Supporting Geotechnical Design with Petri-Net-Based Process Patterns

R. Katzenbach & J. Giere
Institute and Laboratory of Geotechnics · Technische Universität Darmstadt · Germany

ABSTRACT: The paper deals with process patterns based on Coloured Petri-Nets as a process model to coordinate planning in structural engineering, e.g. geotechnical engineering. Coloured Petri-Nets are mathematically well-founded, and are often used to solve problems concerning concurrency in processes in other engineering domains. The paper introduces a new method to represent geotechnical information in Coloured Petri-Nets, thus enabling a content-based control over design process. Elementary process patterns are introduced, which serve as smallest modules in a bottom-up approach to the more complex process models.

1 INTRODUCTION

Design and construction-processes in civil engineering and especially in construction engineering require close cooperation of many planning participants who work in separate domains. The activities of architects, engineers, building owners, authorities and construction companies amongst others have to be coordinated in order to achieve cost and time effective solutions of high quality. The globalization of engineering services even increases the demands of coordination. The developments in internet technology have led to a widespread use of net-based communication systems in the construction industry. They often lack however adequate methods to coordinate planning processes as they do not use formal process models.

In order to overcome the shortcomings of contemporary net-based solutions for project communication and cooperation such as web-portals and cooperation-systems, various research activities in the last years have focused on processes and process-models. With respect to the character of planning processes in civil engineering the following aspects have to be considered:

- In contradiction to production processes in the industry, the design process in civil engineering is a priori only partly known. With the proceeding design, various design activities and planning participants have to be involved in the design process.
- As to have control of the actual planning state connected to costs, time etc. the process model requires the distinction between activities and states.
- Activities in the design process produce information, which is exchanged between the planning participants. Based on this information (plans, expertises, calculations, …) following activities may have to be carried out. The process model should be able to take into account this information.

Colored Petri-Nets have all the necessary properties for a suitable process-model. In this paper a process-model will be presented, which is based on the Petri-nets is steering the process based on information from the

2 PETRI-NETS

For a good introduction to the Petri-nets the reader is referred to [Reisig 1985]; a very good comprehensive report can be found in [Murata 1989].

Petri-nets consist of the disjoint finite set of places P and transitions T as nodes of the graph, which are connected with directed arcs as a flow relation F. The marking M, which is represented by tokens in places represents the actual state of the system. By firing a transition, the token is subtracted from the input place of the transition and added to the output place. Thus firing a transition transforms the state of the system to the subsequent state.

The basic form of a Petri-Net is defined as a tuple $PN = (P, T, F, M_0)$, where

P is a finite set of places

$$P = \{p_1, p_2, \ldots, p_m\}$$

T is a finite set of transitions

$$T = \{t_1, t_2, \ldots, t_n\}$$

$$(P \cup T) = \varnothing \text{ and } (P \cap T) \neq 0$$

F is a set of arcs as flow relations

$$F \subseteq (P \times T) \cap (T \times P)$$

M_0 is the initial marking

$$M_0: P \rightarrow \{0, 1, 2, 3, \ldots \}$$

Petri-nets have a graphical representation as is shown in figure 1. The figure also depicts the connection between Petri-nets and the corresponding elements of the planning process.

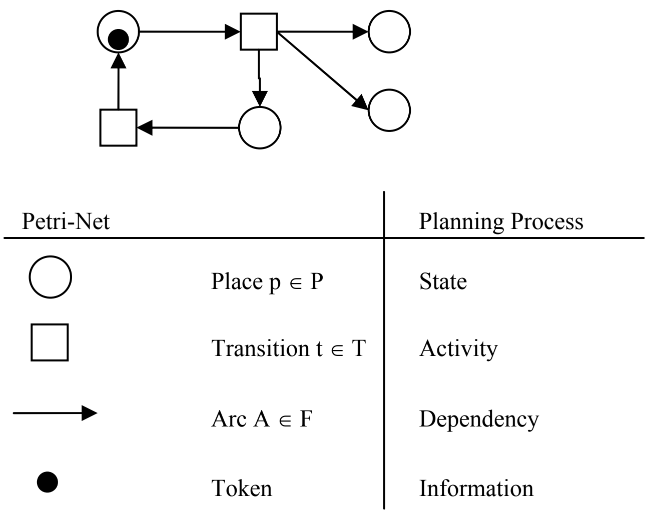


Fig. 1: Graphical representation of Petri-nets and

2.1 *Formal requirements*

As Petri-nets provide a formal framework, they also provide the possibility to analyze them and to define properties of the net, which are appropriate to the given problem.

