HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY

Faculty of Computer Science & Engineering

Assignment

Petri Networks Modeling

Instructor: Mr. Nguyen Tien Thinh
Mr. Nguyen An Khuong

Class: ***CC02***
Cao Hoang Kiet - 2053165
Cao Tuan Kiet - 2053166
Tran Le Trung Chanh - 2052403
Tran Minh Thuc - 2053486
Class: ***CC05***
Dang Cong Khanh - 2053105
Bui Vo Cong Phu - 2053326

Ho Chi Minh City, November 2021

Contents

1 Member list & Workload 4

2 Preliminary knowledge 5
- 2.1 Petri networks - Background 5
  - 2.1.1 The Art of Modeling - motivated from Operation Research 5
  - 2.1.2 Formal definitions 6
  - 2.1.3 On Enabled transition and Marking changes 12
  - 2.1.4 Important usages of **Petri net** via explaining Figure 7 14
- 2.2 Petri networks - Behaviors 17
  - 2.2.1 From Firing, Reachability to Labeled Petri net 17
  - 2.2.2 Representing Petri Nets as Special Transition Systems 21
- 2.3 Petri networks - Structures and Basic Problems 21
  - 2.3.1 Causality 21
  - 2.3.2 Concurrency 22
  - 2.3.3 Synchronization 22
  - 2.3.4 Effect of Concurrency 23
- 2.4 Petri nets - Assignment on modelling 24
  - 2.4.1 Dynamic of Petri nets via Special properties 24
  - 2.4.2 Modeling by Petri networks - Problem 25
- 2.5 Reviewed problems 26

3 Application: Consulting Medical Specialists 39

List of Figures

1 Carl Adam Petri (1926 - 2010) 6
2 A small size transition system 10
3 A Petri net for the process of an X-ray machine. 11
4 The first three markings in a process of the X-ray machine. . . 12
5 The marking **before** firing the enabled transition **t**. 13
6 The marking **after** firing the enabled transition **t**. 13
7 A marked **Petri net** with one initial token 14
8 A Petri net system and its reachability graph. 20
9 Causality in a **Petri Net** 21
10 Concurrency in a **Petri Net** 22
11 Synchronization a **Petri Net** 22
12 An example for k**-bounded** 24
13 A simple **Petri net**, with only two transitions 27
14 A Petri net model of a business process of an X-ray machine . 29
15 An improved Petri net for the business process of an X-ray machine 30
16 The marking of the improved Petri net for the working process of an X-ray room after transition **enter** has fired. 31
17 Transition system for figure 7 32
18 A Petri net with small numbers of places and transitions . . . 33
19 A Petri net with small numbers of places and transitions when we remove place p_1 and its adjacent arcs 35
20 A Petri net with small numbers of places and transitions when we remove place p_3 and its adjacent arcs 35
21 A Petri net allows concurrency 36
22 The transition systems of a specialist (left) and a patient (right) 39
23 A Petri net modeling the state of the specialist. 40
24 A transition system when $M_0 = [1, 0, 0]$ 41
25 A transition system when $M_0 = [1, 1, 0]$ 41
26 A Petri net N_{Pa} that has five patients at place **wait**, no patient at place **inside** and one patient at place **done**. 43
27 The superimposed (merged) **Petri Net** model $\mathrm{N} = N_S \oplus N_{Pa}$ that has four patients **waiting**, one patient at **done** and one specialist at **free**. 44
28 The superimposed (merged) **Petri Net** model $\mathrm{N} = N_S \oplus N_{Pa}$ that has three patients **waiting**, one patient at **done** and one specialist at **free**. 45

29 A **Petri net** for medical examination and treatment of two specialists 48

1 Member list & Workload

No.	Fullname	Student ID	Problem	Percentage
1	Cao Hoang Kiet	2053165	Preliminary knowledge Problem 6,7	17%
2	Cao Tuan Kiet	2053166	Preliminary knowledge Problem 6,7	17%
3	Tran Le Trung Chanh	2052403	Preliminary knowledge Problem 1	17%
4	Tran Minh Thuc	2053486	Preliminary knowledge Problem 6,7	17%
5	Dang Cong Khanh	2053105	Preliminary knowledge Problem 2,3,4,5	16%
6	Bui Vo Cong Phu	2053326	Preliminary knowledge Problem 2,3,4,5	16%

2 Preliminary knowledge

2.1 Petri networks - Background

2.1.1 The Art of Modeling - motivated from Operation Research

2.1.1.1 Operation research(OpRe)

Operation research (OpRe), is a branch of management science heavily relying on modeling. Here a variety of mathematical models ranging from

- Deterministic modeling: integer linear programming,dynamic programming,to
- Stochastic modeling: Markov chains,queueing models,to stochastic dynamic programming,and
- Mixed type one:as simulation [discrete event simulation,MCMC simulation]

2.1.1.2 Goal

Models are used to reason about processes (redesign) and to make decisions inside processes (planning and control).The main purpose of two type of models:

- The models used in operations management are typically tailored towards a particular analysis technique and only used for answering a specific question.
- Process models in Business Process Management typically serve multiple purposes.

However,it caused some difficulties.Therefore, **Petri Net** was created by Carl Adam Petri.
IMPACTS: Petri nets nowadays have brought engineers a breakthrough in their treatment of discretely controlled systems. Petri nets are a key to solve the design problem, as this is the first technique to allow for a unique description, as well as powerful analysis of **discrete control systems**.

2.1.1.3 Information of Carl Adam Petri (1926 - 2010):

Carl Adam Petri (1926- 2010), the first computer scientist to identify concurrency as a fundamental aspect of computing (sketched largely in his seminal PhD thesis, title Communication with Automata,submitted to the Science Faculty of Darmstadt Technical University in 1962, where in fact he outlined a whole new foundations for computer science). Petri's father was a serious scholar. He had a PhD in mathematics and had met Minkowski and Hilbert.

Figure 1: *Carl Adam Petri (1926 - 2010)*

2.1.1.4 Petri Net

A **Petri Net** is a graphical tool [**a bipartite graph** consisting of places and transitions] for the description and analysis of concurrent processes which arise in systems with many components (distributed systems). The graphics, together with the rules for their coarsening and refinement, were invented in August 1939 by Carl Adam Petri.

2.1.2 Formal definitions

2.1.2.1 Definition

Definition 2.1: Transition system or State transition system

A transition system is a triplet $TS = (S, A, T)$ where:

- S is the set of states.
- $A \subseteq \mathcal{A}$ is the set of activities (often referred to as actions).
- $T \subseteq S \times A \times S$ is the set of transitions.

In which:

- $S^{start} \subseteq S$ is the set of initial states (sometimes referred to as 'start' states).
- $S^{end} \subseteq S$ is the set of final states (sometimes referred to as 'accept' states).

For most practical applications the state space S is finite.

2.1.2.2 Goal

Transition systems are used in the process model to decide which activities need to be executed and the executing order. Activities can be executed sequentially, they can be optional or concurrent, also repeated execution of the same activity may be possible.

2.1.2.3 Behavior

A transition system can be studied and expressed through its net structure and dynamic. The transition starts in one of the initial states (S^{start}). Any path in the graph starting in such a state corresponds to a possible execution sequence.

- A path terminates successfully if it ends in one of the final states (S^{end}).
- A path deadlocks if it reaches a non-final state without any outgoing transitions.

(Note that the absence of deadlocks does not guarantee successful termination)

Petri Net is a bipartite directed graph directed graph N of *places* and *transitions*.

Definition 2.2

A Petri net is a triplet $N = (P, T, F)$ where:

- P is a finite set of places.
- T is a finite set of transition such that $P \cap\ T = \emptyset$
- $F \subseteq (P \times T) \cup (T \times P)$ is a set of directed arcs, called the flow relation

1. A token is a special transition node, being graphically rendered as a black dot •. The symbolic tokens generally denote elements of the real world. Places can contain tokens, and transitions **cannot**.
2. A transition is enabled if each of its input places contains a **token**. [Example: node "enter" in figure 3 has input places "wait" and "free".] An **enabled transition** can fire, thereby consuming (energy of) one token [e.g., node "enter" in figure 3] from each input place and producing at least one token for each output place next.
3. A marking is a distribution of tokens across places. A marking of net N is a function $m : P \implies N$ assigning to each place $p \in P$ the number $m(p)$ of tokens at this place. [e.g. $m(wait) = 3$.]
Denote $M = m(P)$ the range of map m, viewed as a multiset.
4. A marked Petri net is a pair (N, M) where $N = (P, T, F)$ is a Petri net and where M is a multi-set (or bag, defined generally in Equation 5.1) over P denoting the marking of the net.
   - We write the set of all multisets over P as $\mathcal{M}(P)$ or $\mathcal{M}$ for short.
   - The set of all marked Petri nets is denoted $\mathcal{N}$.

