Proposal for Master Thesis 02

24 April 2018

William Bombardelli da Silva

Properties of Triple Graph Grammars with Non-Terminal Symbols

Introduction

With the objective to enhance software quality, research and industry have proposed some modeling techniques that utilize graphical modeling languages (e.g. UML). These modeling languages can be viewed as graph languages, which in turn can be described by graph grammars.
In this context, a common problem is the transformation of graphs from different languages based on their grammars. A quite popular approach to this problem is the triple graph grammar, which aims to describe the transformation between two languages in terms of their grammars. Even though TGGs have shown some positive results, the efficiency of transformation algorithms and the usability of the formalism are still some drawbacks (Schürr & Klar, 2008). Nevertheless, we believe that TGGs could profit from some concepts of the theory of formal languages and graph grammars, in special the realm of visual languages.
More specifically, we judge that the TGG formalism is more expressive than it needs to be for the practical use in model transformations, this affects negatively its usability and efficiency. So we propose a restriction on the syntax of the grammar production rules so to create a family of grammars, that is potentially a proper subset of the general TGG. This restriction on the form of the rules includes the concept of non-terminal symbols, analogous to the non-terminal symbols of regular and context-free grammars of Chomsky. Then, we aim to devise a parsing/ transformation algorithm with smaller time and spatial complexity.

Motivation. The restriction on the form of the rules of TGGs may increase significantly its usability, as it reduces the minimum amount of rules necessary to describe a language, and increase the efficiency of the transformation algorithm, like it happens in the context of programming language compilers with the LL languages. This can, in turn, encourage more the use of triple graph grammars in practice and ease the solution of model transformation problems.

Key-words. Graph Grammars, Triple Graph Grammars, Parsing, Graph Transformation, Model Transformation, Model-driven Engineering, Software Engineering.

Methodology

This work is divided into several phases, presented in Table 1. First, we intend to use graph theory, formal languages and abstract algebra concepts to define abstract syntax and semantics of our family of triple graph grammars with non-terminal symbols. This part is heavily supported by the current literature.
Second, we plan to create a specialized parsing/ transformation algorithm and analyze its complexity. After that we intend to study the expressiveness of our formalism, specially in comparison with other current triple graph grammar techniques. Finally, we want to evaluate its practical use in some examples comparing the amount of rules used in the grammars and the efficiency of parsing and transformation algorithms in terms of run-time.
A possible extension of this work could be the characterization of an equivalence between several families of triple graph grammars with different computational models (e.g. finite automata, turing machine), like it exists for the classical formal languages.

Threats. One possible obstacle for this work is the creation of an efficient parsing algorithm. Even though there exist well-known parsing algorithms for string grammars, translating their concepts into our definitions of graph grammar might be difficult. Moreover, we are aware of possible termination problems in the case of directed cyclic graphs.

Start and End Dates	Activities
12/02/2018 to 11/03/2018	Initial research; first drafts on the idea; problem concretization; approach definition and validation
12/03/2018 to 11/04/2018	Vacations
12/04/2018 to 11/05/2018	Definition of the approach; search of literature; proposal;
12/05/2018 to 11/06/2018	Theoretical definitions; sketch of parsing algorithm; write literature review; 1st review
12/06/2018 to 11/07/2018	Parsing algorithm and expressiveness; write introduction; 2nd review
12/07/2018 to 11/08/2018	Algorithm's analysis; further theoretical considerations; write the main chapter
12/08/2018 to 11/09/2018	Evaluation of results; write evaluation; 3rd review
12/09/2018 to 11/10/2018	Finalize writing; prepare presentation/ defense

Table 1: Plan for the researching, developing and writing of the master thesis. This schedule is organized in months, where each month has its respective activities planed.

Related Work

There exists an extensive literature on graph transformation and rewriting systems, specially on the algebraic approach to graph grammars (Ehrig *et al.* , 2015), and triple graph grammars (Schürr, 1994). Moreover, there are several proposals of families of graph grammars applied to the context of visual languages, including regular (Gilroy *et al.* , 2017), context-free (Rekers & Schürr, 1997), and context-sensitive grammars (Zhang *et al.* , 2001; Adachi *et al.* , 1999; Drewes *et al.* , 2010). Marriott & Meyer (1997) proposed a hierarchy for classes of graph grammars, Flasiński in (1998) associated several types of graph grammars with the respective complexity to parse and in (1993) presented a parsing algorithm with complexity $O(n^2)$.
Shi *et al.* (2016) proposed a method for simplifying graph grammars. And several authors report the use of TGGs to solve model transformations (Gottmann *et al.* , 2016). But we have not found any publication that tries to enhance TGGs' efficiency and expressiveness/ usability by means of restrictions on the form of the grammar rules.

References

Adachi, Yoshihiro, Kobayashi, Suguru, Tsuchida, Kensei, & Yaku, Takeo. 1999. An NCE context-sensitive graph grammar for visual design languages. *Pages 228–235 of: Visual Languages, 1999. Proceedings. 1999 IEEE Symposium on*. IEEE.

Drewes, Frank, Hoffmann, Berthold, Janssens, Dirk, & Minas, Mark. 2010. Adaptive star grammars and their languages. *Theoretical Computer Science*, **411**(34-36), 3090–3109.

Ehrig, Hartmut, Ermel, Claudia, Golas, Ulrike, & Hermann, Frank. 2015. Graph and Model Transformation.

Flasiński, Mariusz. 1993. On the parsing of deterministic graph languages for syntactic pattern recognition. *Pattern Recognition*, **26**(1), 1–16.

Flasiński, M. 1998. Power properties of NLC graph grammars with a polynomial membership problem. *Theoretical Computer Science*, **201**(8), 189–231.

Gilroy, Sorcha, Lopez, Adam, & Maneth, Sebastian. 2017. Parsing Graphs with Regular Graph Grammars. *Pages 199–208 of: Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (* SEM 2017)*.

Gottmann, Susann, Nachtigall, Nico, Engel, Thomas, Ermel, Claudia, & Hermann, Frank. 2016. Towards the propagation of model updates along different views in multi-view models. *CEUR Workshop Proceedings*, **1571**(Bx), 45–60.

Marriott, Kim, & Meyer, Bernd. 1997. On the classification of visual languages by grammar hierarchies. *Journal of Visual Languages and Computing*, **8**(4), 375–402.

Rekers, Jan, & Schürr, Andy. 1997. Defining and parsing visual languages with layered graph grammars. *Journal of Visual Languages & Computing*, **8**(1), 27–55.

Schürr, Andy. 1994. Specification of graph translators with triple graph grammars. *Pages 151–163 of: International Workshop on Graph-Theoretic Concepts in Computer Science*. Springer.

Schürr, Andy, & Klar, Felix. 2008. 15 Years of triple graph grammars: Research challenges, new contributions, open problems. *Lecture Notes in Computer Science*, **5214 LNCS**, 411–425.

Shi, Zhan, Zeng, Xiaoqin, Huang, Song, Qi, Zekun, Li, Hui, Hu, Bin, Zhang, Sainan, Liu, Yanyun, & Wang, Cailing. 2016. A method to simplify description and implementation of Graph Grammars. *6th International Conference on Computing, Communications and Networking Technologies, ICCCNT 2015*, 2–7.

Zhang, Da Qian, Zhang, Kang, & Cao, Jiannong. 2001. A context-sensitive graph grammar formalism for the specification of visual languages. *Computer Journal*, **44**(3), 186–200.