With respect to the requirements of process modelling of business processes, v.d. Aalst gives a formal definition of Petri-nets as workflow nets [v.d. Aalst 1996, 1997, 1998, 2002], defining as well a soundness criterion. According to it, a workflow net (wf-net) has one single input place *I* and one single output place *O*. When the wf-net is short-circuited by a transition t*, it is strongly connected. When the output place *O* is marked with a token, all other places in the wf-net must be empty, so no other activities can be carried out (no transitions are enabled to fire). This prevents, that a task has erroneously not been carried out, although the process should already be terminated. In this case the process has not been modelled correctly. Figure 2 shows basic routing primitives in Petri-nets.

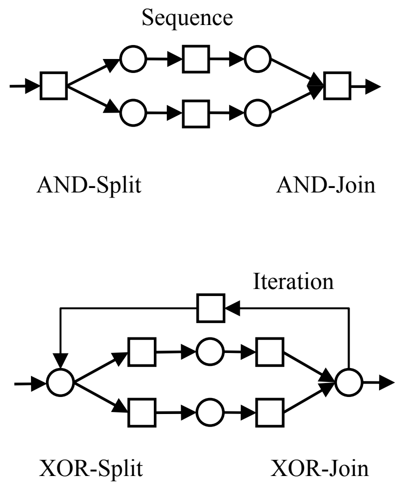


Fig. 2: Basic routing primitives in Petri-nets

In civil engineering the design process is partly unknown at the beginning but evolves with the proceeding planning. It is necessary to adapt the underlying process-model. This is achieved by the hierarchical substitution of transitions in the net as shown in figure 3.

In the coarse net the transition with the adjacent socket-places is substituted by an underlying, finer net with the port places.

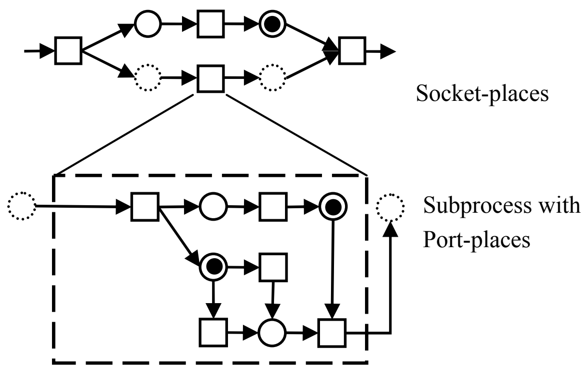


Fig. 3: Hierarchy by substitution

2.2 *Petri-nets with individual tokens*

For the representation of information in the Petri-nets it is necessary to introduce Coloured Petri-nets, where different tokens are distinguishable.

Within the activities the planning information is generated or modified. Depending on the actual planning state the further planning activities have to be carried out. For example the number of basement floors which are designed by the architect imply the required construction for the retaining walls. In order to be able to take into account the exchanged information and the results of each planning activity in the planning process and its effects on the process-model, the Petri-Nets are extended with a formal semantic to individualize the tokens.

The theory of Coloured Petri-Nets [Jensen 1996] expands the formal semantics of the Petri-nets by adding colours to the tokens and a formal semantic to perform operations on the individual tokens. Thus it is possible to transport information through the Petri-Net and define operations based on this information. Guard functions on the transitions can be arbitrarily complex to formulate conditions to fire the transition, i.e. carry out the activity.

3 METAINFORMATION

The information exchange in the design process today is mainly document-based. Product-models exist only for specific domains, such as steel-constructions. In order to access the information which is necessary e.g. for decisions in the process-model, a metainformation is introduced. The idea is, to abstract only the necessary information to control the design-process from the abundance of information, which is generated in the planning process. Of course the difficulty lies in the definition of the adequate amount of information. One approach from the technical basis is to abstract the information from technical standards and regulations [Katzenbach et al. 2004].

The metainformation has to be accessible for the process model, which was programmed using Standard ML (SML, [Milner et al. 1997]). As a first approach, a very simple approach using tuples of labels (identifiers), values and index is used. One tuple carries one construction specific piece of information, coded in string, integer or real. The tuples are listed in one indexed list called the information container.

Table 1 shows the identified requirements from process-modeling in the left column and the developed implementation in the right column, an example is given in fig. 4.

