}$ applied to the R'_i. This type accordingly depends on the arity of parameter tuples $\overline{c}$ that are admitted; we suppress this dependence in our notation.

Let Q_* stand for a quantifier symbol of appropriate type, i.e. a syntactic object that behaves just like one of the Q_n^Γ. With the arguments from above, the following extension of Lemma 5.3 is obtained:

Lemma 5.5. *For any sentence $\varphi \in \mathrm{FP}(Q)[\tau]$ there is a sentence $\varphi_*(Q_*) \in \mathrm{FP}(Q_*)[\tau]$ such that for all sufficiently large n*

$$\Gamma(\mathfrak{A}, n) \models \varphi \quad \Longleftrightarrow \quad \mathfrak{A} \models \varphi_*(Q_n^\Gamma),$$

where $\varphi_*(Q_n^\Gamma)$ is the sentence $\varphi_*(Q_*)$ with the semantics of Q_n^Γ for the dummy quantifier Q_*.

The claim applies similarly to families $\mathcal{Q}$ of quantifiers. A separation of $\mathrm{FP}(\mathcal{Q})$ from a logic $\mathcal{L}$ can be achieved if it can be shown that the complexity of the quantifiers Q_n^Γ falls short of the complexity attainable in $\mathcal{L}$ on the $\Gamma(\mathfrak{A}, n)$ for large n. Since we pass from formulae $\varphi \in \mathrm{FP}(\mathcal{Q})$ to a family of formulae $\varphi_*(Q_n^\Gamma)$ with an a priori non-uniform dependence of the semantics of the Q_n^Γ on n, these arguments are adapted to non-uniform complexity considerations.

We apply this strategy to $\mathcal{L} = \mathrm{FP}^*$ in a context in which FP^* over the $\Gamma(\mathfrak{A}, n)$ captures full PTIME. We shall also use that $|\Gamma(\mathfrak{A}, n)| \geqslant n$ and that $\mathfrak{A}$ itself and n (as a number in the second sort) are uniformly FP^*-interpretable over $\Gamma(\mathfrak{A}, n)^*$.

We show that this is the case for trivial products $\Gamma : (\mathfrak{A}, n) \mapsto \mathfrak{A} \otimes n$, if $\mathfrak{A}$ is linearly ordered. Let $<$ be the symbol for the linear ordering on $\mathfrak{A} \in \mathrm{ord}[\tau]$. Denote by $\leqslant^{\mathfrak{A}}$ the corresponding ordering in the sense of $\leqslant$ on $\mathfrak{A}$. Then $\leqslant^{\mathfrak{A}}\otimes n$ is a pre-ordering on $A \times n$ whose equivalence relation is $=^{\mathfrak{A}}\otimes n$, the quotient interpretation of equality on A over the product $A \times n$. Note that $=^{\mathfrak{A}}\otimes n$ and $\leqslant^{\mathfrak{A}}\otimes n$ are definable on $\mathfrak{A} \otimes n$ from the given $<^{\mathfrak{A}}\otimes n$ according to $(a, a') \in \leqslant^{\mathfrak{A}} \otimes n \Leftrightarrow \mathfrak{A} \otimes n \models \neg a' < a$ and $(a, a') \in =^{\mathfrak{A}}\otimes n \Leftrightarrow \mathfrak{A} \otimes n \models \neg a' < a \wedge \neg a < a'$. It follows that $\mathfrak{A} \in \mathrm{ord}[\tau]$ is interpreted over $\mathfrak{A} \otimes n$ as a quotient with respect to $=^{\mathfrak{A}}\otimes n$ — even in a first-order definable manner.

It follows further that $\mathfrak{A}$ and n and an ordered version of $\mathfrak{A} \otimes n$ are FP^*-interpretable over the second sort of $(\mathfrak{A} \otimes n)^*$, whence FP^* captures PTIME over the $\mathfrak{A} \otimes n$ for $\mathfrak{A} \in \mathrm{ord}[\tau]$.

To make a comparison between the complexity of queries over the $\Gamma(\mathfrak{A}, n)$ and that of their non-uniform description over the $\mathfrak{A}$ precise, we introduce the notion of a *pull-back* with respect to a function γ. This function γ serves to couple the scaling parameter n of Γ to the size of $\mathfrak{A}$.

Definition 5.6. *Let* $\Gamma\colon \mathrm{fin}[\tau] \times \omega \to \mathrm{fin}[\tau]$ *and* $\gamma\colon \omega \to \omega$. *If* $\mathcal{K}$ *is a boolean query on* $\mathrm{fin}[\tau]$ *then the following class is the* pull-back *of* $\mathcal{K}$ *under* Γ *and* γ*:*

$$\mathcal{K}_{\Gamma,\gamma} := \Big\{\mathfrak{A} \in \mathrm{fin}[\tau] \;\Big|\; \Gamma\big(\mathfrak{A}, \gamma(|A|)\big) \in \mathcal{K}\Big\}.$$

A pull-back *of a quantifier Q of type $\sigma = \{R_1, \ldots, R_k\}$ with respect to Γ and γ (and associated encoding scheme) is a quantifier*

$$Q^{\Gamma}_{\gamma} = \left\{ \big(A, [R_1']_{\bar{c}}, \ldots, [R_k']_{\bar{c}}\big) \;\middle|\; \begin{array}{l} \big(\Gamma(\mathfrak{A}, \gamma(|A|)), R_1'|\bar{c}, \ldots, R_k'|\bar{c}\big) \restriction \sigma \in Q, \\ \text{the } R_i' \simeq\text{-closed on } \Gamma(\mathfrak{A}, \gamma(|A|)) \end{array} \right\}.$$

Lemma 5.7. *Assume that Γ is such that $|\Gamma(\mathfrak{A}, n)| \geqslant n$ and that $\mathfrak{A}$ and n are uniformly* FP^**-interpretable over $\Gamma(\mathfrak{A}, n)^*$ for $\mathfrak{A} \in \mathrm{ord}[\tau]$. Then there is for every recursive query $\mathcal{K}_0$ on $\mathrm{ord}[\tau]$ a class $\mathcal{K}$ which is* FP^**-definable over $\big\{\Gamma(\mathfrak{A}, n) \;\big|\; \mathfrak{A} \in \mathrm{ord}[\tau]\big\}$, such that $\mathcal{K}_0$ is the pull-back of $\mathcal{K}$ under Γ and γ, for all sufficiently fast growing γ.*

Sketch of Proof. Let $\mathcal{K}_0 \subseteq \mathrm{ord}[\tau]$ be recursive. It follows that there is a function γ whose graph is in PTIME and such that membership of $\mathfrak{A}$ in $\mathcal{K}_0$ is decidable in time $\gamma(|A|)$. For instance $\gamma(m)$ could be the step counter for the consecutive simulation of some algorithm for $\mathcal{K}_0$ on all $\mathfrak{A}$ over universe m. Put

$$\mathcal{K} := \big\{\Gamma(\mathfrak{A}, n) \;\big|\; \mathfrak{A} \in \mathcal{K}_0, n \geqslant \gamma(|A|)\big\}.$$

We observe first that the class $\{\Gamma(\mathfrak{A},n) \mid \mathfrak{A} \in \mathrm{ord}[\tau], n \geqslant \gamma(|A|)\}$ is FP*-definable over $\{\Gamma(\mathfrak{A},n) \mid \mathfrak{A} \in \mathrm{ord}[\tau]\}$: n and $|A|$ are available over the second sort by the assumptions on Γ, and the graph of γ is in PTIME.

$\mathfrak{A}$ is FP*-interpreted over $\Gamma(\mathfrak{A},n)^*$ by assumption on Γ, and $\mathfrak{A} \in \mathcal{K}_0$ is decidable in time $\gamma(|A|)$ by the choice of γ. It follows that $\mathcal{K}$ is FP*-definable over $\{\Gamma(\mathfrak{A},n) \mid \mathfrak{A} \in \mathrm{ord}[\tau]\}$.

But $\mathcal{K}_0 = \mathcal{K}_{\Gamma,\gamma}$ by construction. Observe that γ may be replaced with any other function γ' that grows at least as fast as γ: $\mathcal{K}_0 = \mathcal{K}_{\Gamma,\gamma'}$ for any γ' such that $\gamma'(m) \geqslant \gamma(m)$ for all m. □

So the pull-backs of FP*-definable queries are of arbitrarily high complexity. In the next section we shall see that in contrast the pull-backs with respect to $\Gamma(\mathfrak{A},n) = \mathfrak{A} \otimes n$ of FP$(\mathcal{Q}_{\mathrm{card}})$-definable queries are in PTIME$/_{\mathrm{poly}}$ — polynomial time with non-uniform polynomial advice.

Recall the definition of PTIME$/_{\mathrm{poly}}$ from complexity theory. A class $\mathcal{K}$ of ordered τ-structures is in PTIME$/_{\mathrm{poly}}$ if there is an *advice function* T defined on ω with values that are polynomially bounded in size and such that membership of $\mathfrak{A}$ in $\mathcal{K}$ can be decided in polynomial time upon input $(\mathfrak{A}, T(|A|))$. PTIME$/_{\mathrm{poly}}$ may equivalently be characterized by computability in polynomial size families of boolean circuits. In any case, standard diagonalization techniques based on counting arguments show that PTIME$/_{\mathrm{poly}}$ is strictly contained in the class of all recursive sets. See for instance [Weg87].

A Lindström quantifier is in PTIME$/_{\mathrm{poly}}$ if there is a polynomially size bounded advice function T such that the class of pairs

$$\{(\mathfrak{A}; T(|A|)) \mid \mathfrak{A} \in Q\}$$

is in PTIME. Think of $T(n)$ as a polynomial size table encoding the semantics of Q over size n structures. FP(Q)-definable queries can obviously be evaluated in PTIME$/_{\mathrm{poly}}$ if Q is in PTIME$/_{\mathrm{poly}}$.

Putting the results of the above considerations together we obtain the following general statement. It will be applied below to the functor $\Gamma\colon (\mathfrak{A},n) \mapsto \mathfrak{A} \otimes n$.

Proposition 5.8. *Let $\Gamma\colon \mathrm{fin}[\tau] \times \omega \to \mathrm{fin}[\tau]$ be a functor that admits a good encoding scheme with parameters. Assume that Γ is such that $|\Gamma(\mathfrak{A},n)| \geqslant n$ and that $\mathfrak{A}$ and n are uniformly* FP**-interpreted over $\Gamma(\mathfrak{A},n)^*$ for $\mathfrak{A} \in \mathrm{ord}[\tau]$. Suppose further that for all quantifiers $Q \in \mathcal{Q}$ and for sufficiently fast growing $\gamma\colon \omega \to \omega$ the pull-backs Q^{Γ}_{γ} are in* PTIME$/_{\mathrm{poly}}$. *Then* FP$^* \not\subseteq$ FP$(\mathcal{Q})$.

Proof. Under the assumptions on Γ we may apply Lemma 5.7 to find that any recursive query on $\mathrm{ord}[\tau]$ is the pull-back of some query that is FP*-definable over $\{\Gamma(\mathfrak{A},n) \mid \mathfrak{A} \in \mathrm{ord}[\tau]\}$. The complexity of pull-backs of FP$(\mathcal{Q})$-definable queries is at most PTIME$/_{\mathrm{poly}}$ by Lemma 5.5 and the assumptions on $\mathcal{Q}$. □

5.2 Cardinality Lindström Quantifiers

Before applying the techniques prepared in the previous section to the proof of Theorem 5.1, we show in an aside that no finite collection of generalized quantifiers can capture FP+C. The argument is an adaptation of the proof by Dawar and Hella [DH94] that no finite extension of FP captures PTIME.

Theorem 5.9 (Dawar, Hella). *For any finite set* $\mathcal{Q}$ *of* PTIME *Lindström quantifiers:* FP+C $\not\subseteq$ FP($\mathcal{Q}$).

Proof. We consider FP+C and FP($\mathcal{Q}$) over pure sets ($\tau = \emptyset$) and show that over these FP+C $\not\subseteq$ FP($\mathcal{Q}$). Because over pure sets FP+C $\equiv$ FP* we even show that FP* $\not\subseteq$ FP($\mathcal{Q}$). Obviously FP* captures PTIME over pure sets.

Consider now definability in FP($\mathcal{Q}$) for finite $\mathcal{Q}$ over pure sets. By invariance under automorphisms, any predicate definable over pure sets has to be quantifier free equality definable, or a union of equality types. In each bounded arity k there are only finitely many equality types, so that it follows (with an argument strictly analogous to that in Corollary 1.32) that over the empty vocabulary FP($\mathcal{Q}$) $\equiv L_{\omega\omega}(\mathcal{Q})$.

Consider a single quantifier $Q \in \mathcal{Q}$ and without loss of generality assume that its type consists of a single relation R of arity r (tuples of predicates can be encoded into single predicates by first-order means, and corresponding transformations of Q do not affect polynomiality).

Let $L^k_{\omega\omega}(Q)$ be that syntactic fragment of $L_{\omega\omega}(Q)$ which uses only first-order variables $x_1, \ldots, x_k$. In $L^k_{\omega\omega}(Q)$ over pure sets, Q can only be applied to r-ary predicates that are quantifier free equality definable (with parameters) in at most k variables. Up to logical equivalence there is a finite list of quantifier free equality formulae $\chi_j(\overline{x}, \overline{x}')$ in variables $x_1, \ldots, x_k$ that provide such definitions. Let the χ_j be of the form

$$\chi_j(\overline{x}, \overline{x}') = \theta_j(\overline{x}) \wedge \eta_j(\overline{x}, \overline{x}'),$$

with $\overline{x}$ and $\overline{x}'$ disjoint, $\overline{x}'$ of arity r, and with θ_j specifying a complete equality type in the parameters $\overline{x}$. Then the semantics of Q in $L^k_{\omega\omega}(Q)$ is exhaustively described over each individual set n by a finite table $T(n)$ that encodes the behaviour of Q on the $\chi_j[n]$. Let $T(n)$ be the finite list of indices j for which

$$\big(n, \big\{\overline{m}' \;\big|\; n \models \chi_j[\overline{m}, \overline{m}']\big\}\big) \in Q \qquad \text{for } \mathrm{eq}\big(\overline{m}\big) = \theta_j.$$

There are only finitely many possibilities $T_1, \ldots, T_l$ for this entire table. For any fixed value T_i the quantifier Q in $L^k_{\omega\omega}(Q)$ becomes uniformly first-order definable over all n with $T(n) = T_i$. A formula $\xi(\overline{x}) = Q\big(\overline{x}'; \varphi(\overline{x}, \overline{x}')\big)$ is equivalent over all n with $T(n) = T_i$ with the disjunction

$$\xi_i(\overline{x}) := \bigvee_{j \in T_i} \big(\theta_j(\overline{x}) \wedge \forall \overline{x}'(\varphi \leftrightarrow \chi_j)\big).$$

It follows that Q can be eliminated (in $L^k_{\omega\omega}(Q)$ over pure sets) at the cost of introducing cardinality quantifiers Q_i of type $\emptyset$ according to

$$Q_i := \{A \mid T(|A|) = T_i\}.$$

For then, $\xi(\overline{x})$ as above becomes equivalent with $\bigvee_i (Q_i \wedge \xi_i(\overline{x}))$. This carries through inductively to eliminate all occurrences of Q.

If the complexity of the original Q is in Ptime of degree d, and if $d \geqslant r$, then the tables $T(n)$ can be computed in Ptime of degree d, too. This is because a standard representation of each $(n, \{\overline{m}' \mid n \models \chi_j[\overline{m}, \overline{m}']\})$ may be constructed in time $O(n^r)$. Therefore, also the Q_i are in Ptime of degree d.

Thus, for any finite set $\mathcal{Q}$ of Ptime quantifiers there is some d, such that over the empty vocabulary $L^k_{\omega\omega}(\mathcal{Q}) \equiv L^k_{\omega\omega}(\mathcal{Q}')$ for some finite set $\mathcal{Q}'$ of quantifiers of type $\emptyset$ whose complexity is in Ptime of degree bounded by d. For d we may take the maximal degree in a set of polynomials that bound the complexities of the $Q \in \mathcal{Q}$. Note that $\mathcal{Q}'$ depends on k, but the bound d does not.

That the quantifiers in $\mathcal{Q}'$ are of type $\emptyset$ means that their semantics only depends on the size of the universe. Let $\varphi \in L^k_{\omega\omega}(\mathcal{Q}')$ be a sentence. Then there is some m such that $n \models \varphi \Leftrightarrow n' \models \varphi$ for all $n, n' \geqslant m$ which satisfy the same $Q \in \mathcal{Q}'$. Asymptotically therefore, and over the empty vocabulary, any boolean query in $L^k_{\omega\omega}(\mathcal{Q}')$ is equivalent with a boolean combination of quantifiers $Q \in \mathcal{Q}'$, and therefore its complexity is of degree bounded by d. Now, since over pure sets

$$\mathrm{FP}(\mathcal{Q}) \equiv L_{\omega\omega}(\mathcal{Q}) \equiv \bigcup_k L^k_{\omega\omega}(\mathcal{Q}),$$

$\mathrm{FP}(\mathcal{Q})$ can only define boolean queries whose complexity is of a constantly bounded degree. It is obvious on the other hand that no such restriction applies to FP^* over pure sets, because there are numerical properties of arbitrarily high polynomial degree in Ptime. □

5.2.1 Proof of Theorem 5.1

Plain cardinality Lindström quantifiers. Consider first the case of $\mathrm{FP}(\mathcal{Q}_{\mathrm{card}})$. $\Gamma: (\mathfrak{A}, n) \mapsto \mathfrak{A} \otimes n$ is now fixed. We want to show the following for all $\gamma: \omega \to \omega$.

(∗) For any $Q \in \mathcal{Q}_{\mathrm{card}}$, the quantifiers Q^Γ_γ — the pull-backs of Q from $\mathfrak{A} \otimes \gamma(|A|)$ to $\mathfrak{A}$ — can be encoded in polynomially size bounded tables $T(|A|)$. In other words: each Q^Γ_γ is in $\mathrm{Ptime}/_{\mathrm{poly}}$.

Recall Definition 5.6 for the Q^Γ_γ. By Proposition 5.8, (∗) suffices to prove that part of Theorem 5.1 that deals with ordinary cardinality Lindström quantifiers.

For the proof of (∗) first observe that the Q^{Γ}_{γ} for $Q \in \mathcal{Q}_{\mathrm{card}}$ are themselves in $\mathcal{Q}_{\mathrm{card}}$. Recall that we write π_i for the projections to the factors in $\mathfrak{A} \otimes n$. The extended encoding scheme $R, \overline{c} \mapsto [R]_{\overline{c}}$ for the fibres of $t{+}r$-ary automorphism closed predicates R with parameter tuples $\overline{c}$ of arity t, takes for $[R]_{\overline{c}}$ the tuple of predicates

$$R^e_{\overline{c}} = \bigl\{\pi_1(\overline{c}, \overline{b}) \bigm| \overline{b} \in R|\overline{c}, \mathrm{eq}\bigl(\pi_2(\overline{c}, \overline{b})\bigr) = e\bigr\}, \quad e \in \mathrm{Eq}(t + r).$$

If $R|\overline{c}$ is non-empty, then the $R^e_{\overline{c}}$ determine $\pi_1(\overline{c})$ and $\mathrm{eq}\bigl(\pi_2(\overline{c})\bigr)$. $\overline{c}$ itself is then determined up to an arbitrary choice of $\pi_2(\overline{c})$ that realizes $\mathrm{eq}\bigl(\pi_2(\overline{c})\bigr)$. Up to this choice, the fibre $R|\overline{c}$ can be recovered from the encoding as

$$R|\overline{c} = \dot{\bigcup_e} \Bigl\{\overline{b} \in (A \times n)^r \Bigm| \pi_1(\overline{c}, \overline{b}) \in R^e_{\overline{c}}, \mathrm{eq}\bigl(\pi_2(\overline{c}, \overline{b})\bigr) = e\Bigr\}.$$

Therefore

$$\bigl|R|\overline{c}\bigr| = \sum_e \bigl|R^e_{\overline{c}}\bigr|\, \nu_e(n),$$

where ν_e is the counting function whose value on n is the number of realizations of e over n that extend any fixed realization of $\mathrm{eq}\bigl(\pi_2(\overline{c})\bigr)$.

Suppose for instance Q is of type $\{R_1\}$, R_1 of arity r, and based on the numerical relation $S \subseteq \omega^2$. Then, for automorphism invariant R and parameters $\overline{c}$,

$$\bigl(A \times n, R|\overline{c}\bigr) \in Q \quad \text{if} \quad \bigl(|A|n, \bigl|R|\overline{c}\bigr|\bigr) \in S$$

and

$$\bigl(A, [R]_{\overline{c}}\bigr) \in Q^{\Gamma}_{\gamma} \quad \text{if} \quad \Bigl(|A|\, \gamma\bigl(|A|\bigr), \textstyle\sum_e \bigl|R^e_{\overline{c}}\bigr|\, \nu_e\bigl(\gamma(|A|)\bigr)\Bigr) \in S.$$

This latter condition constitutes a cardinality quantifier $\widehat{Q}$ of the type of the encoding $[R]_{\overline{c}}$ over the base structures $\mathfrak{A}$. The same applies without any changes to cardinality quantifiers Q of more complex types.

It is obvious, finally, that the semantics of cardinality quantifiers can be fully encoded in polynomial size tables. Let the arities in $\widehat{Q}$ be bounded by $\hat{r}$ and let $\widehat{S}$ be the numerical relation for $\widehat{Q}$. To evaluate $\widehat{Q}$ over a structure of size m, one need only know $\widehat{S} \restriction \{0, \dots, m^{\hat{r}}\}$. This restriction of $\widehat{S}$ is naturally encoded in a polynomial size table. This finishes the proof of Theorem 5.1 as far as $\mathrm{FP}\bigl(\mathcal{Q}_{\mathrm{card}}\bigr)$ is concerned.

The following discussion shows how to extend the argument to $\mathrm{FP}\bigl(\mathcal{Q}^{\sim}_{\mathrm{card}}\bigr)$ where counting of equivalence classes is involved. This is based on a slightly more technical analysis of the encodings.

Quotient cardinality quantifiers. We claim that also for a quotient cardinality quantifier $Q \in \mathcal{Q}^{\sim}_{\mathrm{card}}$ the pull-backs Q^{Γ}_{γ} of Q from $\mathfrak{A} \otimes \gamma(|A|)$ to $\mathfrak{A}$ are in $\textsc{Ptime}/_{\mathrm{poly}}$, or encodable in polynomially size bounded tables $T(|A|)$.

In order not to get overburdened by technical details, let us consider the special case of a parameter free pull-back. This is the case of a pull-back quantifier Q^{Γ}_{γ} that captures the counting of equivalence classes over $\mathfrak{A} \otimes n$ with respect to an equivalence relation R that is interpreted without parameters. The technical lemma on equality defined equivalence relations, that governs this case, may be extended to the general case with parameters in order to prove the full claim.

In the parameter free case we deal as above with the encoding scheme that is based on the mapping

$$\begin{array}{rcl} \Pi\colon (A \times n)^r & \longrightarrow & A^r \times \mathrm{Eq}(r) \\ \bar{b} & \longmapsto & \Big(\pi_1(\bar{b}), \mathrm{eq}\big(\pi_2(\bar{b})\big)\Big). \end{array}$$

Lemma 5.10. *Let $R \subseteq (A \times n)^{2r}$ be closed under automorphisms of $\mathfrak{A} \otimes n$ and assume that R interprets an equivalence relation on the r-th power of $A \times n$. Let $\overline{a}_0 \in A^r$, $e_0 \in \mathrm{Eq}(r)$. Then the following are satisfied:*

- *(i) the index of the restriction of R to $\Pi^{-1}(\overline{a}_0, e_0)$ is of the form $p(n)/r!$ where p is a polynomial of degree at most r and with coefficients in $\{0, \dots, (r!)^2\}$. These coefficients can be determined from the encoding $[R] = (R^e)_{e \in \mathrm{Eq}(2r)}$ on $\mathfrak{A}$ in* Ptime.
- *(ii) if $P \subseteq (A \times n)^r$ is an automorphism closed predicate on $\mathfrak{A} \otimes n$, then the index of the restriction of R to P is of the form $q(n)/r!$ for a polynomial q of degree at most r with coefficients in $\{0, \dots, (r!)^3|A|^r\}$. Again the coefficients are* Ptime *computable from the encodings $[R]$ and $[P]$ on $\mathfrak{A}$.*

Proof. Assume n is much greater than r.
(i) For the first claim consider any quantifier free equality defined equivalence relation $\sim$ on the set $e_0[n] := \big\{\overline{m} \in n^r \bigm| \mathrm{eq}(\overline{m}) = e_0\big\}$. Without loss of generality assume that e_0 is the equality type that forces all r components of the $\overline{m}$ to be distinct. Otherwise the claim is reduced to smaller r. Let $i \in \{1, \dots, r\}$ be called *free* under $\sim$ if there are $\overline{m}, \overline{m}' \in e_0[n]$ with $m'_i \notin \{m_1, \dots, m_r\}$ and $\overline{m} \sim \overline{m}'$. An easy automorphism argument that exploits transitivity and symmetry of $\sim$ shows that, if i is free in $\sim$, then $\overline{m} \sim \overline{m}\frac{m}{i}$ for all $\overline{m} \in e_0[n]$ and all $m \notin \{m_1, \dots, m_r\}$. In this case therefore, $\sim$ is reducible to an equivalence relation $\sim'$ on the remaining components that has the same index as $\sim$: if for instance r is free, let e'_0 be the restriction of e_0 to the first $r - 1$ variables, and put for $\overline{m}, \overline{m}' \in e'_0[n]$

$$\overline{m} \sim' \overline{m}' \quad \text{if} \quad \overline{m}m \sim \overline{m}'m' \text{ for all } m \notin \overline{m} \text{ and } m' \notin \overline{m}'.$$

We may therefore assume without loss of generality that no i is free in $\sim$. This implies that $\overline{m} \sim \overline{m}'$ only if $\overline{m}' = \rho(\overline{m})$ for some permutation $\rho \in S_r$. Let

G be the normal subgroup (!) of S_r consisting of those ρ for which $\overline{m} \sim \rho(\overline{m})$. The index of $\sim$ on $e_0[n]$ is the product of the number $\binom{n}{r}$ of different r-element subsets of n with the index of G in S_r.

