Archives of Control Sciences
Volume 12(XLVIII), 2002
No. 3, *pages 225–241*

Industrial Petri Nets

MIROSŁAW ZABOROWSKI

A new class of Petri nets, called "Industrial Petri Nets" (IPN) has been defined in the paper as a special case of Timed Coloured Petri Nets (TCPN). The purpose of IP-nets is modelling manufacturing processes, maintenance and data processing in CIM systems. For the IPN definition the relations between the basic notions of production planning and control were discussed. Thus a reference model for different types of processes and different control methods has been created. As an example the IPN graph of a stamping process in a car factory has been displayed.

Key words: computer integrated manufacturing (CIM), coloured Petri nets (CPN), production planning and control (PPC), discrete process modeling

1. Motivation

A reference model for discrete production processes is needed. It should be similar to the ones for MRP II systems [6] but it should encompass all layers of a CIM system – from tactical planning in an industrial plant to activity control in particular machines. Activities, operations and processes, belonging to any production process, may be classified as:

– **manufacturing** (manufacturing, transport and quality control),
– **preparatory** (reequipment, repair, maintenance etc.),
– **data processing** (data processing, especially for control, including management).

The general production process model should include all interactions between activities, operations and processes of the same and of different types. With such a model integration of CIM systems [5] would be considerably easier. The model should contain the data structure of production process and its fragments, as well as the functional structure of a multi-layer production control system. The greatest difficulties arise with formal description of events sequence in a production process and in its control system. It is generally known that Petri Nets are often applied to such purposes [3, 4]. Petri nets designed to build general models of production processes should be colored and timed,

The author is with Institute of Automatic Control, Silesian Technical University, ul. Akademicka 16, 44–100 Gliwice, Poland, email: mzaborowski@polsl.gliwice.pl

The work has been supported by Polish State Committee for Scientific Research, grant 8 T11A 02018.
Received 14.06.2002, revised 12.9.2002.

because there are many options of multi-variant product features, many feasible variants of workstations equipment and many feasible output values of data processing algorithms and because it is necessary for many applications to model all situations when the time is consumed. Furthermore, there are many structural features of discrete production processes which are relevant to all of them and, on the other side, distinguish them among other processes modeled with TCPN [1]. Therefore it is possible to define a subclass of TCPN, which are simpler but sufficiently general for industrial applications. Such nets, which have been named "Industrial Petri Nets" (IPN), are defined in this paper.

2. Industrial Petri nets as a special case of TCP-nets

The CPN and TCPN definitions from [1] are mentioned below to facilitate comparison of IP-nets and TCP-nets:

Definition 1: A **non-hierarchical CP-net** is a tuple
CPN = $(\Sigma, \mathrm{P}, \mathrm{T}, \mathrm{A}, \mathrm{N}, \mathrm{C}, \mathrm{G}, \mathrm{E}, \mathrm{I})$ satisfying the requirements below:

1. Σ is a finite set of non-empty types, called **color sets**.
2. P is a finite set of **places**.
3. T is a finite set of **transitions**.
4. A is a finite set of **arcs** such that: $\mathrm{P} \cap \mathrm{T} = \mathrm{P} \cap \mathrm{A} = \mathrm{T} \cap \mathrm{A} = \emptyset$.
5. N is a **node function**. It is defined from A into $\mathrm{P} \times \mathrm{T} \cup \mathrm{T} \times \mathrm{P}$
6. C is a **color function**. It is defined from P into Σ.
7. G is a **guard function**. It is defined from T into expressions such that: $\forall \mathrm{t} \in \mathrm{T} : [\mathrm{Type}(\mathrm{G}(\mathrm{t})) = \mathrm{B} \wedge \mathrm{Type}(\mathrm{Var}(\mathrm{G}(\mathrm{t}))) \subseteq \Sigma]$
8. E is an **arc expression function**. It is defined from A into expressions such that: $\forall \mathrm{a} \in \mathrm{A} : [\mathrm{Type}(\mathrm{E}(\mathrm{a})) = \mathrm{C}(\mathrm{p}(\mathrm{a}))_{\mathrm{MS}} \wedge \mathrm{Type}(\mathrm{Var}(\mathrm{E}(\mathrm{a}))) \subseteq \Sigma]$ where p(a) is the place of N(a)
9. I is an **initialization function**. It is defined from P into closed expressions such that: $\forall \mathrm{p} \in \mathrm{P} : [\mathrm{Type}(\mathrm{I}(\mathrm{p})) = \mathrm{C}(\mathrm{p})_{\mathrm{MS}}]$

Definition 2: A **timed non-hierarchical CP-net** is a tuple TCPN = (CPN, R, r_0) such that:

1. CPN satisfies the requirements of a non-hierarchical CP-net in Def. 1 – when in (viii) and (ix) we allow the type of E(a) and I(p) to be timed or untimed multi-set over C(p(a)) and C(p), respectively.
2. R is a set of **time values**, also called **time stamps**. It is a subset of Re closed under + and containing 0.
3. r_0 is an element of R, called the **start time**.

