Using Net Patterns to Simplify the Application of Graph Transformation Rules

Frank Trollmann and Sahin Albayrak

DAI-Labor, TU Berlin
Faculty of Electrical Engineering
and Computer Science
Frank.Trollmann@dai-labor.de, Sahin.Albayrak@dai-labor.de

1 Introduction

P/T-Systems and related languages are a good technique for modeling distributed executable processes. Reconfiguration Rules for P/T-Systems can be used to change their structure. This is useful for modeling processes that can change during their execution. This constellation is called a Reconfigurable P/T-System [1]. The task of modeling such a system requires the designer to consider all possible applications of the Reconfiguration Rules. Depending on the size of the system to be modeled, this task can be very complex.

This paper proposes a structural restriction of the P/T-System by introducing Net-Patterns. The purpose of these patterns is to simplify the complicated task of rule application and enable the designer to concentrate on the modeling process.

2 Problem

Reconfiguration Rules are a form of Graph Transformation. A Graph Transformation Rule for P/T-Systems consist of two systems, called Left Hand Side (LHS) and Right Hand Side (RHS),as well as a way to transformation LHS into RHS. The application of a Graph Transformation Rule to a P/T-System requires the existence of a morphism between LHS and the system. This morphism defines the scope of the application.

During the modeling process the designer has to assure the existence of such morphism and the non-existence of any morphisms that do not target the correct scope of the reconfiguration. The complexity of this task grows with the size of the P/T-System and number of Reconfiguration Rules. This growing complexity can easily lead to errors in rule applicability.

Additionally, the designer has to be aware of the fact that the application of a Reconfiguration Rule changes the P/T-Systems structure, which can influence the applicability of other Reconfiguration Rules. The designer also has to ensure the correct applicability of the Reconfiguration Rules to the set of all possible structures of the P/T-System. This task is even more complex. Especially in cases where there is a large or even infinite amount of possible structures.

The original task of the designer is to model her system under study. Rule applicability can divert her from this primary task and even lead to compromises between easy rule application and truthful representation of the system under study.

3 Approach

The applicability of a P/T-System Reconfiguration Rule mainly depends on the structure of the P/T-System and the LHS of the rule. Our approach is to provide structural design patterns for these nets that simplify the task of finding morphisms between them. This idea partially follows the work of Christopher Alexander [2], who introduced the design pattern paradigm as a way to communicate reoccurring patterns to other developers.

The difference is that the purpose of our Net-Patterns is to alleviate rule applicability. Furthermore, Net-Patterns are not voluntary. This assures the occurrence of certain fixed structures that can be targeted by the Reconfiguration Rule. Net-Patterns can also be seen as the definition of a Doman-Specific Modeling Language (DSML) based on P/T-Systems that consists of all P/T-Systems that conform to the Net-Patterns. By using this language the designer is able to abstract from issues that are already targeted by the patterns.

One example for such a DSML is described in [3]. In this paper the workflow of an emergency operation is modeled with Algebraic High Level (AHL-) Systems [4], a modeling language related to P/T-Systems. In this publication a Net-Pattern for all tasks, executed by the team members is established. This pattern is instantiated for every specific task. The types of the places within each tasks net are unique. This enables the Reconfiguration Rules to target a specific task by referring to the corresponding place types. Additionally, rules that operate solely on the task-pattern are defined. These rules are able to handle the high level task structure by inserting or moving tasks.

This example shows one possibility to define a Domain-Specific Modeling Language out of P/T-Systems and related languages and serves as proof of concept that such Patterns can be used to ease the problems of rule applicability.

References

1. Ehrig, H., Hoffmann, K., Padberg, J., Prange, U., Ermel, C.: Independence of net transformations and token firing in reconfigurable place/transition systems. In: ICATPN'07: Proceedings of the 28th international conference on Applications and theory of Petri nets and other models of concurrency, Berlin, Heidelberg, Springer-Verlag (2007) 104–123
2. Alexander, C., Ishikawa, S., Silverstein, M.: A Pattern Language: Towns, Buildings, Construction. Oxford University Press, New York (1977)
3. Trollmann, F.: Modeling emergency scenarios using albegraic higher order nets. Technical report, Technische Universitaet Berlin (2009) Diploma Thesis, to appear.
4. Padberg, J., Ehrig, H., Ribeiro, L.: Algebraic high-level net transformation systems. Mathematical Structures in Computer Science **5** (1995) 217–256