Theoretical Computer Science 554 (2014) 217–228

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

On the treewidth of dynamic graphs ☆

Bernard Mans, Luke Mathieson *

Department of Computing, Macquarie University, Sydney, NSW 2109, Australia

ARTICLE INFO

Available online 8 January 2014

Keywords:
Dynamic graphs
Metatheorems
Treewidth
Local treewidth
Bounded expansion

ABSTRACT

Dynamic graph theory is a novel, growing area that deals with graphs that change over time and is of great utility in modeling modern wireless, mobile and dynamic environments. As a graph evolves, possibly arbitrarily, it is challenging to identify the graph properties that can be preserved over time and understand their respective computability.
In this paper we are concerned with the *treewidth* of dynamic graphs. We focus on *metatheorems*, which allow the generation of a series of results based on general properties of classes of structures. In graph theory two major metatheorems on treewidth provide complexity classifications by employing structural graph measures and finite model theory. Courcelle's Theorem gives a general tractability result for problems expressible in monadic second order logic on graphs of bounded treewidth, and Frick and Grohe demonstrate a similar result for first order logic and graphs of bounded local treewidth.
We extend these theorems by showing that dynamic graphs of bounded (local) treewidth where the length of time over which the graph evolves and is observed is finite and bounded can be modeled in such a way that the (local) treewidth of the underlying graph is maintained. We show the application of these results to problems in dynamic graph theory and dynamic extensions to static problems. In addition we demonstrate that certain widely used dynamic graph classes have bounded local treewidth.

1. Introduction

Graph theory has proven to be an extremely useful tool for modeling computational systems and with the advent and growing preponderance of mobile devices and dynamic systems it is natural that graph theory is extended and adapted to capture the evolving aspects of these environments. Dynamic graphs have been formalized in a number of ways: e.g., time-varying graphs [7,8,20], carrier-based networks [4], evolving graphs [6,19,5], delay-tolerant networks [26], dynamic networks [29,28], scheduled networks [1], temporal networks [27], opportunistic networks [9,25], Markovian [10]. When considering the dynamic aspects of a dynamic graph, even classically simple properties such as shortest paths become more complex to compute and may even become definitionally ambiguous [5] (Fig. 1).

In this paper, we are not particularly interested in any particular dynamic model. Initially we will loosely use the term *dynamic graph* and for our purpose we will define the term formally using the simplest possible definition as it is a generalized and reasonably assumption free model for a dynamic graph. For this paper, one of the key questions in moving from static graph theory to dynamic graph theory concerns the preservation of properties and their computability.

One important general approach to the complexity and computability of properties in (static) graphs is the application of *metatheorems* which classify large classes of problems. An important metatheorem is *Courcelle's Theorem* (stated and

☆ This paper contains results published in preliminary form in [30].

* Corresponding author.
E-mail addresses: bernard.mans@mq.edu.au (B. Mans), luke.mathieson@mq.edu.au (L. Mathieson).

0304-3975/$ – see front matter
http://dx.doi.org/10.1016/j.tcs.2013.12.024

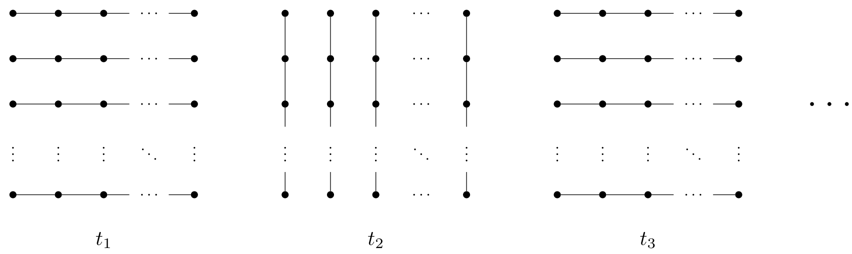

Fig. 1. An example of a simple dynamic graph on a static set of vertices, where the edges oscillate between "horizontal" and "vertical" paths. Some of the properties that the graph exhibits include being disconnected at any given time, but having a path *over time* (a journey) between any two vertices, and at each time the graph has treewidth 1, but the union of edges over all times (or even just two consecutive times) gives a grid graph, which is a canonical example of a graph with unbounded treewidth.