Definition 3: A **marking** is a timed multi-set over a set of token elements. The **initial marking** M_0 is the marking obtained by evaluating the initialization expressions:

$$\forall p \in P : M_0(p) = I(p)_{r_0}$$

A **state** is a pair (M,r) where M is a marking and r a time value. The **initial state** is the pair (M_0,r_0)

Relations between IP-nets and equivalent TCP-nets are described by following replacements and their interpretations, which are presented in the next part of the paper:

(i) $\Sigma = (\mathrm{FO}, \mathrm{VV}, \mathrm{IV}, \mathrm{DV}, \mathrm{RV}, \mathrm{IE}, \mathrm{AE})$
(ii) $\mathrm{P} = \mathrm{AS} \cup \mathrm{PS}$
(iii)
(iv) $\mathrm{A} = \mathrm{IN} \cup \mathrm{MO} \cup \mathrm{AO} \cup \mathrm{FS(AS)} \cup \mathrm{FC(AS)}$
(v)
(vi) $\forall(a, i) \in \mathrm{AS} : \mathrm{C}(a, i) = \mathrm{AE}(a, i)$
$\forall(s, l) \in \mathrm{PS} : \mathrm{C}(s, l) = \mathrm{IE}(s, l)$
(vii)
(viii) $\forall((a, i), (s, l)) \in \mathrm{IN} :$ [EI((a,i),(s,l))) $\in \mathrm{IE}(s, l)_{\mathrm{MS}}$]
$\forall((a, i), (s, l)) \in \mathrm{MO} :$ [EM((a,i),(s,l))) $\in \mathrm{IE}(s, l)_{\mathrm{MS}}$]
$\forall((a, i), (s, l)) \in \mathrm{AO} :$ [EO((a,i),(s,l))) $\in \mathrm{IE}(s, l)_{\mathrm{MS}}$]
$\forall(a, i) \in \mathrm{AS} :$ [ES(a,i) $\in \mathrm{AE}(a, i)_{\mathrm{TMS}}$]
$\forall(a, i) \in \mathrm{AS} :$ [EC(a,i) $\in \mathrm{AE}(a, i)_{\mathrm{TMS}}$]
(ix) $\forall(a, i) \in \mathrm{AS} : \mathrm{I}(a, i) \in \mathrm{AE}(a, i)_{\mathrm{TMS}}$
$\forall(s, l) \in \mathrm{PS} : \mathrm{I}(s, l) \in \mathrm{IE}(s, l)_{\mathrm{MS}}$

where

FO – a set of feature options of products and resources
VV – a set of variable values of data items
IV – a set of variants of multi-variant products and resources
DV – a set of variants of data items
RV – a set of readiness variants of organizational units
IE – a set of executive objects
AE – a set of executive agents
AS – a set of active stages
PS – a set of passive stages
IN – a relation of active stage inputs
MO – a main output function of active stages
AO – a relation of active stage accessory outputs
FS(AS) – a hidden start arc function of active stages
FC(AS) – a hidden completion arc function of active stages
EI – an input expression function
EM – a main output expression function
EO – an accessory output expression function
ES – a start expression function
EC – a completion expression function

I – an initialization function
$IE(s,l)_{MS}$ – a set of all multi-sets over the set of executive items of the passive stage (s,l)
$AE(a,i)_{TMS}$ – a set of all timed multi-sets over the set of executive agents of the active stage (a,i).

3. Assumptions about structure of a production process model

It is assumed that:

- **workstations**, which are established or virtual aggregates of renewable resources, are elementary organizational units of an industrial plant,
- **activities** are elementary pieces of a discrete production process,
- **organizational units** and material **balance nodes** (including buffer stocks and warehouses) mutually separate each other in the logistic system structure of an industrial plant,
- **data processing workstations** and **data memory areas** mutually separate each other in the production planning and control system structure,
- in the sequential structure of a production process **active** and **passive stages** separate each other,
- a discrete production process in an industrial plant is composed of **operations** executed in workstations and operations consist of activities. In hierarchical production systems operations are aggregated to **processes** which are often assigned to **workcells**, e.g. established or virtual organizational units, composed of workstations. In certain models there are also **operation phases** or **work operations**. During operation phase no resources nor input products are introduced to the operation, as well as no resources nor output products are issued [9].Work operations are executed in **workcenters**, which consist of parallel identical workstations.

Activities, operation phases, operations, work operations and processes are special cases of **active stages** which are applied to the models relevant for different levels of production planning and control system. **Passive operation stages, stages between operations** and **stages between processes** are special cases of **passive stages**.

To facilitate analysis of complex IPN graphs different symbols for six classes of stages are introduced (fig. 1). Among them the symbols of manufacturing active and passive stages are the same as symbols of operations and buffer stocks in the common organizational charts.

It is assumed each active stage has exactly one input transition, one output transition and one place between them (fig. 2). The arcs from the input transition to the place and from the place to the output transition are called a **start arc** and a **completion arc** respectively. They represent the state changes of the active stage at start and completion times of the elementary **stage action**. Arcs from input passive stages represent their state