Table 1: Requirements to the metainformation

Requirements deriving from Process-Modelling	**Implementation in Petri-Nets with individual Tokens**
Metainformation Single, construction specific information • label (string) • value (integer/real/string) • version (integer)	Tuple (label, value, index) Example („foundation_type“, S(„shallow“), 1) („excavation_depth“, R(6.5), 1)
Information container All metainformation necessary for the control of the process	List of tuples (Tuple 1, Tuple 2, …, Tuple n)

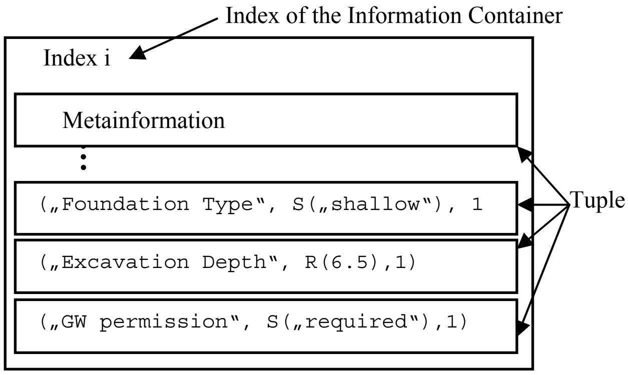


Figure 4: Information Container in SML (Example)

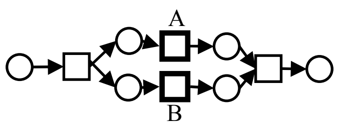
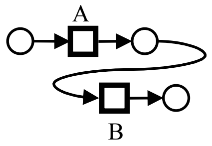
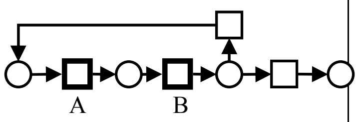
4 DESIGN PATTERNS

On the basis of the introduced metainformation the routing primitives shown in fig. 1 can be extended towards the use of semantics. The work was done using CPNtools [Ratzer et. al 2003] and Standard ML.

The aim of the use of design patterns is to build up the whole complex process-model from small parts in a bottom-up approach. Each design pattern has to meet the formal requirements.

The six routing primitives (Sequence, Iteration, XOR-Split, XOR-Join, AND-Split, AND-Join) are arranged into elementary design patterns, taking into consideration the kind of dependency between two activities in the design process. The possible dependencies of two activities A and B and the corresponding arrangements are given in table 2. For a better overview not all inscriptions of the net are shown in the figures of the table.

Table 2: Design patterns in Petri-nets

dependency	primitives	Design patterns in Petri-nets
no dependency	concurrency	
unilateral dependency	sequence	
mutual dependency	sequence + iteration	

For the mutual dependency of activities A and B it is necessary to carry out both activities until the result of the activity B has no effect on A. An example for such a mutual dependency is the design of a combined piled-raft foundation, where the load-settlement behaviour of the whole foundation interacts with the rising structure. Structural engineer and Geotechnical Engineer have to cooperate iteratively to optimize the construction. Both planning domains have to approve the design before the next steps can be taken. In the design pattern this is symbolized by the XOR-split following activity B.

```
color Int_Real_String = union I:Int + R:Real + S:String;
color Meta = Product String * Int_Real_String * Int;
color Info = list Meta;
...
var info, info1, info2: Info;
var b, b1, b2: Int;
...
(* list_union unifies two lists of Metainformation *)
fun list_union ([], list2) = list2
|list_union (hd::tl, list2) = exists_l hd (list_union(tl,list2)) ;
```

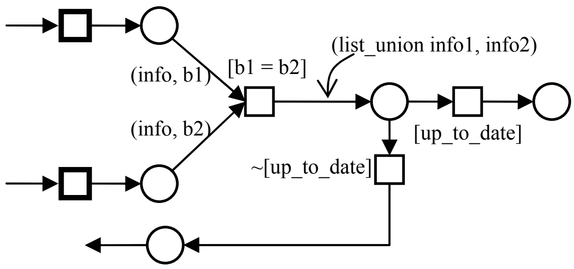


Figure 5: Detail of Petri-net

A library of functions has been installed in SML to carry out the necessary operations on the meta-information in the information container (token). These operations include splitting, unifying and versioning of the information.

Figure 5 shows a detail of an example of a design pattern including some net inscriptions. The arc inscriptions are in brackets, the guards of the transitions are shown in angular brackets. There are two concurrent design activities, marked with the thick line. In both activities information is generated and the list of metainformation in the tuples is modified. After finishing the information is merged under the condition, that the index of the information container is identical (b1=b2). It is then checked, whether the information is up to date. If applicable the design process is to be checked again, indicated by the iteration.

