БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ . BULGARIAN ACADEMY OF SCIENCES

ПРОБЛЕМИ НА ТЕХНИЧЕСКАТА КИБЕРНЕТИКА И РОБОТИКАТА, 52
PROBLEMS OF ENGINEERING CYBERNETICS AND ROBOTICS, 52

София . 2001 . Sofia

Modeling and Synthesis of Information Interactions*

Tasho Tashev, Hristo Hristov

Institute of Information Technologies, 1113 Sofia

Introduction

The present paper is directly connected with previous study on the development of the functional structure of a software system for modeling and synthesis of information interactions. The models are formally described with the help of Petri nets (PN), accepting the design of hierarchical structures [1, 2]. PN models enable the complete investigation of the interactions behaviour, as well as the synthesis of new structures with prescribed properties**.

In this synthesis, based on tensor transformations of PN models, the application of the arcs weight matrix W has been suggested, which allows finding unambiguous structural solution [3]. The tensor transformations pass several stages, representing the initial model into a matrix form because the tensor equations for PN transformations are based on their matrix form [4, 5]. The transformations are accomplished on a model, specified as basic PN (BPN), defined in [6].

The whole mechanism of synthesis of new structures from the initial model up to the type of BPN consists of four successive stages [7], which represent the methodology of the tensor approach in its use for PN. The matrix D_{β}^{γ} of BPN is introduced as input data, and as a result matrix $D_{\beta'}^{\gamma'}$ is obtained, which is an analytical description of the new net considered. D_{β}^{γ} and $D_{\beta'}^{\gamma'}$ are incidence matrices. Nevertheless matrix $D_{\beta'}^{\gamma'}$ is hardly accepted and evaluated. In the cases when comparison of several variants is implied, it is more efficient to evaluate and compare the graphically represented PN models. This is one of the considerations to suggest the use of a graphically oriented software system.

---

[*] Synthesis of a structure with additional interaction, for example. A similar problem has been solved in [3].

** The investigations are partially supported by the National Scientific Fund, contract No И-703/97 and BAS-IIT, theme 010041.

Graphically oriented software structure for modeling and synthesis of PN models

The computer analysis of the properties of PN-models is accomplished with already established software packets and hence the analysis is not a subject of consideration. The synthesis of a new PN model follows the approach of tensor transformation on BPN, using the weights matrix W_{β}^{γ}, in order to obtain unambiguous solution [3]. When the new model is obtained it is analyzed in order to check whether it has preserved BPN properties.

The properties are classified into several main groups:

– Structural properties: Bounds, Liveness, Conservation, Repetiveness;
– Graph properties: Traps, Deadlocks;
– Linear properties: P- and T-semi-flows, called invariants also;
– Time analysis: liveness, safety, etc, with time limits introduced.

Fig. 1 shows the block scheme of the software packet designed for analysis of the properties of a PN model (1a) and for the synthesis of BPN of a model or a group of models with set properties (1b).

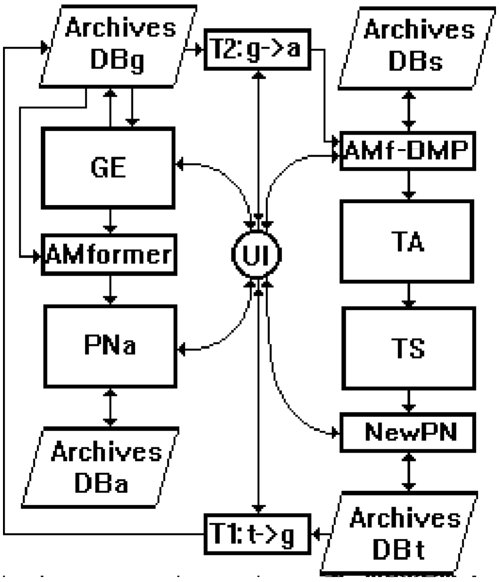


Fig. 1. Block-scheme of analysis and synthesis process of PN

The blocks are denoted as follows:

DBg – a file structure storing the image of graphical PN-models;

GE – a graphic editor, which enables the use of graphic symbolic for transition (Fig. 2) (represented horizontally or vertically), position and arc. The last one can be generated as dotted line, defined by the points ai (1, 2, 3,..., *n*), as well as by the symbolic of hierarchical structures.

A/Mf – transducer of the graphic PN model into analytical, respectively into matrix or list type, appropriate for next processing.

PNa is a block for analysis of the properties of PN models entered into the system, the results being stored in Dba.

UI – user's interface.