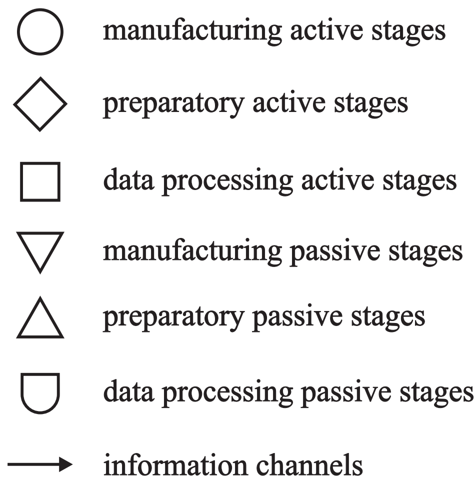

Figure 1. Symbols applied to the IPN graphs

changes at the start time. Similarly, arcs to output passive stages correspond to their state changes at the stage action completion time. One of output passive stages is the main, while the others are accessory output stages. From assumption of mutual separation of active and passive stages results that in a Petri Net model of a production process all transitions are input or output transitions of certain active stages. Furthermore, connections between transitions and input or output passive stages may be univocally identified by relevant pairs (active stage, passive stage). Therefore the relation of places and transitions incidence from the Petri Net definition may be replaced by the relation of active and passive stages sequence in a production process. Formally, in the so changed PN based model, transitions are hidden inside active stage symbols (fig. 2). Of course, hidden arcs and related to them expressions, which describe changes of the stage marking, may not be forgotten. Thus, instead of one arc expression set E(A) for TCPN arcs, there are five expression sets EI(IN), EM(MO), EO(AO), ES(AS), EC(AS) for the IPN arcs.

For activities and operation phases the active stage model with one input transition and one output transition is correct but for operations, work operations and processes is not. In a general case the simplifying assumption that marking of all input passive stages of a given active stage changes at the same time is not true. The same is for the output passive stages. However this assumption is applied to all MRP II systems [2] and to many other production control systems. To avoid it one can model complex manufacturing processes as hierarchical IP-nets. It was demonstrated that it is much easier for the IP-nets than for general TCP-nets [8].

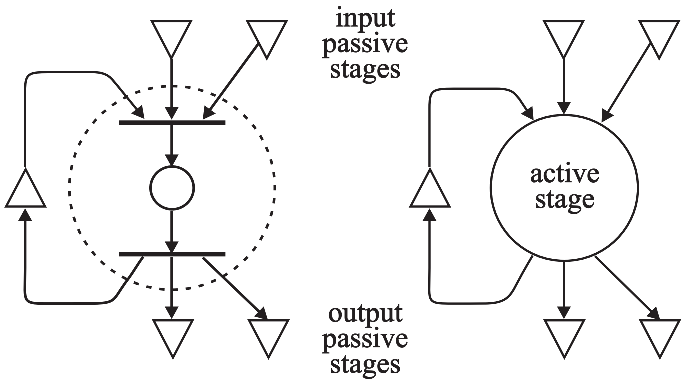

Figure 2. Hiding transitions in an active process stage

4. The IPN graph structure

The first six points of the CPN definition (chapter 2) deal with the structure of a CP-net, while remaining points display its initial marking and expressions, which describe how the marking changes when transitions occur. Imitating this definition one can see that for the IPN models of production systems it is necessary to specify sets of its active and passive stages, relations between them, as well as sets of their attributes and attribute values. To do it one must know objects, which are produced or consumed by active stages, and organizational units, where stage actions are performed, as well as places between workstations, where objects remain before or after stage actions. Thus, the general IPN definition must contain following sets:

1. UU – a finite set of **organizational units**, $\mathrm{UU} \ni \mathrm{i}$. Workstations $(\mathrm{i} \in \mathrm{SN})$ are relevant to activities, operation phases and operations. Workcenters $(\mathrm{h} \in \mathrm{CR})$ are relevant to work operations. Routings (r ∈ RTG) or routings in workcells $((\mathrm{g}, \mathrm{r}) \in \mathrm{RTG}, \mathrm{g} \in \mathrm{CL})$ are relevant to processes.
2. MM – a finite set of **places between workstations**, $\mathrm{MM} \ni \mathrm{l}$. Places between workcells are special cases of places between workstations and are identified by the same identifier "l".
3. II – a finite set of **objects**, $\mathrm{II} \ni \mathrm{s}$. Operation objects $(\mathrm{k} \in \mathrm{IO})$ are relevant to operation phases, operations and work operations. Process objects $(\mathrm{j} \in \mathrm{IP})$ are relevant to processes. Activity objects $((\mathrm{q}, \mathrm{s}) \in \mathrm{IA}$, where $\mathrm{q} \in \mathrm{IO}$ is the main object of the operation) are relevant to active stages of operations. For each level of a production control system the objects may be classified to one of three separate sets:

$$\mathrm{II} = \mathrm{IM} \cup \mathrm{ID} \cup \mathrm{IG}$$

where

IM – a set of **manufacturing items**, i.e. manufacturing products and resources,
ID – a set of **data items**, especially messages and decisions,
IG – a set of **readiness states** of organizational units.

For instance:

$$IO = IOM \cup IOD \cup IOG$$

where

IOM – a set of **manufacturing operation items**, i.e. manufacturing operation products and resources,
IOD – a set of **operation data items** related to places between workstations, i.e. data packets that may appear at inputs or outputs of operations.
IOG – a set of **workstation readiness states**, as well as messages and decisions of them, which may be coded in a different way, for instance IOG = $\{1\}\in\{2\}\in\{5\}\in\{6\}\in\{7\}\in\{8\}$

where

k = 1 – message of the workstation readiness state,
k = 2 – decision of the workstation readiness state,
k = 5 – workstation readiness state after reequipment,
k = 6 – workstation readiness state after a breakdown occurrence,
k = 7 – workstation readiness state after repair,
k = 8 – workstation readiness state after maintenance.

