Tractable Recursion over Geometric Data

(Extended Abstract)

Stéphane Grumbach\textsuperscript{1*} and Gabriel Kuper\textsuperscript{2**}

\textsuperscript{1} I.N.R.I.A., Rocquencourt BP 105, 78153 Le Chesnay, France, stephane.grumbach@inria.fr
\textsuperscript{2} U. Libre de Bruxelles Informatique, C.P. 165, 50 Avenue F.D. Roosevelt, 1050 Brussels, Belgium, kuper@cs.ulb.ac.be

Abstract. We study the issue of adding a recursion operator to constraint query languages for linear spatial databases. We introduce a language with a bounded inflationary fixpoint operator which is closed and captures the set of polynomial time computable queries over linear constraint databases. This is the first logical characterization of the class of PTIME queries in this context. To prove the result, we develop original techniques to perform arithmetical and geometric operations with constraints.

1 Introduction

The area of constraint databases is an important area of current research. Since the initial work by Kanellakis, Kuper and Revesz [KKR90], which described the basic methodology for combining constraint solvers and traditional database query languages in a single framework, many related issues concerning expressive power, query languages, potential applications, and other aspects, have been studied.

One area that has received particular attention is that of linear constraints. These are particularly promising for many applications, e.g., for geographical databases [VGV95]. The complexity of the linear relational language (i.e., first-order logic) has been shown to be in NC\textsuperscript{1} [GS97]. A lower complexity, AC\textsuperscript{0}, can be shown for the case of finite, as opposed to finitely representable, databases. There has also been a significant amount of work that studies the expressive power of first-order logic with linear constraints [GST94, PVV95, GS97]. As with the standard relational first-order language, the language of linear constraints with first-order logic turns out to be rather restricted. For example, transitive closure and topological connectivity, the latter a very important query for spatial databases, cannot be expressed in the language.

\textsuperscript{*} Work supported in part by TMR Project Chorochronos.
\textsuperscript{**} Work supported by Inria, FNRS and ULB.
As with the relational calculus, it is therefore natural to ask whether the language of first-order logic with linear constraints can be extended to a more powerful language. Such a language should be powerful enough to express connectivity queries, but not too powerful — for example, we would like such a language to have a relatively low complexity. One possibility would be to simply add recursion to the language. Unfortunately, as was pointed out in [KKR90], the resulting language is not closed, i.e., the result of a query is not always finitely expressible using linear constraints. Besides this, it is well-known that in the presence of arithmetic, a fixpoint operator leads in fact to Turing computability. One could solve this problem by allowing recursion only over an order relation, not over the arithmetic predicates, but such a language does not seem to be powerful enough to express connectivity. (A formal proof of this is still an open problem.)

Despite this, it is reasonable to look for a restricted form of recursion that is sufficiently expressive, but has low complexity. If we look at the examples of “bad” recursion, we see that the problem is that they have the ability to create “new” objects (speaking informally) without limit. If we had some formal way to restrict queries to those that do not generate new objects or only in some limited way, we might be able to obtain a reasonable query language with recursion.

This paper describes an attempt to define such a language. We propose a notion of bounded recursion: Such a recursion performs, in parallel, two separate recursions, only one of which may involve arithmetic, and terminates as soon as either recursion terminates. Furthermore, the recursion is designed in such a way that only a constant number of new points can be created at each recursive step (non-recursive first-order formulas can still create finite representations of infinite objects, as this does not create any problems). While the language itself is not particularly elegant, it does have nice formal properties. First of all, the language is closed: the result of such a query on a linear constraint database is itself finitely representable by linear constraints. Secondly, the language is quite expressive, for example, natural topological properties such as connectivity can be expressed in this language.

Finally, there is a precise characterization of the expressive power of our language. In the tradition of studying complexity classes over ordered finite structures [Imm86, Var82], we show that our language with bounded recursion over linear constraints expresses precisely the class of linear constraint queries that are computable in polynomial time.

The paper is organized as follows. In the next section, we review two standard formalisms to represent linear geometric data, the linear constraint databases, and the point-based model of computational geometry. In section 3, we introduce the bounded inflationary fixpoint, and give some examples of queries. Finally, in Section 4, we prove the main result of the paper, namely the PTIME characterization.