5 DESIGN PROCESSES AND CONCLUSION

The given process-model has been implemented in a new cooperation platform called ProMiSE (Process Model in Structural Engineering) [Rueppel et al. 2004, Katzenbach et al. 2002] and has been tested for design scenarios with several planning participants. For the domain of Geotechnical Engineering the design processes for the geotechnical constructions such as retaining walls, foundations, slopes etc. have been transferred to the process model. The model comprises all design activities in the domain from the technical point of view. It showed, that the hierarchical approach is well suited to integrate processes from the granularity of single proofs up to the very coarse process of the whole scenario. The standard processes of Geotechnical Engineering can be provided by a library and be integrated during runtime into the process model, thus enabling the change and adaptation to the proceeding design process.

Design patterns including the function library to operate on the metainformation are a universal approach to design processes. Although the information is kept in a rather simple way, the functionality of the model is sufficient. With a process model of the design process taking into consideration the actual design via the metainformation and a formal framework, cooperation between the planning participants is easier to achieve and to control.

REFERENCES

van der Aalst, W. M. P. 1996. Three Good Reasons for Using a Petri-net-based Workflow Management System. In Navathe, S. and Wakayama, T. (eds.), *Proc. of the Internationsl Working Conference on Information an Process Integration in Enterprises (IPIC 96)*: 179-201, Cambridge, Mass., USA

van der Aalst, W. M. P. 2002. Making Work Flow: On the Application of Petri Nets. *Lecture notes in computer science 2360*: 1-22. Berlin, Heidelberg: Springer van der Aalst, W. M. P. 1998. The Application of Petri Nets to Workflow Management. *Journal of Circuits, Systems, and Computers 8 (1)*: 21-66. Singapore: World Scientific Publishing Company

van der Aalst, W. M. P. 1997. Verification of Workflow Nets. *Lecture notes in computer science 1248*: 407-426. Berlin, Heidelberg: Springer

Jensen, K 1996. *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use*. Vol. 1-3, Berlin, Heidelbarg: Springer

Katzenbach, R.; Meissner, U.F.; Rueppel, U.; Savidis, S.; Giere, J.; Greb, S.; Mejstrik, M. 2004. Abstraction of Process Relevant Information from Geotechnical Standards and Regulations. In Beucke, K. et al. (eds.) *Proceedings of the Xth International Conference on Computing in Civil and Building Engineering ICCCBE.* Bauhaus Universität Weimar, Germany

Katzenbach, R.; Meissner, U.F.; Rueppel, U.; Giere, J. und Greb, S. 2002. Process-Oriented Network-based Collaboration in Geotechnical Engineering. *Proceedings of the 9th International Conference on Computing in Civil and Building Engineering*: 1063-1068, National Taiwan University, Taipei, Taiwan

Milner, R.; Tofte, M.; Harper R.; MacQueen, D. 1997. *The Definition of Standard ML*. Cambridge, MA USA: The MIT Press

Murata, T. 1989. Petri Nets: Properties, Analysis and Applications. *Proceedings of the IEEE 77 (4)*: 541-580. New York: EEE

Petri, C. A. 1962. *Kommunikation mit Automaten*. Schriften des IIM 3, Universität Bonn, Institut für Instrumentelle Mathematik

Ratzer A..V.; Wells, L.; Lassen, H. M.; Laursen, M.; Qvortrup J. F.; Stissing, M. F.; Westergaard, M.; Christensen, S.; Jensen, K. 2003, CPN Tools for Editing, Simulating, and Analysing Coloured Petri Nets. In W. v.d. Aalst (ed.): *Applications and Theory of Petri Nets 2003: 24th International Conference, ICATPN 2003.* LNCS 2679, pp 450 – 462, Berlin: Springer

Reisig, W. 1985. *Petri Nets, An Introduction*. Monographs on Theoretical Computer Science. W. Brauer, G. Rozenberg, A. Salooma (Eds.). Berlin: Springer

Rueppel, U.; Meissner, U.F.; Greb, S. 2002. A Petri Net Based Method for Distributed Process Modelling in Structural Engineering. In Beucke, K. et al. (eds.) *Proceedings of the Xth International Conference on Computing in Civil and Building Engineering ICCCBE.* Bauhaus Universität Weimar, Germany