Most applications of Petri Nets to discrete manufacturing processes deal with structural problems of resource allocation [3, 4], e.g. deadlock avoidance. The word "resources" in these models stands for renewable resources, because nonrenewable resources do not cause any structural problems. In practice workstations and workcells, instead of individual resources, are allocated to operations and processes. What's more, organizational units must be properly prepared before allocation. Distinction between resources and organizational units and assigning preparatory operations to the last ones are significant qualities of the IP-nets. Renewable and nonrenewable resources are considered to be production process objects. Renewable resources may be fixed, replaceable or free. Fixed resources (e.g. machines) are associated to their workstations for ever and are not displayed in the models. Replaceable resources (e.g. tools) are associated to workstations for the time intervals between reequipment operations. Free resources may be immediately allocated to particular organizational units. If an organizational unit consists of free resources only, then it is called a "virtual" one.

Petri net models of manufacturing process control systems usually do not display control algorithms themselves. For instance, the scheduling is presented as making decisions concerning the input sequence of products into a given "resource" and the selection of a "resource" to perform an operation on a given product [4]. The schedule is determined by allocation of firing sequences of incident transitions to each place, which needs these decisions. One can only imagine that the relevant control system contains the control operations which generate messages resulting from the sequences and send them to "decision stages" that are additional inputs to controlled transitions. On the other side, assumptive control operations receive messages from "information stages" that are additional outputs from certain transitions (see the example below). The general CPN definition contains the guard function (chapter 2), which enables formal description of interactions between the CPN model and outside systems. In the IP-nets there are no guards, but instead the control operations and the data item stages are displayed explicite. Thus, the entire control system is modeled in terms of Petri nets, what facilitates its investigation, e.g. stability analysis.

4. AS – a finite set of **active stages**, i.e. a set of pairs (a,i) where $a \in II$ is the active stage main product and $i \in UU$ is the organizational unit, in which the active stage is located. In particular for operations $(q, i) \in OP$ "q" is the operation main product ($q \in IO$) and $i \in SN$ is the workstation; for active operation stages $(q, i, a) \in AOS$ "a" is the active stage identifier inside the operation $(q, i) \in OP$; for operation phases $(q, i, f) \in OF$ "f" is the phase identifier inside the operation $(q, i) \in OP$; for work operations $(q, h) \in OR$ "h" is the workcenter ($h \in CR$); for processes $(p, r) \in PR$ or $(p, g, r) \in PR$ "p" is the process main product ($p \in IP$); $r \in RTG$ or $(g, r) \in RTG$ is the routing and $g \in CL$ is the workcell. Furthermore

$$\begin{aligned} &AS \subset II \times UU, && ASM \subset IM \times UU, \\ &AS = AM \cup AD \cup AG, && ASD \subset ID \times UU, \\ & && ASG \subset IG \times UU, \end{aligned}$$

where
ASM – a set of **manufacturing active stages**,
ASD – a set of **data processing active stages**,
ASG – a set of **preparatory active stages**,

In particular

$$\begin{aligned} &OP \subset IO \times SN, && OPM \subset IOM \times SN, \\ &OP = OM \cup OD \cup OG, && OPD \subset IOD \times SN, \\ & && OPG \subset IOG \times SN, \end{aligned}$$

where
OPM – a set of **manufacturing operations**,
OPD – a set of **data processing operations**,
OPG – a set of **preparatory operations**.

5. PS – a finite set of **passive stages**, i.e. a set of pairs (s,l) where $s \in II$ is the object and $l \in (UU \cup MM)$ is the organizational unit or the place between workstations, in which the passive stage is located. In particular for stages between operations $(k, l) \in SO$ "k" is the operation item ($k \in IO$) and $l \in (SN \cup MM)$ is the workstation or the place between workstations; for stages between processes $(j, l) \in SP$ "j" is the process item ($j \in IP$) and $l \in (CL \cup MM)$ is the workcell or the place

between workstations; for passive operation stages $(q,i,s) \in POS$ "s" is the passive stage identifier inside the operation $(q,i) \in OP$. Furthermore,

$$\begin{array}{ll} PS \subset II \times (UU \cup MM), & PSM \subset IM \times MM, \\ PS = PSM \cup PSD \cup PSG, & PSD \subset ID \times MM, \\ & PSG \subset IG \times UU, \end{array}$$

where

PSM – a set of **manufacturing passive stages**,

PSD – a set of **data processing passive stages**,
PSG – a set of **preparatory passive stages**,

In particular

$$\begin{array}{ll} SO \subset IO \times (SN \cup MM), & SOM \subset IOM \times MM, \\ SO = SOM \cup SOD \cup SOG, & SOD \subset IOD \times MM, \\ & SOG \subset IOG \times SN, \end{array}$$

where
SOM – a set of **manufacturing operation item stages,**
SOD – a set of **data processing operation item stages,**
SOG – a set of **workstation readiness stages,**

To avoid confusions between identifiers of organizational units and places between workstations it is assumed they have a common set of identifying numbers.

6. IN – an **active stage input** relation, i.e. a set of pairs ((a,i),(s,l)), such that $(a,i) \in AS$ is an active stage and $(s,l) \in PS$ is one of its input passive stages. Hence

$$IN \subset AS \times PS.$$

For instance $INPM \subset PRM \times SPM$ is a relation **of material inputs to manufacturing processes**, i.e. a set of pairs ((p,r),(j,l)), such that $(p,r) \in PRM$ is a manufacturing process and $(j,l) \in SPM$ is one of its inventory locations. The well known **Bill of Material** from the MRP II method [2], which is a set of pairs $(p,j) \in BOM$ such that $p,j \in IP$ are items of the Inventory Item File (ITM), is an aggregated INPM relation, for which "p" represent main process products, i.e. "parent items", and "j" represent their input materials, i.e. "component items".