The claim about the form of the index as a polynomial in n follows. A representation of this polynomial by its coefficients is PTIME computable over the encodings in $\mathfrak{A}$ because the above sequence of reductions is governed by even first-order definable properties of the given equivalence relation R.
(ii) For a preliminary observation let $M \subseteq A^r \times \mathrm{Eq}(r)$, $(\overline{a}, e) \notin M$. Then exactly one of the following holds:

(a) any $\overline{b} \in \Pi^{-1}(\overline{a}, e)$ is R-equivalent with some $\overline{b}_1 \in \Pi^{-1}(M)$.
(b) no $\overline{b} \in \Pi^{-1}(\overline{a}, e)$ is R-equivalent with any $\overline{b}_1 \in \Pi^{-1}(M)$.

Again, a simple automorphism argument proves this claim: for $\overline{b}, \overline{b}' \in \Pi^{-1}(\overline{a}, e)$ there is an automorphism of $\mathfrak{A} \otimes n$ which maps $\overline{b}$ to $\overline{b}'$ while leaving $\Pi^{-1}(M)$ invariant as a set. The distinction between cases (a) and (b) is first-order in terms of R, $\Pi^{-1}(M)$ and $\Pi^{-1}(\overline{a}, e)$. It is therefore first-order and in PTIME also in terms of $[R]$, M, $\overline{a}$ and e over $\mathfrak{A}$.

Let R and P be as required in the lemma. The index $\left|P^{\mathfrak{A}\otimes n}/R^{\mathfrak{A}\otimes n}\right|$ can be determined by going through all $(\overline{a}, e) \in A^r \times \mathrm{Eq}(r)$ in some arbitrarily fixed enumeration as $(\overline{a}, e)_i = (\overline{a}_i, e_i)$, and summing over the indices $\left|\Pi^{-1}(\overline{a}_i, e_i)/R\right|$ whenever $\overline{a}_i \in P^{e_i}$ and case (b) above applies to $(\overline{a}_i, e_i)$ with respect to $M = \left\{(\overline{a}, e)_{i'} \;\middle|\; i' < i\right\}$. This proves claim (ii) of the lemma, since $\left|A^r \times \mathrm{Eq}(r)\right| \leqslant r!|A|^r$. □

With this lemma the quantifier free pull-back of a quotient cardinality quantifier is seen to be in PTIME/$_{\text{poly}}$ as follows. The lemma shows that the indices over $\mathfrak{A} \otimes n$ of $\simeq$-closed interpreted predicates with respect to $\simeq$-closed interpreted equivalences can be represented as polynomials in n, of constantly bounded degree and with a range for the coefficients that is polynomially bounded in $|A|$ ((ii) of the lemma). All these indices are therefore uniquely encodable as numbers to base n, of bounded length and with entries corresponding to the above ranges for the coefficients. The numerical predicate S of Q can therefore — to the extent that matters over structures $\mathfrak{A} \otimes n$ with $|A| = m$ — be encoded in tables of size polynomial in m, with entries to be understood as (tuples of) numbers expressed to base n.

5.3 Aside on Further Applications

Though not directly related to issues of fixed-point with counting, we present two other simple applications of the technique developed in this chapter. Namely we can prove that *sparse* and *rigid* quantifiers do not suffice to capture PTIME.

A Lindström quantifier Q is called rigid if all structures in its defining class are rigid, i.e. possess no non-trivial automorphisms.

A relational structure $\mathfrak{B}$ is called f-sparse if the number of elements of B that occur in any of the predicates in $\mathfrak{B}$ is at most $f(|B|)$. We call f sub-linear if $f(cn)/n \to 0$ for $n \to \infty$ for all c. Q is sparse if there is a sub-linear function f such that all structures in Q are f-sparse

Let $\mathcal{Q}_{\mathrm{sparse}}$ and $\mathcal{Q}_{\mathrm{rigid}}$ be the classes of all sparse or rigid Lindström quantifiers, respectively.

Theorem 5.11. *Neither* $\mathrm{FP}(\mathcal{Q}_{\mathrm{sparse}})$ *nor* $\mathrm{FP}(\mathcal{Q}_{\mathrm{rigid}})$ *comprise all* FP^*, *in particular* $\textsc{Ptime} \not\subseteq \mathrm{FP}(\mathcal{Q}_{\mathrm{sparse}}), \mathrm{FP}(\mathcal{Q}_{\mathrm{rigid}})$.

Sketch of Proof. The proof is straightforward if we consider once more the functor $\Gamma\colon (\mathfrak{A}, n) \mapsto \mathfrak{A} \otimes n$ and the associated pull-backs.

Consider rigid quantifiers first. Let $(\mathfrak{A} \otimes n, \bar{c})$ be such that n exceeds the arity of $\bar{c}$. Then $(\mathfrak{A} \otimes n, \bar{c})$ has non-trivial automorphisms and no structure that is interpreted with parameters $\bar{c}$ over $\mathfrak{A} \otimes n$ can be rigid. In other words, the pull-back of any rigid quantifier corresponds to the trivially false quantifier $Q_n^\Gamma = \emptyset$ for all sufficiently large n.

Consider now a sparse quantifier and its pull-backs involving parameter tuples $\bar{c}$ of arity t. For sufficiently large n, any relation that is interpreted over $(\mathfrak{A} \otimes n, \bar{c})$ either only contains subtuples of $\bar{c}$, or it contains a non-trivial orbit under the automorphism group of $(\mathfrak{A} \otimes n, \bar{c})$, which grows at least linearly with n. But for sub-linear f the bound $f(|A|n)$ grows slower than n, so that for sufficiently large n, Q can evaluate to true at most on those trivial structures whose relations consist of subtuples of $\bar{c}$. These are finitely bounded in the size of their relations and in number. In restriction to their fields, these relations can thus be distinguished up to isomorphism even in first-order. For sufficiently large n the entire information in the Q_n^Γ thus is, which of these trivial structures are in Q, when embedded in the universe of size $|A|n$. Therefore the Q_n^Γ reduce to cardinality quantifiers of type $\emptyset$.

We thus find that the pull-backs of $\mathrm{FP}(\mathcal{Q}_{\mathrm{rigid}})$-definable classes are FP-definable. The pull-backs of $\mathrm{FP}(\mathcal{Q}_{\mathrm{sparse}})$-definable classes are definable in the extension of FP with cardinality quantifiers of type $\emptyset$ if only the pull-back function γ is sufficiently fast growing. In particular the latter are in $\textsc{Ptime}/_{\mathrm{poly}}$ once more. This proves the desired separations. □

Remarks. In a paper [Ott94] on *simple Lindström extensions* the above results (with the exception of the case of sparse quantifiers) have been presented under a slightly different angle. The emphasis there is on quantifiers that express simple properties in the sense that these properties themselves are robust with respect to certain trivial extensions and can be decided in terms of invariants of sub-exponential range. In the case of counting quantifiers such invariants consist of numerical functions that count tuples in predicates; their range is clearly polynomial. I have here chosen to stress the technical basis of the separation proofs rather than a notion of simplicity. This basis is the same really for the applications here and in [Ott94], apart from the small difference that here we work with trivial products rather than with trivial sums. This variation is motivated by the formally smoother encoding schemes available over trivial products. The new application to sparse quantifiers is also due to this change. It relies on the property of trivial products that the pull-backs of sparse relations are sparse themselves. This is not true for trivial sums. Conceivably the general technique applies to other natural classes of quantifiers that might require yet other scaling functors Γ.

6. Canonization Problems

This is the first of two chapters dealing with canonization. In this chapter we consider canonization up to logical equivalences $\equiv^{\mathcal{L}}$, in particular for the logics $\mathcal{L} = L^k_{\infty\omega}$ and $C^k_{\infty\omega}$. We investigate the relation between PTIME canonization, PTIME inversion of the invariants, and the existence of recursive presentations and normal forms for related fragments of PTIME. It is shown for instance that PTIME invertibility for the I_{C^k} for all k would imply that FP+C captures exactly all queries that are PTIME computable and $C^\omega_{\infty\omega}$-definable. This and similar implications are of a hypothetical status, however: the problem of PTIME invertibility — and of PTIME canonization for C^k and L^k — remains open for arbitrary k. We show in this chapter that the general case essentially reduces to that for the *three* variable fragments. An explicit solution to the problem for the *two* variable fragments will be presented in the next chapter.

• Section 6.1 reviews the general notion of canonization and discusses canonization with respect to isomorphism in connection with algorithms on structures.

• In Section 6.2 PTIME canonization for $\equiv^{\mathcal{L}}$ is related to recursive presentations of fragments of PTIME.

• Section 6.3 discusses PTIME inversion of the I_{C^k} and I_{L^k} in relation to canonization and normal forms for the related fragments of PTIME. In particular we present theorems on the impact of PTIME invertibility of all I_{C^k}, respectively all I_{L^k} (in the sense of Definition 6.9), on the classes PTIME $\cap\, C^\omega_{\infty\omega}$ and PTIME $\cap\, L^\omega_{\infty\omega}$.

• The reduction of these results to the three variable fragments is presented in Section 6.4.

6.1 Canonization

For the general notion of canonization compare Definition 1.57 and related remarks in Section 1.7.1. Formally a function H provides canonization for $\sim$ if it satisfies two conditions. For all x we want $H(x) \sim x$ and whenever

$x \sim x'$ then $H(x) = H(x')$. Dealing with finite structures as basic objects and considering computable canonization with respect to an equivalence relation on some $\mathrm{fin}[\tau]$, we require $H(\mathfrak{A})$ to be a structure with standard domain, $H(\mathfrak{A}) \in \mathrm{stan}[\tau]$. Compare in particular Definition 1.61.

An important case is the canonization problem of combinatorial graph theory, namely the problem of *canonization of finite graphs up to isomorphism*. This is often also termed *graph normalization*. The same problem applies to any other class of finite structures, in particular to the entire classes $\mathrm{fin}[\tau]$ for arbitrary finite relational vocabularies τ. Normalization for any $\mathrm{fin}[\tau]$, however, reduces to graph normalization for most purposes. This is because there are natural encoding schemes mapping relational structures of an arbitrary fixed vocabulary to graphs in a way that would be compatible with normalization. With encodings by means of relativized interpretations of τ-structures in graphs, standardization of the parent structure (the graph) immediately induces a corresponding standardization of the interpreted τ-structure.

The problem of finding a standard representative up to isomorphism for relational structures is closely related with the analysis of algorithms over structures as discussed in the introduction (compare also Section 1.2). Standard models of computation require the input structure to be represented as a string over some alphabet. This is possible in a canonical way for ordered structures since these admit a trivial low complexity normalization procedure. Let $< \in \tau$ and recall that $\mathrm{ord}[\tau]$ stands for the class of finite τ-structures that are linearly ordered by $<$. The natural canonization then is

$$\begin{aligned} H\colon \mathrm{ord}[\tau] &\longrightarrow \mathrm{stan}[\tau] \\ \mathfrak{A} = (A, <^{\mathfrak{A}}, \ldots) &\longmapsto (|A|, <^{|A|}, \ldots), \end{aligned}$$

where $(|A|, <^{|A|}, \ldots)$ is the unique structure in $\mathrm{stan}[\tau]$ with the natural ordering, that is isomorphic with $(A, <^{\mathfrak{A}}, \ldots)$. If a priori we admit an arbitrary representation also for $\mathfrak{A} \in \mathrm{ord}[\tau]$ through an arbitrary isomorphic representative in $\mathrm{stan}[\tau]$ we find that this functor H is computable in LOGSPACE.

It is difficult to imagine any feasible representation of the isomorphism type of finite structures for standard computational models and in particular the Turing model, that does not implicitly introduce a linear order on the domain of the given structure. In general one therefore has to admit representations of the abstract structure $\mathfrak{A} \in \mathrm{fin}[\tau]$ through arbitrary isomorphic representative in $\mathrm{stan}[\tau]$ — or, equivalently, through the introduction of an arbitrary ordering for representational purposes. Uniqueness of the representative is given up and the notorious invariance problems have to be dealt with. Algorithms for structures have to satisfy a semantic invariance condition, since the outcome of the computation must be independent of the input representation.

This problem can be side-stepped, however, if there should be a feasible construction of unique representatives after all. In the general case this requires a feasibly computable functor

$$\begin{array}{rcl} H\colon \mathrm{fin}[\tau] & \longrightarrow & \mathrm{stan}[\tau] \\ \mathfrak{A} & \longmapsto & H(\mathfrak{A}), \end{array}$$

$$\text{satisfying} \quad \begin{array}{l} \forall \mathfrak{A} \quad H(\mathfrak{A}) \simeq \mathfrak{A}, \\ \forall \mathfrak{A} \forall \mathfrak{A}' \quad \mathfrak{A} \simeq \mathfrak{A}' \rightarrow H(\mathfrak{A}) = H(\mathfrak{A}'). \end{array}$$

It is here irrelevant whether we regard fin[τ] or stan[τ] as the domain of this functor. A functor H with these properties is a *computable canonization functor with respect to isomorphism* on fin[τ] in the sense of Definition 1.61. In the context of logics for fragments of PTIME "feasible" here means "PTIME computable". It is not known whether there is a PTIME normalization procedure for all finite relational structures, or equivalently for the class of all finite graphs. It is clear that PTIME graph normalization would immediately yield a PTIME algorithm for the *graph isomorphism problem*. The status with respect to complexity of the graph isomorphism problem, however, is a notorious open problem.

For an upper bound on the complexity of graph normalization one can at least show that it is contained in Δ_2^{pol} at the second level of the polynomial hierarchy. Δ_2^{pol} is the class of those problems that admit a PTIME solution relative to an oracle in NPTIME (and NPTIME $= \Sigma_1^{\mathrm{pol}}$).

Example 6.1. There is a graph normalization functor H in Δ_2^{pol}.

For the oracle we choose the *weak subgraph isomorphism problem*. The weak subgraph relation $\mathfrak{G}_1 \subseteq_{\mathrm{w}} \mathfrak{G}_2$ holds if the universe of $\mathfrak{G}_1$ is a subset of the universe of $\mathfrak{G}_2$ and if all edges of $\mathfrak{G}_1$ are also edges of $\mathfrak{G}_2$. Let $\mathcal{O}$ be the set of all standard encodings of pairs of graphs $(\mathfrak{G}_1, \mathfrak{G}_2)$ where $\mathfrak{G}_1$ is isomorphic with some $\mathfrak{G}_1' \subseteq_{\mathrm{w}} \mathfrak{G}_2$. Obviously $\mathcal{O}$ is in NPTIME, in fact it is NPTIME-complete.

Relative to the oracle $\mathcal{O}$ we get the following PTIME algorithm $\mathcal{A}$ for graph normalization.

On input (n, E), a graph on standard domain n, $\mathcal{A}$ successively computes edge relations $E_m \subseteq m \times m$ for $m = 1, \ldots, n$, where $E_1 = \emptyset$ and , for $m > 1$, E_m is the lexicographically maximal element of the set

$$S^m = \big\{ R \subseteq m \times m \; \big| \; E_{m-1} \subseteq R \text{ and } \big((m, R), (n, E)\big) \in \mathcal{O} \big\}.$$

The lexicographic ordering on the $R \subseteq m \times m$ is the usual one if R is identified with the sequence of values of its characteristic function $\chi_R(0,0)$, $\chi_R(0,1)$, $\ldots$, $\chi_R(m-1, m-1)$. It is easily shown inductively that the S^m are non-empty. All the (m, E_m) will actually be isomorphic with *subgraphs* of (n, E), as any addition of more edges to some R is an upward move in the lexicographic ordering. In fact (m, E) automatically is the lexicographically maximal graph of size m that is isomorphic with a subgraph of (n, E).

Therefore $H(n, E) := (n, E_n)$ is as desired. It remains to argue that the E_m can be determined in PTIME relative to $\mathcal{O}$, which is not quite obvious at first as in general S^m is of exponential size in m.

But to compute E_m from E_{m-1} it suffices to settle the values of $\chi = \chi_{E_m}$ at $(0, m-1), \ldots, (m-2, m-1)$. All other entries are in fact determined:

(i) $\chi(m-1, m-1) = 0$, by irreflexivity of E_m.
(ii) $\chi(m-1, j) = \chi(j, m-1)$ for $j < m-1$, by symmetry of E_m.
(iii) $\chi \restriction (m-1) \times (m-1) = \chi_{E_{m-1}}$,
because $E_m \supseteq E_{m-1}$ and $E_m \restriction (m-1) \times (m-1) \leqslant_{\text{lex}} E_{m-1}$.

That sequence $\chi(0, m-1), \ldots, \chi(m-2, m-1)$ that leads to the lexicographically maximal E_m can be constructed as follows.

Put $\chi^0(0, m-1) = \ldots = \chi^0(m-2, m-1) = 0$; the resulting E_m^0 equals E_{m-1} and thus is in S^m. Proceeding inductively, let χ^{j+1} be χ^j with the value at $(j, m-1)$ changed to 1 if the E_m^{j+1} that is so obtained is in S^m, and $\chi^{j+1} = \chi^j$ otherwise. Then $\chi := \chi^{m-1}$ is as desired.

If PTIME canonization up to isomorphism is unlikely to be attained, it is sensible to consider canonization with respect to rougher, and in particular logical notions of equivalence instead of isomorphism.

6.2 PTIME Canonization and Fragments of PTIME

Definition 6.2. *Let $\mathcal{L}$ be a logic, $\equiv^{\mathcal{L}}$ the induced notion of equivalence on* $\text{fin}[\tau]$ *and on* $\text{fin}[\tau; r]$. *A* PTIME *computable functor* $H\colon \text{fin}[\tau] \to \text{stan}[\tau]$ *provides* PTIME canonization up to $\equiv^{\mathcal{L}}$ *on* $\text{fin}[\tau]$ *or* canonization for $\mathcal{L}$ on $\text{fin}[\tau]$ *if the following are satisfied:*

$$\begin{aligned} &\forall \mathfrak{A} \quad H(\mathfrak{A}) \equiv^{\mathcal{L}} \mathfrak{A}, \\ &\forall \mathfrak{A} \forall \mathfrak{A}' \quad \mathfrak{A} \equiv^{\mathcal{L}} \mathfrak{A}' \to H(\mathfrak{A}) = H(\mathfrak{A}'). \end{aligned}$$

The analogous requirements are imposed on a functor $H\colon \text{fin}[\tau; r] \to \text{stan}[\tau; r]$ *for* PTIME *canonization on* $\text{fin}[\tau; r]$.

Canonization up to $\equiv^{\mathcal{L}}$ determines a unique standard representative within each class of $\mathcal{L}$-equivalent finite structures, respectively of finite structures with parameters. The difference between canonization for plain structures and structures with parameters is inessential for the logics under consideration, because $\equiv^{C^k}$ and $\equiv^{L^k}$ satisfy the requirements of the following lemma. Recall that $\equiv^{C^k}$ and $\equiv^{L^k}$ are in PTIME as relations on $\text{fin}[\tau; r]$ for all $r \leqslant k$, because the invariants for L^k and C^k on the $\text{fin}[\tau; r]$ are PTIME computable. Compare Corollaries 3.9 and 3.14.

Lemma 6.3. *Let $\mathcal{L}$ be such that $\equiv^{\mathcal{L}}$ is in* PTIME *as a relation on* $\text{fin}[\tau; r]$ *and such that $\mathfrak{A} \equiv^{\mathcal{L}} \mathfrak{A}'$ implies that $\mathfrak{A}$ and $\mathfrak{A}'$ realize the same $\mathcal{L}$-types of r-tuples:* $\mathfrak{A} \equiv^{\mathcal{L}} \mathfrak{A}' \Rightarrow \text{Tp}^{\mathcal{L}}(\mathfrak{A}; r) = \text{Tp}^{\mathcal{L}}(\mathfrak{A}'; r)$.

Then any PTIME *canonization functor* $H\colon \text{fin}[\tau] \to \text{stan}[\tau]$ *extends naturally to a* PTIME *canonization* $H\colon \text{fin}[\tau; r] \to \text{stan}[\tau; r]$ *on* $\text{fin}[\tau; r]$.

Sketch of Proof. The following extension of H satisfies the requirements — we denote it H as well. Let $H(\mathfrak{A},\overline{a}) := \big(H(\mathfrak{A}),\overline{b}\big)$ where $\overline{b}$ is the lexicographically least r-tuple over the standard domain of $H(\mathfrak{A})$ for which $\big(H(\mathfrak{A}),\overline{b}\big) \equiv^{\mathcal{L}} (\mathfrak{A},\overline{a})$. □

PTIME canonization bears the following simple yet fundamental relationship with recursive presentations of fragments of PTIME.

Lemma 6.4. *Let H provide* PTIME *canonization up to $\equiv^{\mathcal{L}}$ on* fin$[\tau]$ *and on the* fin$[\tau; r]$. *Then the class of all those queries over* fin$[\tau]$ *that are* PTIME *computable in the usual sense* and *closed with respect to $\equiv^{\mathcal{L}}$, is recursively enumerable. In fact the following are equivalent for any boolean query $Q \subseteq$* fin$[\tau]$*:*

(i) Q is closed with respect to $\equiv^{\mathcal{L}}$ and Q is in PTIME.
(ii) $Q = \big\{\mathfrak{A} \bigm| H(\mathfrak{A}) \in Q\big\}$ and there is a PTIME *algorithm that recognizes $Q \cap$* stan$[\tau]$.

For an r-ary global relation R on fin$[\tau]$ *the following are equivalent:*

(i) R is closed with respect to $\equiv^{\mathcal{L}}$ and PTIME*-computable.*
(ii) $R^{\mathfrak{A}} = \big\{\overline{a} \in A \bigm| \overline{b} \in R^{\mathfrak{B}}$ where $(\mathfrak{B},\overline{b}) = H(\mathfrak{A},\overline{a})\big\}$ and there is a PTIME *algorithm that, applied to $H(\mathfrak{A},\overline{a}) = (\mathfrak{B},\overline{b})$, decides whether $\overline{b} \in R^{\mathfrak{B}}$.*

Note that the algorithms in (ii) are not subject to any semantic constraints, since these algorithms need merely realize boolean *functions* on stan$[\tau]$ or stan$[\tau; r]$, respectively. A natural recursive set of representatives consists of all algorithms that first check the input size, then initialize some counter to a fixed polynomial in this size and terminate their computation after this pre-set number of steps (polynomially clocked algorithms).

Sketch of Proof. We indicate the proof for boolean queries. Observe that $\equiv^{\mathcal{L}}$–closure is equivalent with $Q = \big\{\mathfrak{A} \bigm| H(\mathfrak{A}) \in Q\big\}$. Therefore, any PTIME algorithm $\mathcal{A}$ that recognizes an $\equiv^{\mathcal{L}}$–closed class Q is semantically equivalent with $\mathcal{A} \circ H$. For (i) ⇒ (ii) use $\mathcal{A}$ in restriction to stan$[\tau]$. For the converse use $\mathcal{A}$ as given in (ii) and compose it with H to get the PTIME algorithm $\mathcal{A} \circ H$ which computes the boolean query Q over fin$[\tau]$. □

Definition 6.5. *Let* PTIME$\cap\mathcal{L}$ *stand for the class of all those global relations that are both in* PTIME *and $\mathcal{L}$-definable.*

Recall from Lemma 1.33 that for logics $\mathcal{L}$ that are closed under countable disjunctions and conjunctions and under negation, $\mathcal{L}$-definability coincides with closure under $\equiv^{\mathcal{L}}$. Lemma 6.4 therefore yields a connection between PTIME canonization for $\mathcal{L}$ and a recursive presentation for PTIME$\cap\mathcal{L}$. Assume for the following definition that $H\colon$fin$[\tau] \to$ stan$[\tau]$ provides canonization on fin$[\tau]$ and extends to functors $H\colon$fin$[\tau; r] \to$ stan$[\tau; r]$ on the fin$[\tau; r]$ in the sense of Lemma 6.3 above.