NOTES on using transition system:

1. **Any process model with executable semantics** can be mapped onto a transition system. Therefore, many notions defined for transition systems can easily be translated to higher-level languages such as Petri nets...
2. Transition systems, however, are simple but have problems expressing concurrency succinctly, as 'state explosion'. But a Petri Net can be used much more compactly and efficiently.

3. Indeed, suppose that there are n parallel activities, i.e., all n activities need to be executed but any order is allowed.
There are n! possible execution sequences. The transition system requires 2^n states and $n \times 2^{n-1}$ transitions. When $n = 10$, the number of reachable states is $2^n = 1024$, and the number of transitions is $n \times 2^{n-1} = 5120$. A **Petri Net** needs only 10 transitions and 10 places to model the 10 parallel activities.

On the set of all multi-sets $\mathcal{M}$ over a domain $\mathcal{D}$
Given a finite domain $D = \{x_1, x_2, ..., x_k\}$, a map $X : D \Rightarrow \mathbb{N}$ defines a multi-set on D as follow: for each $x \in D, X(x) = m$ denotes the number of times x is included in the multi-set,i.e.,

$$M = \{\underbrace{(x_1, ..., x_1)}_{m_1 \text{ times}} ... \underbrace{(x_k, ..., x_k)}_{m_k \text{ times}}\} \tag{1}$$

Evidently, $support(M) \subseteq D$ and we could use multiplicative format for

$$M = \{x_1^{m_1}, x_2^{m_2}, ..., x_k^{m_k}\}$$

and so M is defined with the list $[m_1, m_2, ..., m_k]$. Here frequencies $m_i \geqslant 0, m_i = 0$ means that
x_i does not appear in M, and $support(M)$ consists of different elements in the multi-set M.

Definition 2.3

Let $N = (P, T, F)$ be a Petri net. Elements of $P \cup T$ are called nodes.

- A node x is an input node of another node y if and only if there is a directed arc from x to y (i.e.,$(x, y) \in F$.
- For any $x \in P \cup T$, write $\bullet x = \{|(y, x) \in F\}$ - the preset of x, and $x \bullet \{y|(x, y) \in F\}$- the postset of x.

Question 2.1.
Can you give • c1 =?, c5 • =? in Figure 7 ?
Observe a transition system in Figure 7 we can see:
Answer:
• c1 = {a.f}, c5 • = {g,h,f}

2.1.2.4 Examples

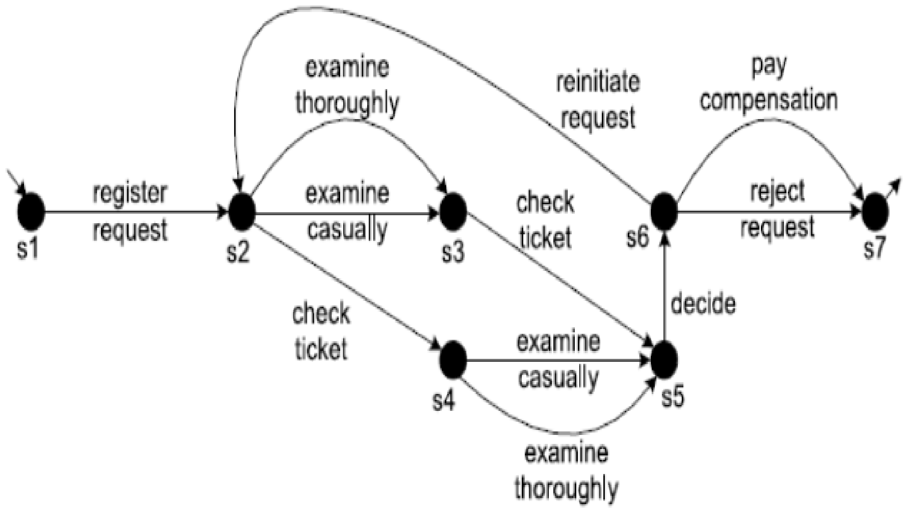


Figure 2: *A small size transition system*

Example 2.1.

Observe a transition system in figure 2 we see the state space $S = \{s_1, s_2, \cdots_7\}$. Here $S^{start} = \{s_1\}$,$S^{end} = \{s_7\}$.Could the reader fill in the set of activities.$A = \{$ register request, examine thoroughly, examine casually,...$\}$,and determine the set $T = \{(s_1,$ register request, $s_2), (s_2,$ examine casually, $s_3), ...\}$ of all transitions?

Solution:

The set of activities $A = \{$ register request, examine thoroughly, examine casually, check ticket, decide, reinitiate request, pay compensation, reject request$\}$.

The set of all transitions $T = \{(s_1$, register request, $s_2)$, $(s_2$, examine casually, $s_3)$, $(s_2$, examine thoroughly, $s_3)$, $(s_2$, check ticket, $s_4)$, $(s_2$, reinitiate request, $s_6)$, $(s_3$, check ticket, $s_5)$, $(s_4$, examine casually, $s_5)$, $(s_4$, examine thoroughly, $s_5)$, $(s_5$, decide, $s_6)$, $(s_6$, reject request, $s_7)$, $(s_6$, pay compensation, $s_7)\}$.

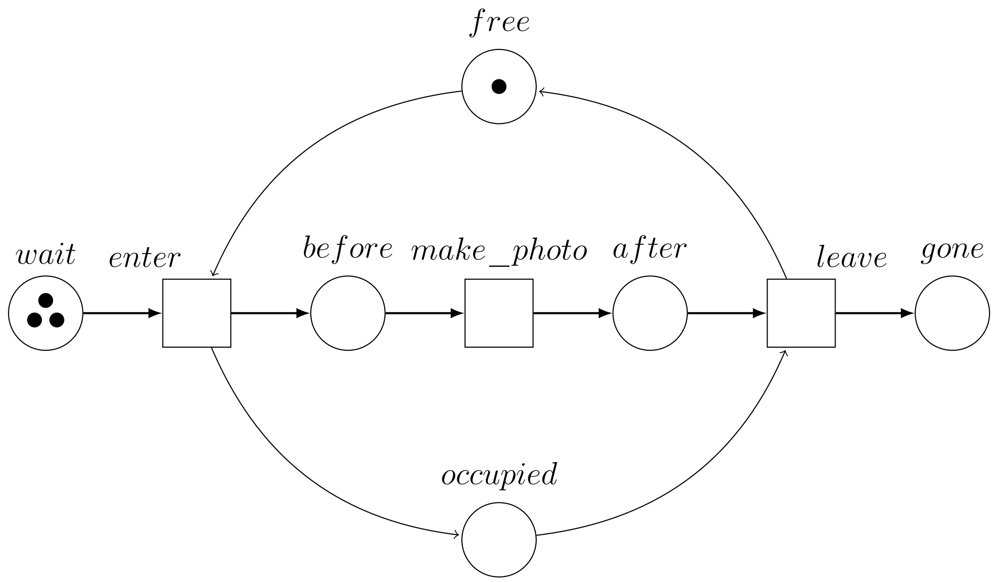


Figure 3: *A Petri net for the process of an X-ray machine.*

Example 2.2. The Petri net in figure 3 has three transitions (drawn as $\square$):$T = \{enter, make - photo, leave\}$. Transition **enter** is **enable** if there is at least one token in place **wait** and at least one token in place **free**. In the making of this net, these conditions are fulfilled . Transition **make-photo** is enable if place **before** holds at least one token. This condition is **not** fulfilled.

List the place P, and give the marking of Petri Net in figure 3.

Determine fully the flow relation $F \subseteq (P \times T) \cup (T \times P)$.

Solution:

Places P $=$ {wait, before, after, free, occupied, gone}

The marking of Petri Net in the figure: $m(wait) = 3, m(free) = 1$.