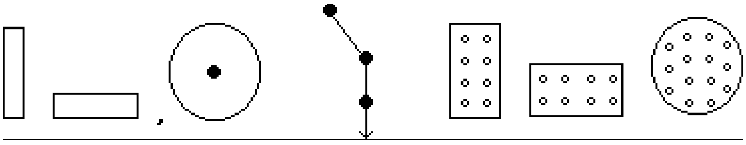


Fig. 2. Graphic symbols

The block Amf-BMP transfers the analytically defined model into BPN [4], which is applied to the input of the block for tensor analysis (TA) – Stages I and II [2]. The block for tensor analysis (TS) realizes stages III and IV. The archive DBs contains BPN models, and DBT – the new models synthesized in the form of incidence matrices of PN. The incidence matrix D of the new model obtained is represented into graphic form for convenience. The presence of mutual unambiguous correspondence between the matrix and graphic representation of PN enables the transformation automation.

Two blocks are necessary in the general case. One of them – (T1), transforming the matrix representation of PN into the form of data base Dbg (Ttamsfotm1: tensor → graphics) and the other one – (T2), transforming the form of Dbg into analytical form – the form of BPN block (Ttransform 2: graphics → analytical).

The separating of the final phase of synthesis into a separate block (NewPN) in the new PN obtained is quite appropriate, since a group of models can be obtained from one initial model with the help of tensor transformations. For this reason the formation of block T1 is an important task.

An algorithm for the transformation of PN from a matrix into a graphical form

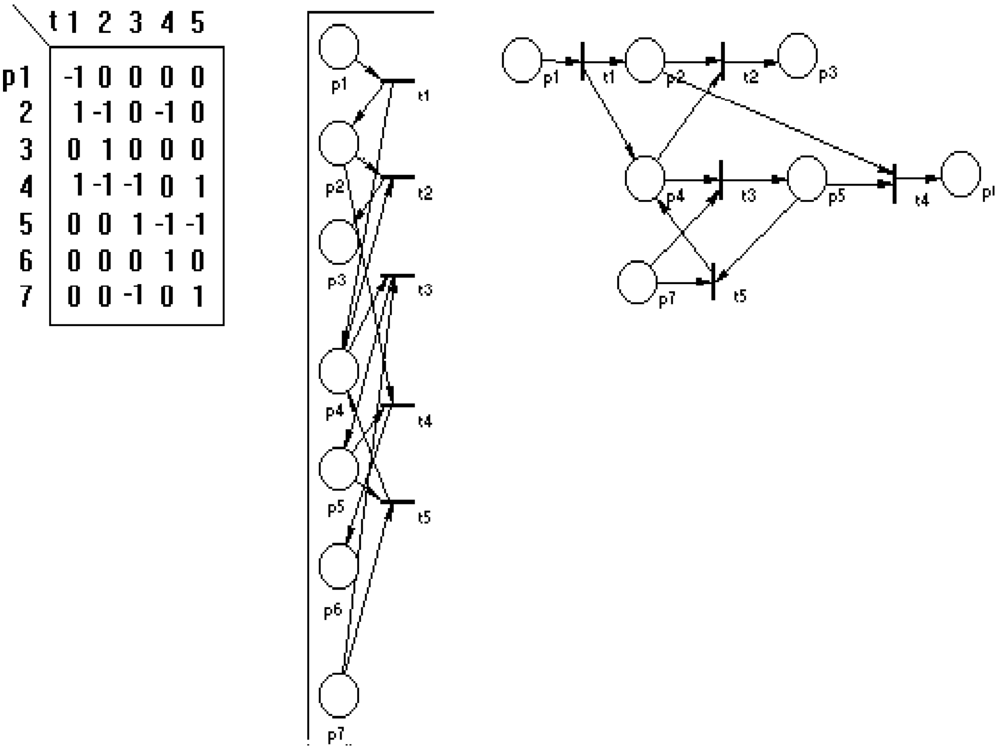
The specified in Fig. 3 block (T1) for transformation requires software support for its efficient use by the investigators. This requires the specification of an algorithm that is to be programmed for computer execution. It follows from the block-scheme (Fig. 3) that we cannot talk for one (the best in general) algorithm because it cannot be optimized with respect to all the disciplines for screen design. That is why the problem is divided into two parts – basic algorithm (general screen parameters) and specific module (for the discipline selected). In this way the requirements (and formats) of the concrete GE will be respectively reflected.