Definition 6.6. *Let* PTIME(H) *stand for the class of all queries that are* PTIME *computable in terms of the images under* H*. More precisely,*

(i) a boolean query Q *on* $\mathrm{fin}[\tau]$ *is in* PTIME(H) *if membership of* $\mathfrak{A}$ *in* Q *is a* PTIME *property of* $H(\mathfrak{A})$*.*
(ii) an r*-ary query* R *on* $\mathrm{fin}[\tau]$ *is in* PTIME(H) *if membership of* $\overline{a}$ *in* $R^{\mathfrak{A}}$ *is a* PTIME *property of* $H(\mathfrak{A}, \overline{a})$*.*

Again, as the PTIME algorithms mentioned in the definition are not subject to additional semantic constraints, PTIME(H) is recursively presented through all compositions of polynomially clocked algorithms with some fixed algorithm for H.

Lemma 6.4 can be rephrased with this notion of PTIME(H) as follows. We state it for $\mathcal{L} = C^k_{\infty\omega}$ or $L^k_{\infty\omega}$. Note that for these any PTIME canonization on $\mathrm{fin}[\tau]$ extends to all $\mathrm{fin}[\tau; r]$ with $r \leqslant k$ by Lemma 6.3. This is sufficient for the statement below since there are no $L^k_{\infty\omega}$- or $C^k_{\infty\omega}$-definable queries in arities greater than k.

Corollary 6.7. *Let* $\mathcal{L} = C^k_{\infty\omega}$ *or* $L^k_{\infty\omega}$ *and let* H *provide* PTIME *canonization for* $\mathcal{L}$ *on* $\mathrm{fin}[\tau]$*. Then* H *extends to the* $\mathrm{fin}[\tau; r]$ *for* $r \leqslant k$ *and*

$$\text{PTIME} \cap \mathcal{L} \equiv \text{PTIME}(H).$$

In particular PTIME $\cap\, \mathcal{L}$ *is recursively enumerable (i.e. admits a recursive presentation).*

6.3 Canonization and Inversion of the Invariants

As sketched in the abstract setting in Lemma 1.60, canonization problems are generally related with inversion problems for complete invariants. While a canonization H must assign representatives, complete invariants may assign any kind of values that are characteristic of classes. Canonization may be obtained from an invariant if it is possible to reconstruct a typical member of each class on the basis of the value of that class under the invariant. The mere existence of such an inverse is obvious from the definitions. Its complexity, however, is critical. Different complete invariants for the same equivalence relation might lead to entirely different inversion problems in particular with respect to complexity.

We return to the canonization problem for the $C^k_{\infty\omega}$ and $L^k_{\infty\omega}$. The functors I_{C^k} and I_{L^k} provide complete invariants. Recall that we write I_{C^k} and I_{L^k} for the complete invariants on $\mathrm{fin}[\tau]$ as well as for their natural extensions to the $\mathrm{fin}[\tau; r]$ for $r \leqslant k$. We shall see below that also with respect to the corresponding inversion problems a solution for I_{L^k} or I_{C^k} on $\mathrm{fin}[\tau]$ naturally extends to a solution over the $\mathrm{fin}[\tau; r]$. We restate for convenience the definition of an inverse to a complete invariant, Definition 1.59, in the present context.

Definition 6.8. *Let $\mathcal{L} = C^k_{\infty\omega}$ or $L^k_{\infty\omega}$, $I_{\mathcal{L}}$ the corresponding invariant. A function*

$$F: \{ I_{\mathcal{L}}(\mathfrak{A}) \mid \mathfrak{A} \in \mathrm{fin}[\tau] \} \longrightarrow \mathrm{stan}[\tau]$$

is an inverse for $I_{\mathcal{L}}$ *on* $\mathrm{fin}[\tau]$ *if it satisfies:* $\forall \mathfrak{A}\ F(I_{\mathcal{L}}(\mathfrak{A})) \equiv^{\mathcal{L}} \mathfrak{A}$. *The analogous condition applies for inverses of $I_{\mathcal{L}}$ on $\mathrm{fin}[\tau; r]$ for $r \leqslant k$. Equivalently these conditions can be put as $I_{\mathcal{L}} \circ F = id$ on $image(I_{\mathcal{L}})$.*

Generally an inversion of a complete invariant yields a canonization simply through composition of the inverse with the invariant itself (Lemma 1.60). Also a PTIME computable inversion F here yields PTIME canonization, since the $I_{\mathcal{L}}$ themselves are PTIME computable. Note however that the converse need not a priori be true. It is conceivable that H provides PTIME canonization while the associated F defined by the requirement that $H = F \circ I_{\mathcal{L}}$ might not be in PTIME. In fact, for the $L^k_{\infty\omega}$ with $k \geqslant 3$ we already know that inversion of I_{L^k} cannot be in PTIME in the usual sense, simply because the image under F might necessarily be of a size that is exponential in the size of the argument. See Example 3.23. The following definition takes care of this obvious obstacle and defines PTIME inversion for the I_{L^k} as an inversion that is polynomial time computable in terms of the size of the desired image.

Definition 6.9. *We say that I_{C^k}* admits PTIME inversion *if there is an inverse F for I_{C^k} that is* PTIME *computable in the usual sense.*

I_{L^k} admits PTIME inversion *if there is an inverse F for I_{L^k}, such that for all $\mathfrak{A}$, F is computable on $I_{L^k}(\mathfrak{A})$ in time polynomial in* $\min\{ |\mathfrak{B}| \mid \mathfrak{B} \equiv^{L^k} \mathfrak{A} \}$. *To mark the difference in the complexity requirement let us say that such F is computable in* PTIME*.

In either case we shall speak, however, of F as a PTIME inverse of the invariant.

PTIME canonization and PTIME inversion for the L^k are discussed in Dawar's dissertation [Daw93] and in [DLW95]. The appropriate notion of PTIME inversion of I_{L^k} is put forward there and the question whether I_{L^k} admits PTIME inversion in this sense is formulated as an open problem.

A natural and intuitively stronger definition of PTIME inversion for I_{L^k} would be to require an algorithm that takes as its input pairs $(I_{L^k}(\mathfrak{A}), n)$ and produces in time polynomial in $\max\{|I_{L^k}(\mathfrak{A})|, n\}$ a structure $\mathfrak{B} \in \mathrm{stan}[\tau]$ of size n with $\mathfrak{B} \equiv^{L^k} \mathfrak{A}$ if such exists. From such an algorithm a PTIME inverse in the sense of the preceding definition is obtained through application to $(I_{L^k}(\mathfrak{A}), n)$ for growing n until a successful output is constructed. This exhaustive search for a standardized pre-image under I_{L^k} (of minimal size even) is still polynomial in the size of a minimal solution.

We come to the extension of inverses to the I_{L^k} and I_{C^k} on $\mathrm{fin}[\tau]$ to inverses of the extended invariants on $\mathrm{fin}[\tau; r]$, $r \leqslant k$.

Lemma 6.10. *Let $\mathcal{L} = C^k_{\infty\omega}$ or $L^k_{\infty\omega}$ and $I_{\mathcal{L}} = I_{C^k}$ or I_{L^k}, respectively.* PTIME *inversion of $I_{\mathcal{L}}$ on* $\mathrm{fin}[\tau]$ *extends naturally to* PTIME *inversion of $I_{\mathcal{L}}$ on* $\mathrm{fin}[\tau; r]$ *for $r \leqslant k$.*

Sketch of Proof. Assume $F\colon \{I_{\mathcal{L}}(\mathfrak{A}) \mid \mathfrak{A} \in \mathrm{fin}[\tau]\} \to \mathrm{stan}[\tau]$ is an inverse to $I_{\mathcal{L}}$ on $\mathrm{fin}[\tau]$. Recall that the extension of $I_{\mathcal{L}}$ to $\mathrm{fin}[\tau; r]$ maps $(\mathfrak{A}, \overline{a})$ to the expansion of $I_{\mathcal{L}}(\mathfrak{A})$ in which the $\mathcal{L}$-type of $\overline{a}$ is marked: $I_{\mathcal{L}}(\mathfrak{A}, \overline{a}) = \big(I_{\mathcal{L}}(\mathfrak{A}), [\overline{a}]\big)$. Extend F to $\mathrm{fin}[\tau; r]$ by putting $F\big(I_{\mathcal{L}}(\mathfrak{A}, \overline{a})\big) := \big(F(I_{\mathcal{L}}(\mathfrak{A})), \overline{b}\big)$ for the lexicographically least tuple $\overline{b}$ in the standard domain of $F(I_{\mathcal{L}}(\mathfrak{A}))$ which satisfies $I_{\mathcal{L}}\big(F(I_{\mathcal{L}}(\mathfrak{A})), \overline{b}\big) = I_{\mathcal{L}}(\mathfrak{A}, \overline{a})$. The search for this tuple is polynomially bounded in the size of $F(I_{\mathcal{L}}(\mathfrak{A}))$. Therefore if F is PTIME, respectively PTIME* in the sense of the preceding definition, on $\mathrm{fin}[\tau]$, then so is its extension to $\mathrm{fin}[\tau; r]$. □

Theorem 6.11. *Let $\mathcal{L} = C^k_{\infty\omega}$ or $L^k_{\infty\omega}$, and correspondingly $I_{\mathcal{L}} = I_{C^k}$ or I_{L^k}. If F is a* PTIME *inverse for $I_{\mathcal{L}}$, then $H := F \circ I_{\mathcal{L}}$ provides* PTIME *canonization for $\mathcal{L}$. Moreover this composition is compatible with the respective natural extensions of the $I_{\mathcal{L}}$, H and F to the* $\mathrm{fin}[\tau; r]$ *for $r \leqslant k$.*

Sketch of Proof. We check the requirements in the case of I_{L^k}. H maps $\mathfrak{A} \in \mathrm{fin}[\tau]$ to a standard structure equivalent with $\mathfrak{A}$, since $I_{L^k} \circ F \circ I_{L^k} = I_{L^k}$ by the definition of inverses. As a composition with I_{L^k}, H certainly maps L^k-equivalent structures to the same image. It remains to check that H is in PTIME, even if F is computable only in PTIME* in the sense of Definition 6.9. Since $\min\{|\mathfrak{B}| \mid \mathfrak{B} \equiv^{L^k} \mathfrak{A}\} \leqslant |\mathfrak{A}|$, the computation of F on $I_{L^k}(\mathfrak{A})$ is still polynomial in terms of $|\mathfrak{A}|$.

Compatibility with the extensions to cover $\mathrm{fin}[\tau; r]$ instead of $\mathrm{fin}[\tau]$ follows directly from the definition of these extensions. See in particular the above lemma and compare with Lemma 6.3. □

Combining Corollary 6.7 with Theorem 6.11 we get the following connection between PTIME inversion of the I_{C^k} and the capturing of PTIME $\cap\, C^k_{\infty\omega}$. Recall Definition 4.17 for the classes PTIME$\big(I_{C^k}\big)$. The global relations in PTIME$\big(I_{C^k}\big)$ are those that are PTIME computable over the invariants I_{C^k}. Logically the same class is representable by the logics $\mathrm{FP}\big(I_{C^k}\big)$ also discussed in connection with Definition 4.17 and Theorem 4.18.

Theorem 6.12. *If I_{C^k} admits* PTIME *inversion, then*

$$\mathrm{PTIME} \cap C^k_{\infty\omega} \equiv \mathrm{PTIME}\big(I_{C^k}\big) \equiv \mathrm{FP}\big(I_{C^k}\big).$$

Proof. Let F be a PTIME inverse for I_{C^k}. The non-trivial inclusion PTIME $\cap\, C^k_{\infty\omega} \subseteq$ PTIME$\big(I_{C^k}\big)$ follows from Corollary 6.7 if we observe that $F \circ I_{C^k}$ provides PTIME canonization for $C^k_{\infty\omega}$: PTIME $\cap\, C^k_{\infty\omega} \subseteq$ PTIME$\big(F \circ I_{C^k}\big) \subseteq$ PTIME$\big(I_{C^k}\big)$, as F is in PTIME. □

Putting Theorem 4.18 — $\mathrm{FP{+}C} \equiv \bigcup_k \mathrm{FP}(I_{C^k})$ — and the last theorem together, we obtain the following hypothetical theorem.

Corollary 6.13. *If the I_{C^k} admit* PTIME *inversion for all k, then*

$$\text{PTIME} \cap C^{\omega}_{\infty\omega} \equiv \text{FP+C}.$$

For the $L^k_{\infty\omega}$ the situation is somewhat less smooth because of the possible collapse in size that can occur in the passage from $\mathfrak{A}$ to $I_{L^k}(\mathfrak{A})$. Note that, because of this potential collapse, it is not true that — as in the proof of Theorem 6.12 — $\text{PTIME}(F \circ I_{L^k}) \subseteq \text{PTIME}(I_{L^k})$ for a PTIME inverse F of I_{L^k} (which need actually only be computable in PTIME*).

Theorem 6.14. *Assume I_{L^k} admits* PTIME *inversion through F. Then* PTIME $\cap\, L^k_{\infty\omega}$ *is recursively enumerable. A boolean query is in* PTIME $\cap\, L^k_{\infty\omega}$ *if it is computable on the basis of the $I_{L^k}(\mathfrak{A})$ in time polynomial in the size of $F\big(I_{L^k}(\mathfrak{A})\big)$. Similarly for the computation of an r-ary global relation in terms of the $I_{L^k}(\mathfrak{A}, \overline{a})$.*

Let us say that I_{L^k} *is bounded* on a class $\mathcal{K} \subseteq \text{fin}[\tau]$ if there is a polynomial p such that $|\mathfrak{A}| \leqslant p(|I_{L^k}(\mathfrak{A})|)$ for all $\mathfrak{A} \in \mathcal{K}$. Obviously, if I_{L^k} is bounded on $\mathcal{K}$, then so is $I_{L^{k'}}$ for all $k' > k$. Suppose that I_{L^k} is bounded on $\mathcal{K}$ and that I_{L^k} admits PTIME inversion through F. Then F must in fact be computable in PTIME rather than in PTIME*: $F\big(I_{L^k}(\mathfrak{A})\big)$ must be polynomial time computable in terms of $|A|$ by definition, and $|A|$ is polynomial in the size of $I_{L^k}(\mathfrak{A})$ for bounded I_{L^k}. The following is then proved in precise analogy with Corollary 6.13 above.

Corollary 6.15. *If the I_{L^k} admit* PTIME *inversion for all k, then*

$$\text{PTIME} \cap L^{\omega}_{\infty\omega} \equiv \text{FP} \qquad \textit{on } \mathcal{K}$$

for all classes $\mathcal{K}$ on which I_{L^k} is bounded for some k.

6.4 A Reduction to Three Variables

We exhibit a reduction technique that shows that PTIME canonization and PTIME inversion for the $L^k_{\infty\omega}$ and $C^k_{\infty\omega}$ with arbitrary k essentially reduce to the three variable cases. *'Essentially'* because the proposed reduction does not work in a k-by-k fashion but rather introduces a shift in the number of variables of the following kind. Assuming for instance PTIME invertibility of I_{C^3} we get a PTIME construction that, given $I_{C^m}(\mathfrak{A})$ for certain $m > k$, yields a standard structure that is C^k-equivalent with $\mathfrak{A}$. The effect of this mismatch is smoothed out, however, if we consider the effect with respect to the unions across all levels k. For instance, from PTIME invertibility of I_{C^3} we shall still get FP+C $\equiv$ PTIME $\cap\, C^{\omega}_{\infty\omega}$.

Here are the precise statements concerning the reduction, first in terms of canonization, then in terms of inversion of the invariants. Note that a priori these statements might be of independent interest, since the existence of a PTIME canonization procedure does not, as far as we can see, imply PTIME invertibility of the particular invariants considered here.

Theorem 6.16. *Suppose $C^3_{\infty\omega}$ admits* PTIME *canonization. Then there is for each τ and each k a* PTIME *functor $K^{3k,k}\colon \mathrm{fin}[\tau] \longrightarrow \mathrm{stan}[\tau]$ such that*

$$\begin{aligned} &\forall\mathfrak{A} \quad K^{3k,k}(\mathfrak{A}) \equiv^{C^k} \mathfrak{A}, \\ &\forall\mathfrak{A}\forall\mathfrak{A}' \quad \mathfrak{A} \equiv^{C^{3k}} \mathfrak{A}' \;\rightarrow\; K^{3k,k}(\mathfrak{A}) = K^{3k,k}(\mathfrak{A}'). \end{aligned}$$

The same statement holds of the $L^k_{\infty\omega}$ and under the assumption that $L^3_{\infty\omega}$ admits PTIME *canonization.*

Note the difference between $K^{3k,k}$ and a canonization functor with respect to $C^k_{\infty\omega}$. While $K^{3k,k}$ also produces C^k-equivalent standard structures, these representatives may depend not only on the C^k-theory of the given structure but on its C^{3k}-theory. The analogous reduction result for the inversion problem is the following.

Theorem 6.17. *Suppose I_{C^3} admits* PTIME *inversion. Then there is for each τ and each k a* PTIME *functor $G^{3k,k}\colon \mathit{image}(I_{C^{3k}}) \longrightarrow \mathrm{stan}[\tau]$ such that*

$$\forall\mathfrak{A} \quad G^{3k,k}\big(I_{C^{3k}}(\mathfrak{A})\big) \equiv^{C^k} \mathfrak{A},$$

or equivalently, $I_{C^3} \circ G^{3k,k} = \Pi^{3k,k}$ where $\Pi^{3k,k}$ is the obvious projection that sends $I_{C^{3k}}(\mathfrak{A})$ to $I_{C^k}(\mathfrak{A})$. Again, the same holds (for PTIME* *computability) with respect to the $L^k_{\infty\omega}$ and I_{L^k} and under the assumption that I_{L^3} admits* PTIME *inversion.*

The appropriate notion of PTIME* computability for $G^{3k,k}\colon \mathrm{image}(I_{L^{3k}}) \rightarrow \mathrm{stan}[\tau]$ is the following: $G^{3k,k}\big(I_{L^{3k}}(\mathfrak{A})\big)$ has to be computable in time polynomial in the size of a minimal $\mathfrak{B}$ that is L^{3k}-equivalent with $\mathfrak{A}$.

For our present purposes we thus have the following corollaries. The first is in terms of PTIME canonization for the three variable case, the second in terms of PTIME inversion for the three variable invariants. In both settings we find that the general statements of Corollaries 6.7, and Corollaries 6.13 and 6.15 respectively, reduce to the three variable cases if we consider the overall effect on the unions across all k, PTIME $\cap\, C^\omega_{\infty\omega}$ and PTIME $\cap\, L^\omega_{\infty\omega}$.

Corollary 6.18. *Let $\mathcal{L} = C^\omega_{\infty\omega}$ or $L^\omega_{\infty\omega}$, respectively. Assume that $C^3_{\infty\omega}$, respectively $L^3_{\infty\omega}$, admits* PTIME *canonization. Then* PTIME$\cap\mathcal{L}$ *is recursively enumerable; in fact*

$$\textsc{Ptime} \cap \mathcal{L} \equiv \bigcup_k \textsc{Ptime}\big(K^{3k,k}\big),$$

where PTIME$\big(K^{3k,k}\big)$ *is formally defined in analogy with Definition 6.6 for the functors $K^{3k,k}$ as characterized in Theorem 6.16 (and their natural extensions to the* $\mathrm{fin}[\tau; r]$ *where r-ary queries rather than just boolean ones are concerned).*

Sketch of Proof. Consider $\mathcal{L} = C^{\omega}_{\infty\omega}$. Both inclusions are in fact obvious. Any query in some $\text{PTIME}\big(K^{3k,k}\big)$ is in PTIME, and also in $C^{3k}_{\infty\omega}$ since the image under $K^{3k,k}$ only depends on the $C^{3k}_{\infty\omega}$-theory of structures. For the converse inclusion note that any query in $\text{PTIME} \cap C^{k}_{\infty\omega}$ can without affecting its semantics be evaluated after application of $K^{3k,k}$ because $K^{3k,k}$ preserves C^k-equivalence. □

Corollary 6.19. *(i) Assuming that I_{C^3} admits* PTIME *inversion we get:* $\text{PTIME} \cap C^{\omega}_{\infty\omega} \equiv \text{FP+C}$.
(ii) If I_{L^3} admits PTIME *inversion, then* $\text{PTIME} \cap L^{\omega}_{\infty\omega} \equiv \text{FP}$ *on all classes on which I_{L^k} is bounded for some k.*

Sketch of Proof. We indicate how the claim for boolean queries in $\text{PTIME} \cap C^{\omega}_{\infty\omega}$ follows from Theorem 6.17. Suppose that $Q \subseteq \text{fin}[\tau]$ is in $\text{PTIME} \cap C^{k}_{\infty\omega}$. Let $\mathcal{A}$ be a PTIME algorithm that recognizes Q. Then $\mathfrak{A} \in Q$ if and only if $G^{3k,k}\big(I_{C^{3k}}(\mathfrak{A})\big) \in Q$ if and only $\mathcal{A} \circ G^{3k,k} \circ I_{C^{3k}}$ accepts $\mathfrak{A}$. The latter composition is in FP+C because $I_{C^{3k}}$ is FP+C-interpretable over the $\mathfrak{A}^*$ and $\mathcal{A} \circ G^{3k,k}$ is FP-interpretable as a PTIME functor on the ordered $I_{C^{3k}}$. Closure of FP+C under interpretations (Proposition 4.8) yields $Q \in \text{FP+C}$. The converse inclusion $\text{FP+C} \subseteq \text{PTIME} \cap C^{\omega}_{\infty\omega}$ is obvious anyway (compare Corollary 4.20).

For the case of $\text{PTIME} \cap L^{\omega}_{\infty\omega}$ compare the appropriate modifications in Corollary 6.15 to adapt the argument to obtain (ii). □

Whether or not the three variable cases are solvable, remains open. The reduction achieved here therefore remains hypothetical. The two variable case is settled positively in the next chapter. In view of the above statements, a positive solution in the three variable case would be a major break-through in the understanding of the bounded-variable fragments of PTIME. The reduction argument itself is of interest because it also applies to other model theoretic questions about the $L^{k}_{\infty\omega}$ and $C^{k}_{\infty\omega}$, in particular we think of questions related to spectrum properties for these fragments, cf. [Ott96b]. For the present investigation it also illustrates where the essential power of *three*, as compared to *two* variables lies. At a more technical level it may also indicate potential obstacles for three variable canonization.

It might be worth pointing out that 3 is just the minimal number of variables for which we can show the reduction to go through. The reduction argument applies, essentially unchanged, to any other number of variables above 3. (And indeed, it is not clear why for instance k-variable canonization for some $k > 3$ should directly yield 3-variable canonization.)

None of the material in the rest of this chapter will be used in the last chapter on two-variable canonization.

6.4.1 The Proof of Theorems 6.16 and 6.17

The following definition of the *k-th power* of a relational structure resembles the definition of the game k-graphs, Definition 2.26. Here we include more complete information about the equality types of pairs of k-tuples for reasons that will become apparent below.

For finite relational τ let $\tau^{[k]}$ consist of unary predicates P_θ for $\theta \in \mathrm{Atp}(\tau;k)$ and binary predicates $\stackrel{i,j}{=}$ for $1 \leqslant i,j \leqslant k$. The intended interpretation for the latter — over some A^k — is that $(\overline{a},\overline{a}') \in \stackrel{i,j}{=}$ if $a_i = a'_j$. We shall write $\overline{a} \stackrel{i,j}{=} \overline{a}'$ instead of $(\overline{a},\overline{a}') \in \stackrel{i,j}{=}$.

Definition 6.20. *For $\mathfrak{A} \in \mathrm{fin}[\tau]$ let the k-*th power *of $\mathfrak{A}$ be the following structure $\mathfrak{A}^{[k]}$ in vocabulary $\tau^{[k]}$:*

$$\mathfrak{A}^{[k]} = \Big(A^k, (\stackrel{i,j}{=}), (P_\theta)\Big),$$

with the natural interpretations for the P_θ and the $\stackrel{i,j}{=}$. Denote by Γ_k the functor that takes $\mathfrak{A}$ to its k-th power $\mathfrak{A}^{[k]}$. Let $\Gamma_k\big(\mathrm{fin}[\tau]\big) \subseteq \mathrm{fin}[\tau^{[k]}]$ denote the closure under isomorphisms of the class of all k-th powers $\mathfrak{A}^{[k]}$ for $\mathfrak{A} \in \mathrm{fin}[\tau]$.

Just as the game k-graphs $\mathfrak{A}^{(k)}$, the $\mathfrak{A}^{[k]}$ are quantifier free interpretable in the k-th power over the given structures $\mathfrak{A}$.[1] Moreover, the game k-graphs $\mathfrak{A}^{(k)}$ are quantifier free (and directly) interpretable over the $\mathfrak{A}^{[k]}$: the edge relation E_j of the game k-graphs is the intersection of the $\stackrel{i,i}{=}$ for all $i \neq j$. Note, however, that conversely the $\stackrel{i,j}{=}$ are not quantifier free definable from the E_j.