The Petri net is a directed bipartite graph consisting a finite set of places P, a finite set of transition T with $P \cap T = \emptyset$, $P \cup T \neq \emptyset$, and a flow relation F that represents a set of directed arcs $F \subseteq (P \times T) \cup (T \times P)$, going from places to transition or transition to places.

Flow relations: $F =$ { (wait, enter), (enter, before), (before, make_photo), (make_photo, after), (after, leave), (leave, gone), (leave, free), (free, enter), (enter, occupied), (occupied, leave) }.

Example 2.3.

Figure 4 shows a Petri net with three different markings.

Find the places P and give the transitions T of the net.

Write down completely three different markings in format of lists or tables

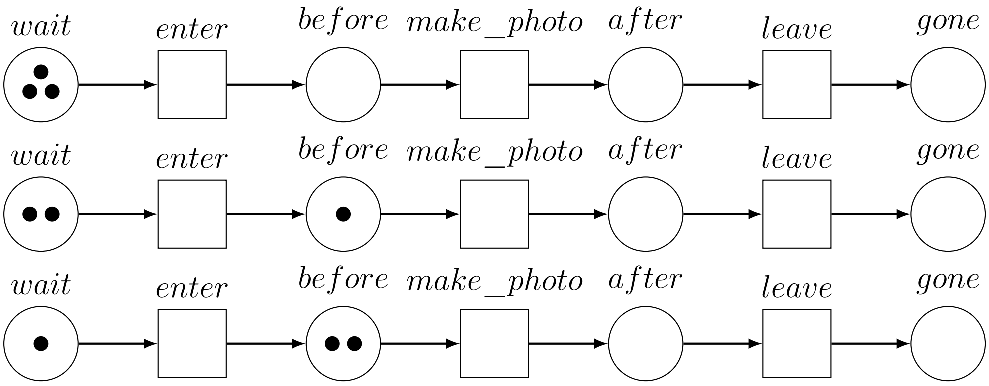


Figure 4: *The first three markings in a process of the X-ray machine.*

Solution:
Observe a transition system in Figure 4 we can see:

- Places $P = \{wait, before, after, gone\}$.
- Transition $T = \{enter, make\_photo, leave\}$.

Three different markings of the process of the X-ray machine in format of lists:

1. Marking: $M = [3.wait, 0.before, 0.after, 0.gone]$
2. Marking: $M = [2.wait, 1.before, 0.after, 0.gone]$
3. Marking: $M = [1.wait, 2.before, 0.after, 0.gone]$

2.1.3 On Enabled transition and Marking changes

2.1.3.1 Enable rule and Firing rule

The execution of a **Petri net** is controlled by the number and distribution of tokens in the **Petri net**. And changing distribution of tokens in places, which may reflect the occurrence of events or executions of many operations. A **Petri net** executes by firing transitions.

Now we will introduce enable rule and firing rule.

1. **Enable rule**: A transition **t** is said to be **enabled** if each input place **p** of transition **t** contains tokens.

2. **Firing rule**: Only enabled transition can fire. The firing of an enabled transition **t** removes from each input place **p** the number of tokens equal to the **weight** of the **directed arc** connecting place **p** to transition **t**. It also deposits in each **output place p** the number of tokens **equal** to the **weight** of the **directed arc** connecting transition **t** to place **p**.

2.1.3.2 Example of enable and firing rule

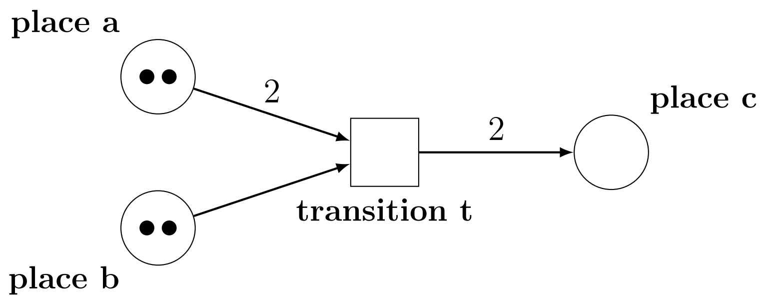


Figure 5: *The marking **before** firing the enabled transition **t**.*

In figure 5, the transition **t** is enabled because both **place a** and **place b** which are the input places of transition **t** also have more than one token in each place.

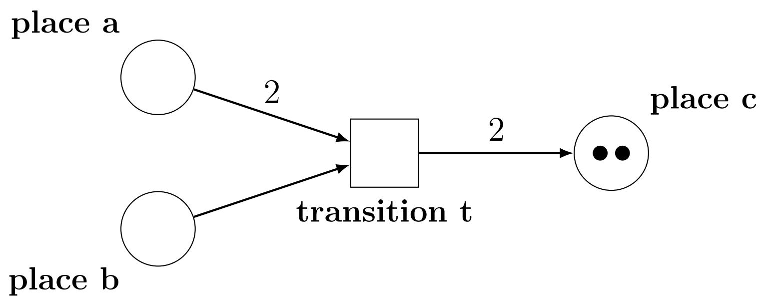


Figure 6: *The marking **after** firing the enabled transition **t**.*

In figure 6, the firing rule for enabled transition **t** will remove 2 tokens in **place a** which is equals to the weight of transition **t**, and put them into the output **place c**.

Note that: The marking in figure 6 now isn't enabled because **place a** doesn't contain any token.

2.1.4 Important usages of Petri net via explaining Figure 7

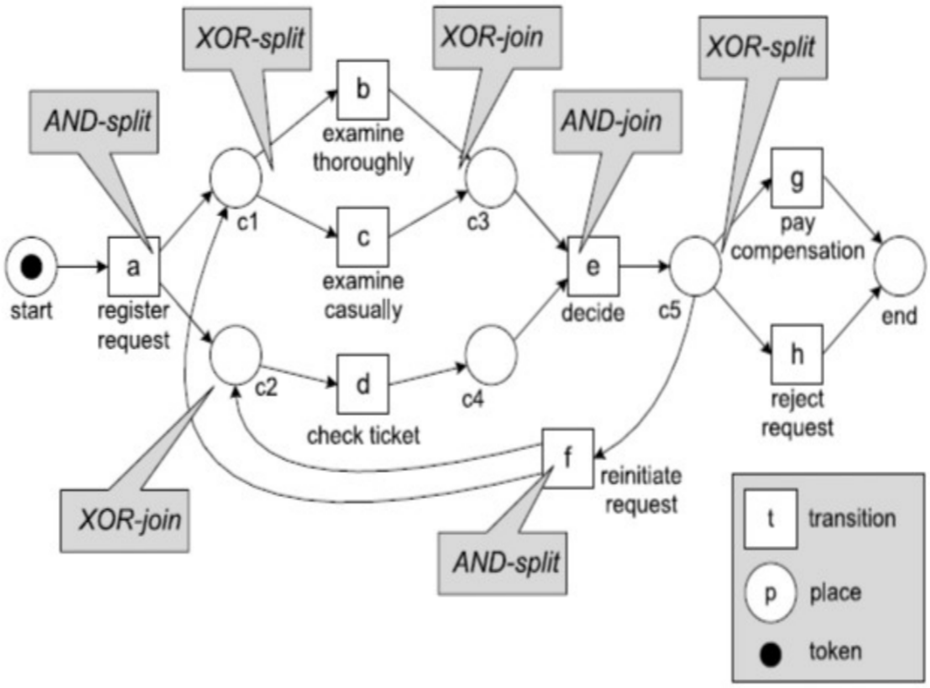


Figure 7: A marked Petri net with one initial token

Important usages of Petri nets: Information systems (IS) and business process modeling (BPM).

Business process (BP)

An organization is a system consisting of humans,machines, materials, buildings, data, knowledge, rules and other means, with a set of goals to be met. Most organizations have, as one of their main goals, the creation of delivery of (physical) products or (abstract) services.

The creation of services and products is performed in business process (BP). A BP is a set of tasks with causal dependencies between tasks.

Task ordering principles The five basic task ordering principles (Figure 7) are

1. Sequence pattern: putting tasks in a linear order;
2. Or-split pattern: selecting one branch to execute;
3. And-split patterns: all branches will be executed;
4. Or-join patterns: one of the incoming branches should be ready in order to continue; and
5. And-join pattern: all incoming branches should be ready in order to continue.