The input data for the basic algorithm is the matrix D with dimension $n \times m$ (n rows and m columns) of integer elements d_{ij} (positions × transitions); A1, B1, C1, M1, N1, O1>0, integers.

```
A1 Input Dnew[n×m]: R=n, S=m: A(I, J) = Dnew[n×m]
A2 Input A1, :B1, :C1, M1, :N1, O1
A3  B1  =  M1+C1
A4  FOR  I  =  1  TO  R
A5 K(I) = A1 +(I–1)*N1
A6 SUB{P(I) =<centerP(I)_X>,<centerP(I)_Y>}
A7  NEXT  I
A8  X1  =  M1+C1:  Y1=B1+O1:
A9  FOR  J=1  TO  S
A10 L(J) = Y1 +(J–1)*N1
A11 SUB{T(J) =<centerT(J)_X>, <centerT(J)_Y>}
```

```
A12 NEXT J
A13 FOR J=1 TO S:
A14 FOR I = 1 TO R:
A15 IF A(I, J) =–1 THEN SUB{Input_arc(I, J)=
=[<frontP(I)_X>,<frontP(I)_Y>], [<endT(J)_X>,<endT(J)_Y>]}
A16 IF A(I, J) =–1 THEN SUB{Input_arc(I, J)=
=[<frontP(I)_X>,<frontP(I)_Y>], [<endT(J)_X>,<endT(J)_Y>],
   [weight_ABS(A(I, J))]}
A17 NEXT I
A18 NEXT J
A19 FOR J = 1 TO S
A20 FOR I = 1 TO R:
A21 IF A(I, J) = 1 THEN SUB{Output_arc(I, J)=
=[<endP(I)_X>,<endP(I)_Y>], [<frontT(J)_X>,<frontT(J)_Y>]}
A22 IF A(I, J) >1 THEN SUB{Output_arc(I, J)=
=[<endP(I)_X>,<endP(I)_Y>], [<frontT(J)_X>,<frontT(J)_Y>),
[weight_ABS(A(I, J))]}
A23 NEXT I
A24 NEXT J
A25  END
```

Here A1 is the radius of the circle of a primitive position; B1 is the width, and C1– the length of the primitive transition. M1 is the length of one cell (along X-axis), and N1 – the width (along Y-axis). O1 is the shift of the primitive transition along Y-axis.



	t1	2	3	4	5
p1	-1	0	0	0	0
2	1	-1	0	-1	0
3	0	1	0	0	0
4	1	-1	-1	0	1
5	0	0	1	-1	-1
6	0	0	0	1	0
7	0	0	-1	0	1

Fig. 3. Incidence matrix

The arrays K(I), L(J) contain the screen coordinates along the Y-axis of the centers of cells on the screen network (for positions, transitions). The discipline chosen for filling the screen cells is two vertical columns – the positions are to the left, the transitions – to the right. This simplest discipline is possible due to the presence of PN – its graphic representation in the type of bichromatic oriented multigraph.

The subroutines SUB() fill the lists with screen coordinates: of the positions P(I), the transitions T(J), the input for the transition arcs Input_arc(I, J), and the output ones – Output_arc(I, J). They are parts of the specific module. The lists thus pointed out contain sufficient information for graphic construction of PN-model, which is the task of GE.

As illustration Fig. 3 shows the incidence matrix for synthesized PN-model of a timer mechanism, and Fig. 4 – the graphic form, constructed according to the algorithm described.

Conclusion

The paper discusses the main problems that have to be solved in the design of a graphic system aiding PN tensor transformations.

A model of the system interactions is suggested and its structure is discussed. The connections between the functional blocks are defined. An algorithm transforming PN from a matrix towards a graphic form is specified.

The present research continues some earlier problems studied by the authors' team. The structural schemes and algorithmic procedures obtained are a basis for the software realization of the system

References

1. Hristov, H. R., R. K. Kunchev, T. D. Tashev. Computer-aided modeling and investigation of interactions described by Petri nets. — Problems of Engineering Cybernetics and Robotics, **47**, Sofia, 1998, 69-74.
2. Hristov, H. R., T. D. Tashev. Computer-aided synthesis for interconnection process model. — Problems of Engineering Cybernetics and Robotics, **51**, Sofia, 2000.
3. Tashev, T. D., H. R. Hristov. On one approach for modification and expansion of the information interaction models. — Problems of Engineering Cybernetics and Robotics, **49**, Sofia, 2000, 78-87.
4. Petrov, A. E. Tensor Methodology in Systems Theory. M., Radio i Sviaz, 1985, p.152. (in Russian).
5. Kulagin V. P. Algebra of the network models describing parallel computing structures. — Automation and Modern Technology, 1993, No 2, 25-30 (in Russian).
6. Tashev, T. D., M. D. Marinov. Basic structures and operations in tensor transformations of Petri nets. — In: Proc. of Bulgarian-Rusian Seminar "Methods and Algorithms for Distributed Information Systems Design. Theory and Applications" May, 1996, Sofia, 152-162. (in Russian).
7. Tashev, T. D., E. V. Dimova. Algorithmic conditions for tensor transformations on matrix representation of Petri nets. — In: National Conference "Automation and Informatics — 97". 13-16 Oct., 1997, Sofia, vol. 6, 46-49 (in Bulgarian).

Моделирование и синтез информационных взаимодействий

Ташо Ташев, Христо Христов

Институт информационных технологий, 1113 София

(Резюме)

Рассматриваются проблемы, связянные с функциональной структурой программного пакета, осуществляющий анализ основных параметров PN-моделей и определенный тип синтеза PN-моделей при сохранении параметров начальной модели. Основное внимание направлено к двум из функциональных блоков пакета: графический редактор и преобразователи PN-моделей, задающие форму описания модели (аналитическую или граффическую). Предложена модель взаимодействий в системе и дефинируются связы между функциональными блоками.