The crucial fact for the desired reduction is that the C^k-theory (respectively L^k-theory) of $\mathfrak{A}$ is fully captured by the C^2-theory (respectively L^2-theory) of the k-th power $\mathfrak{A}^{[k]}$ of $\mathfrak{A}$. This follows directly from Proposition 3.25 where it was shown that even the C^2-theory of the game k-graph $\mathfrak{A}^{(k)}$ determines the C^k-theory of $\mathfrak{A}$. Clearly, the C^2-theory of $\mathfrak{A}^{[k]}$ determines that of $\mathfrak{A}^{(k)}$ owing to quantifier free interpretability of $\mathfrak{A}^{(k)}$ in $\mathfrak{A}^{[k]}$. In fact Proposition 3.25 says that the k-variable invariants $I_{C^k}(\mathfrak{A})$ or $I_{L^k}(\mathfrak{A})$ are PTIME computable (FP-interpretable) in the 2-variable invariants of the game k-graphs, $I_{C^2}\big(\mathfrak{A}^{(k)}\big)$ or $I_{L^2}\big(\mathfrak{A}^{(k)}\big)$. This carries over to the $\mathfrak{A}^{[k]}$ as well as for instance $I_{C^2}\big(\mathfrak{A}^{(k)}\big)$ is PTIME computable (FP-interpretable) in $I_{C^2}\big(\mathfrak{A}^{[k]}\big)$. We thus have the following, as a corollary to Proposition 3.25.

Proposition 6.21. *The two-variable theories of the k-th powers fully determine the k-variable theories of the base structures:*

$$\mathfrak{A}^{[k]} \equiv^{L^2} \mathfrak{A}'^{[k]} \;\Rightarrow\; \mathfrak{A} \equiv^{L^k} \mathfrak{A}', \quad \text{and} \quad \mathfrak{A}^{[k]} \equiv^{C^2} \mathfrak{A}'^{[k]} \;\Rightarrow\; \mathfrak{A} \equiv^{C^k} \mathfrak{A}'.$$

Moreover, $I_{C^k}(\mathfrak{A})$ and $I_{L^k}(\mathfrak{A})$ are PTIME *computable from $I_{C^2}\big(\mathfrak{A}^{[k]}\big)$ and $I_{L^2}\big(\mathfrak{A}^{[k]}\big)$, respectively.*

[1] There is no conflict with the notion of *interpretability in the k-th power*: this notion may be identified with (direct) interpretability over the (interpreted) $\mathfrak{A}^{[k]}$.

Canonization or inversion of the invariants in the three variable case will prove to be sufficient for Theorems 6.16 and 6.17 because being a k-th power is definable in three variables, in fact even in $L^3_{\omega\omega}$.

Lemma 6.22. *Let the arities in τ be at most k.*

(i) There is a sentence φ in $L^3_{\omega\omega}[\tau^{[k]}]$ such that $\mathrm{fmod}(\varphi) = \mathit{\Gamma}_k\big(\mathrm{fin}[\tau]\big)$.
(ii) $\mathfrak{A}$ is PTIME *computable from $\mathfrak{A}^{[k]}$.*

More precisely, (ii) is to say that there is a PTIME algorithm that maps $\mathfrak{C} \in \mathrm{stan}[\tau^{[k]}] \cap \mathit{\Gamma}_k\big(\mathrm{fin}[\tau]\big)$ to a structure $\mathfrak{B} \in \mathrm{stan}[\tau]$ such that $\mathfrak{B}^{[k]} \simeq \mathfrak{C}$.

The proof of the lemma is postponed — we first show how it applies to prove Theorems 6.16 and 6.17. For this we need one more simple lemma about an interpretability relation between certain invariants.

Lemma 6.23. *Let $m \geqslant 2$. Then the m-variable theories of of the k-th powers are fully determined by the mk-variable theories of the base structures:*

$$\mathfrak{A} \equiv^{L^{mk}} \mathfrak{A}' \;\Rightarrow\; \mathfrak{A}^{[k]} \equiv^{L^m} \mathfrak{A}'^{[k]}, \qquad \textit{and} \qquad \mathfrak{A} \equiv^{C^{mk}} \mathfrak{A}' \;\Rightarrow\; \mathfrak{A}^{[k]} \equiv^{C^m} \mathfrak{A}'^{[k]}.$$

Moreover, $I_{C^m}\big(\mathfrak{A}^{[k]}\big)$ and $I_{L^m}\big(\mathfrak{A}^{[k]}\big)$ are PTIME *computable from $I_{C^{mk}}\big(\mathfrak{A}\big)$ and $I_{L^{mk}}\big(\mathfrak{A}\big)$, respectively.*

Proof. The proof is similar to that of Proposition 3.25: it suffices to check that the entire inductive generation of the pre-ordering underlying the m-variable invariant of $\mathfrak{A}^{[k]}$ can be simulated over the mk-variable invariant of $\mathfrak{A}$. Let $\approx_i$ and $\preccurlyeq_i$ be the stages in the generation of $\equiv^{C^m}$ and $\preccurlyeq$ over $\mathfrak{A}^{[k]}$ as required for $I_{C^m}\big(\mathfrak{A}^{[k]}\big)$.

Writing $\overline{a} = \big(\overline{a}^{(1)}, \dots, \overline{a}^{(m)}\big)$ for mk-tuples over A we indicate their identification with m-tuples over A^k. A $\approx_i$-class α can be represented over $I_{C^{mk}}\big(\mathfrak{A}\big)$ as

$$\underline{\alpha} = \big\{ \mathrm{tp}^{C^{mk}}_{\mathfrak{A}}(\overline{a}) \;\big|\; \big(\overline{a}^{(1)}, \dots, \overline{a}^{(m)}\big) \in \alpha \big\}.$$

At the atomic level, $i = 0$, this representation is sound because the atomic $\tau^{[k]}$-type of $\big(\overline{a}^{(1)}, \dots, \overline{a}^{(m)}\big)$ is directly determined by the atomic τ-type of $\overline{a}$. It remains to consider the refinement step — soundness of the representation and PTIME computability in terms of $I_{C^{mk}}\big(\mathfrak{A}\big)$. Let α be a $\approx_i$-class, $\underline{\alpha}$ its representation. The refinement step is governed by the counting functions

$$\begin{aligned} \nu^{\alpha}_j(\overline{a}) &= \Big| \big\{ \overline{b}^{(j)} \;\big|\; (\overline{a}^{(1)}, \dots, \overline{a}^{(m)}) \tfrac{\overline{b}^{(j)}}{j} \in \alpha \big\} \Big| \\ &= \Big| \big\{ \overline{b}^{(j)} \;\big|\; \mathrm{tp}^{C^{mk}}_{\mathfrak{A}}\big((\overline{a}^{(1)}, \dots, \overline{a}^{(m)}) \tfrac{\overline{b}^{(j)}}{j} \big) \in \underline{\alpha} \big\} \Big|. \end{aligned}$$

These values clearly only depend on the C^{mk}-type of $\overline{a}$, and they are PTIME computable from $\big(I_{C^{mk}}\big(\mathfrak{A}\big), \underline{\alpha}\big)$. In fact $\nu^{\alpha}_j(\overline{a})$ is the cardinality of a definable k-ary predicate, definable in terms of a union of C^{mk}-equivalence classes that is represented over $I_{C^{mk}}\big(\mathfrak{A}\big)$ through $\underline{\alpha}$. □

Proof of Theorem 6.16. Let H^3 be a PTIME canonization functor for C^3 on $\mathrm{fin}[\tau^{[k]}]$. By definition H^3 satisfies for all $\mathfrak{A}^{[k]}$: $H^3(\mathfrak{A}^{[k]}) \equiv^{C^3} \mathfrak{A}^{[k]}$. With Lemma 6.22 we conclude that $H^3(\mathfrak{A}^{[k]}) \in \Gamma_k(\mathrm{fin}[\tau])$, since $\mathfrak{A}^{[k]} \in \Gamma_k(\mathrm{fin}[\tau])$ and $\Gamma_k(\mathrm{fin}[\tau])$ is closed under C^3-equivalence. Therefore $H^3(\mathfrak{A}^{[k]}) \simeq \mathfrak{B}^{[k]}$ for some $\mathfrak{B} \in \mathrm{stan}[\tau]$ that is PTIME computable from $H^3(\mathfrak{A}^{[k]})$ by Lemma 6.22. We infer from Proposition 6.21 that $\mathfrak{B} \equiv^{C^k} \mathfrak{A}$. By Lemma 6.23, $H^3(\mathfrak{A}^{[k]})$ and therefore the resulting $\mathfrak{B}$ are fully determined by the C^{3k}-theory of $\mathfrak{A}$. The composite mapping $\Gamma_k^{-1} \circ H^3 \circ \Gamma_k$ is thus seen to satisfy the requirements on $K^{3k,3}$ in the theorem. The statement concerning the L^k rather than the C^k is obtained in exactly the same manner. □

In complete analogy we also prove Theorem 6.17.

Proof of Theorem 6.17. Consider first the case with counting quantifiers. Assume that F^3 is a PTIME inverse for I_{C^3}. For all $\mathfrak{B}$: $F^3(I_{C^3}(\mathfrak{B})) \equiv^{C^3} \mathfrak{B}$. Since membership in $\Gamma_k(\mathrm{fin}[\tau])$ is a C^3-property by Lemma 6.22, F^3 restricts to $\{I_{C^3}(\mathfrak{B}) \mid \mathfrak{B} \in \Gamma_k(\mathrm{fin}[\tau])\}$ such that

$$F^3 : \{I_{C^3}(\mathfrak{B}) \mid \mathfrak{B} \in \Gamma_k(\mathrm{fin}[\tau])\} \longrightarrow \mathrm{stan}[\tau^{[k]}] \cap \Gamma_k(\mathrm{fin}[\tau]).$$

Let I be the mapping I: $I_{C^{3k}}(\mathfrak{A}) \mapsto I_{C^3}(\mathfrak{A}^{[k]})$, which is in PTIME according to Lemma 6.23. The composite mapping $G^{3k,k} = \Gamma_k^{-1} \circ F^3 \circ I$ satisfies the requirement of Theorem 6.17:

$$\forall \mathfrak{A} \quad G^{3k,k}\Big(I_{C^{3k}}(\mathfrak{A})\Big) \equiv^{C^k} \mathfrak{A}.$$

This is because

$$F^3 \circ I(I_{C^{3k}}(\mathfrak{A})) = F^3(I_{C^3}(\mathfrak{A}^{[k]})) \equiv^{C^3} \mathfrak{A}^{[k]} = \Gamma_k(\mathfrak{A}).$$

$\Gamma_k^{-1} \circ F^3 \circ I(I_{C^{3k}}(\mathfrak{A})) \equiv^{C^k} \mathfrak{A}$ now follows with Proposition 6.21.

For the case of L^k one merely checks in addition that the modified notion of PTIME inversion adapted to the I_{L^k} carries over from the corresponding given F^3 to the composite mapping $G^{3k,k}$. Note that the minimal size of structures $\mathfrak{B}$ that are L^3-equivalent with $\mathfrak{A}^{[k]}$ is bounded from above by the k-th power of the size of any structure $\mathfrak{B}'$ that is L^{3k}-equivalent with $\mathfrak{A}$. □

Proof of Lemma 6.22. Obviously any $\mathfrak{A}^{[k]}$ satisfies the following axioms that are all in $L^3_{\omega\omega}$.

(1) $\displaystyle\bigwedge_{i,j} \forall x \forall y \Big(x \overset{i,j}{=} y \longleftrightarrow y \overset{j,i}{=} x \Big).$

(2) $\displaystyle\forall x \forall y \Big(\bigwedge_{i} x \overset{i,i}{=} y \longleftrightarrow x = y \Big).$

(3) $\displaystyle\bigwedge_{i,j,l} \forall x \forall y \forall z \Big(x \overset{i,j}{=} y \wedge y \overset{j,l}{=} z \longrightarrow x \overset{i,l}{=} z \Big).$

(4) $\bigwedge_{s\subseteq\{1,\dots,k\}} \forall x\forall y\exists z\Big(\bigwedge_{i\in s} z \overset{i,i}{=} x \wedge \bigwedge_{i\notin s} z \overset{i,i}{=} y\Big).$

Axioms (1) – (4) exclusively concern the equality structure. We add a finite schema of axioms that formalize compatibility conditions between the atomic types as encoded in the P_θ and the equality structure described by the $\overset{i,j}{=}$. Let Ψ be the set of all quantifier free $\tau^{[k]}$-formulae in two variables x and y that are valid in $\Gamma_k\big(\mathrm{fin}[\tau]\big)$; we can restrict these to some syntactic normal form to keep the set finite without changing its semantics. In fact, the quantifier free kernels of (1) and (2) above are also represented in Ψ. Thus (1) and (2) become redundant when we now further put

(5) $\forall x\forall y \bigwedge_{\psi\in\Psi} \psi.$

We first show that

(A) any $(\overset{i,j}{=})$-structure satisfying (1) – (4) is isomorphic with a structure $\Big(\{0,\dots,n-1\}^k, (\overset{i,j}{=})\Big)$ with the natural interpretation for the $\overset{i,j}{=}$.

The isomorphism is unique up to a permutation of $n = \{0,\dots,n-1\}$.

To prove (A) let $\mathfrak{B} = \big(B, (\overset{i,j}{=}{}^{\mathfrak{B}})\big)$ be a model of (1) – (4). Observe that (1) – (3) imply that the $\overset{i,i}{=}$ are equivalence relations on B whose common refinement is equality. Denote by $[b]_i$ the equivalence class of b with respect to $\overset{i,i}{=}$. It follows from (2) that

$$\begin{array}{rcl}\pi\colon B & \longrightarrow & \prod_i B/\overset{i,i}{=} \\ b & \longmapsto & \big([b]_1,\dots,[b]_k\big)\end{array}$$

is an injection. We show that (4) implies π is surjective. Assume to the contrary that some $\big([b_1]_1,\dots,[b_k]_k\big)$ is not in the image of π. Then at least one of

$$\begin{array}{rl} & \Big([b_1]_1,\dots,[b_{k-2}]_{k-2},[b_{k-1}]_{k-1},[b_{k-1}]_k\Big) \\ \text{or} & \Big([b_1]_1,\dots,[b_{k-2}]_{k-2},[b_k]_{k-1},[b_k]_k\Big)\end{array}$$

is not in the image of π. Otherwise, choosing pre-images under π of these for x and y and applying (4) with $s = \{1,\dots,k-1\}$, one would get a pre-image of $\big([b_1]_1,\dots,[b_k]_k\big)$. Proceeding inductively we obtain that for some b, $\big([b]_1,\dots,[b]_k\big)$ is not in the image of π, which is clearly absurd. Therefore π is a bijection. By definition it maps $\overset{i,i}{=}$ to equality in the i-th component.

Finally, the $\overset{i,j}{=}{}^{\mathfrak{B}}$ induce bijections between the different factors $B/\overset{i,i}{=}$. This follows from (3): (3) implies that $\overset{i,j}{=}$ is closed under $\overset{i,i}{=}$ on the left and under $\overset{j,j}{=}$ on the right, so that it factorizes to yield a binary relation between $B/\overset{i,i}{=}$ and $B/\overset{j,j}{=}$. We claim that in this sense it becomes the graph of a bijection. For reasons of symmetry (1) it suffices to show injectivity, or that

$$b_1 \overset{i,j}{=} b \wedge b_2 \overset{i,j}{=} b' \wedge b \overset{j,j}{=} b' \implies b_1 \overset{i,i}{=} b_2,$$

which is immediate from (3) and (1). Let $\rho^{ij}: B/\overset{i,i}{=} \longrightarrow B/\overset{j,j}{=}$ be this bijection. Then

$$\begin{array}{rcl} \widehat{\pi}: B & \longrightarrow & \underbrace{(B/\overset{1,1}{=}) \times \cdots \times (B/\overset{1,1}{=})}_{k} \\ b & \longmapsto & \big(\rho^{i1}([b]_i)\big)_{i=1,\dots,k} \end{array}$$

is an isomorphism between $\mathfrak{B}$ and $(B/\overset{1,1}{=})^k$ with the standard interpretation for the $\overset{i,j}{=}$. We have shown that (A) holds. The proof also shows that the desired isomorphism with some $\big(\{0,\dots,n-1\}^k, (\overset{i,j}{=})\big)$ is unique up to the choice of an identification of $B/\overset{1,1}{=}$ with the appropriate set n. If $\mathfrak{B}$ is itself presented as a standard structure over some $\{0,\dots,n^k-1\}$, then the natural order on B induces an ordering of $B/\overset{1,1}{=}$ which can be used to determine a unique isomorphism of $\mathfrak{B}$ with $\big(\{0,\dots,n-1\}^k, (\overset{i,j}{=})\big)$. This isomorphism is constructible in PTIME.

Assume now that $\mathfrak{B}$ carries interpretations for the P_θ and is a model also of (5). For the full claim of the lemma it remains to translate the information in the P_θ to a τ-interpretation over n.

Let now $\pi: \big(\mathfrak{B}\restriction(\overset{i,j}{=})\big) \longmapsto \big(\{0,\dots,n-1\}^k, (\overset{i,j}{=})\big)$ be an isomorphism with the standard model of the equality part. (5) implies in particular that

(a) the $P_\theta^{\mathfrak{B}}$ form a partition of B. Introduce the mapping $\Theta: B \to \mathrm{Atp}(\tau;k)$ which sends b to that θ with $b \in P_\theta^{\mathfrak{B}}$.
(b) $\pi(b) = (m_1,\dots,m_k)$ implies that the equality type of $(m_1,\dots,m_k)$ is as prescribed in $\Theta(b)$.
(c) if $\pi(b) = (m_1,\dots,m_k)$ and $\pi(b') = (m'_1,\dots,m'_k)$, then the instantiations $\big(\Theta(b)\big)[m_1,\dots,m_k]$ and $\big(\Theta(b')\big)[m'_1,\dots,m'_k]$ are consistent, i.e. respect the equality type of the tuple $(m_1,\dots,m_k,m'_1,\dots,m'_k)$.

It follows that $\{0,\dots,n-1\}$ can be expanded to a τ-structure $\mathfrak{A}$ in a unique and consistent way by the stipulation that $\mathrm{atp}_{\mathfrak{A}}\big(\pi(b)\big) = \Theta(b)$. Thus π becomes an isomorphism between $\mathfrak{B}$ and $\mathfrak{A}^{[k]}$. $\mathfrak{A}$ can obviously be computed from π and $\mathfrak{B}$ in polynomial time, so that the second claim of Lemma 6.22 also follows. □

It is interesting to note that in the above $L^3_{\omega\omega}$-axiomatization three variables are necessary for the transitivity conditions (3) and the sentence (4), which ensures surjectivity of π. A condition to the effect of (4) can actually also be formalized in $C^2_{\infty\omega}$. Let χ_n be the sentence

$$\exists^{=m} x\, x = x \ \wedge\ \bigwedge_i \forall x\, \exists^{=s} y\ x \overset{i,i}{=} y$$

with $m = n^k$, $s = n^{k-1}$. Then (4) above can be replaced by $\bigvee_n \chi_n$. One obtains an axiomatization of $\Gamma_k\big(\mathrm{fin}[\tau]\big)$ in $C^3_{\infty\omega}$ that uses three variables only in the transitivity conditions for the $\overset{i,j}{=}$ in (3).

6.4.2 Remarks on Further Reduction

This aside is of a more technical nature. The proofs of Theorems 6.16 and 6.17 given in Section 6.4.1 need seemingly weaker assumptions than full PTIME canonization (or inversion of the invariants) in the three variable cases. We strengthen the formulation accordingly in this section. As all these considerations remain hypothetical — we have no well founded conjecture whether $\text{PTIME} \cap C^{\omega}_{\infty\omega}$ or $\text{PTIME} \cap L^{\omega}_{\infty\omega}$ can indeed be captured — the interest in these ramifications mainly is a technical one. We explicitly address the C^k in this aside, but once more everything translates to the L^k. Consider the situation with respect to canonization. The proof of Theorem 6.16 rests on the existence of PTIME computable functors

$$H_0 \colon \text{fin}[\tau^{[k]}] \longrightarrow \text{stan}[\tau^{[k]}] \cap \varGamma_k\big(\text{fin}[\tau]\big) \tag{6.1}$$

such that for all $\mathfrak{C}$ and $\mathfrak{C}'$ in the domain of H_0:

$$\begin{aligned} &H_0\big(\mathfrak{C}\big) \equiv^{C^2} \mathfrak{C} \\ &\mathfrak{C} \equiv^{C^m} \mathfrak{C}' \;\rightarrow\; H_0\big(\mathfrak{C}\big) = H_0\big(\mathfrak{C}'\big) \end{aligned} \tag{6.2}$$

for some m. From these we obtain 'weak canonization functors' $K^{mk,k}$ from $\text{fin}[\tau]$ to $\text{stan}[\tau]$ that satisfy

$$\begin{aligned} &\forall \mathfrak{A} \quad K^{mk,k}(\mathfrak{A}) \equiv^{C^k} \mathfrak{A}, \\ &\forall \mathfrak{A} \forall \mathfrak{A}' \quad \mathfrak{A} \equiv^{C^{mk}} \mathfrak{A}' \;\rightarrow\; K^{mk,k}(\mathfrak{A}) = K^{mk,k}(\mathfrak{A}'). \end{aligned}$$

In the proof of Theorem 6.16 we have explicitly used this construction for a proper canonization functor H for $C^3_{\infty\omega}$ in place of H_0 and with $m = 3$. Note that in this special case both $H\big(\mathfrak{A}^{[k]}\big) \in \varGamma_k\big(\text{fin}[\tau]\big)$ and $H\big(\mathfrak{A}^{[k]}\big) \equiv^{C^2} \mathfrak{A}^{[k]}$ are consequences of the stronger requirement that

$$H\big(\mathfrak{A}^{[k]}\big) \equiv^{C^3} \mathfrak{A}^{[k]}.$$

Surprisingly, a twofold application of the reduction schema leads to a further reduction in the assumptions expressed in equations 6.1 and 6.2. One need only assume the existence of such H_0 for $k = 3$. In particular this amounts to a reduction to vocabularies $\tau^{[3]}$ with a fixed set of binary relations $(\overset{i,j}{=})_{1 \leqslant i,j \leqslant 3}$ and only unary predicates besides.

Proposition 6.24. *Assume that for each τ there is a* PTIME *functor*

$$H \colon \text{fin}\big[\big(\tau^{[k]}\big)^{[3]}\big] \longrightarrow \text{stan}\big[\big(\tau^{[k]}\big)^{[3]}\big] \cap \varGamma_3\big(\text{fin}\big[\tau^{[k]}\big]\big)$$

such that for all $\mathfrak{C}, \mathfrak{C}' \in \text{fin}\big[\big(\tau^{[k]}\big)^{[3]}\big]$ and some fixed m:

$$\begin{aligned} &H\big(\mathfrak{C}\big) \equiv^{C^2} \mathfrak{C} \\ &\mathfrak{C} \equiv^{C^m} \mathfrak{C}' \;\rightarrow\; H\big(\mathfrak{C}\big) = H\big(\mathfrak{C}'\big). \end{aligned}$$

Then $H_0 := \Gamma_3^{-1} \circ H \circ \Gamma_3$ *satisfies the conditions in equations 6.1 and 6.2, with* $3m$ *in place of* m *and consequently* $K^{3mk,k} := \Gamma_k^{-1} \circ \Gamma_3^{-1} \circ H \circ \Gamma_3 \circ \Gamma_k$ *provides 'weak canonization' for* $C^k_{\infty\omega}$ *in the sense of Theorem 6.16 on* $\mathrm{fin}[\tau]$*:*

$$\begin{aligned}
&\forall \mathfrak{A} \quad K^{3mk,k}(\mathfrak{A}) \equiv^{C^k} \mathfrak{A},\\
&\forall \mathfrak{A} \forall \mathfrak{A}' \quad \mathfrak{A} \equiv^{C^{3mk}} \mathfrak{A}' \;\rightarrow\; K^{3mk,k}(\mathfrak{A}) = K^{3mk,k}(\mathfrak{A}').
\end{aligned}$$

Sketch of Proof. The crucial observation is that $H_0 := \Gamma_3^{-1} \circ H \circ \Gamma_3$ is well defined, has image in $\Gamma_k\big(\mathrm{fin}[\tau]\big)$ and satisfies for all $\mathfrak{A} \in \mathrm{fin}[\tau]$: $H_0\big(\mathfrak{A}^{[k]}\big) \equiv^{C^2} \mathfrak{A}^{[k]}$.