Execution of tasks For the execution of tasks resources are required.
Resources can either be durable or consumable. The first kind is available again after execution of one or more tasks, like a catalyst in a chemical process. Typical examples of this kind are humans, machines, computer systems, tools, information and knowledge. Consumable resources disappear during the task execution. Examples are energy, money, materials, components and data.

The result or output of a task can be considered as resources for subsequent tasks or as final products or services. Two kinds of durable resources are of particular importance: The humans as a resource, called human resources, and information and knowledge, which we will call data resources.

Since human activities are sometimes replaced by computer systems, we use the term agents as a generic term for human and system resources.

Information Systems and Modeling Business Processes

Definition 2.4

To study **Petri net** we first formalize the above concepts.

1. A **business process** consists of a set of activities that is performed in an organizational and technical environment. These activities are coordinated to jointly realize a business goal. Each business process is enacted by a single organization, but it may interact with business processes performed by other organizations.

2. An **information system** is a software system to capture, transmit, store, retrieve, manipulate, or display information, thereby supporting people, organizations, or other software systems.

The awareness of this importance of business processes has triggered the introduction of concept of process-aware information systems. The most notable implementations of the concept of process-aware information systems are workflow management systems. A workflow management system is configured with a **process model**, its graphical visualization is **workflow net**.

SUMMARY 1.

A **Petri net** is a triplet structure (P, T, F). The structure of a Petri net is determined if we know the places P, the transitions T, and the flow relation F of the ways in which they are connected with each other (i.e arcs connecting places and transitions.)

1. A Petri net contains zero or more places. Each place has a unique name, the place label. We cam describe the places of a Petri net by the set P of place labels. We can describe the transitions in a Petri net in the same way. Each transition has a unique name, the transition label.

   We describe the transitions of a Petri net by the set T of transition labels.

2. **Transitions** are the active nodes of a Petri net, because they can change the marking through firing. We therefore name transitions with *verbs* to express action.

   E.g., see node **b**, **c** and **d** in figure 7.

   Places are the passive nodes of a Petri net because they **cannot** change the marking. We name places using nouns, adjectives, or adverbs.

3. In addition to the places and transitions, we must describe the arcs. Like a transition in a transition system, we can represent an arc as an ordered pair (x, y). The set of arcs is a binary relation.

   As a **Petri net** has two kinds of arcs, we obtain two binary relations.

   (i) The binary relation $R_I \subseteq P \times T$ contains all arcs connecting transitions and their input paces.

   (ii) The binary relation $R_O \subseteq T \times P$ contains all arcs connecting transitions and their output places.

   The union $R_I \cup R_O$ represents all arcs of a Petri net. This union is again a relation

   $$F = R_I \cup R_O \subseteq (P \times T) \cup (T \times P)$$

   called the **flow relation**, where $(p, t) \in F$ defines the arc from p to t.

4. **Labeling of Arcs and Transitions**: Arcs and transitions can be labeled with expressions (for instance, **-**, a subtractions, and variables **x** and **y**). These expressions have two central properties:

(i) If all variables in an expression are replaced by elements, it becomes possible evaluate the expression in order to obtain yet another element.

(ii) The variables in these expressions are parameters describing different instance ("modes") of a transition. Such a transition can only occur if its labeling evaluates to the logical value '**true**'.

5. We can summarize the possible roles of tokens, places, and transitions with the following modeling guideline:

   We represent events as transitions, and we represent states as places and tokens.

   We represent the evolving states of a system by the distribution of tokens over the places. Each token in a place is part of the state. A token can model a physical object, information, or a combination of the two, but it can also model the state of an object or a condition.

   In the remainder we only consider Petri nets (special class of workflow nets), to model BPs. For many purposes it is sufficient to consider classical Petri nets, i.e. with '**black**' tokens.

2.2 Petri networks - Behaviors

2.2.1 From Firing, Reachability to Labeled Petri net

Definition 2.5: Firing rule

Let $(N, M) \in \mathcal{N}$ be a marked Petri net with $N = (P, T, F)$ and $M \in \mathcal{M}$

- A transition is enabled if there is at least one token in each of its input places.
- Transition $t \in T$ is enabled at marking M , denoted $(N, M)[t\rangle$, if and only if $\bullet\ \mathrm{t} \leq M$
- The firing rule $\alpha[t\rangle\beta \subseteq \mathcal{N} \times T \times \mathcal{N}$ is the smallest relation satisfying

  $(N, M)[t\rangle \longrightarrow (N, M)[t\rangle(N, (M \backslash \bullet t) \uplus t\bullet)$ where

  for any $(N, M) \in \mathcal{N}$ and any $t \in T$

For example, $(N, [start])[a\rangle(N, [c1, c2])$ means that firing this enabled transition **a** results in marking $[c1, c2]$. And $(N, [c3, c4])[e\rangle(N, [c5])$ denotes that firing this enabled transition **e** results in marking $[c5]$.

Definition 2.6: Firing sequence

Let $(N, M_0) \in \mathcal{N}$ be a marked Petri net with $N = (P, T, F)$

1. A sequence $\sigma \in T^*$ is called a firing sequence of (N, M_0) **if and only if**, for some natural number $n \in M$, there exist markings $M_1, M_2, \cdot, M_n$ and transitions $T_1, T_2, \cdot, T_n$ such that

- $\sigma = (t_1, t_2, t_3, \cdot, t_n) \in T^*$
- And for all i with $0 \leq i < n$, then $(N, M_i)[t_{i+1}\rangle$ and $(N, M_i)[t_{i+1}\rangle(N, M_{i+1})$.

2. A marking M is reachable from the initial marking M_0 **if and only if** there exists a sequence of enabled transitions whose firing leads from M_0 to M.
The set of reachable markings of (N, M_0) is denoted by $[N, M_0\rangle$.

Example 2.4.
In figure 7, the initial marking $M_0 = [start] = [1, 0, 0, 0, 0, 0, 0]$, and we get the marked Petri net (N, M_0)
• The empty sequence $\sigma = \langle\rangle$ - being enabled in (N, M_0) - obviously is a firing sequence of (N, M_0).

• The sequence $\sigma_1 = \langle a, b\rangle$ is also enabled in (N, M_0) and firing σ_1 results in marking [c2,c3]. We can write $(N, [start][ab\rangle(N, [c2, c3]$ or $(N, M_0[_1\rangle(N, [c2, c3]$.

• The sequence $\sigma_2 = \langle a, b, d, e\rangle$ is another possible firing sequence, and we will get $(N, M_0)[\sigma_2\rangle(N, [c5])$ because:

- We define marking like $M = [start, c1, c2, c3, c4, c5, end]$
- Firing transition **a** we will get a marking $M_1 = [0, 1, 1, 0, 0, 0, 0]$
- Firing transition **b** we will get a marking $M_2 = [0, 0, 1, 1, 0, 0, 0]$
- Firing transition **d** we will get a marking $M_3 = [0, 0, 0, 1, 1, 0, 0]$
- Firing transition **e** we will get a marking $M_4 = [0, 0, 0, 0, 0, 1, 0]$

• Is $\sigma = \langle a, c, d, e, f, b, d, e, g\rangle$ a firing? What is the reachable marking M in the output $(N, M_0[\rangle(N, M)$

- σ is a firing.
- The reachable marking M is $(N, M_0[\rangle(N, [end])$.

- The set $[N, M_0\rangle$ has seven reachable markings, which are

1. $M_0 = [1, 0, 0, 0, 0, 0, 0]$ using the empty sequece $\sigma_0 = \langle\rangle$.
2. $M_1 = [0, 1, 1, 0, 0, 0, 0]$ using the sequece $\sigma_1 = \langle a\rangle$.
3. $M_2 = [0, 0, 1, 1, 0, 0, 0]$ using the sequece $\sigma_2 = \langle a, b\rangle$.
4. $M_3 = [0, 1, 0, 0, 1, 0, 0]$ using the sequece $\sigma_3 = \langle a, d\rangle$.
5. $M_4 = [0, 0, 0, 1, 1, 0, 0]$ using the sequece $\sigma_4 = \langle a, b, d\rangle$.
6. $M_5 = [0, 0, 0, 0, 0, 1, 0]$ using the sequece $\sigma_5 = \langle a, b, d, e\rangle$
7. $M_6 = [0, 0, 0, 0, 0, 0, 1]$ using the sequece $\sigma_6 = \langle a, b, d, e, g\rangle$

Definition 2.7: Labeled Petri net

A **labeled Petri net** with $N = (P, T, F, A, l)$ where (P, T, F) is a Petri net.