7. MO – an **active stage main output** function, which assigns to each active stage $(a,i) \in AS$ its main output passive stage $(s,l) \in PS$. To display the tabular expression of the MO it is convenient to present it (and other functions from the IP-nets definition) as a relation, that is a set of pairs ((a,i),(s,l)),

$$MO \subset AS \times PS.$$

8. AO – an **active stage accessory output** relation, i.e. a set of pairs ((a,i),(s,l)), such that $(a,i) \in AS$ is an active stage and $(s,l) \in PS$ is one of its accessory output passive stages. Hence

$$IN \subset AS \times PS.$$

5. Example. IPN of a stamping control system

The IPN graph of a follow-up production control system [7] with 2 presses designed for stamping 3 car body elements is displayed in the fig. 3. Detailed control decisions follow executive orders, which – in turn – follow shop orders from an MRP II system [6]. The system encompasses 4 manufacturing, 8 preparatory and 3 control operations. For the example simplicity each operation consists of only one phase. (Examples of operations with several phases are shown in [8].) The meaning of particular operations and stages between operations is presented in the tables below.

SN – Workstations

i	Description
1	press line control unit
11	press 1
12	press 2
16	press 1 control unit
17	press 2 control unit

MM – Places between Workstations

l	Description
1	sheets stock area
2	place behind the press 1
3	warehouse
7	gantry rest place

OP – Operations

q	**i**	Description
1	16	detailed control of the press 1
1	17	detailed control of the press 2
2	1	executive planning of the press line
5	11	press 1 reequipment
5	12	press 2 reequipment
6	11	press 1 waiting for breakdown
6	12	press 2 waiting for breakdown
7	11	press 1 repair
7	12	press 2 repair
8	11	press 1 maintenance
8	12	press 2 maintenance
11	11	stamping body part 1
21	12	stamping body part 2
31	11	stamping operation 1 of body part 3
32	12	stamping operation 2 of body part 3

SO – Stages between Operations

SOG		SOM		SOD		SOD	
k	**i**	**k**	**l**	**k**	**l**	**k**	**l**
1	11	11	3	111	3	321	3
1	12	21	3	112	3	322	3
2	11	31	2	113	3	323	3
2	12	32	3	211	3	405	1
5	11	40	1	212	3	406	1
5	12	50	1	213	3	407	1
6	11	70	7	310	2	505	1
6	12			311	2	506	1
7	11			315	2	507	1
7	12						
8	11						
8	12						
9	11						
9	12						

IO – Operation Objects

k	Description
1	message of a workstation readiness state
2	decision of a workstation readiness state
5	workstation readiness state after reequipment
6	workstation readiness state after a breakdown occurrence