Let $\mathfrak{A} \in \mathrm{fin}[\tau]$, $\mathfrak{A}^{[k]}$ its k-th power. Let $H\Big(\big(\mathfrak{A}^{[k]}\big)^{[3]}\Big) \simeq \mathfrak{B}^{[3]}$. By Proposition 6.21 we know that $\big(\mathfrak{A}^{[k]}\big)^{[3]} \equiv^{C^2} \mathfrak{B}^{[3]}$ implies that $\mathfrak{A}^{[k]} \equiv^{C^3} \mathfrak{B}$. By Lemma 6.22, $\mathfrak{B}$ therefore is itself a k-th power. A further application of Proposition 6.21 yields $\Gamma_k^{-1}(\mathfrak{B}) \equiv^{C^k} \mathfrak{A}$. As $H(\mathfrak{C})$ only depends on the C^m-theory of $\mathfrak{C}$ by assumption, it follows that $\big[H \circ \Gamma_3\big](\mathfrak{A})$ is determined by the C^{3m}-theory of $\mathfrak{A}$ and finally that $\big[H \circ \Gamma_3 \circ \Gamma_k\big](\mathfrak{A})$ is fully determined by the C^{3mk}-theory of $\mathfrak{A}$ (compare Lemma 6.23). □

7. Canonization for Two Variables

In this chapter we prove that both $L^2_{\infty\omega}$ and $C^2_{\infty\omega}$ admit PTIME canonization. We do so by exhibiting PTIME inverses for I_{L^2} and I_{C^2}. The inversion for I_{L^2} is even PTIME in terms of the size of the I_{L^2}, a phenomenon that we know to be peculiar to the two variable case. These are the main theorems:

Theorem 7.1. *I_{L^2} admits* PTIME *inversion in the strong sense that for each finite relational τ there is a* PTIME *functor $F \colon \{I_{L^2}(\mathfrak{A}) \mid \mathfrak{A} \in \mathrm{fin}[\tau]\} \to \mathrm{stan}[\tau]$, which is an inverse for I_{L^2}:*

$$\forall \mathfrak{A} \quad F\big(I_{L^2}\mathfrak{A}\big) \equiv^{L^2} \mathfrak{A}.$$

It follows that

(i) the range of I_{L^2} can be recognized in PTIME.
(ii) $L^2_{\infty\omega}$ admits PTIME *canonization.*
(iii) PTIME $\cap\, L^2_{\infty\omega}$ *is recursively enumerable (has a recursive presentation).*
(iv) PTIME $\cap\, L^2_{\infty\omega} \equiv \mathrm{FP}\big(I_{L^2}\big) \equiv \mathrm{PTIME}\big(I_{L^2}\big)$.

Compare the general Theorems 6.11 and 6.14 for (ii) and (iii). (i) is obvious: for $\mathfrak{I}$ of the format of an L^2-invariant, $\mathfrak{I} \in \{I_{L^2}(\mathfrak{A}) \mid \mathfrak{A} \in \mathrm{fin}[\tau]\}$ if and only if $F(\mathfrak{I}) \in \mathrm{fin}[\tau]$ and $I_{L^2}\big(F(\mathfrak{I})\big) = \mathfrak{I}$. (i) and the strong form of (iv) (if compared to the statement of Theorem 6.14) are consequences of polynomiality of F in the usual sense.

Theorem 7.2. *I_{C^2} admits* PTIME *inversion. For each finite relational τ there is a* PTIME *functor $F \colon \{I_{C^2}(\mathfrak{A}) \mid \mathfrak{A} \in \mathrm{fin}[\tau]\} \to \mathrm{stan}[\tau]$, which is an inverse for I_{C^2}:*

$$\forall \mathfrak{A} \quad F\big(I_{C^2}\mathfrak{A}\big) \equiv^{C^2} \mathfrak{A}.$$

It follows that

(i) the range of I_{C^2} can be recognized in PTIME.
(ii) $C^2_{\infty\omega}$ admits PTIME *canonization.*
(iii) PTIME $\cap\, C^2_{\infty\omega}$ *is recursively enumerable (has a recursive presentation).*
(iv) PTIME $\cap\, C^2_{\infty\omega} \equiv \mathrm{FP}\big(I_{C^2}\big) \equiv \mathrm{PTIME}\big(I_{C^2}\big)$.

The construction of the inverses is reduced to a combinatorial problem that only deals with the abstract information about the corresponding two-pebble games as represented in the invariants. The relational information encoded in the invariants through the identification of atomic types is at first suppressed for this purpose. The reduced invariants, stripped of the relational particulars, are what we shall call *game tableaux*.

- These game tableaux are introduced in Section 7.1. The inversion problems for I_{L^2} and I_{C^2} are reduced to the problem of constructing realizations for tableaux.
- In Section 7.2 such realizations are constructed in the case of C^2.
- Section 7.3 deals with the corresponding constructions in the case of L^2.

7.1 Game Tableaux and the Inversion Problem

For the start $L^2_{\infty\omega}$ and $C^2_{\infty\omega}$ can be treated in parallel. Recall from Definitions 3.3 and 3.11 the format of the invariants I_{C^2} and I_{L^2}. The special situation in dimension two allows for certain simplifications in their presentation. Consider I_{C^2} first. In the original format:

$$I_{C^2}(\mathfrak{A}) \;=\; \Big(A^2/{\equiv^{C^2}}, \leqslant, \big(P_\theta\big)_{\theta\in \mathrm{Atp}(\tau;2)}, \big(E_j\big)_{j=1,2}, \big(S_\rho\big)_{\rho\in S_2}; \big(\nu_j\big)_{j=1,2}\Big).$$

As pointed out in the general case, it suffices to retain one of the E_j and ν_j each, since the other one remains definable with the help of $S_{(1,2)}$, the encoding of the permutation that exchanges first and second component. This permutation is the only member of S_2 apart from the identity. In the following we denote by T (for transposition) both this exchange of first and second component as a member of S_2 and its operation on the elements of I_{C^2}. The graph of this operation, $S_{(1,2)}$, is also denoted T. This is not likely to cause any confusion, since the transition between these representations is trivial. Retaining $E := E_2$ and $\nu := \nu_2$, we get:

$$\begin{array}{rcccl} E_1 &=& E^T &:=& \big\{(\alpha,\alpha') \,\big|\, (T\alpha, T\alpha')\in E\big\},\\ \nu_1 &=& \nu^T &:=& \nu\circ T. \end{array}$$

We separate the equality type information from the remaining relational atomic information in the P_θ by putting

$$\Delta := \big\{\alpha \,\big|\, (a_1,a_2)\in\alpha \Rightarrow a_1 = a_2\big\}.$$

For notational convenience finally, the partition of the universe into the P_θ is replaced by a function $\Theta\colon A^2/\equiv^{C^2} \to \mathrm{Atp}(\tau;2)$. We thus obtain the following format for the I_{C^2}, which is obviously interdefinable at first-order level with the former one:

$$I_{C^2}(\mathfrak{A}) \;=\; \Big(A^2/\equiv^{C^2}, \leqslant, E, T, \Delta; \Theta, \nu\Big). \tag{7.1}$$

The same modifications apply to I_{L^2}:

$$I_{L^2}(\mathfrak{A}) \;=\; \Big(A^2/\equiv^{L^2}, \leqslant, E, T, \Delta; \Theta\Big). \tag{7.2}$$

Proviso. For the purposes of this chapter we fix the special format for the two-variable invariants according to equations 7.1 and 7.2 above. We regard both I_{C^2} and I_{L^2} as (standard representations of) ordered weighted κ-structures, where $\kappa := \big\{\leqslant, E, T, \Delta\big\}$.

We collect a few obvious facts about the κ-reducts of two-variable invariants, no matter whether I_{C^2} or I_{L^2}, in the following lemma.

Lemma 7.3. *Let $\mathfrak{Q} = \big(Q, \leqslant, E, T, \Delta\big)$ be the κ-reduct of some $I_{C^2}(\mathfrak{A})$ or $I_{L^2}(\mathfrak{A})$. Then $\mathfrak{Q}$ satisfies the following:*

(i) E is an equivalence relation on Q.
(ii) T is (the graph of) an involutive function from Q to Q: $T \circ T = \mathrm{id}_Q$.
(iii) Δ consists of points fixed under T: $T \restriction \Delta = \mathrm{id}_\Delta$.
(iv) each E-class contains exactly one element from Δ.

Proof. (i) – (iii) are obvious on the basis of the definitions. Note in connection with (i) that for the underlying invariant $I_{C^2}(\mathfrak{A})$ or $I_{L^2}(\mathfrak{A})$ an E-class exactly corresponds to the type of a single element of $\mathfrak{A}$. This may be seen as follows. Let $\mathcal{L} = C^2_{\infty\omega}$ or $L^2_{\infty\omega}$, $I_\mathcal{L}$ the corresponding invariant, $\mathfrak{Q} = I_\mathcal{L}(\mathfrak{A}) \restriction \kappa$. Let q be the E-class of an element $\alpha \in Q$. Fix some (a_1, a_2) such that $\alpha = \mathrm{tp}^\mathcal{L}_\mathfrak{A}\big(a_1, a_2\big)$. Then by definition q consists of exactly those α' with $\alpha' = \mathrm{tp}^\mathcal{L}_\mathfrak{A}\big(a_1, a_2'\big)$ for some $a_2' \in A$. Let $\beta = \mathrm{tp}^\mathcal{L}_\mathfrak{A}\big(a_1\big)$. β is fully determined by q since it exactly consists of all those formulae $\varphi(x_1) \in \mathcal{L}[\tau]$ that are members of all $\alpha' \in q$. Conversely, q itself is completely determined by β, since $\alpha' \in Q$ if and only if $\exists x_2 \varphi_{\alpha'}(x_1, x_2) \in \beta$, for some formula $\varphi_{\alpha'}(x_1, x_2)$ that isolates α'.

For (iv) first observe that there must be an element from Δ in each E-class of a real invariant. If $\alpha = \mathrm{tp}^\mathcal{L}_\mathfrak{A}\big(a_1, a_2\big)$, then $\mathrm{tp}^\mathcal{L}_\mathfrak{A}\big(a_1, a_1\big)$ is in Δ and E-related with α. For uniqueness as claimed in (iv) consider $\delta_1, \delta_2 \in \Delta$ and assume that δ_1 and δ_2 are E-related. $\delta_1 = \mathrm{tp}^\mathcal{L}_\mathfrak{A}\big(a_1, a_1\big)$ for some a_1, and by E-relatedness there must be some a_2 such that $\delta_2 = \mathrm{tp}^\mathcal{L}_\mathfrak{A}\big(a_1, a_2\big)$. Since $\delta_2 \in \Delta$, $a_1 = a_2$ and therefore $\delta_1 = \delta_2$. □

The following definition introduces the term *game tableaux* for those κ-structures that are candidates for the relational parts of two-variable invariants according to the last lemma. Note that Θ, the assignment of relational atomic types, is not made part of the game tableaux.

Definition 7.4. *A finite κ-structure $\mathfrak{Q} = \big(Q, \leqslant, E, T, \Delta\big)$ is called a* game tableau *if and only if $\leqslant$ is a linear ordering on Q and $\mathfrak{Q}$ satisfies conditions (i) – (iv) of Lemma 7.3. A* weighted game tableau *is a game tableau $\mathfrak{Q}$ together with a weight function $\nu\colon Q \to \omega \setminus \{0\}$.*

The size of a game tableau $\mathfrak{Q}$ is its size as a relational structure, i.e. the size of its universe Q. The size of a weighted game tableau $(\mathfrak{Q}; \nu)$ is taken to be $\sum_{\alpha \in Q} \nu(\alpha)$. These conventions ensure that the size of the original invariant is polynomially related to the size of the abstracted tableau.

As the $\mathfrak{Q}$ and $(\mathfrak{Q}; \nu)$ are linearly ordered, we may think of them as standard objects. The standardization is then implicitly assumed to be the same as for the I_{L^2} and I_{C^2} in their new format.

Note that the class of game tableaux is first-order definable. Game tableaux and weighted game tableaux are recognizable in LOGSPACE. The following lemma isolates some obvious conditions on the function Θ that in real invariants associates relational types with the elements of the invariant. The proof is immediate and similar to that of Lemma 7.3 above.

Lemma 7.5. *Let $\mathfrak{Q}$ be the κ-reduct of $I_{C^2}(\mathfrak{A})$ or $I_{L^2}(\mathfrak{A})$ for some $\mathfrak{A}$, $\Theta\colon Q \to$ $\mathrm{Atp}(\tau; 2)$ the mapping that associates the relational atomic types with the elements of the invariant. Then Θ satisfies the following conditions:*

(i) if $\alpha \in \Delta$ then $\Theta(\alpha)$ is the type of an identity pair: $x_1 = x_2 \in \Theta(\alpha)$.
(ii) for all $\alpha \in Q$, $\Theta\big(T(\alpha)\big)$ is the atomic type obtained from $\Theta\big(\alpha\big)$ by exchanging x_1 and x_2 in all formulae.
(iii) if δ is the unique element of Δ that is in the E-class of α, then $\Theta(\delta)$ contains all formulae $\varphi(x_1)$ from $\Theta(\alpha)$.

Note that the syntactic conditions in (i), (ii) and (iii) completely determine $\Theta\big(T(\alpha)\big)$ in terms of $\Theta(\alpha)$ in (ii) and $\Theta(\delta)$ in terms of $\Theta(\alpha)$ in (iii).

Definition 7.6. *Let $\mathfrak{Q}$ be a game tableau, Θ a function from its domain to* $\mathrm{Atp}(\tau; 2)$ *for some τ. Θ is a* good extension *of $\mathfrak{Q}$ if conditions (i) – (iii) of Lemma 7.5 are satisfied.*

It can be checked in LOGSPACE whether Θ is a good extension of $\mathfrak{Q}$.

The inversion of an invariant asks for the construction of a relational structure over some n such that the types of pairs in this structure fit the specifications laid down in the given invariant. We first approach this problem at the level of the underlying game tableaux or weighted game tableaux — the relational atomic types, as encoded in the Θ, are disregarded at first. Correspondingly, the result of this approach is somewhat less than a relational structure. We shall call it a *realization* of the given game tableau. It turns out that these realizations govern the combinatorial pattern of relational structures to such an extent that the plain relational information in Θ need only be added in later. Formally we describe the desired realizations as mappings that associate pairs over some standard domain with elements of the game tableau. The intention is that — once we also plug in relational information — this mapping will actually be the projection sending pairs to their types.

Definition 7.7. *Let $\mathfrak{Q}$ be a game tableau. A surjective mapping $\pi\colon n\times n \to Q$ is called a* realization of $\mathfrak{Q}$ over the standard domain n, *if the following conditions are satisfied, where m_1, m_2 range over the elements of n:*

(i) π respects the diagonal: $\pi(m_1, m_2) \in \Delta$ if and only if $m_1 = m_2$.
(ii) π respects T: $\pi(m_1, m_2) = T\big(\pi(m_2, m_1)\big)$, i.e. π commutes with T.
(iii) π respects E: The E-class of $\pi(m_1, m_2)$ is the set of all $\pi(m_1, m_2')$ for $m_2' \in n$.

If $\nu\colon Q \to \omega \setminus \{0\}$ is a weight function on $\mathfrak{Q}$, then we further say that π realizes the weighted tableau $(\mathfrak{Q}; \nu)$ if also

(iv) π is compatible with ν:
$\nu\big(\pi(m_1, m_2)\big) = \big|\big\{m_2' \in n \;\big|\; \pi(m_1, m_2') = \pi(m_1, m_2)\big\}\big|$.

Obviously the definition states a number of conditions that are always satisfied in case that $\mathfrak{Q}$ (and ν) are derived from a real invariant of a structure over n and if π is the natural projection sending pairs of elements to their types. We state this fact as a lemma; the proof is trivial.

Lemma 7.8. *Let $\mathfrak{A} \in \mathrm{stan}[\tau]$ be a τ-structure over universe n. Let $\mathcal{L} = C^2_{\infty\omega}$ or $L^2_{\infty\omega}$, $I_{\mathcal{L}}(\mathfrak{A})$ the corresponding invariant. Let $\mathfrak{Q}$ be the induced game tableau, so that $Q = \mathrm{Tp}^{\mathcal{L}}(\mathfrak{A}; 2)$. Put*

$$\begin{array}{rcl} \pi\colon\; n \times n & \longrightarrow & Q \\ (m_1, m_2) & \longmapsto & \mathrm{tp}^{\mathcal{L}}_{\mathfrak{A}}\big(m_1, m_2\big). \end{array}$$

Then π is a realization of the tableau $\mathfrak{Q}$. In case $\mathcal{L} = C^2_{\infty\omega}$ and if ν is the weight function of $I_{C^2}(\mathfrak{A})$, π is a realization of the weighted tableau $(\mathfrak{Q}; \nu)$.

A realization of a game tableau over n, together with attributions of atomic τ-types (in the form a some good extension) uniquely determines a τ-structure with domain n. Let $\mathfrak{Q}$ be realized by $\pi\colon n \times n \to Q$ and let $\Theta\colon Q \to \mathrm{Atp}(\tau; 2)$ be good in the sense of Definition 7.6. Assume first that τ contains no relation symbols of arity greater than 2. Then there is a unique structure $\mathfrak{A}\big(\pi, \Theta\big) \in \mathrm{stan}[\tau]$ over n for which

$$\forall m_1 \forall m_2 \;\; \mathrm{atp}_{\mathfrak{A}}\big(m_1, m_2\big) = \Theta\big(\pi(m_1, m_2)\big).$$

Uniqueness is obvious. For the existence claim one has to check that the conditions expressed in the above equations are compatible. The requirements for realizations and for good Θ are designed just to guarantee this compatibility. For instance if $m_1 = m_2$, we have, by a corresponding condition on realizations, that $\pi(m_1, m_1) \in \Delta$, whence it follows that $\Theta\big(\pi(m_1, m_1)\big)$ is an atomic type of an identity pair. For any $m_1, m_2 \in n$, compatibility of $\Theta\big(\pi(m_1, m_2)\big)$ with $\Theta\big(\pi(m_1, m_1)\big)$ follows from the fact that $\pi(m_1, m_1)$ must be the unique element of Δ in the E-class of $\pi(m_1, m_2)$, since π respects E. But then $\Theta\big(\pi(m_1, m_1)\big)$ corresponds to the restriction of $\Theta\big(\pi(m_1, m_2)\big)$ to the first component as Θ is good.

In order to extend the definition of $\mathfrak{A}(\pi, \Theta)$ in a well-defined way to the general case in which relation symbols of arity greater than 2 are admitted in τ, we stipulate that no tuple involving more than two distinct components is put into the interpretation of such relations.

Definition 7.9. *For a realization $\pi: n \times n \to Q$ of a game tableau $\mathfrak{Q}$ and good $\Theta: Q \to \mathrm{Atp}(\tau; 2)$ let $\mathfrak{A}(\pi, \Theta)$ be the unique τ-structure over standard universe n induced by π and Θ as described above.*

Note that $\mathfrak{A}(\pi, \Theta)$ is constructible from π and Θ in PTIME.

The following is an obvious statement to the effect that in the intended case — the case that all the data are obtained from a real structure over some standard universe — $\mathfrak{A}(\pi, \Theta)$ essentially reproduces that original structure.

Lemma 7.10. *Let $\mathfrak{A} \in \mathrm{stan}[\tau]$, $\mathcal{L} = C^2_{\infty\omega}$ or $L^2_{\infty\omega}$, $\mathfrak{Q}$ the game tableau induced by $I_{C^2}(\mathfrak{A})$ or $I_{L^2}(\mathfrak{A})$, respectively. Let $\Theta: Q \to \mathrm{Atp}(\tau; 2)$ be the good extension induced by the invariant itself. Let $\pi: n \times n \longrightarrow Q$ be the realization that is the natural projection $\pi: A^2 \to A^2/\equiv^{\mathcal{L}}$. If τ contains no relation symbols of arity greater than 2 then $\mathfrak{A}(\pi, \Theta) = \mathfrak{A}$. Otherwise $\mathfrak{A}$ and $\mathfrak{A}(\pi, \Theta)$ agree on all atoms involving at most two elements, so that at least $\mathfrak{A} \equiv^{\mathcal{L}} \mathfrak{A}(\pi, \Theta)$.*

The following proposition is crucial for showing that inversion for the I_{C^2} and I_{L^2} reduces to the construction of realizations for (weighted) tableaux.

Proposition 7.11. *Let $\mathfrak{Q}$ be a game tableau, $\Theta: Q \to \mathrm{Atp}(\tau; 2)$ a good extension of $\mathfrak{Q}$.*

(i) If π and π' are any two realizations of $\mathfrak{Q}$ then $\mathfrak{A}(\pi, \Theta) \equiv^{L^2} \mathfrak{A}(\pi', \Theta)$.
(ii) If $\nu: Q \to \omega \setminus \{0\}$ is a weight function on $\mathfrak{Q}$ and π and π' are any two realizations of the weighted tableau $(\mathfrak{Q}; \nu)$ then $\mathfrak{A}(\pi, \Theta) \equiv^{C^2} \mathfrak{A}(\pi', \Theta)$.

Proof. Consider (ii), the case of C^2. It has to be shown that $\mathfrak{A} := \mathfrak{A}(\pi, \Theta)$ and $\mathfrak{A}' := \mathfrak{A}(\pi', \Theta)$ satisfy exactly the same C^2-types. Using the game characterization for C^2-equivalence, Theorem 2.2, we show that player **II** has a strategy to maintain the condition that $\pi(a_1, a_2) = \pi'(a_1', a_2')$ throughout all stages $(\mathfrak{A}, (a_1, a_2); \mathfrak{A}', (a_1', a_2'))$ in the infinite C^k-game. This is the natural condition since realizations are modelled to describe the projections to the C^2-types. This condition is also sufficient for a strategy in the game since the atomic types of pairs over $\mathfrak{A}$ and $\mathfrak{A}'$ are determined by $\Theta \circ \pi$ and $\Theta \circ \pi'$ respectively.

Assume that $\pi(a_1, a_2) = \pi'(a_1', a_2') = \alpha_0$ in the current position. Everything is explicitly symmetric with respect to $\mathfrak{A}$ and $\mathfrak{A}'$ and implicitly also with respect to first or second component, since realizations and good extensions respect T. Let therefore without loss of generality player **I** choose pebble 2 and put forward the challenge $B \subseteq A$. Let $q \subseteq Q$ be the E-class of α_0. For each $\alpha \in q$, let $B_\alpha := \{b \in B \mid \pi(a_1, b) = \alpha\}$. It follows that B is the disjoint union of the B_α — only $\alpha \in q$ need be considered since π respects E. Since π is also compatible with ν, $|B_\alpha| \leqslant \nu(\alpha)$. π' respects E and ν as well so that in $\mathfrak{A}'$ there are disjoint subsets $B'_\alpha \subseteq \{b' \in A' \mid \pi'(a'_1, b') = \alpha\}$ with $|B_\alpha| = |B'_\alpha|$. In fact, for each $\alpha \in q$ we have that the size of $\{b' \in A' \mid \pi'(a'_1, b') = \alpha\}$ must be $\nu(\alpha)$. Let **II** respond with $B' := \bigcup_{\alpha \in q} B'_\alpha$. In the second exchange of this round **I** now chooses $b' \in B'_\alpha$ for some $\alpha \in q$, so that **II** can answer with any $b \in B_\alpha$ and the desired equality $\pi(a_1, b) = \pi'(a'_1, b') = \alpha$ is maintained. □

Together with Lemma 7.10 this proposition yields the main preparatory result for the construction of inverses: full reduction of the inversion problem to that of finding realizations. We give separate statements for $L^2_{\infty\omega}$ and $C^2_{\infty\omega}$.

Theorem 7.12. *Let $\mathfrak{Q}$ be a game tableau, $\Theta: Q \to \mathrm{Atp}(\tau; 2)$ a function. The following are equivalent:*

(i) $(\mathfrak{Q}; \Theta) = I_{L^2}(\mathfrak{A})$ *for some* $\mathfrak{A} \in \mathrm{fin}[\tau]$.
(ii) Θ *is a good extension of* $\mathfrak{Q}$ *and there is a realization* π *of* $\mathfrak{Q}$ *such that* $I_{L^2}(\mathfrak{A}(\pi, \Theta)) = (\mathfrak{Q}; \Theta)$.
(iii) Θ *is a good extension of* $\mathfrak{Q}$, *there is a realization of* $\mathfrak{Q}$, *and for all realizations* π *of* $\mathfrak{Q}$: $I_{L^2}(\mathfrak{A}(\pi, \Theta)) = (\mathfrak{Q}; \Theta)$.

Theorem 7.13. *For a weighted game tableau $(\mathfrak{Q}; \nu)$ and a function $\Theta: Q \to \mathrm{Atp}(\tau; 2)$ the following are equivalent:*

(i) $(\mathfrak{Q}; \Theta, \nu) = I_{C^2}(\mathfrak{A})$ *for some* $\mathfrak{A} \in \mathrm{fin}[\tau]$.
(ii) Θ *is a good extension of* $\mathfrak{Q}$ *and there is a realization* π *of* $(\mathfrak{Q}; \nu)$ *such that* $I_{C^2}(\mathfrak{A}(\pi, \Theta)) = (\mathfrak{Q}; \Theta, \nu)$.
(iii) Θ *is a good extension of* $\mathfrak{Q}$, *there is a realization of* $(\mathfrak{Q}; \nu)$, *and for all realizations* π *of* $(\mathfrak{Q}; \nu)$: $I_{C^2}(\mathfrak{A}(\pi, \Theta)) = (\mathfrak{Q}; \Theta, \nu)$.