- $A \in \mathcal{A}$ is a set of activity labels, and the map $l \in \{L : T \longrightarrow A\}$ is a labeling function.
- Use the label $\mathcal{T}$ for a special activity label, called 'invisible'. A transition $t \in T$ with $\mathrm{l(t)} = \mathcal{T}$ is said to be unobservable, silent or invisible .

Example 2.5. (Reachability graph).
Consider the **Petri net** system in Figure 8 modeling the four seasons.

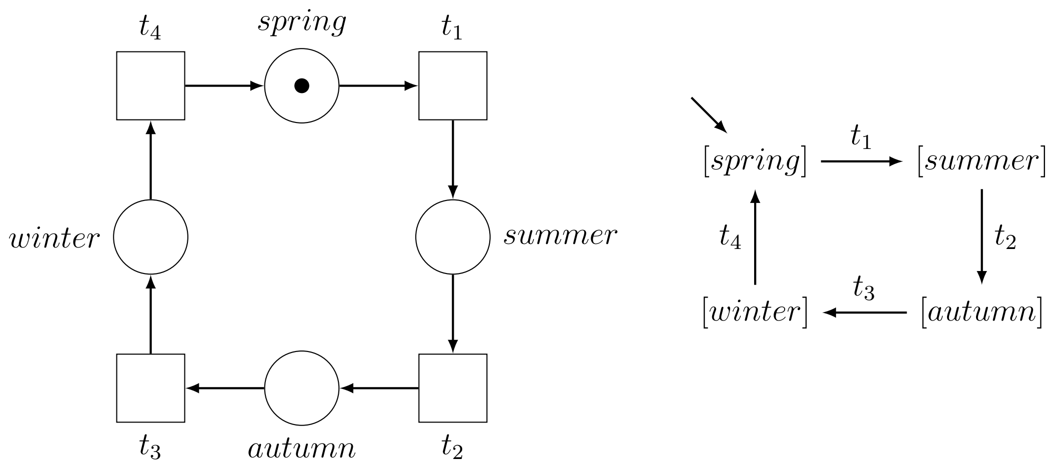


Figure 8: *A Petri net system and its reachability graph.*

Recall that, each of the reachable markings is represented as a multiset (where the same element may appear multiple times). Multiset [spring] thus represents the marking in figure 8(a).

• The incoming edge without source pointing to this node denotes that this marking is the initial marking. We labeled each edge of the reachability graph on the right with the transition that fired in the corresponding marking.

• Figure 8(b) depicts the accompanying reachability graph that represents the set of markings that are reachable from the initial marking shown in figure 8(a). We can conclude that the net in figure 5.12(a) has four reachable markings.

• If a marking M is reachable from the initial marking M_0, then the reachability graph has a path from the start node to the node representing marking M. This path represents a sequence of transitions that have to be fired to reach marking M from M_0.

We refer to this transition sequence as a run (as an execution in finite automaton). A run is finite if the path and hence the transition sequence is finite. Otherwise, the run is infinite

Question 2.2.

The path from marking [spring] to marking [winter] is a finite run (t_1, t_2, t_3) of the net in figure 8(a). Does it have infinite run?

Answer:

It does not have infinite run. As you can see from reachability graph, it will be fired $(t_1, t_2, t_3)^n$ times with $n \in \mathbb{N}^*$, so it does not have infinite run.

2.2.2 Representing Petri Nets as Special Transition Systems

Let (N, M_0) with $N = (P, T, F, A, l)$ marked labeled Petri net.

Definition 2.8: Reachability Graph

(N, M_0) defines a transition system $TS = (S, A_1, TR)$ with

$S = [N, M_0\rangle, S^{start} = M_0, A_1 = A$ and

$TR = (M, M_1) \in S \times S | \exists t \in T \quad (N, M)[t\rangle(N, M_1)$, or with label $l(t)$:

$TR = (M, l(t), M_1) \in S \times A \times S | t \in T \quad (N, M)[t\rangle(N, M_1)$

TS is often referred to as the **reachability graph** of (N, M_0)

2.3 Petri networks - Structures and Basic Problems

2.3.1 Causality

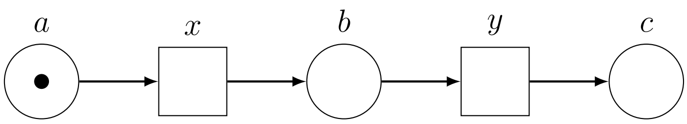


Figure 9: *Causality in a* ***Petri Net***

Causality is formally understood as a relationship between two events in a system that must take place in a certain order. In a **Petri net N**,we may represent this relationship by two transitions connected through an intermediate place.

2.3.2 Concurrency

Concurrency (i.e., parallelism) is an important feature of (information) systems. In a concurrent system, several events **can occur simultaneously**. For example, several users may access an information system like a database at the same time.

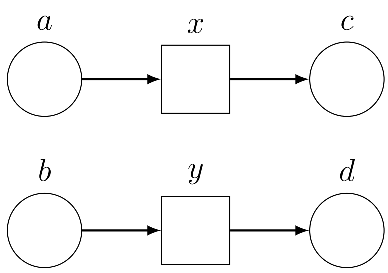


Figure 10: *Concurrency in a* ***Petri Net***

2.3.3 Synchronization

We can model synchronization in a **Petri Net** as a transition with at least two input places.

In industry or any process, assume that transitions x and y epresent two concurrent production steps. Transition z can then represent an assembly step that can take place only after the results of the two previous production steps are available.

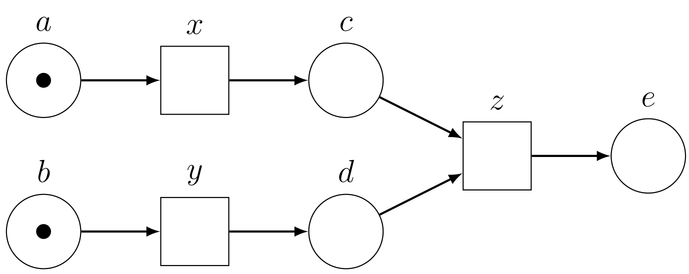


Figure 11: *Synchronization a* ***Petri Net***

In figure 11, there is transition z after firing transitions x and y. Therefore, transition z only fire after transitions x and y have fired.

2.3.4 Effect of Concurrency

If the model of a process contains a lot of concurrency or multiple tokens reside in the same place, then the transition system TS is much bigger than the Petri net $N = (P, T, F)$.

Generally, a marked Petri net (N, M_0) may have infinitely many reachable states.

Question 2.3.

Could we quantify the concurrency for a given process or **Petri Net**?

Answer:

The answer is YES. By using the transition system, we quantify the concurrency for a given process or Petri net.

If the model of a process contains a lot of concurrencies or multiple tokens reside in the same place, then the transition system TS is much bigger than the Petri net $TS = (S, A, T)$. Generally, a marked Petri net (N, M_0) may have infinitely many reachable states.

Remind that, a transition system is formally a triplet $TS = (S, A, T)$ where:

- S is the set of states.
- $A \subseteq \mathcal{A}$ is the set of activities (often referred to as actions).
- $T \subseteq S \times A \times S$ is the set of transitions.

2.4 Petri nets - Assignment on modelling

2.4.1 Dynamic of Petri nets via Special properties

1. A marked Petri net (N, M_0) is k**-bounded** if no place ever holds more than k tokens. Formally, for any $p \in P$ and any $M \in [N, M_0\rangle : M(p) \leqslant k$.

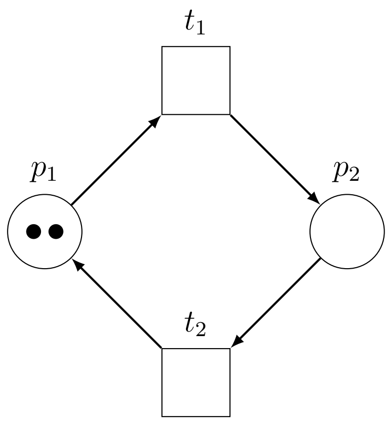


Figure 12: *An example for* k***-bounded***

This Petri net is 2-bounded because each place can contain maximum 2 tokens.