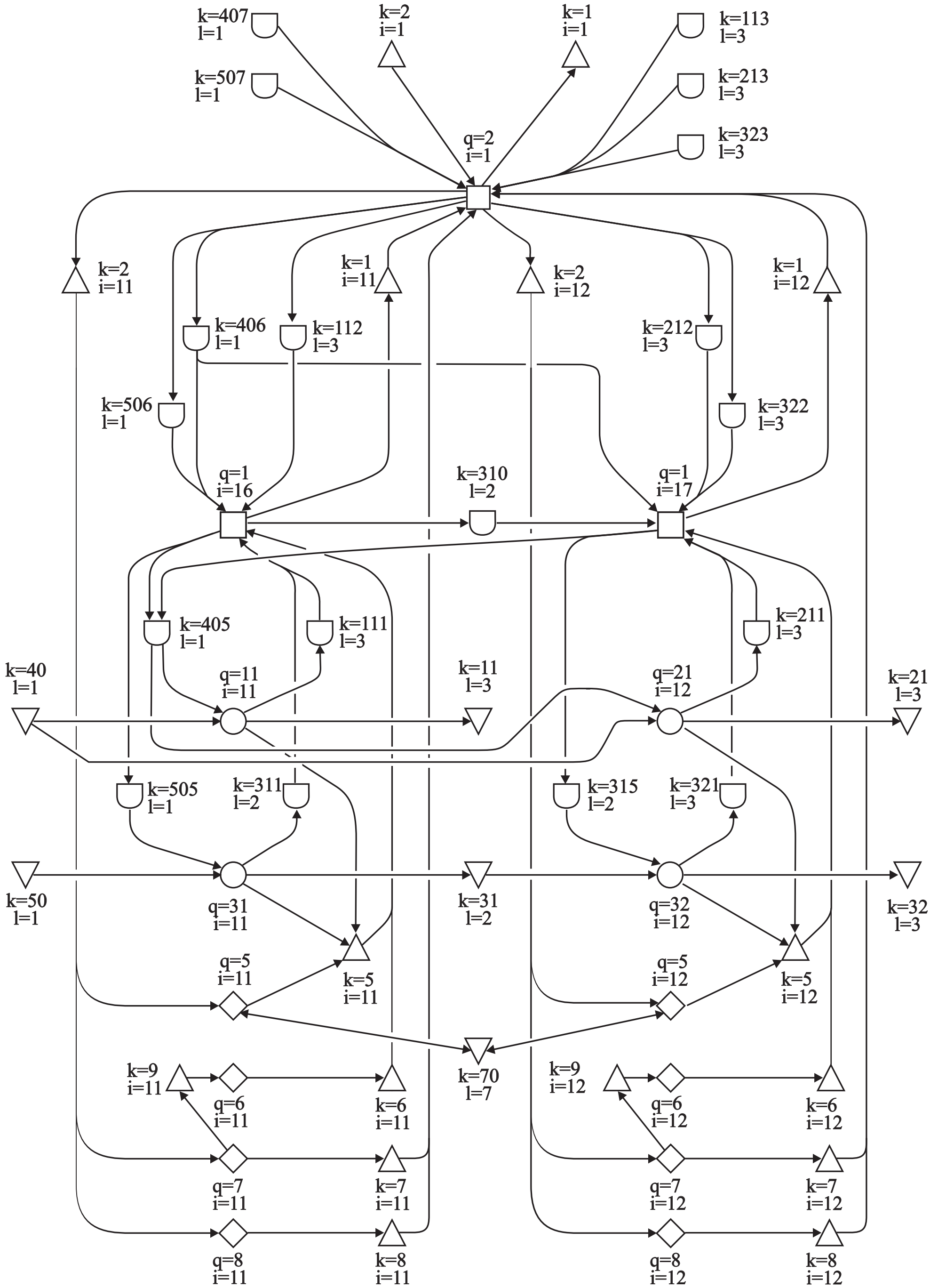

Figure 3. IPN graph for a stamping process

7 workstation readiness state after repair
8 workstation readiness state after maintenance
9 message of a workstation readiness state after repair
11 body part 1 in the warehouse
21 body part 2 in the warehouse
31 body part 3 after stamping operation 1
32 body part 3 in the warehouse
40 sheet for body parts 1 and 2
50 sheet for body part 3
70 gantry in the rest place
111 detailed report of stamping body part 1
112 waiting executive order of stamping body part 1
113 waiting shop order of stamping body part 1
211 detailed report of stamping body part 2
212 waiting executive orders of stamping body part 2
213 waiting shop orders of stamping body part 2
310 detailed report of body part inventory between stamping operations
311 detailed report of stamping operation 1 of body part 3
315 waiting detailed task of body part 3 withdrawing for operation 2
321 detailed report of stamping operation 2 of body part 3
322 waiting executive orders of stamping body part 3
323 waiting shop orders of stamping body part 3
405 waiting detailed task of sheet withdrawing for body parts 1 or 2
406 waiting executive orders of sheet withdrawing for body part 1 or 2
407 waiting shop orders of sheet withdrawing for body part 1 or 2
505 waiting detailed task of sheet withdrawing for body part 3
506 waiting executive order sof sheet withdrawing for body part 3
507 waiting shop orders of sheet withdrawing for body part 3

6. Executive objects and agents

Like for all CP-nets, an IP-net state depends on attribute values of tokens remaining in its places. In shortened description all feasible combinations of attribute values of a given object in a passive stage are represented by its variants. Furthermore, for a given agent in an active stage there are three independent kinds of variants, namely a workstation readiness variant, a variant of its main item and a variant of its free resource allocation. It is assumed that other features of objects and agents are functions of mentioned above attributes. Thus, in the general IPN definition a description of the IPN graph structure should be supplemented by following sets:

9. RV – a finite set of **readiness variants of organizational units**, i.e. a set of triplets (s,i,w), where $(s, i) \in PSG$ is a preparatory passive stage, and "w" is one of feasible readiness variants of an organizational unit; RV(s,i) is a set of readiness variants in the stage (s,i).
10. IV – a set of **variants of multi-variant products and resources**, i.e. a set of pairs (s,u), where $s \in IM$ is a multi-variant manufacturing item and "u" is one

of its variants. IV(s) is a set of variants of the item "s". Different variants of a manufacturing item exist, if its features have different feasible options. Generally,

$$\underset{s \in IM}{\forall} IV(s) = \underset{(s,\varphi) \in FF(s)}{P} FO(s,\varphi)$$

where ***P*** is the symbol of the generalized Cartesian product and

FF – a finite set of **facultative features** of manufacturing items, i.e. a set of pairs (s,φ), where "s" is an item and "φ" is its facultative feature,

FO – a finite set of **options of facultative features**, $FO \ni (s,\varphi,o), \text{where} (s,\varphi)$ is a feature and "o" is one of its options,

$$FF(s) = \{(j,\varphi) \in FF | j = s\},$$
$$FO(s,\varphi) = \{(j,f,o) \in FO | (j,f) = (s,\varphi)\}.$$

11. DV – a set of **data item variants**, $DV \ni$ (s,u), $DV(s) = \{(s,u) \in$ DV | j = s },

$$\underset{s \in ID}{\forall} DV(s) = \underset{(V,s,id) \in DA(s)}{P} VV(V,s,id)$$

DA – a finite set of **data item attributes**, i.e. a set of triplets (V,s,id), where "s" is a data item, "V" is the name of one of its attributes and "id" is the identifier (an index or a tuple of indices or an empty symbol), which is needed in the case of a multidimensional attribute variable V,

VV – a finite set of **data item attribute values**, i.e. a set of 4-tuples (V,s,id,v), where v is one of feasible values of the attribute (V,s,id),

$$DA(s) = \{(V,j,id) \in DA | j = s\},$$
$$VV(V,s,id) = \{(X,j,i,v) \in VV | (X,j,i) = (V,s,id)\}.$$

12. IE – a set of **executive objects**, i.e. a set of triplets (s,l,u), where (s,u) is an object variant, feasible for a passive stage $(s,l) \in PS$.

$$IE(s,l) = \{(s,l,u) \in IE | (s,u) \in IV(s)\}, \qquad \text{for} (s,l) \in PSM,$$
$$IE(s,l) = \{(s,l,u) \in IE | (s,u) \in DV(s)\}, \qquad \text{for} (s,l) \in PSD,$$
$$IE(s,l) = RV(s,l), \qquad \text{for} (s,l) \in PSG,$$

For instance a particular variant "u" of a multi-variant product "k", stored in a particular buffer "l" is an executive operation object. Other example is a particular readiness variant "w" of a particular workstation "i" after its reequipment (k=5 in the above example).