Proof. The proof is indicated for the case of C^2: (iii) ⇒ (ii) ⇒ (i) is obvious. Assume (i). Without loss of generality $\mathfrak{A} \in \mathrm{stan}[\tau]$. Then the natural projection $\pi: A^2 \to A^2 /\!\equiv^{C^2}$ yields a realization, see Lemma 7.8. Any two realizations of $(\mathfrak{Q}; \nu)$ lead to C^2-equivalent structures $\mathfrak{A}(\pi, \Theta)$ by Proposition 7.11; we therefore get (iii). □

These theorems reduce the proof of the main theorems on Ptime inversion for I_{L^2} and I_{C^2} to the following claims. Recall for complexity considerations that the size of a tableau $\mathfrak{Q}$ is the size of its universe Q as usual, while the size of a weighted tableau $(\mathfrak{Q}; \nu)$ is $\sum_{\alpha \in Q} \nu(\alpha)$.

Theorem 7.14. *There are* Ptime *algorithms $\mathcal{A}$ and $\mathcal{A}^*$ defined on all κ-structures $\mathfrak{Q}$, respectively on all κ-structures with positive weights $(\mathfrak{Q}; \nu)$, such that*

(a) if $\mathfrak{Q}$ is a game tableau that admits any realization then $\mathcal{A}$ applied to $\mathfrak{Q}$ yields a realization of $\mathfrak{Q}$.
(b) if $(\mathfrak{Q}; \nu)$ is a weighted game tableau that admits any realization then $\mathcal{A}^*$ applied to $(\mathfrak{Q}; \nu)$ yields a realization of $(\mathfrak{Q}; \nu)$.

Such algorithms provide the basis for Theorems 7.1 and 7.2. We sketch an algorithm F as required in Theorem 7.1 with respect to $L^2_{\infty\omega}$. The case of $C^2_{\infty\omega}$ is entirely analogous. The input is a structure $\mathfrak{I} = (\mathfrak{Q}; \Theta)$ of the format of an $I_{L^2}(\mathfrak{A})$. The following diagram describes the desired algorithm:

- $\mathfrak{I} = (\mathfrak{Q}; \Theta)$; check whether $\mathfrak{Q}$ is a game tableau and whether Θ is a good extension of $\mathfrak{Q}$

  > if $\mathfrak{Q}$ is not a game tableau, or if Θ is not a good extension of $\mathfrak{Q}$, then $\mathfrak{I} \notin \text{range}(I_{L^2})$

- In the positive case apply $\mathcal{A}$ to $\mathfrak{Q}$ and check whether the output $\mathcal{A}(\mathfrak{Q})$ is a realization of $\mathfrak{Q}$

  > if the output is not a realization of $\mathfrak{Q}$, then $\mathfrak{I} \notin \text{range}(I_{L^2})$

- Let in the positive case $\pi = \mathcal{A}(\mathfrak{Q})$ be that realization
- Construct $\mathfrak{A}(\pi, \Theta) \in \text{stan}[\tau]$
- Compute $I_{L^2}(\mathfrak{A}(\pi, \Theta))$ and compare with $\mathfrak{I}$

  > if $I_{L^2}(\mathfrak{A}(\pi, \Theta)) \neq \mathfrak{I}$, then $\mathfrak{I} \notin \text{range}(I_{L^2})$

- In the positive case output $F(\mathfrak{I}) := \mathfrak{A}(\pi, \Theta)$.

Correctness essentially depends on Theorem 7.12, which says that any realization of the game tableau leads to a successful construction of an inverse to the invariant if there is any! The rest of this chapter is devoted to the proof of Theorem 7.14.

7.1.1 Modularity of Realizations

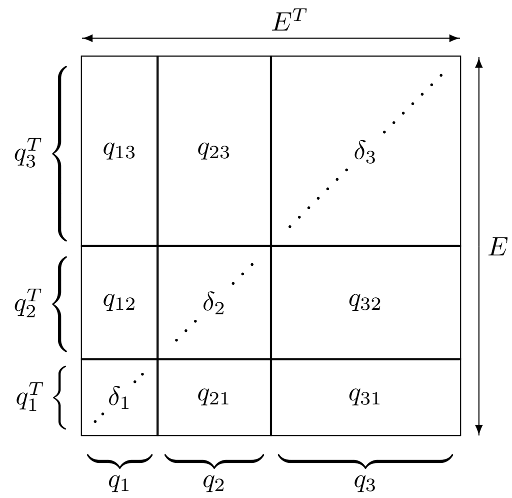
This section exhibits an important modularity property of the game tableaux that facilitates the construction of realizations. The overall problem can be decomposed into simpler subproblems, whose solutions form the building blocks for the desired realization.

Definition 7.15. *Let $\mathfrak{Q}$ be a game tableau. We enumerate the E-classes as $q_1, \dots, q_l$. Here $l = |Q/E|$ and the ordering is that induced by $\leqslant^{\mathfrak{Q}}$ in terms of $\leqslant$-least elements of the classes.*

(i) We denote by δ_i the unique element of $q_i \cap \Delta$.
(ii) Let $q_i^T := \{T\alpha \mid \alpha \in q_i\}$.
(iii) Let $q_{ij} := q_i \cap q_j^T$.

Note that the q_i^T are the equivalence classes with respect to E^T, defined by $E^T = \{(\alpha, \alpha') \mid (T\alpha, T\alpha') \in E\}$. For real invariants $E^T = E_1$ is accessibility via a move in the first component. Note also that $T(q_{ij}) = q_{ji}$ and that therefore $q_{ij} \cap q_{ji} = \emptyset$ unless $i = j$; in this case, $\delta_i \in q_{ii} \neq \emptyset$. Simple examples show, however, that by no means need $q_{ii} = \{\delta_i\}$. Consider a directed cycle of length 4 as a graph. For the associated C^2-invariant there is only one type in Δ, E and E^T are both trivial, but there are 4 different C^2-types.

Fig. 7.1



The following characterization of the q_{ij} is technically very useful, for a pictorial presentation see Figure 7.1. In Figure 7.1 the fine structure of $\mathfrak{Q}$ is depicted as projected onto some $n \times n$ square that would be a realization.

Lemma 7.16. *Let $\mathfrak{Q}$ be a game tableau, π be a realization of $\mathfrak{Q}$. Then*

$$\pi(m_1, m_2) \in q_{ij} \quad \text{if and only if} \quad \pi(m_1, m_1) = \delta_i \wedge \pi(m_2, m_2) = \delta_j.$$

Proof. Observe that $\pi(m_1, m_2) \in q_{ij}$ implies that $\pi(m_1, m_2)$ and δ_i are E-related. Therefore there must be some m_2' such that $\pi(m_1, m_2') = \delta_i$. As π respects Δ, $m_2' = m_1$, so that $\pi(m_1, m_1) = \delta_i$. Applying the same argument to E^T we get $\pi(m_2, m_2) = \delta_j$.

Conversely, $\pi(m_1, m_1) = \delta_i \in q_i$ implies $\pi(m_1, m_2) \in q_i$ for all m_2. Repeating the same argument we get that $\pi(m_2, m_2) = \delta_j$ implies $\pi(m_2, m_1) \in q_j$ so that $\pi(m_1, m_2) \in q_j^T$. Putting these together, $\pi(m_1, m_1) = \delta_i$ and $\pi(m_2, m_2) = \delta_j$ imply $\pi(m_1, m_2) \in q_{ij}$. □

We define the restrictions of a game tableau $\mathfrak{Q}$ to its subdomains q_{ij}. Note that in restriction to each q_{ij} the equivalence relations E and E^T become trivial since the q_{ij} are the classes of the common refinement of the two. As for Δ it is obvious that $\Delta \cap q_{ij} = \emptyset$ if $i \neq j$, and $\Delta \cap q_{ii} = \{\delta_i\}$. T is an involutive mapping from q_{ii} to itself, and turns into a bijection between q_{ij} and q_{ji} for $i \neq j$. If we also consider weighted tableaux, it makes sense to retain both weight functions, ν and ν^T over each q_{ij} with $i \neq j$ since T is no longer internal to q_{ij}.

Definition 7.17. *Let $\mathfrak{Q}$ be a game tableau, the q_{ij} as defined in Definition 7.15. The restriction of $\mathfrak{Q}$ to q_{ij} is defined to be*

$$\mathfrak{Q}_{ij} := \begin{cases} \big(q_{ii}, \leqslant\restriction q_{ii}, T \restriction q_{ii}, \{\delta_i\}\big) & \textit{for the diagonal case } j = i \\ \big(q_{ij}, \leqslant\restriction q_{ii}\big) & \textit{for the off-diagonal case } i \neq j. \end{cases}$$

For the restrictions of a weighted tableau $(\mathfrak{Q};\nu)$, put

$$(\mathfrak{Q};\nu)_{ij} := \begin{cases} \big(\mathfrak{Q}_{ii}; \nu \restriction q_{ii}\big) & \textit{for } j = i \\ \big(\mathfrak{Q}_{ij}; \nu \restriction q_{ij}, \nu^T \restriction q_{ij}\big) & \textit{for } i \neq j. \end{cases}$$

This decomposition calls for an adapted notion of realizations. The modifications and simplifications required with respect to Definition 7.7 are canonical. The diagonal restrictions to the q_{ii} can in fact be regarded as special cases of game tableaux, with trivial E. It is only for the off-diagonal boxes that formal modifications are required.

Definition 7.18. *Let $\mathfrak{Q}$ be a game tableau, $\mathfrak{Q}_{ij}$ its restriction to a subdomain q_{ij}. Assume first $i \neq j$. A surjective mapping $\pi\colon s \times t \to q_{ij}$ is a* realization *of $\mathfrak{Q}_{ij}$, if for all $m_1 \in s$ and all $m_2 \in t$:*

$$\big\{\pi(m_1, m_2') \bigm| m_2' \in t\big\} = \big\{\pi(m_1', m_2) \bigm| m_1' \in s\big\} = q_{ij}.$$

π realizes the weighted restriction $(\mathfrak{Q};\nu)_{ij} = (\mathfrak{Q}_{ij}; \nu \restriction q_{ij}, \nu^T \restriction q_{ij})$ *if for all $\alpha \in q_{ij}$, $m_1 \in s$ and all $m_2 \in t$:*

$$\big|\big\{m_2' \in t \bigm| \pi(m_1, m_2') = \alpha\big\}\big| = \nu(\alpha),$$
$$\big|\big\{m_1' \in s \bigm| \pi(m_1', m_2) = \alpha\big\}\big| = \nu^T(\alpha).$$

For $i = j$ the conditions that a surjective mapping $\pi\colon s \times s \to q_{ii}$ realizes $\mathfrak{Q}_{ii}$ or $(\mathfrak{Q},\nu)_{ii}$ are those of Definition 7.7 applied to the game tableau $\big(q_{ii}, \leqslant\restriction q_{ii}, q_{ii} \times q_{ii}, T \restriction q_{ii}, \{\delta_i\}\big)$ and to the weighted game tableau $\big(q_{ii}, \leqslant\restriction q_{ii}, q_{ii} \times q_{ii}, T \restriction q_{ii}, \{\delta_i\}; \nu \restriction q_{ii}\big)$.

Suppose $\pi\colon n \times n \to Q$ is a realization of $\mathfrak{Q}$ or $(\mathfrak{Q},\nu)$ over n. Let n be decomposed into the subsets $\big\{m \in n \bigm| \pi(m,m) = \delta_i\big\}$. Obviously, n is the disjoint union of these. Without loss of generality we may assume that n is the disjoint ordered sum $n = \sum_i n_i$, $n_i = \big\{m \in n \bigm| \pi(m,m) = \delta_i\big\}$. Formally this means that n_1 is identified with an initial subset of n, n_2 consists of a consecutive interval following that initial segment and so on. In particular *identifying n with the disjoint ordered sum* of the n_i implies that the subsets n_i are embedded in a well-defined way into n such that n is the disjoint union of the embedded n_i. In the present case this situation may be assumed without loss of generality because a realization over n can be composed with any permutation of n in the obvious manner to yield a new realization, equivalent with the former one for our purposes.

With such a presentation of $n = \sum_i n_i$, $n_i = \{m \in n \mid \pi(m,m) = \delta_i\}$, we immediately have that the restrictions of π to the subsets $n_i \times n_j \subseteq n \times n$ provide realizations for the restrictions $\mathfrak{Q}_{ij}$ and $(\mathfrak{Q};\nu)_{ij}$. It is straightforward to check the conditions mentioned in the last definition. Let us supply the argument for surjectivity of $\pi_{ij} := \pi \restriction n_i \times n_j \to q_{ij}$: by Lemma 7.16, $\pi(m_1, m_2) \in q_{ij}$ if and only if $\pi(m_1, m_1) = \delta_i$ and $\pi(m_2, m_2) = \delta_j$, i.e. if and only if $m_1 \in n_i$ and $m_2 \in n_j$. Thus surjectivity of π_{ij} follows from surjectivity of π itself.

The interesting fact is that, conversely, realizations of the individual restrictions can be fit together to form a realization of the whole (weighted) tableau if they satisfy just the most obvious compatibility conditions relating the sizes of the subdomains.

Lemma 7.19. *Let $\mathfrak{Q}$ be a game tableau, $\mathfrak{Q}_{ij}$, $1 \leqslant i,j \leqslant l$, its restrictions to the q_{ij} defined as above. Assume that for some tuple $(n_i)_{1\leqslant i\leqslant l}$ of positive numbers there are surjective mappings $\pi_{ij}: n_i \times n_j \to q_{ij}$ for each $1 \leqslant i \leqslant j \leqslant l$, such that π_{ij} is a realization of $\mathfrak{Q}_{ij}$. Then the following is a realization of $\mathfrak{Q}$ on the disjoint ordered sum $n := \sum_i n_i$:*

$$\pi(m_1, m_2) := \begin{cases} \pi_{ij}(m_1, m_2) & \text{if } m_1 \in n_i,\ m_2 \in n_j,\ i \leqslant j, \\ T\pi_{ji}(m_2, m_1) & \text{if } m_1 \in n_i,\ m_2 \in n_j,\ j < i. \end{cases}$$

The same holds for realizations of a weighted tableau $(\mathfrak{Q};\nu)$.

Before giving a proof, let us note that together with the preceding considerations we have thus found that a (weighted) game tableau is realizable if and only if its restrictions are realizable over subdomains of matching sizes. In terms of at first arbitrary domains for the realization of the restrictions, $\pi_{ij}: n_1^{ij} \times n_2^{ij} \to q_{ij}$ the conditions for *matching size* are that the n_1^{ij} are independent of j, and that the n_2^{ij} are independent of i.

Proposition 7.20. *A game tableau $\mathfrak{Q}$ has a realization over n if and only if $n = \sum_i n_i$ for $n_i > 0$ such that each of its restrictions $\mathfrak{Q}_{ij}$ admits a realization over $n_i \times n_j$.*

Similarly, a weighted game tableau $(\mathfrak{Q};\nu)$ has a realization over n if and only if $n = \sum_i n_i$ such that its restrictions $(\mathfrak{Q};\nu)_{ij}$ admit realizations over $n_i \times n_j$, for all i,j.

Proof (of Lemma 7.19). Recall from Definition 7.7 the conditions on a realization π of $\mathfrak{Q}$. It is clear that π as defined above is surjective, since the π_{ij} are surjective mappings to the q_{ij} and since $Q = \bigcup q_{ij} = \bigcup_{i \leqslant j} q_{ij} \cup \bigcup_{j<i} Tq_{ji}$ as $q_{ij} = Tq_{ji}$. π respects the diagonal, because the π_{ii} do. π commutes with T, because the π_{ii} do and because the appropriate transformation under T is explicitly built into π on the off-diagonal boxes. We check that π is correct with respect to E. By construction, $\pi(m_1, m_2) \in q_{ij}$ if and and only if $m_1 \in n_i$ and $m_2 \in n_j$. Varying $m_2' \in n_{j'}$ we get $\{\pi(m_1, m_2') \mid m_2' \in n_{j'}\} = q_{ij'}$, since the corresponding behaviour is required of the $\pi_{ij'}$ or $\pi_{j'i}$. Therefore $\{\pi(m_1, m_2') \mid m_2' \in n\} = q_i$ as required.

The proof for a weighted tableau is similar. The multiplicity requirements for the realizations π_{ij} of the restrictions immediately imply that also the composition π realizes the multiplicities prescribed by the overall weight function. □

We now pursue the actual constructions of realizations in separate presentations for $L^2_{\infty\omega}$ and $C^2_{\infty\omega}$. There are more constraints in the case of $C^2_{\infty\omega}$, so that the constructions are more difficult. On the other hand these constructions appear more straightforward because there are more data available and correspondingly fewer arbitrary choices to be made. We treat $C^2_{\infty\omega}$ or the realization of weighted tableaux first and specialize and modify this treatment in Section 7.3 to obtain the results for $L^2_{\infty\omega}$.

7.2 Realizations for I_{C^2}

7.2.1 Necessary Conditions

The numerical information contained in the I_{C^2}, and in the weighted game tableaux that derive from these, fixes the size of a possible realization and the n_i as in the compatibility conditions in Proposition 7.20. Fix a weighted game tableau $(\mathfrak{Q}; \nu)$, with q_i, q_{ij}, δ_i for $1 \leqslant i, j \leqslant l$ defined according to Definition 7.15. Further define the numbers

$$n_j := \sum_{\alpha \in q_{ij}} \nu(\alpha) \quad \text{and} \quad n := \sum_i n_i.$$

An equivalent definition of n_i in terms of $(\mathfrak{Q}; \nu)_{ij}$ is $n_i = \sum_{\alpha \in q_{ij}} \nu^T(\alpha)$. Equivalence with the above is a consequence of the fact that T is an involutive bijection between q_{ij} and q_{ji} and that, by definition, $\nu^T = \nu \circ T$.

Lemma 7.21. *Let the n_i and n be as just defined. If $(\mathfrak{Q}; \nu)$ has any realization, then it must be over n. The induced realizations of the restrictions $(\mathfrak{Q}; \nu)_{ij}$ must be over domains $n_i \times n_j$.*

Proof. Let π be a realization of $(\mathfrak{Q}; \nu)$ over s. By Lemma 7.16, $\pi(m_1, m_2) \in q_{ij}$ if and only if $\pi(m_1, m_1) = \delta_i$ and $\pi(m_2, m_2) = \delta_j$. Choose $m_1 \in s$ such that $\pi(m_1, m_1) = \delta_i$. Then

$$\left|\{m_2 \in s \mid \pi(m_1, m_2) \in q_{ij}\}\right| \quad = \quad \textstyle\sum_{\alpha \in q_{ij}} \nu(\alpha) \quad = \quad n_j\,.$$

It further follows that also $n_j = \left|\{m_2 \in s \mid \pi(m_2, m_2) = \delta_j\}\right|$.

Note in particular, that these numbers depend on j and not on i. This is just the compatibility condition of Proposition 7.20. Applying the same argument to variations in the first component, and with fixed m_2 for which $\pi(m_2, m_2) = \delta_j$, we obtain

$$\begin{aligned}\left|\{m_1 \in s \mid \pi(m_1, m_2) \in q_{ij}\}\right| \quad &= \quad \textstyle\sum_{\alpha \in q_{ij}} \nu^T(\alpha) \\ &= \quad \textstyle\sum_{\alpha \in q_{ji}} \nu(\alpha) \quad = \quad n_i.\end{aligned}$$

This shows that the induced realization of $(\mathfrak{Q}; \nu)_{ij}$ must be over $n_i \times n_j$. The first claim of the lemma, $s = \sum_i n_i$, follows directly from Proposition 7.20. □

The following lemma states some necessary conditions for the realizability of $(\mathfrak{Q}; \nu)$ in terms of the restrictions $(\mathfrak{Q}; \nu)_{ij}$. Sufficiency of these conditions will be shown in the sequel.

Lemma 7.22. *Any realization of $(\mathfrak{Q}; \nu)_{ij}$ is over $n_i \times n_j$. Recall that the numbers n_i and n_j are defined in terms of $(\mathfrak{Q}; \nu)_{ij}$ as*

$$n_i = \sum_{\alpha \in q_{ij}} \nu^T(\alpha) \qquad \textit{and} \qquad n_j = \sum_{\alpha \in q_{ij}} \nu(\alpha).$$

If $(\mathfrak{Q}; \nu)_{ij}$ has a realization then for all $\alpha \in q_{ij}$:

$$(*) \qquad \frac{\nu(\alpha)}{\nu^T(\alpha)} = \frac{n_j}{n_i}.$$

For realizability of a diagonal restriction $(\mathfrak{Q}; \nu)_{ii}$ it is necessary that in addition $\nu(\delta_i) = 1$, and that if n_i is odd, then for all $\alpha \in q_{ii} \setminus \{\delta_i\}$:

$$(**) \qquad T\alpha = \alpha \implies \nu(\alpha) \text{ is even.}$$

Proof. Suppose that $\pi\colon s \times t \to q_{ij}$ realizes $(\mathfrak{Q}; \nu)_{ij}$ (cf. Definition 7.18). That $s \times t = n_i \times n_j$ is shown by an argument similar that in Lemma 7.21, but in restriction to the individual q_{ij}. We show that $t = n_j$. For all $m_1 \in s$ and all $\alpha \in q_{ij}$, $\left|\{m_2 \in t \mid \pi(m_1, m_2) = \alpha\}\right| = \nu(\alpha)$. Therefore $t = \left|\{m_2 \in t \mid \pi(m_1, m_2) \in q_{ij}\}\right| = \sum_{\alpha \in q_{ij}} \nu(\alpha) = n_j$.

For the quotient conditions $(*)$ it suffices to count the number of pairs that are mapped to α, first in column-wise fashion, then row-wise and equate the two:

$$\begin{aligned}\left|\{(m_1, m_2) \mid \pi(m_1, m_2) = \alpha\}\right| \quad &= \quad \textstyle\sum_{m_1 \in s} \left|\{m_2 \mid \pi(m_1, m_2) = \alpha\}\right| \\ &= \quad \textstyle\sum_{m_1 \in s} \nu(\alpha) \;=\; s\,\nu(\alpha),\end{aligned}$$

$$\begin{aligned}\left|\{(m_1, m_2) \mid \pi(m_1, m_2) = \alpha\}\right| \quad &= \quad \textstyle\sum_{m_2 \in t} \left|\{m_1 \mid \pi(m_1, m_2) = \alpha\}\right| \\ &= \quad \textstyle\sum_{m_2 \in t} \nu^T(\alpha) \;=\; t\,\nu^T(\alpha).\end{aligned}$$

Consider now the additional constraints expressed for the diagonal case. Necessity of $\nu(\delta_i) = 1$ is obvious, since π must respect $\Delta \restriction q_{ii} = \{\delta_i\}$. For $(**)$ assume $s = t = n_i$ is odd and that $\alpha \neq \delta_i$ is a fixed point under T. $\pi^{-1}(\alpha)$ must be disjoint from the diagonal $\{(m, m) | m \in s\}$, because $\pi(m, m) = \delta_i$. This implies that T operates as a fixed-point free involutive bijection on $\pi^{-1}(\alpha)$. Therefore $|\pi^{-1}(\alpha)|$ must be even. The above counting equations imply that this number equals $|\{(m_1, m_2) \mid \pi(m_1, m_2) = \alpha\}| = s\,\nu(\alpha)$. If s is odd, therefore, $\nu(\alpha)$ must be even. $\square$

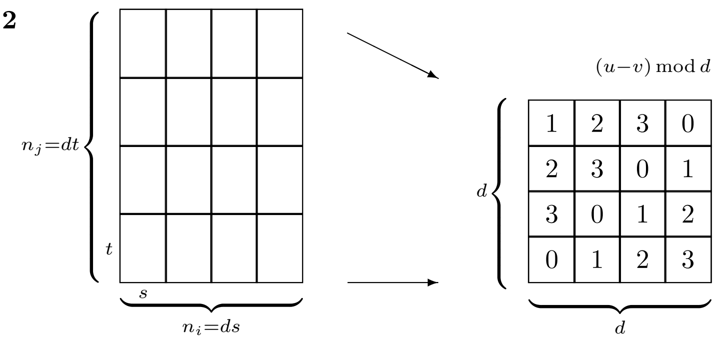
7.2.2 Realizations of the Off-Diagonal Boxes

We turn to the proof of sufficiency of the conditions expressed in the last lemma. The realization of off-diagonal restrictions turns out to be quite straightforward.

Lemma 7.23. *Let $(\mathfrak{Q}; \nu)_{ij}$, $i \neq j$ satisfy condition $(*)$ of Lemma 7.22: for all α $\nu(\alpha)/\nu^T(\alpha) = n_j/n_i$. Then there is a realization $\pi: n_i \times n_j \to q_{ij}$ of $(\mathfrak{Q}; \nu)_{ij}$. Such realizations are constructible in time polynomial in $n_i n_j$.*

Proof. Let t/s be the reduced presentation of n_j/n_i. Let $\alpha_1, \ldots, \alpha_r$ be an enumeration of q_{ij} as ordered by $\leqslant$. By assumption there exist numbers d_k, for $1 \leqslant k \leqslant r$, such that $\nu(\alpha_k) = d_k t$ and $\nu^T(\alpha_k) = d_k s$. Putting $d = \sum d_k$ we have $ds = n_i$ and $dt = n_j$ by definition of n_i and n_j. For the following compare Figure 7.2.