2. A marked Petri net is **safe** if and only if it is 1-bounded.

3. A marked Petri net is **bounded** if and only if there exists a $k \in \mathbb{N}$ such that it is k**-bounded**. For example, Pretri net in Figure 12 is bounded because it is 2-bounded.

To represent the dynamic of Petri nets we could talk about the followings.

- A marked Petri net (N, M_0) is **deadlock free** if at every reachable marking at least one transition is enabled. For example, Figure 12 is deadlock free because this Petri net can fire p_1 or p_2 in each reachable marking.

- A transition $t \in T$ in a marked Petri net (N, M_0) is **live** if from every reachable marking it is possible to enable t. Formally, for any $M \in [N, M_0\rangle$ there exists a marking $M_1 \in [N, M\rangle$ such that $(N, M_1)[t\rangle$.

- A marked Petri net is live if each of its transitions is **live**.

Note that: A deadlock-free Petri net does not need to be live.

2.4.2 Modeling by Petri networks - Problem

Definition 2.9: The Superimposition (Merging) Operator

Consider a system of only two agent types, denote $N1, N2$ to be their own Petri Nets. Assume that $N_1 = (P_1, T_1, F_1, M_0)$ and $N_2 = (P_2, T_2, F_2, M_0)$, where the places P_i could be disjoint, but with the same initial marking M_0.

The superimposed (merged) Petri net is determined by $N = (N_1 \oplus N_2) = (P_1 \cup P_2, T, F_1 \cup F_2, M_0)$. Where

- The **superimposition** operator: $\oplus : T_1 \times T_2 \longrightarrow T$, where $T = T_1 \cup T_2$
- Two transitionsevents of two nets can be identified into one node of the merged Petri net $N = N_1 \oplus N_2$ if the events act on the same physical token.

Question 2.4.
How could we build the grand Petri net of a large system without losing essential and useful information/knowledge of constituents' nets, as well as showing the true dynamic of the whole process/system?

Answer:
We need to know interaction of agents because of the different business activities of themselves and provide enough information to allow for the simulation of a system. In general, tokens in each places and interactions primarily provide a static description of a system. They define relevant activities and connectivity so the dynamics of a system can be emerged from these interactions.

2.5 Reviewed problems

Problem 1: Explain the following terms for Petri nets, and provide a specific example for each term:

1. Enable transition
2. Firing of a transition
3. Reachable marking
4. Terminal marking
5. Non-deterministic choice

Solution:

1. Enable transition: A transition is enabled if all places connected to it as inputs contain at least one token.

2. Firing of a transition: The firing rule for a transition can be characterized by subtracting a number of tokens from its input places equal to the multiplicity of the respective input arcs and accumulating a new number of tokens at the output places equal to the multiplicity of the respective output arcs. A transition can only fire when there is at least one token in each of its input places.

3. Reachable marking: Let m be a marking of a Petri net N.

   The reachable marking of m (the reachable set of m, denoted by reach(m)), is the smallest set of markings such that:

   - m $\in$ reach(m)
   - if m'$\Rightarrow$m" by t for some t $\in$ T, m' $\in$ reach(m), then m" $\in$ reach(m).

4. Terminal marking: The transitions keep firing until the net reaches a marking state that does not enable any transition. This marking is called a terminal marking.

5. Non-deterministic choice: When several transitions are enabled at the same moment, it is not determined which of them will fire. This situation is called a non-deterministic choice

Problem 2: Consider the **Petri net** in figure below.

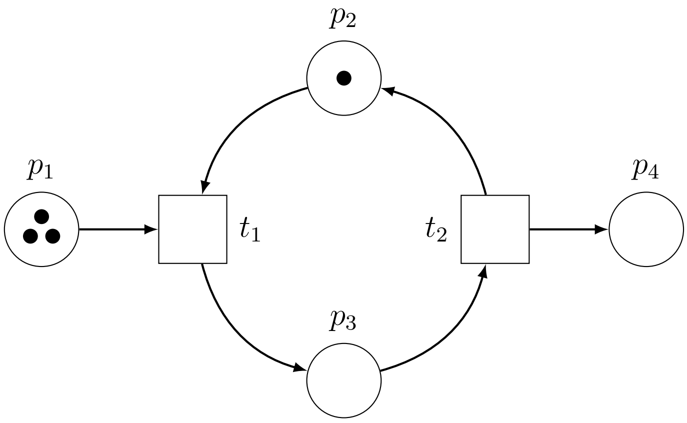


Figure 13: *A simple* ***Petri net****, with only two transitions*

1. Define the net formally as a triple (P, T, F).
2. List presets and postsets for each transition.
3. Determine the marking of this net.
4. Are the transitions t_1 and t_2 enabled in this net.

Solution:

1. Define the net formally as a triple (P, T, F).

   – P = $\{p_1, p_2, p_3, p_4\}$
   – T = $\{t_1, t_2\}$
   – We have: $F = R_i \cup R_o \subseteq (P \times T) \cup (T \times P)$.
   · $R_i(t_1, p_j) = 1$ for j = 1, 2.
   · $R_i(t_1, p_j) = 0$ for j = 3, 4.
   · $R_i(t_2, p_j) = 1$ for j = 3.
   · $R_i(t_2, p_j) = 0$ for j = 1, 2, 4.
   · $R_o(t_1, p_j) = 1$ for j = 3.
   · $R_o(t_1, p_j) = 0$ for j = 1, 2, 4.
   · $R_o(t_2, p_j) = 1$ for j = 2, 4.
   · $R_o(t_2, p_j) = 0$ for j = 1, 3.

2. List presets and postsets for each transition.

   - The presets of t_1 are $\bullet\ t_1 = \{(p_1, t_1), (p_2, t_1)\}$.
   - The postset of t_1 is $t_1\ \bullet = \{(t_1, p_3)\}$.
   - The preset of t_2 is $\bullet\ t_2 = \{(p_3, t_2)\}$.
   - The postset of t_2 are $t_2\ \bullet = \{(t_2, p_2), (t_2, p_4)\}$.

3. Determine the marking of this net.

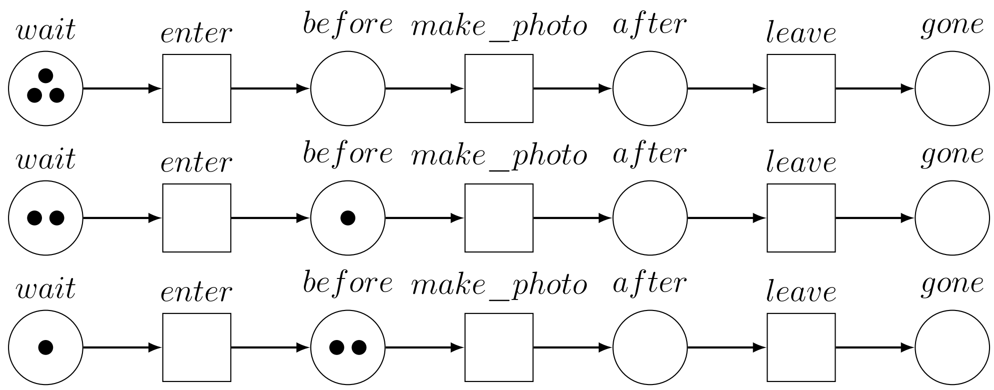
   - The marking of this net is M = [3, 1, 0, 0].

4. Are the transitions t_1 and t_2 enabled in this net.

   - There are three tokens in place p_1, and one token in place p_2. Hence, the transition t_1 is enabled.
   - However, in place p_3, there is no token in that place so transition t_2 is not enabled.

Problem 3

Given a process of a X-ray machine in which we assume the first marking in Figure 14.a shows that there are three patients in the queue waiting for an X-ray. Figure 14.b depicts the next marking, which occurs after the firing of transition enter.



The first three markings in a process of Xray machine
a) [top, transition **enter** not fired]; b) [middle, transition **enter** fired]; and c) [down, transition **enter** has fired again];

Figure 14: *A Petri net model of a business process of an X-ray machine*

1. Determine the two relations R_I and R_O and the flow relation $F = R_I \cup R_O$

Solution:

R_I = {(wait, enter), (before, make_photo), (after, leave)}

R_O = {(enter, before), (make_photo, after), (leave, gone)}

F = {(wait, enter), (enter, before), (before, make_photo), (make_photo, after), (after, leave), (leave, gone)}

2. A patient may enter the X-ray room only after the previous patient has left the room. We must make sure that places **before** and **after** together do not contain more than one token.