13. AE – a set of **executive agents**, i.e. a set of 5-tuples (a,i,w,v,r), where "r" is a variant of free resource allocation to an active stage (a,i), which is feasible in the organizational unit readiness variant (i,w) for the variant "v" of the main item "s" (such that $\exists (s,l) = MO(a,i)$) of the active stage (a,i). A data processing active stage has only one readiness variant and one resource set.

$$AE(a,i) = \{(s,l,w,v,r) \in AE | (s,l) = (a,i)\}.$$

In terms of CPN the executive objects and the executive agents are token elements [1] that may appear in active and passive stages respectively.

7. The IPN state

Let TT is a set of **time values**, $t \in TT$, among which t_0 is the **IPN start time**, and T: AE $\rightarrow$ TT is a function of **executive stage action durations**.

A **state** $X(s,l,t)$ **of the passive stage** (s,l) at the time "t" is an untimed multi-set over the executive objects set of this passive stage:

$$\underset{(s,l,t)\in PS\times TT}{\forall} \mathrm{Type}(X(s,l,t)) = IE(s,l)_{MS},$$
$$\underset{(s,l,t)\in PS\times TT}{\forall} X(s,l,t) = \sum_{(s,l,u)\in IE(s,l)} X(s,l,u,t)'(s,l,u).$$

If the size |X(s,l,t)| = 0, then the passive stage (s,l) at the time "t" is in the empty state.

A **stage action** (a,i,n), where (a,i) – an active stage, n – a stage action number at the stage (a,i), is a realization of a determined executive agent (a,i,w,v,r), in a determined time interval between S(a,i,n), C(a,i,n), i.e. start and completion times of the action.

For active stages a **state** $A(a,i,t)$ **of the active stage** (a,i) at the time t is a timed multi-set over the executive agents set of this active stage:

$$\underset{(a,i,t)\in AS\times TT}{\forall} \mathrm{Type}(A(a,i,t)) = AE(a,i)_{TMS}$$

$$\underset{(a,i,t)\in AS\times TT}{\forall} A(a,i,t) = \sum_{(a,i,w,v,r)\in AE(a,i)} A(a,i,w,v,r,t)'(a,i,w,v,r)@[t_C(a,i,w,v,r,t)]$$

where t_C is the completion time, which is expected at the current time "t" for one of stage actions running in the active stage (a,i). The attribute t_C satisfies the following formula:

$$t_C(a,i,w,v,r,t) = \begin{cases} t + T(a,i,w,v,r), & \text{if } A(a,i,w,v,r,t) = 1 \text{ and } A(a,i,w,v,r,t\text{-}1) = 0 \\ 0, & \text{if } A(a,i,w,v,r,t) = 0 \text{ and } A(a,i,w,v,r,t\text{-}1) = 1 \\ t_C(a,i,w,v,r,t\text{-}1), & \text{in other case} \end{cases}$$

For a given manufacturing or preparatory activity, operation phase or operation, at a given time only one realization of an executive agent may exist. Thus, the size $|A(a,i,t)| \leqslant 1$ and the equation A(a,i,w,v,r,t)=1 may be true only for one, selected at a given time, executive agent (a,i,w,v,r), whereas for all of the others A(a,i,w,v,r,t)=0. For work operations and processes it is not true, because more than one executive agent may exist in such an active stage at the same time. It is not true also for data processing operations, because such an operation may generate many decisions of the same type but with different time stamps. If the size |A(a,i,t)|=0, then the active stage (a,i) at the time "t" is in the idle state.

8. Expressions describing changes of the IPN state

At the stage action start time the executive agent "appears" in the given active stage and some executive objects are removed from its input passive stages (fig.2). At the completion time the executive agent "is removed" from the active stage and some executive objects are added to its output passive stages. Quantities of the executive objects that are removed or added to passive stages, as well as the choice of an executive agent at the start time, may be calculated from expressions that are assigned to relevant arcs of the TCPN, in other words – to relevant pairs of active and passive IPN stages and to active stages themselves (fig.2). Therefore the general IPN definition from chapters 4 and 6 should be supplemented by following functions:

14. EI – an **input expression** function,
EM – a **main output expression** function,
EO – an **accessory output expression** function,
such that

$$\begin{array}{ll}\forall((a,i),(s,l)) \in IN: & [Type(EI((a,i),(s,l))) = IE(s,l)_{MS}] \\ \forall((a,i),(s,l)) \in MO: & [Type(EM((a,i),(s,l))) = IE(s,l)_{MS}] \\ \forall((a,i),(s,l)) \in AO: & [Type(EO((a,i),(s,l))) = IE(s,l)_{MS}]\end{array}$$

which means that every value of the expression related to an arc between an active stage and one of its incident passive stages is an untimed multi-set over the executive objects set of this passive stage.