proved over a series of articles from [12] to [13]) which gives a polynomial time algorithm for any monadic second-order expressible property on graphs of bounded treewidth. More precisely the model checking problem for monadic second-order logic is fixed-parameter tractable with parameter $|\phi| + k$ where ϕ is the sentence of logic and k is the treewidth of the input structure. Frick and Grohe [23] give a similar metatheorem for first-order logic and structures of locally bounded treewidth, which allows a greater class of structures (all structures of bounded treewidth have locally bounded treewidth), but constrains the logical expressibility. In fact Dvořák et al. [16] show that properties expressible in first-order logic are decidable in linear time for graphs of bounded expansion, a superclass of several classes of sparse graphs, including those with bounded local treewidth. Stewart [34] demonstrates that Frick and Grohe's result holds if the bound on the local treewidth is dependent on the parameter of the input problem, rather than simply being constant.

Such metatheorems are extremely useful classification tools, and having them available for use in the context of dynamic graphs would be highly desirable. Two questions immediately arise when considering such an extension — are there any restrictions that have to be made to the logic and are there any further constraints on the structures (in this case the graphs)? Answering the first question is simple: no. An apparent natural match for dynamic graphs is temporal logic (a form of modal logic), however temporal logic has a simple canonical translation into first-order and monadic second-order logics. The addition of a "time" relation, that provides a temporal ordering is sufficient to capture the ideas of temporal logic. Thus we are not limited (more than before) with the logics that are applicable. With regards to the structure, we must first consider the setting of these metatheorems. They are both true in the context of finite model theory, emphasizing the *finite*. Therefore we are immediately limited to finite temporal domains (a finite domain for the vertices and hence edges of the graph is expected of course). Furthermore the tractability results we derive (at least in some cases) rely on taking the size of the temporal domain as a parameter. For a number of interesting application areas of dynamic graphs this is quite natural: periodic and recurrent graphs (see [7] and [20] for details) intrinsically bounded temporal domains. In other cases where we have a total order on the temporal domain, the parameter dependence on the size of the temporal domain may be removed. However, as we require the length of the logical sentence to be bounded by some function of the parameter, we still encounter the usual problem with logically expressing a relation between elements that are an unbounded distance apart. Of course this is even a problem encountered when trying to express simple connectivity in first order logic. Beyond this if we are to capture the temporal aspect in the structure, we must do so in a manner that is both usable and respects the constraints on the structure necessary for the metatheorems — namely bounded treewidth, local treewidth and bounded expansion.

Dynamic problems have been approached before in a number of ways. For instance, Hagerup [24] examines the situation where the logical expression changes (i.e. the question being asked changes), and Bodlaender [3], Cohen et al. [11] and Frederickson [22] give a variety of results that deal with graphs that change, requiring changes in the tree decompositions used. In contrast this work deals with problems that include the temporal dimension as an intrinsic aspect, rather than problems that can be answered at an instant in time (and then may need to be recomputed). A simple example is a *journey*, the temporal extension of a path, where the route between two vertices may not exist at any given point of time, but as the graph changes the schedule of edges may allow traversal.

The remainder of this paper deals with the translation of dynamic graphs into structures that maintain these bounds if the original graph did so at every point in time. In addition we demonstrate the utility of these metatheorems in classification by application to some open problems on dynamic graphs.

2. Preliminaries

2.1. Graphs and dynamic graphs

The graphs we employ are simple graphs, both directed and undirected. For the usual graph properties we use the usual notation (*qq.v.* Diestel [14]). In particular, given a graph G, we denote the number of vertices of G by $|G|$ and the number of edges by $\|G\|$.