There are two possible states: the room can be **free** or **occupied**. We model this by adding theses two places to the model, to get the improved the Petri net, in Figure 15

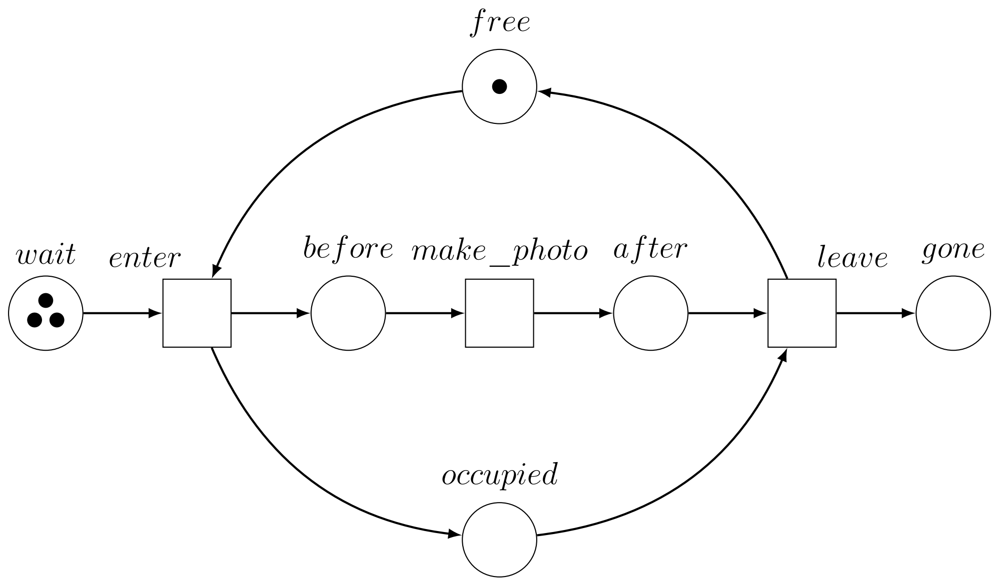


Figure 15: *An improved Petri net for the business process of an X-ray machine*

Now for this Petri net, can place **before** contain more than one token? Why? Rebuild the set P of place labels.

Solution:

Place **before** can not contain more than one token. Transition **enter** is firing if and only if places **wait** and **free** have a token. When transition **enter** was firing, place **free** will have another token when transition **leave** is firing. Transition **leave** is firing after transitions **before** and **after** firing. Therefore, transition **before** always has less than or equal to one token.

P = {wait, enter, before, make_photo, after, leave, gone, free, occupied}

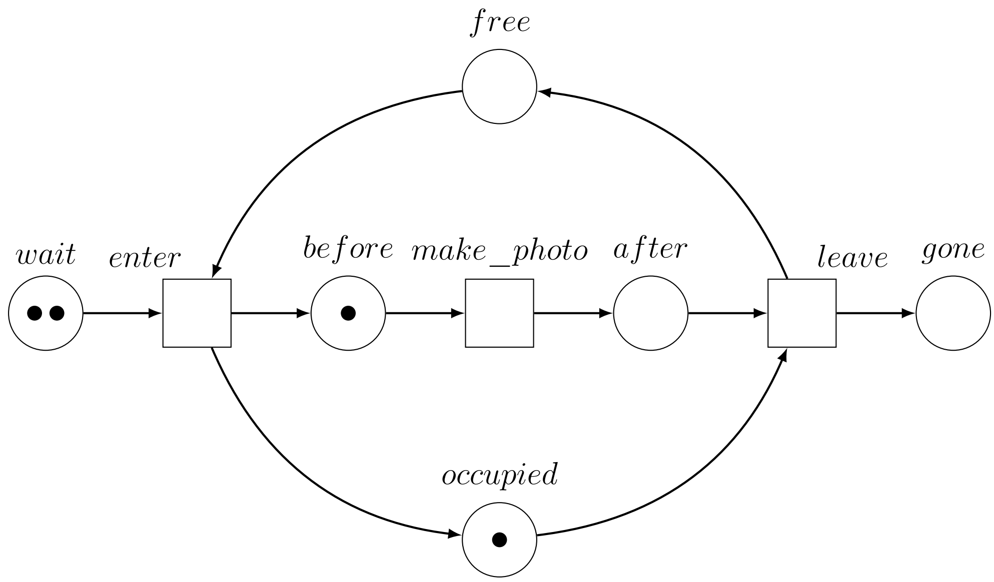


Figure 16: *The marking of the improved Petri net for the working process of an X-ray room after transition **enter** has fired.*

3. As long as there is no token in place **free** [Figure 16], can transition **enter** fire again? Explain why or why not. Remake the two relations R_I and R_O

Solution:

If there is no token in place **free**, transition **enter** can not fire again. Transition **enter** can fire if and only if places **free** and **wait** have a token. Thus, transition **enter** can not fire.

R_I = {(wait,enter), (before,make_photo), (after,leave), (free, enter), (occupied, leave)}

R_O = {(enter,before), (make_photo, after), (leave,gone), (enter, occupied), (leave,free)}

Problem 4

Build up the transition system TS generated from the labeled marked Petri net shown in figure 7.

Solution:

S = {[start], [c1,c2], [c2,c3], [c1,c4], [c3,c4], [c5], [end]}

A = {register request (a), examine thoroughly (b), examine casually (c), check ticket (d), decide (e), pay compensation (g), reject request (h)}

TR = {([start], a, [c1,c2]), ([c1,c2], b, [c2,c3]), ([c1,c2], c, [c2,c3]), ([c2,c3], d, [c3,c4]), ([c1,c2], d, [c1,c4]), ([c1,c4], b, [c3,c4]), ([c1,c4], c, [c3,c4]), ([c3,c4], e, [c5]), ([c5], g, [end]), ([c5], h, [end])}

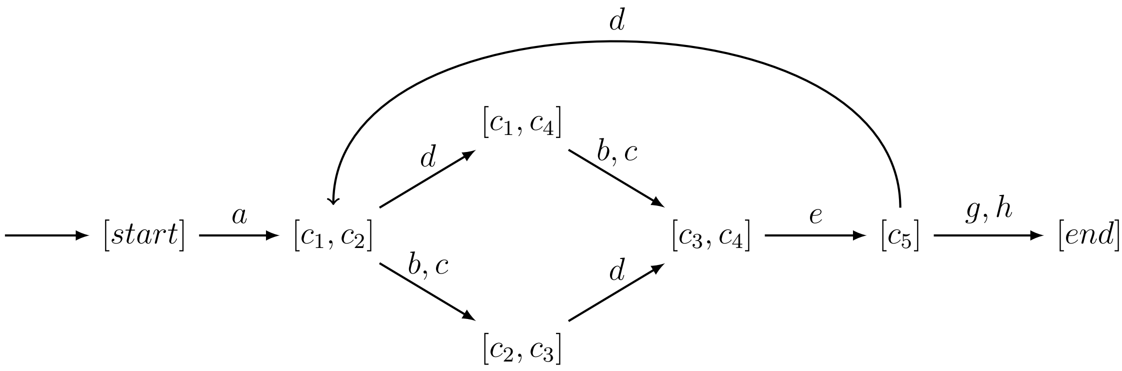


Figure 17: *Transition system for figure 7*

Problem 5: Consider the Petri net system in figure below.

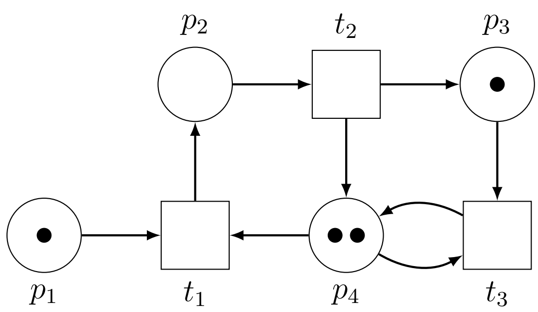


Figure 18: *A Petri net with small numbers of places and transitions*

1. Formalize this net as a quadruplet (P, T, F, M_0).
2. Give the preset and the postset of each transition.
3. Which transitions are enabled at M_0?
4. Give all reachable markings. What are the reachable terminal markings?
5. Is there a reachable marking in which we have a non-deterministic choice?
6. Does the number of reachable markings increase or decrease if we remove

   (1) place p_1 and its adjacent arcs and

   (2) place p_2 and its adjacent arcs?