Denoting the value of the expression EI((a,i),(s,l)) at a start time "t" of the active stage (a,i) by U(a,i,s,l,t), the following pattern for the state decrease in the input passive stage (s,l) is obtained:

$$U(a,i,s,l,t) = \sum_{(s,l,u)\in IE(s,l)} U(a,i,s,l,u,t)'(s,l,u)$$

Analogously, denoting the values of the expressions EM((a,i),(s,l)), EO((a,i),(s,l)) at a completion time "t" of this active stage by Z(a,i,s,l,t) and Y(a,i,s,l,t), respectively, the state increase in the output item place (s,l) must satisfy the following pattern:

$$Z(a,i,s,l,t) = \sum_{(s,l,u)\in IE(s,l)} Z(a,i,s,l,u,t)'(s,l,u)$$

$$Y(a,i,s,l,t) = \sum_{(s,l,u)\in IE(s,l)} Y(a,i,s,l,u,t)'(s,l,u)$$

15. ES – a **start expression** function,
EC – a **completion expression** function,
such that

$$\begin{array}{ll}\forall(a,i) \in AS: & [Type(ES(a,i)) = AE(a,i)_{TMS}] \\ \forall(a,i) \in AS: & [Type(EC(a,i)) = AE(a,i)_{TMS}]\end{array}$$

which means that every value of the expression describing a state increase or decrease in an active stage is a timed multi-set over the executive agents set of this active stage.

Denoting the value of the expression ES(a,i) at a start time "t" of the stage action (a,i) by G(a,i,t), the following pattern is obtained:

$$G(a,i,t) = \sum_{(a,i,w,v,r)\in AE(a,i)} G(a,i,w,v,r,t)'(a,i,w,v,r)@[t + T(a,i,w,v,r)]$$

For a given manufacturing or preparatory operation at a given time "t" only one realization of only one executive agent may start. In this case the start expression should point out alternatively the idle state of the operation or the executive agent selected for it at the start time. Then the coefficient of the selected executive agent in the multi-set of the state increase G(a,i,w,v,r,t)=1, whereas coefficients of other executive agents are equal to 0. For control operations it is not always true, because sometimes more than one piece of a given executive agent may be generated with different time stamps. The decrease of an operation state at the completion time "t" is described by the formula:

$$H(q,i,t) = 1'(q,i,w,v,r)@[t],$$

where (a,i,w,v,r) is the executive agent completed at the time t. The general formula for the value of the expression EC(a,i) is analogous to the pattern mentioned above for ES(a,i).

9. Conclusions

Transitions, the node function and the guard function, as well as conditions related to types of expression variables in the general TCPN definition, have been excluded from the IPN definition. Transitions are identified by active stages, where they are hidden in. The node function is not needed because places related to specific arcs are already pointed out as passive stages in relationships IN, MO, AO and – for hidden arcs between transitions and active stages - by active stages themselves. Guard expressions and the guard function are useless because of assumption that all actions influencing the state of an IPN must be represented by expressions related to active stages or to arcs between them and passive stages. All variable types are classified as several color types for active and passive stages, then there is no need to verify types of expression variables. Thus, IP-nets are structurally simpler than TCP-nets.

The main practical purpose of Industrial Petri Nets is to facilitate integration of planning and control systems for manufacturing, transport, maintenance etc. on all hierarchical levels, from tactical production planning to control of elementary activities executed in particular machines. They also may be used for formal or experimental comparison of different planning, scheduling and control methods applied to current production control in industrial plants.

It is expected that all (known to the author) discrete production planning and control problems may be formally expressed in terms of IPN. They are a good tool for constructing models of special problems, such as multivariant production management, dynamic products grouping in flexible manufacturing systems or the follow-up production scheduling [7]. Furthermore, they enable a unified formal approach to many well known problems which seem to be totally different, such as operational planning in MRP II systems, activity control of a machine reequipment, synchronization of a flow production line or job shop scheduling. What's more, they facilitate specification of simplifying assumptions, which are made when building models of production planning and control problems.

References

[1] K. JENSEN: Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. Springer–Verlag, 1997.

[2] D.V. LANDVATER and C.D GRAY: MRP II Standard System. Oliver Wight Publications, 1989.

[3] J.O. MOODY J.O and P.J. ANTSAKLIS: Supervisory Control of Discrete Event Systems Using Petri Nets. Kluwer Academic Publishers, 1998.

[4] J.-M. PROTH and XIAOLAN XIE: Petri Nets. A Tool for Design and Management of Manufacturing Systems. J. Wiley & Sons, 1996.

[5] A.–W. SCHEER: CIM (Computer Integrated Manufacturing) – Towards the Factory of the Future. Springer–Verlag, 1994.

[6] A.–W. SCHEER: Business Process Engineering. Reference Models for Industrial Enterprises. Springer–Verlag, 1994.

[7] M. ZABOROWSKI: The Follow-up Scheduling in a Production Control System. *Proc.* 6^{th} *IFAC Workshop on Intelligent Manufacturing Systems*, Poznań, Poland, Elsevier's IFAC Publications, (2001), 161-166.

[8] M. ZABOROWSKI: A Petri Net Based Model of Discrete Production Operations. *Proc.* 3^{rd} *Int. Symp. on Robotics and Automation*, Toluca, Mexico, (2002), 646-653.

[9] M. ZABOROWSKI: A Petri Net Based Model for Control of Complex Manufacturing Operation Systems. *Proc. IASTED Int. Conf. on Applied Modelling and Simulation*, Cambridge, USA, (2002), 381-388.