Fig. 7.2



Identify d with the ordered disjoint sum of the d_k and let $g: d \to \{1, \ldots, r\}$ be the function that characterizes the embedded d_k: $g(u)$ is that k with $u \in d_k$. Also identify n_i with the product $d \times s$, and similarly n_j with $d \times t$. Note that the sum and product identifications can be uniquely defined with the help of the natural orderings (we have done this explicitly for the sum above). Define a surjective function

$$\begin{array}{rcl} f\colon (d \times s) \times (d \times t) & \longrightarrow & m \\ \big((u,x),(v,y)\big) & \longmapsto & (u-v) \bmod d\,. \end{array}$$

Figure 7.2 sketches the situation in an example with $d = 4$. The passage from the left to the right indicates the effect of the projections $n_i = d \times s \to d$ and $n_j = d \times t \to d$ involved in the definition of f. On the right-hand side the distribution of values for the function $(u - v) \bmod d$ is indicated.

A realization π can now be defined on $n_i \times n_j = (d \times s) \times (d \times t)$ by

$$\pi(m_1, m_2) := \alpha_{g(f(m_1,m_2))}.$$

π factorizes with respect to f and maps all those blocks, whose values under f fall into $d_k \subseteq d$, to α_k. Let us check that π realizes the multiplicities for columns as specified by $\nu \restriction q_{ij}$. Consider $m_1 = (u, x) \in d \times s$:

$$\big\{m_2 \in d \times t \;\big|\; \pi(m_1, m_2) = \alpha_k\big\} = \big\{m_2 \in d \times t \;\big|\; f(m_1, m_2) \in d_k\big\}.$$

$f(m_1, (v, y)) = (u - v) \bmod d$ so that there are exactly $d_k t = \nu(\alpha_k)$ many $m_2 = (v, y)$ such that $f(m_1, m_2) \in d_k$. The multiplicity conditions on rows are checked to be in accordance with $\nu^T \restriction q_{ij}$ in exactly the same way.

Note that the proposed construction of a realization is quite definite: we have sketched how to construct a particular solution to the realization problem for an off-diagonal restriction. This construction is clearly in PTIME with respect to the product $n_i n_j$. □

7.2.3 Magic Squares

Sufficiency of the conditions of Lemma 7.22 and the construction of realizations is combinatorially more demanding for diagonal restrictions $(\mathfrak{Q}; \nu)_{ii}$ because of the symmetries imposed by T. We first present a preparatory lemma on certain colourings of squares.

The most complicated case in the construction of realizations for diagonal restrictions $(\mathfrak{Q}; \nu)_{ii}$ — the case of even n_i with fixed points $\alpha \neq \delta_i$ under T, as we shall see — reduces to the construction of such colourings. The colourings described in the following lemma in fact present the worst case for the construction of a realization. The symmetry requirements for these colourings are reminiscent of magic squares and related number puzzles (and call for a try with paper and pencil).

Lemma 7.24. *Let n be even. Then there is a colouring $c\colon n \times n \longrightarrow n$ of the n-square $n \times n$ with n colours $0, \ldots, n-1$ with the following properties:*

(i) the main diagonal, i.e. all identity pairs, are coloured 0.
(ii) each colour occurs exactly once in each row and in each column.
(iii) the entire colouring is mirror symmetric with respect to the main diagonal. In other words the colouring is invariant under T.

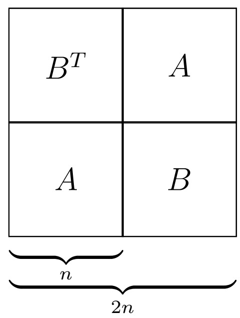
A colouring of this kind can be constructed in time polynomial in n.

Observe that the same puzzle cannot be solved for odd n: each colour apart from 0 has to occur an even number of times because T operates as a fixed-point free involutive mapping on the points of this colour.

An example of a colouring of the 6×6 square according to the requirements of the lemma is given in Figure 7.5 (a) below.

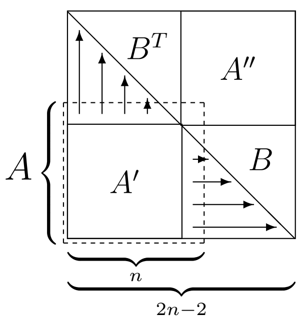
Proof. We give an inductive existence proof that can immediately be turned into a PTIME construction. The claim is obvious for $n = 2$ and we now show how to construct good colourings of the $2n$-square and the $(2n-2)$-square from a given good colouring of the n-square. This yields a valid inductive proof, because for even $m > 2$ at least one of $m/2$ and $(m+2)/2$ is even and smaller than m. Let $c\colon n \times n \to n$ be a good colouring.

Fig. 7.3



A good colouring $C\colon (2n)\times(2n) \to 2n$ is easily obtained by gluing four copies of trivially modified c-coloured squares together. The pattern is indicated in Figure 7.3. The box A represents an n-square coloured according to c, B an n-square coloured by $c'\colon n\times n \to \{n,\dots,2n-1\}$, $c'(m_1,m_2) = c(m_1,m_2)+n$. B^T finally is coloured $c' \circ T$.

Fig. 7.4



Consider now the $(2n-2)$-square. Assume without loss of generality that the c-coloured n-square A has top row (from left to right) coloured $n-1$, $n-2$, …, 0. By symmetry of c this implies that the rightmost column of A

is coloured 0, 1, ..., $n-1$ (from top to bottom). Let A' be the $(n-1)$-square coloured by $c' = c \upharpoonright (n-1)\times(n-1)$, or A with top row and rightmost column removed. Let A'' be the mirror image of A' across the second diagonal. As a second building block we use an $(n-1)$-square B coloured as follows. The second diagonal of B is coloured 1, ..., $n-1$ from top left to bottom right. The remaining places are coloured with colours $n, \dots, 2n-3$ such that each of these colours occurs exactly once in each row and in each column. This can be done with a cyclic permutation of colours following the second diagonal. B^T finally is the mirror image of B across the main diagonal. A good colouring of the $(2n-2)$-square is obtained by gluing the four $(n-1)$-squares squares A', A'', B and B^T as indicated in Figure 7.4. Note that the second diagonals in B and B^T exactly supply those colours from n in each row and in each column, that are missing in A' and A''. The arrows in the figure indicate how these second diagonals in B and B^T replace the rows and columns cut away from A. □

Lemma 7.25. *Let $D \subseteq n \times n$ be a subset of the n-square that is symmetric with respect to the main diagonal (invariant under T), disjoint from the main diagonal, and contains exactly two elements of each row and of each column. Then there is a colouring of this subset with two colours $c\colon D \to \{0,1\}$ such that each colour occurs exactly once in each row and in each column and such that c is antisymmetric with respect to the main diagonal: $T \circ c = 1 - c$. Such c is* PTIME *computable from D.*

In Figure 7.5 (b) the set D consisting of those points that are coloured 2 or 5 in (a) is split according to these requirements.

Proof. Consider the relation S of belonging to the same row *or* to the same column of $n \times n$ in restriction to D. Since each row and each column contains exactly 2 elements of D, D must be the disjoint union of even-length S-cycles. Since S and D are T-invariant, it follows that for each such cycle C either $T(C) \cap C = \emptyset$ or $T(C) = C$. Two different cycles cannot contain points of the same row or of the same column, since for instance there are only two points of D in any row and these necessarily belong to the same cycle. It follows that the requirements on c can be satisfied if for each cycle C there is a colouring $c'\colon C \cup T(C) \to \{0,1\}$, which is antisymmetric for T and contains at most one point coloured 0, respectively 1, in each row and each column. To obtain c take the union of these c'. Consider first a single cycle C. C can be coloured alternately with colours 0 and 1, starting with colour 0 say from the lexicographically least member in C and proceeding in the direction of the horizontal S-neighbour of this point. If $T(C) \neq C$, then c' on $C \cup T(C)$ can be taken as the union of this colouring of C with the antisymmetric image under T on $T(C)$. In case $T(C) = C$ we have to check that the colouring we have obtained is antisymmetric itself. Let C be enumerated in the order used in the colouring procedure as $c_0, c_1, \dots, c_{2n} = c_0$, so that c_0 and c_1 are in the same row, c_1 and c_2 in the same column, etc. Assume for contradiction that for example $Tc_{2i} = c_{2j}$ (c_{2i} and c_{2j} are both coloured 0), where $i < j$. Then c_{2i+1}, the element of D in the same row as c_{2i}, must be T-related with c_{2j-1}, which is the element of D that is in the same column with c_{2j}. This is because D is symmetric with respect to T. Proceeding in this manner we would find $Tc_k = c_k$ for $k = i + j$, which is impossible since D is disjoint from the diagonal. □

Fig. 7.5 (a)

3	5	4	2	1	0
4	2	5	3	0	1
5	4	1	0	3	2
2	3	0	1	5	4
1	0	3	4	2	5
0	1	2	5	4	3

(b)

	●		○		
	○	●			
○					●
●				○	
				●	○
		○	●		

7.2.4 Realizations of the Diagonal Boxes

With the help of these lemmas about colourings we finally construct realizations for diagonal restrictions $(\mathfrak{Q}; \nu)_{ii}$ that satisfy the conditions derived in Lemma 7.22.

Lemma 7.26. *Let $(\mathfrak{Q}; \nu)_{ii}$ satisfy the following conditions:*

(i) $\nu(\delta_i) = 1$.
(ii) for all $\alpha \in q_{ii}$: $\nu(\alpha) = \nu^T(\alpha)$.
(iii) if $n_i = \sum_{\alpha \in q_{ii}} \nu(\alpha)$ *is odd, then* $\nu(\alpha)$ *is even for all* $\alpha \neq \delta_i$ *with* $T\alpha = \alpha$.

Then there is a realization $\pi \colon n_i \times n_i \to q_{ii}$ *of* $(\mathfrak{Q}; \nu)_{ii}$ *over* $n_i = \sum_{\alpha \in q_{ii}} \nu(\alpha)$. *Moreover, such a realization can be constructed in time polynomial in* n_i.

Proof. Recall from Definitions 7.18 and 7.7 that a realization π of $(\mathfrak{Q}; \nu)_{ii}$ has to satisfy the following:

(a) $\pi(m_1, m_2) = \delta_i$ exactly for $m_1 = m_2$.
(b) $T \circ \pi = \pi \circ T$.
(c) For all m_1: $\big|\{m_2 \mid \pi(m_1, m_2) = \alpha\}\big| = \nu(\alpha)$.

(c) is a combination of (iii) and (iv) in Definition 7.7 applied to the present case with trivial E. Note that (c) together with (b) also implies that for all m_2: $\big|\{m_1 \mid \pi(m_1, m_2) = \alpha\}\big| = \big|\{m_1 \mid \pi(m_2, m_1) = T\alpha\}\big| = \nu(T\alpha)$, which by assumption (ii) of the lemma is the same as $\nu(\alpha)$.

The construction of π depends on whether n_i is even or odd. The odd case is the easier one.

Case A: n_i odd. Let $q_{ii} \setminus \{\delta_i\} = q_0 \dot\cup q_1$ where q_0 consists of those points that are fixed under T. Since T is a fixed-point free involutive permutation on q_1, $q_1 = q \dot\cup Tq$ for some $q \subseteq q_1$. For a definite construction it is important that q can be specified in a unique way, with the help of the ordering $\leqslant$ on q_{ii}. For instance we may take q as the lexicographically least subset $q \subseteq q_1$ for which $q_1 = q \dot\cup Tq$. This q can be determined in PTIME. Let now q_0 be enumerated as $\alpha_1, \ldots, \alpha_s$, and q as $\beta_1, \ldots, \beta_t$, both in $\leqslant$-order. So $\delta_i, \alpha_1, \ldots, \alpha_s, \beta_1, \ldots, \beta_t, T\beta_1, \ldots, T\beta_t$ is an enumeration of q_{ii} without repetitions. By the assumptions of the lemma, $\nu(\alpha_j) = 2k_j$ for suitable k_j, $\nu(\beta_j) = \nu(T\beta_j) =: l_j$. Let $d_1 := \sum k_j$, $d_2 := \sum l_j$, $d := d_1 + d_2$, where we take these as identifications with the disjoint ordered sums. Note that $n_i = 1 + 2d$. Put $D_0 := \big\{(u,v) \in n_i \times n_i \bigm| (u-v) \bmod n_i \in \{1, \ldots, d\}\big\}$. It follows that $n_i \times n_i = \big\{(u,u) \bigm| u \in n_i\big\} \dot\cup D_0 \dot\cup T(D_0)$. See the sketch in Figure 7.6 with $d = 3$, $n_i = 7$, where the values $(u-v) \bmod n_i$ in $\{1, \ldots, d\}$ are indicated. The desired realization π can be defined as follows:

$$\pi(u,v) := \begin{cases} \delta_i & \text{for } u = v \\ \alpha_j & \text{if } (u,v) \in D_0 \text{ and } (u-v) \bmod n_i \in k_j \\ \beta_j & \text{if } (u,v) \in D_0 \text{ and } (u-v) \bmod n_i \in l_j \\ T(\pi(v,u)) & \text{if } (u,v) \in T(D_0)\,. \end{cases}$$

Conditions (a) and (b) are obviously satisfied.

To check the multiplicity requirements (c), note that for each $u \in n_i$ and $s \in \{1, \ldots, d\}$, $\big|\{v | (u-v) \bmod n_i = s\}\big| = \big|\{v | (u-v) \bmod n_i = -s\}\big|$ and that the operation of T on (u,v) translates into $(u-v) \bmod n_i \mapsto -(u-v) \bmod n_i$. It follows that for all $u \in n_i$ indeed $\big|\{v | \pi(u,v) = \alpha_j\}\big| = 2k_j = \nu(\alpha_j)$ and $\big|\{v | \pi(u,v) = \beta_j\}\big| = l_j = \nu(\beta_j)$ as required.

Fig. 7.6

1	2	3				
2	3					1
3					1	2
				1	2	3
			1	2	3	
		1	2	3		
	1	2	3			

Case B: n_i even. Enumerate q_{ii} as $\delta_i, \alpha_1, \ldots, \alpha_s, \beta_1, \ldots, \beta_t, T\beta_1, \ldots, T\beta_t$ as above. In particular $T\alpha_j = \alpha_j$. Let $\nu(\alpha_j) = k_j$, $\nu(\beta_j) = \nu(T\beta_j) = l_j$. Thus $n_i = 1 + d_1 + 2d_2$, where $d_1 = \sum k_j$, $d_2 = \sum l_j$. We identify n_i with the disjoint ordered sum $1 + \sum k_j + 2d_2$.

Let $c\colon n_i \times n_i \to n_i$ be a colouring function as constructed in Lemma 7.24. The crucial properties are symmetry, $c \circ T = c$, exactly one occurrence of every colour in each row and in each column, and $c(u,u) = 0$ on the diagonal. With the above identification we consider c as a function to the disjoint sum $1 + \sum k_j + 2d_2$. Thus c can directly be used to define π partially as

$$\pi(u,v) := \begin{cases} \delta_i & \text{for } u = v \\ \alpha_j & \text{if } c(u,v) \in k_j \,. \end{cases}$$

Symmetry and multiplicities for the α_j are as required. It remains to define $\pi \restriction \big\{(u,v) \bigm| c(u,v) \in 2d_2\big\}$ with values in $\big\{\beta_1, \dots, \beta_t, T\beta_1, \dots, T\beta_t\big\}$. Note that this remaining subdomain is disjoint from the diagonal. We now further identify $2d_2$ with the product $\{0,1\} \times \sum l_j$ (in some canonical and definite way), so that on the remaining domain c takes values $(0,d)$ and $(1,d)$ for $d \in \dot{\bigcup} l_j$. Put $D_d := \big\{(u,v) \bigm| c(u,v) = (0,d) \vee c(u,v) = (1,d)\big\}$ for $d \in \dot{\bigcup} l_j$. D_d is T-symmetric, disjoint from the diagonal and contains exactly two elements of each row and each column. By Lemma 7.25 each D_d can be coloured by some $c_d \colon D_d \to \{0,1\}$ in such a way that each column and each row contains colour 0 and 1 exactly once, and such that $T \circ c_d = 1 - c_d$ corresponds to an inversion of the colouring. To complete the definition of π put

$$\pi(u,v) := \begin{cases} \beta_j & \text{if } (u,v) \in D_d, d \in l_j \text{ and } c_d(u,v) = 0 \\ T\beta_j & \text{if } (u,v) \in D_d, d \in l_j \text{ and } c_d(u,v) = 1 \,. \end{cases}$$

Compatibility with T follows, since T preserves D_d and inverts c_d. The multiplicities for the β_j are realized correctly because each row and each column contains exactly one element $(u,v) \in D_d$ such that $c_d(u,v) = 0$ (respectively 1) for each d. Therefore $\big|\{v | \pi(u,v) = \beta_j\}\big| = \big|\{v | \pi(u,v) = T\beta_j\}\big| = l_j$ as required. □

Putting those results of the preceding sections, that relate to the case of $C^2_{\infty\omega}$, together we have the following.

Proposition 7.27. *Let $(\mathfrak{Q};\nu)$ be a weighted game tableau. Let the E-classes of Q be $q_1, \dots, q_l$, let $q_{ij} := q_i \cap T(q_j)$ and put $n_i := \sum_{\alpha \in q_{ii}} \nu(\alpha)$. $(\mathfrak{Q};\nu)$ admits a realization if and only if the following conditions are satisfied:*

(i) $\sum_{\alpha \in q_{ij}} \nu(\alpha) = n_j$ independent of i.
(ii) $\nu(\delta) = 1$ for all $\delta \in \Delta$.
(iii) $\nu(\alpha) \,\big/\, \nu^T(\alpha) = n_j \,\big/\, n_i$ for all $\alpha \in q_{ij}$.
(iv) For all odd n_i, and all $\alpha \in q_{ii} \setminus \Delta$, if $T\alpha = \alpha$, then $\nu(\alpha)$ is even.

In this case a realization on $n = \sum n_i$ can be constructed in time polynomial in n, thus proving the $C^2_{\infty\omega}$-related part of Theorem 7.14.

We review the arguments that lead to this statement: (i) is the compatibility condition for fitting together realizations of the restrictions $(\mathfrak{Q};\nu)_{ij}$; necessity follows from Proposition 7.20 together with Lemma 7.22. (ii) is obviously necessary. (iii) is necessary for realizability of each $(\mathfrak{Q};\nu)_{ij}$, (iv) is necessary for $(\mathfrak{Q};\nu)_{ii}$ to admit a realization, both by Lemma 7.22. Sufficiency follows from realizability of the $(\mathfrak{Q};\nu)_{ij}$: (iii) suffices for $(\mathfrak{Q};\nu)_{ij}$, $i \neq j$, see Lemma 7.23; (ii) – (iv) suffice for $(\mathfrak{Q};\nu)_{ii}$ according to Lemma 7.26; (i) suffices to compose these individual realizations.

7.3 Realizations for I_{L^2}

7.3.1 Necessary and Sufficient Conditions

We prove the following analogue of Proposition 7.27 in the case of game tableaux without weights.

Proposition 7.28. *Let $\mathfrak{Q}$ be a game tableau. Let the E-classes of Q be $q_1, \ldots, q_l$. Put $q_{ij} := q_i \cap T(q_j)$. Then $\mathfrak{Q}$ admits a realization if and only if the following conditions are satisfied:*

(i) all q_{ij} are nonempty.
(ii) if $|q_{ii}| = 1$ then $|q_{ij}| = 1$ for all j.

In this case a realization — one of minimal size even — can be constructed in time polynomial in $|Q|$. This proves that part of Theorem 7.14 that relates to $L^2_{\infty\omega}$.

Proof (of necessity of (i) and (ii)). (i) is trivial: if $\mathfrak{Q}_{ij}$ is to have a realization, then q_{ij} must not be empty. For (ii) assume that $|q_{ii}| = 1$, i.e. that $q_{ii} = \{\delta_i\}$. It follows that $\mathfrak{Q}_{ii}$ can only admit the trivial realization $\pi\colon 1 \times 1 \to \{\delta_i\}$ on the one-element square, since no off-diagonal pair may be mapped to δ_i by any realization. In the terminology of Proposition 7.20 it follows that $n_i = 1$ and that all $\mathfrak{Q}_{ij}$ must have realizations on domains $1 \times n_j$. It follows directly from Definition 7.18 that any realization $\pi\colon s \times t \to q_{ij}$ satisfies $s, t \geqslant |q_{ij}|$, so that $s = 1$ implies $|q_{ij}| = 1$. □

The rest of this section is devoted to the proof of the sufficiency claim of Proposition 7.28. Again Proposition 7.20 is invoked to reduce the construction of a realization for $\mathfrak{Q}$ to the realization of the restrictions $\mathfrak{Q}_{ij}$. In fact we shall see that (ii) in the proposition reflects what remains of the numerical compatibility conditions in Proposition 7.20 in the case of $L^2_{\infty\omega}$: $L^2_{\infty\omega}$ can only count "$0, 1, \mathrm{many}$". For the restrictions, we first treat the off-diagonal ones, then the diagonal ones. Fix a game tableau $\mathfrak{Q}$.

The off-diagonal restrictions.

Lemma 7.29. *For $i \neq j$. If $q_{ij} \neq \emptyset$ then there are realizations $\pi\colon s \times t \to q_{ij}$ of $\mathfrak{Q}_{ij}$ exactly for all $s, t \geqslant |q_{ij}|$.*

Sketch of Proof. The condition $s, t \geqslant |q_{ij}|$ is necessary since π has to attain each $\alpha \in q_{ij}$ at least once in every row and in every column.

Let $s_0 = |q_{ij}|$ and first construct a realization on $s_0 \times s_0$. Let q_{ij} be enumerated as $\alpha_0, \dots \alpha_{s_0-1}$ in increasing order with respect to $\leqslant$. Put

$$\pi_0(u, v) := \alpha_k \qquad \text{for } k = (u + v) \bmod s_0.$$

Obviously each α_j occurs once in each row and in each column as required. To obtain realizations for $s \times t$, $s, t \geqslant s_0$ put

$$\pi(u, v) := \begin{cases} \pi_0(u, v) & \text{for } u, v < s_0 \\ \alpha_k & \text{for } k = v \bmod s_0, u \geqslant s_0, v \leqslant s_0 \\ \alpha_k & \text{for } k = u \bmod s_0, v \geqslant s_0, u \leqslant s_0 \\ \alpha_0 & \text{for } v, u \geqslant s_0\,. \end{cases}$$

π extends π_0 through repetition of (extensions of) the first row and first column. □

The diagonal restrictions. For the diagonal restrictions $\mathfrak{Q}_{ii}$ the size of a minimal realization may depend on the existence of fixed points under T other than δ_i. We show that the minimal size is equal to $|q_{ii}|$ if there are no such fixed points, and equal to the least even number greater than or equal to $|q_{ii}|$ otherwise. Put

$$d_i := \begin{cases} |q_{ii}| & \text{if } T\alpha \neq \alpha \text{ for all } \alpha \in q_{ii} \setminus \{\delta_i\} \\ 2\lceil \frac{1}{2}|q_{ii}|\rceil & \text{otherwise.} \end{cases}$$

Lemma 7.30. *If $q_{ii} \neq \{\delta_i\}$, then there are realizations $\pi \colon s \times s \to q_{ii}$ of $\mathfrak{Q}_{ii}$ exactly for all $s \geqslant d_i$.*

Proof. First we argue that $s \geqslant d_i$ is necessary. Trivially $s \geqslant |q_{ii}|$ is necessary, since each $\alpha \in q_{ii}$ has to occur at least once in every row and column. $d_i > |q_{ii}|$ if and only if $|q_{ii}|$ is odd and there is some $\alpha \neq \delta_i$ such that $T\alpha = \alpha$. In this case $d_i = |q_{ii}|+1$. But then this α has to occur an even number of times under π, whence either s has to be even (and therefore $s \geqslant |q_{ii}| + 1 = d_i$ in this case), or, if s is odd, α occurs at least twice in at least one row. This row still has to realize all other elements of q_{ii}, and it follows that $s \geqslant |q_{ii}| + 1 = d_i$ in that case as well.

Now for the existence of realizations as claimed. First consider realizations over $d_i \times d_i$.

If $|q_{ii}|$ is even or if $T\alpha \neq \alpha$ for all $\alpha \in q_{ii} \setminus \{\delta_i\}$, consider $(\mathfrak{Q}_{ii}; \nu)$, with ν identically put to 1, as a weighted game tableau. $(\mathfrak{Q}_{ii}; \nu)$ satisfies the requirements of Lemma 7.26 so that we obtain a realization on $d_i \times d_i$ since $d_i = |q_{ii}| = \sum_{\alpha \in q_{ii}} \nu(\alpha)$.

Otherwise $|q_{ii}|$ is odd, there is some $\alpha \neq \delta_i$ with $T\alpha = \alpha$, and $d_i = |q_{ii}|+1$. Let α_0 be the least $\alpha \in q_{ii} \setminus \{\delta_i\}$ that is fixed by T. Put $\nu(\alpha_0) = 2$ and $\nu(\alpha) = 1$

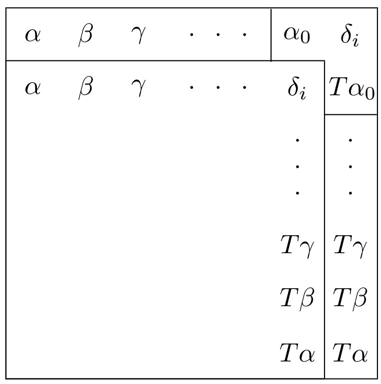
for all $\alpha \neq \alpha_0$. Again, $\sum_{\alpha \in q_{ii}} \nu(\alpha) = d_i$ is even and Lemma 7.26 applies to give a realization on $d_i \times d_i$.

From these minimal realizations on $d_i \times d_i$ one may again obtain realizations over $s \geqslant d_i$ simply by extensions that essentially repeat one row and one column. Let $\pi_0 : d_i \times d_i \to q_{ii}$ be the minimal realization, let α_0 be the minimal element of $q_{ii} \setminus \{\delta_i\}$. Then the following is a realization over $s \geqslant d_i$:

$$\pi(u,v) := \begin{cases} \pi_0(u,v) & \text{for } u,v < d_i \\ \pi_0(d_i - 1, v) & \text{for } u \geqslant d_i, v < d_i - 1 \\ \pi_0(u, d_i - 1) & \text{for } v \geqslant d_i, u < d_i - 1 \\ \delta_i & \text{for } u = v \geqslant d_i \\ \alpha_0 & \text{for } u > v \geqslant d_i - 1 \\ T\alpha_0 & \text{for } v > u \geqslant d_i - 1\,. \end{cases}$$

Figure 7.7 illustrates this extension of the domain by one row and one column to $s = d_i + 1$. □

Fig. 7.7



It remains to determine the size of a minimal realization for the entire game tableau $\mathfrak{Q}$. Put

$$n_{ij} := \begin{cases} |q_{ij}| & \text{for } i \neq j \\ |q_{ii}| & \text{if } i = j \text{ and } T\alpha \neq \alpha \text{ for all } \alpha \in q_{ii} \setminus \{\delta_i\} \\ 2\lceil \frac{1}{2}|q_{ii}| \rceil & \text{if } i = j \text{ and } T\alpha = \alpha \text{ for some } \alpha \in q_{ii} \setminus \{\delta_i\} \end{cases} \tag{7.3}$$

so that according to the last lemmas n_{ij} is the minimal number such that $\mathfrak{Q}_{ij}$ admits a realization over $n_{ij} \times n_{ij}$. Note that for any game tableaux $n_{ij} = n_{ji}$ because T acts as a fixed-point free bijection between q_{ij} and q_{ji}. Assume that the q_{ij} are all non-empty. Then

– each $\mathfrak{Q}_{ij}$ for $i \neq j$ admits a realization over $s \times t$ for all $s, t \geqslant n_{ij}$,
– each non-trivial $\mathfrak{Q}_{ii}$ admits a realization on $s \times s$ for all $s \geqslant n_{ii}$, and

– each trivial $\mathfrak{Q}_{ii}$, $|q_{ii}| = \{\delta_i\}$, admits only the trivial singleton realization.

Optimal values for realizations that fit together in the sense of the condition in Proposition 7.20 are therefore given by:

$$\begin{array}{rcl} n_i & := & \max\{n_{ij} \mid 1 \leqslant j \leqslant l\} \\ n & := & \sum_i n_i \,. \end{array} \tag{7.4}$$

Notice that (ii) of Proposition 7.28 implies that $n_i = 1$ whenever $|q_{ii}| = 1$. If $\mathfrak{Q}$ therefore satisfies all conditions of Proposition 7.28 we do get a realization of size n.

Proposition 7.31. *Let $\mathfrak{Q}$ be a game tableau that satisfies the conditions of Proposition 7.28. Then either $|q_{ii}| = 1$ for all i and the only realization of $\mathfrak{Q}$ is over $n = \sum n_i = |\Delta|$; or there is at least one $q_{ii} \neq \{\delta_i\}$ and in this case $\mathfrak{Q}$ has realizations exactly over all $s \geqslant n$. (The n_i and n are as determined by equations 7.3 and 7.4.)*

The explicit constructions of realizations for the individual $\mathfrak{Q}_{ij}$ presented above and the general procedure for the composition of these according to Proposition 7.20 yield a PTIME algorithm as required for Proposition 7.28. This finishes the proof of Theorem 7.14.

It might be interesting to find a simple bound on the size n of a minimal realization of $\mathfrak{Q}$ also in terms of $|Q| = \sum_{1\leqslant i,j\leqslant l} |q_{ij}|$. Recall that $|Q|$ is the size of the L^2-invariant of the desired structure, or — in more model theoretic terms — the number of distinct L^2-types of pairs that the desired structure has to realize. We claim that actually $|Q| + 1$ is such a bound. In particular this is a *linear bound*, whereas for $k \geqslant 3$ we know that there cannot even be a sub-exponential bound on the size of minimal realizations of L^k-invariants by Example 3.23.

The following proposition gives a somewhat tighter bound in terms of both the number of 2-types and the number of 1-types that are to be realized.

Proposition 7.32. *For all $\mathfrak{A} \in \mathrm{fin}[\tau]$ there is some $\mathfrak{B} \in \mathrm{fin}[\tau]$ such that*

$$\mathfrak{B} \equiv^{L^2} \mathfrak{A} \quad \textit{and} \quad |B| \leqslant |\mathrm{Tp}^{L^2}(\mathfrak{A};2)| + 1 - \big(|\mathrm{Tp}^{L^2}(\mathfrak{A};1)| - 1\big)^2.$$

In particular $|B| \leqslant |\mathrm{Tp}^{L^2}(\mathfrak{A};2)| + 1 = |I_{L^2}(\mathfrak{A})| + 1$.

Proof. Let $\mathfrak{Q} = \big(Q, \leqslant, E, T, \Delta\big)$ be the game tableau associated with $I_{L^2}(\mathfrak{A})$. The desired $\mathfrak{B}$ is obtained from a realization of $\mathfrak{Q}$ of minimal size. Let $q_1, \ldots, q_l$, the q_{ij}, n_{ij}, n_i and n be as described in Proposition 7.28 and equations 7.3 and 7.4.

As $|Q| = |\mathrm{Tp}^{L^2}(\mathfrak{A};2)|$ and $|\mathrm{Tp}^{L^2}(\mathfrak{A};1)| = |Q/E| = l$ it suffices to show

$$n \leqslant |Q| + 1 - (l-1)^2,$$

since n is the minimal size of a realization of $\mathfrak{Q}$ as determined above. We first observe that $n_i = \max\{n_{ij} | 1 \leqslant j \leqslant l\} \leqslant \sum_{1\leqslant j\leqslant l} n_{ij} - (l-1)$ because all n_{ij} are positive. Recall that $n_{ij} \leqslant |q_{ij}| + 1$ and $n_{ij} = |q_{ij}|$ at least for all $i \neq j$. Therefore

$$\begin{aligned} n = \textstyle\sum_i n_i &\leqslant \textstyle\sum_{i,j} n_{ij} - l(l-1) \\ &\leqslant \textstyle\sum_{i\neq j} |q_{ij}| + \sum_i \big(|q_{ii}| + 1\big) - l(l-1) \\ &\leqslant \textstyle\sum_{i,j} |q_{ij}| + l - l(l-1) = |Q| + 1 - (l-1)^2, \end{aligned}$$

as desired. □

The given bound is essentially optimal among bounds that are independent of the vocabulary. This is demonstrated in the following example.

Example 7.33. Let $k \geqslant 2$ and let τ_k consist of k binary relation symbols R_0, ..., R_{k-1}. Let $\mathfrak{A}_k$ consist of $2k-2$ points arranged in a cycle and with R_i interpreted by the set of pairs at distance i. The following sentence axiomatizes the complete L^2-theory of $\mathfrak{A}_k$.

$$\begin{aligned} \varphi_k \quad = \quad & \forall x \forall y \Big(\bigvee_i R_i xy \wedge \bigwedge_{i\neq j} \neg (R_i xy \wedge R_j xy) \Big) \\ & \wedge \forall x \forall y \Big(R_0 xy \leftrightarrow x = y \wedge \bigwedge_i (R_i xy \leftrightarrow R_i yx) \Big) \\ & \wedge \forall x \bigwedge_i \exists y R_i xy \,. \end{aligned}$$

Models of φ_k exactly correspond to realizations of the game tableau $\big(\mathfrak{Q}_k, \leqslant, E, T, \Delta\big)$ where $Q_k = k$, $E = k \times k$, $T = \mathrm{id}_k$, $\Delta = \{0\}$. In fact, if $\mathfrak{B}$ is a τ_k-structure over universe n, then $\mathfrak{B} \models \varphi_k$ if and only if

$$\begin{aligned} \pi \colon n \times n & \longrightarrow Q_k \\ (b, b') & \longmapsto i \ \text{ if } (b, b') \in R_i^{\mathfrak{B}} \end{aligned}$$

is a realization of $\mathfrak{Q}_k$. By Lemma 7.30 φ_k has models exactly in sizes greater than or equal to $n = 2\lceil \frac{1}{2} k \rceil$. Clearly $|\mathrm{Tp}^{L^2}(\mathfrak{A}_k; 2)| = k$ and $|\mathrm{Tp}^{L^2}(\mathfrak{A}_k; 1)| = l = 1$, so that $|Q| + 1 - (l-1)^2 = k + 1 = n$ for all odd k.

For situations with $l > 1$ one obtains similar examples by considering structures

$$\mathfrak{A} = \big(\mathfrak{A}_{k_1} \dot\cup \ldots \dot\cup \mathfrak{A}_{k_l}, P_1, \ldots, P_l\big)$$

with extra unary P_i to encode the partition into the $\mathfrak{A}_{k_i}$. Here $|\mathrm{Tp}^{L^2}(\mathfrak{A}; 2)| = \sum k_i + l(l-1)$, $|\mathrm{Tp}^{L^2}(\mathfrak{A}; 1)| = l$ so that $|Q| + 1 - (l-1)^2 = \sum k_i + l$. This bound is shown to be exact as above if all k_i are odd.

Another, more simple corollary to our findings about realizations concerns the spectrum of complete $L^2_{\infty\omega}$-theories.

Corollary 7.34. *Any complete $L^2_{\infty\omega}$-theory (in a finite relational vocabulary) that has any finite models, either has exactly one finite model up to isomorphism, or has models exactly in all cardinalities above some finite threshold value n.*

Sketch of Proof. Completeness of the theory together with existence of at least one finite model implies that all models (in fact finite and infinite ones) have the same value for their L^2-invariant. We know from Theorem 7.12 that each finite model of the theory is obtained from a realization of the underlying game tableau and vice versa. For realizations the corresponding spectrum property is expressed in Proposition 7.31 above. In the case of at least one non-trivial q_{ii} the above constructions for the extensions of realizations can easily be extended to yield models in arbitrary infinite cardinalities as well. □

7.3.2 On the Special Nature of Two Variables

Combinatorially, and with respect to the solution of the inversion problem presented here, the two-variable invariants and their induced tableaux are special. The trivialization of the accessibility relations E_1 and E_2 and modularity of the solutions as discussed in Section 7.1.1 are peculiar to the two-dimensional case in this sharp form. Basically the easy decomposition can be attributed to the fact that the two-variable type of a pair is fully determined by the individual two-variable types of its components together with the atomic type of the pair — a property that technically is reflected in Lemma 7.16. Combinatorially more sophisticated techniques may be required to approach the three-variable case. We have no well-founded conjecture at this stage whether indeed the k-variable case can be settled positively for any $k \geqslant 3$. In view of the general theorems above these canonization and inversion problems with and without counting remain challenging open problems.

The most important aspect with respect to classical logical concerns, in which two variables are very special, is decidability.

Theorem 7.35 (Mortimer). *$L^2_{\omega\omega}$ has the finite model property, i.e. any satisfiable sentence of $L^2_{\omega\omega}$ in a relational vocabulary has a finite model. Consequently, the* satisfiability problem *for $L^2_{\omega\omega}$ is decidable.*

Decidability of the satisfiability problem for $L^2_{\omega\omega}$ was earlier announced by Scott [Sco62], but the argument Scott gave was based on the erroneous assumption that the Gödel case with equality is decidable. So [Sco62] proves the claim only for $L^2_{\omega\omega}$ without equality. A version of Mortimer's proof [Mor75] can be found in [EF95]. There is also a new proof (with better complexity bounds) by Grädel, Kolaitis and Vardi [GKV96, BGG96].

Let us consider in a brief sketch what Mortimer's result implies about the inversion problem for I_{L^2}. To this end one may transform the information in a given candidate L^2-invariant

$$\mathfrak{I} = \big(Q, \leqslant, E, T; \Theta\big)$$

into an $L^2_{\omega\omega}$-sentence as follows (compare also Example 7.33). Introduce new predicates R_α for $\alpha \in Q$. Then the following sentence represents the full information in $\mathfrak{I}$. It may in fact be considered as an axiomatization of those realizations of the underlying game tableau that also respect atomic types as prescribed by Θ, whence it exactly axiomatizes the canonical R_α-expansions of all τ-structures $\mathfrak{A}$ with $I_{L^2}(\mathfrak{A}) = \mathfrak{I}$.

$$\begin{aligned} & \forall x \forall y \bigvee_\alpha R_\alpha xy \ \wedge \ \forall x \forall y \bigwedge_{\alpha \neq \alpha'} \neg\big(R_\alpha xy \wedge R_{\alpha'} xy\big) \\ \wedge \ & \bigwedge_\alpha \forall x \forall y \big(R_\alpha xy \leftrightarrow R_{T\alpha} yx\big) \ \wedge \bigwedge_{\alpha \in \Delta, (\alpha, \alpha') \in E} \forall x (R_\alpha xx \leftrightarrow \exists y R_{\alpha'} xy) \\ \wedge \ & \bigwedge_\alpha \forall x \forall y \big(R_\alpha xy \rightarrow \Theta\big(\alpha\big)(x,y)\big). \end{aligned}$$

This sentence may be regarded as a variant of a Scott sentence, with the crucial difference that it is over an extended vocabulary but requires only quantifier rank 2 and is of quadratic length in the size of the given invariant.

From the proof in [Mor75] one can infer that owing to the special format of this $L^2_{\omega\omega}$-sentence the size of its minimal models is bounded by a polynomial in the size of the given invariant.

The inversion problem thus reduces to the satisfiability problem for these associated compressed Scott sentences. This reduction induces an exponential time decision procedure for image$\big(I_{L^2}\big)$ and a corresponding solution through exhaustive search to the inversion problem in exponential time.

Quite recently it has been shown in [GOR96b] that also the satisfiability problem for $C^2_{\omega\omega}$ is decidable (although $C^2_{\omega\omega}$ does not have the finite model property, compare Example 1.19).

Theorem 7.36 (Grädel, Otto, Rosen). *The satisfiability problem for $C^2_{\omega\omega}$ is decidable.*

It is remarkable on the other hand that even decidability of image$\big(I_{L^k}\big)$ for $k > 2$ is an open problem. The corresponding problem for the I_{C^k} is trivial, since the size of candidate structures is easily determined from the proposed invariant. It should be stressed that also for the I_{L^k} there is no obvious connection between the decidability of the set of all invariants of actual structures and PTIME invertibility of I_{L^k} in the sense of Definition 6.9. In fact the size of prospective realizations may always in this context be thought of as a given parameter, in which case decidability becomes trivial through exhaustive search (compare the remarks following Definition 6.9).

Bibliography

[AV89] S. Abiteboul and V. Vianu. Fixpoint extensions of first-order logic and Datalog-like languages. In *Proc. 4th IEEE Symp. on Logic in Computer Science*, pages 71–79, 1989.

[AV91] S. Abiteboul and V. Vianu. Generic computation and its complexity. In *Proc. 23rd ACM Symp. on Theory of Computing*, pages 209–219, 1991.

[BES80] L. Babai, P. Erdös, and M. Selkow. Random graph isomorphism. *SIAM Journal of Computing*, 9:628–635, 1980.

[BK80] L. Babai and L. Kučera. Canonical labellings of graphs in linear average time. In *Proc. 21st IEEE Symp. on Foundations of Computer Science*, pages 39–46, 1980.

[Bar77] J. Barwise. On Moschovakis closure ordinals. *Journal of Symbolic Logic*, 42:292–296, 1977.

[BG84] A. Blass and Y. Gurevich. Equivalence relations, invariants, and normal forms, II. In E. Börger, G. Hasenjäger and D. Rödding, editors, *Logic and Machines: Decision Problems and Complexity, Lecture Notes in Computer Science 171*, pages 24–42. Springer-Verlag, 1984.

[BGG96] E. Börger, E. Grädel, and Y. Gurevich. *The Classical Decision Problem*, Perspectives in Mathematical Logic, Springer-Verlag, 1996.

[Büc60] J. R. Büchi. Weak second-order arithmetic and finite automata. *Z. Math. Logik Grundlagen Math.*, 6:66–92, 1960.

[CFI89] J. Cai, M. Fürer, and N. Immerman. An optimal lower bound on the number of variables for graph identification. In *Proc. 30th IEEE Symp. on Foundations of Computer Science*, pages 612–617, 1989.

[CFI92] J. Cai, M. Fürer, and N. Immerman. An optimal lower bound on the number of variables for graph identification. *Combinatorica*, 12:389–410, 1992.

[CH80] A. Chandra and D. Harel. Computable queries for relational databases. *Journal of Computer and System Sciences*, 21:156–178, 1980.

[CH82] A. Chandra and D. Harel. Structure and complexity of relational queries. *Journal of Computer and System Sciences*, 25:99–128, 1982.

[Dah87] E. Dahlhaus. Skolem normal forms concerning the least fixpoint. In E. Börger, editor, *Computation Theory and Logic, Lecture Notes in Computer Science 270*, pages 101–106. Springer-Verlag, 1987.

[Daw93] A. Dawar. *Feasible computation through model theory.* PhD thesis, University of Pennsylvania, 1993.

[Daw95a] A. Dawar. Generalized quantifiers and logical reducibilities. *Journal of Logic and Computation*, 5:213–226, 1995.

[Daw95b] A. Dawar. A restricted second-order logic for finite structures. In D. Leivant, editor, *Logic and Computational Complexity*, pages 393–413. Springer-Verlag, 1995.

[DH94] A. Dawar and L. Hella. The expressive power of finitely many generalized quantifiers. In *Proc. 9th IEEE Symp. on Logic in Computer Science*, pages 20–29, 1994.

[DLW95] A. Dawar, S. Lindell, and S. Weinstein. Infinitary logic and inductive definability over finite structures. *Information and Computation*, 119:160–175, 1995.

[Ebb85] H.-D. Ebbinghaus. Extended logics: the general framework. In J. Barwise and S. Feferman, editors, *Model–Theoretic Logics*, pages 25–76. Springer-Verlag, 1985.

[EF95] H.-D. Ebbinghaus and J. Flum. *Finite Model Theory.* Perspectives in Mathematical Logic, Springer-Verlag, 1995.

[EFT94] H.-D. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic.* Springer-Verlag, 2nd edition, 1994.

[Elg61] C. Elgot. Decision problems of finite-automata design and related arithmetics. *Trans. Amer. Math. Soc.*, 98:21–51, 1961.

[Fag74] 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.

[Fag90] R. Fagin. Finite-model theory — a personal perspective. In S. Abiteboul and P. Kanellakis, editors, *Proc. 1990 International Conference on Database Theory, Lecture Notes in Computer Science 470*, pages 3–24. Springer-Verlag, 1990. Also in *Theoretical Computer Science*, 116:3–31, 1993.

[GG95] E. Grädel and Y. Gurevich. Tailoring recursion for complexity. *Journal of Symbolic Logic*, 60:952–969, 1995.

[GG95] E. Grädel and Y. Gurevich. Metafinite model theory. In D. Leivant, editor, *Logic and Computational Complexity*, pages 313–366. Springer-Verlag, 1995.

[GKV96] E. Grädel, Ph. G. Kolaitis, and M. Y. Vardi. On the decison problem for two-variable first-order logic. Submitted, 1996.

[GO93] E. Grädel and M. Otto. Inductive definability with counting on finite structures. In E. Börger et al., editors, *Computer Science Logic, CSL '92, Selected Papers, Lecture Notes in Computer Science 702*, pages 231–247. Springer-Verlag, 1993.

[GOR96a] E. Grädel, M. Otto, and E. Rosen. Undecidability results on two-variable logics. Preprint, 1996.

[GOR96b] E. Grädel, M. Otto, and E. Rosen. Two-variable logic with counting is decidable. Preprint, 1996.

[Gro95] M. Grohe. Complete problems for fixed-point logics. *Journal of Symbolic Logic*, 60:517–527, 1995.

[Gur84] Y. Gurevich. Toward logic tailored for computational complexity. In M. M. Richter et al., editors, *Computation and Proof Theory, Lecture Notes in Mathematics 1104*, pages 175–216. Springer-Verlag, 1984.

[Gur88] 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.

[GS86] Y. Gurevich and S. Shelah. Fixed-point extensions of first-order logic. *Annals of Pure and Applied Logic*, 32:265–280, 1986.

[GS96] Y. Gurevich and S. Shelah. On finite rigid structures. *Journal of Symbolic Logic*, 61:549–562, 1996.

[Här65] H. Härtig. Über einen Quantifikator mit zwei Wirkungsbereichen. In L. Kalmar, editor, *Colloque sur les Fondements des Mathematiques, les Machines Mathematiques, et leur Applications*, pages 31–36, 1965.

[Imm81] N. Immerman. Number of quantifiers is better than number of tape cells. *Journal of Computer and System Sciences*, 22:384–406, 1981.

[Imm82] N. Immerman. Upper and lower bounds for first-order expressibility. *Journal of Computer and System Sciences*, 25:76–98, 1982.

[Imm86] N. Immerman. Relational queries computable in polynomial time. *Information and Control*, 68:86–104, 1986.

[Imm87a] N. Immerman. Expressibility as a complexity measure: results and directions. In *Proc. 2nd Structure in Complexity Conference*, pages 194–202, 1987.

[Imm87b] N. Immerman. Languages that capture complexity classes. *SIAM Journal of Computing*, 16:760–778, 1987.

[Imm89] N. Immerman. Descriptive and computational complexity. In J. Hartmanis, editor, *Computational Complexity Theory, Proc. Symp. Applied Math., Vol. 38*, pages 75–91. American Mathematical Society, 1989.

[IL90] N. Immerman and E. S. Lander. Describing graphs: a first-order approach to graph canonization. In A. Selman, editor, *Complexity Theory Retrospective*, pages 59–81. Springer-Verlag, 1990.

[KVä95] Ph. G. Kolaitis and J. Väänänen. Generalized quantifiers and pebble games on finite structures. *Annals of Pure and Applied Logic*, 74:23–75, 1995.

[KVa92a] Ph. G. Kolaitis and M. Y. Vardi. Infinitary logics and 0–1 laws. *Information and Computation*, 98:258–294, 1992.

[KVa92b] Ph. G. Kolaitis and M. Y. Vardi. Fixpoint logic vs. infinitary logic in finite-model theory. In *Proc. 7th IEEE Symp. on Logic and Computer Science*, pages 46–57, 1992.

[Lei90] D. Leivant. Inductive definitions over finite structures. *Information and Computation*, 89:95–108, 1990.

[Mor75] M. Mortimer. On languages with two variables. In *Z. Math. Logik und Grundlagen Math.*, 21:135–140, 1975.

[Ott94] M. Otto. Generalized quantifiers for simple properties. In *Proc. 9th IEEE Symp. on Logic and Computer Science*, pages 30–39, 1994.

[Ott95a] M. Otto. Ptime canonization for two variables with counting. In *Proc. 10th IEEE Symp. on Logic and Computer Science*, pages 342–352, 1995.

[Ott95b] M. Otto. Canonization for two variables and puzzles on the square. To appear in *Annals of Pure and Applied Logic*.

[Ott96a] M. Otto. The expressive power of fixed-point logic with counting. *Journal of Symbolic Logic*, 61:147–176, 1996.

[Ott96b] M. Otto. Bounded-variable logics: two, three, and more. Preprint, 1996.

[Pap94] C. H. Papadimitiou. *Computational Complexity*. Addison-Wesley Publishing Company, 1995.

[Poi82] B. Poizat. Deux ou trois choses que je sais de L_n. *Journal of Symbolic Logic*, 47:641–658, 1982.

[Res62] N. Rescher. Plurality quantification. *Journal of Symbolic Logic*, 27:373–374, 1962.

[Sco62] D. Scott. A decision method for validity of sentences in two variables. *Journal of Symbolic Logic*, 27:377, 1962.

[Tra61] B. Trakhtenbrot. Finite automata and the logic of monadic predicates. (in Russian) *Dokl. Akad. Nauk SSSR*, 140:326–329, 1961.

[Var82] M. Y. Vardi. The complexity of relational query languages. In *Proc. 14th ACM Symp. on Theory of Computing*, pages 137–146, 1982.

[Weg87] I. Wegener. *The Complexity of Boolean Functions*. John Wiley and Teubner, 1987.

Index