Solution:

1. Formalize this net as a quadruplet (P, T, F, M_0).

   $P = \{p_1, p_2, p_3, p_4\}$.

   $T = \{t_1, t_2, t_3\}$.

   $F = \{(p_1, t_1), (t_1, p_2), (p_2, t_2), (t_2, p_3), (p_3, t_3), (t_3, p_4), (p_4, t_3), (p_4, t_1)\}$.

   $M_0 = [p_1, p_3, 2.p_4]$

2. Give the preset and the postset of each transition.

   The presets of t_1 are $\bullet t_1 = \{p_1, p_4\}$.

   The postset of t_1 is $\bullet t_1 = \{p_2\}$.

   The preset of t_2 is $\bullet t2 = \{p_2\}$.

   The postsets of t_2 are $\bullet t_2 = \{p_3, p_4\}$.

   The presets of t_3 are $\bullet t_3 = \{p_3, p_4\}$.

   The postset of t_3 is $\bullet t_3 = \{p_4\}$.

3. Which transitions are enabled at M_0?

   Transitions t_1 and t_3 are enabled at M_0. Only transition t_2 is not enabled because there is no available token at place p_2.

4. Give all reachable markings. What are the reachable terminal markings?

   There are total 7 reachable markings from M_0.

   (1) Marking: $[p_1, p_3, 2.p_4]$
   (2) Marking: $[p_2, p_3, p_4]$
   (3) Marking: $[p_1, 2.p_4]$
   (4) Marking: $[p_2, p_4]$
   (5) Marking: $[2.p_3, 2.p_4]$
   (6) Marking: $[p_3, 2.p_4]$
   (7) Marking: $[2.p_4]$

   From the set of reachable markings, marking $[2.p_4]$ is the reachable terminal marking because it does not enable any transitions.

5. Is there a reachable marking in which we have a non-deterministic choice?
   A nondeterministic choice is available when more than a single transition is enable at a given state. There is a reachable marking (1.p1, 1.p3, 2.p4) which has an option to fire, transition t1 or t3. So, this Petri net has a nondeterministic choice.

6. Does the number of reachable markings increase or decrease if we remove

(1) place p_1 and its adjacent arcs. There are 8 reachable marking: $[p_3, 2.p_4]$, $[p_2, p_3, p_4]$, $[2.p_4]$, $[p_2, p_4]$, $[2.p_2]$, $[2.p_2, p_3]$, $[2.p_2, 2.p_4]$, and $[p_2, 2.p_3, p_4]$. Therefore, the number of reachable markings increases.

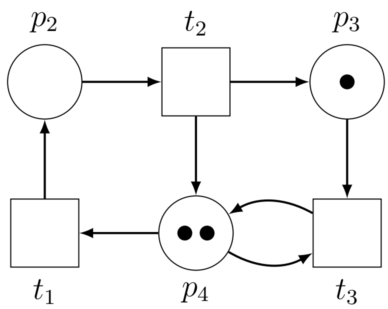


Figure 19: *A Petri net with small numbers of places and transitions when we remove place p_1 and its adjacent arcs*

(2) place p_2 and its adjacent arcs. There are only 3 reachable markings: $[p_1, 2.p_4]$, $[p_2, p_4]$ and $[2.p_4]$. Then, the number of reachable marking decreases.

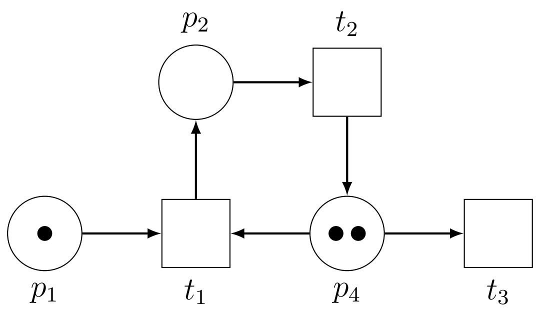


Figure 20: *A Petri net with small numbers of places and transitions when we remove place p_3 and its adjacent arcs*

Problem 6: (From small marked Petri net to bigger transition system).

Consider a marked Petri net $N = (P, T, F)$ with $|P| = 4 = n$, with 4 start places each initially get 3 tokens, and the exit node is specially designated as place OUT.

Besides, assume $|T| = 4$, and the initial marking is M_0 as seen in figure 21

Denote TS for the whole transition system made by the Petri net $N = (P, T, F)$

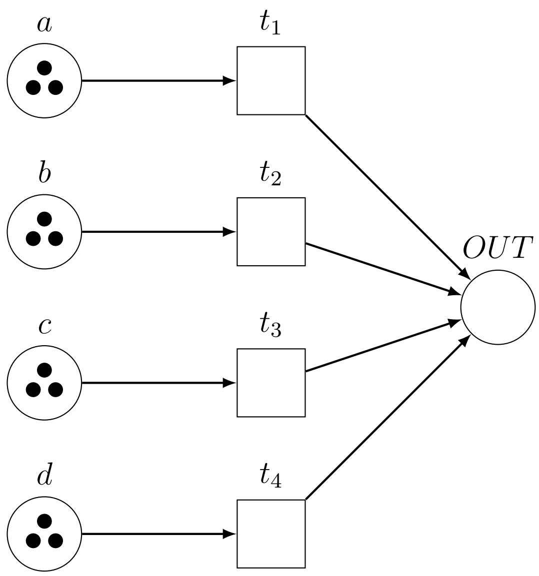


Figure 21: *A Petri net allows concurrency*

1. Write down M_0, P, T of N.

2. If not allow concurrency in this process (marked Petri net) then how many states of the transition system TS can created? How many transitions are there?

3. If we allow concurrency in this net, how many states and how many transitions of the transition system TS could built?

Solution:

1. Write down M_0, P, T of N.

    - $M_0 = [3, 3, 3, 3, 0]$

- $P = \{a, d, c, d, OUT\}$
- $T = \{t_1, t_2, t_3, t_4\}$

2. If not allow concurrency in this process (marked Petri net) then how many states of the transition system TS can created? How many transitions are there?

   - State of the transition system TS: In each place, there are 4 possible cases. Therefore, we have $4 \cdot 4 \cdot 4 \cdot 4 = 4^4 = 256$ (states).

   - Transition of the transition system TS: There are 4 possible cases occured.

   - The number of tokens in places are greater than 0: There are 3^4 states and 4 ways to fire, so we have $3^4 \cdot 4 = 324$ (transitions).
   - One place has 0 token, the number of tokens in other places are greater than 0: There are $\binom{1}{3} \cdot 3^3$ states and 3 ways to fire so that we have $\binom{1}{4} \cdot 3^3 \cdot 3 = 243$ (transitions).
   - Two places have 0 token, the number of tokens in other places are greater than 0: There are $\binom{2}{4} \cdot 3^2$ states and 2 ways to fire so that we have $\binom{2}{4} \cdot 3^2 \cdot 2 = 108$ (transitions).
   - Three places have 0 tokens and the number of tokens in other places are greater than 0: There are $\binom{3}{4} \cdot 3^1$ states and 1 ways to fire so that we have $\binom{3}{4} \cdot 3^1 \cdot 1 = 12$ (transitions).

   Thus, there are $324 + 243 + 108 + 12 = 687$ (transitions)

3. If we allow concurrency in this net, how many states and how many transitions of the transition system TS could built?

   - State of the transition system TS: Because the number of states is unchanged (similar to (2)), so there are 256 (states).

   - Transition of the transition system TS: There are 4 possible cases occured.

   - The number of tokens in places are greater than 0: There are 3^4 states and $\binom{1}{4} + \binom{2}{4} + \binom{3}{4} + \binom{4}{4} = 15$ ways to fire, so we have $3^4 \cdot 15 = 1215$ (transitions).
   - One place has 0 token, the number of tokens in other places are greater than 0: There are $\binom{1}{3} \cdot 3^3$ states and $\binom{1}{3} + \binom{2}{3} + \binom{3}{3} = 7$ ways to fire so that we have $\binom{1}{4} \cdot 3^3 \cdot 7 = 756$ (transitions).

- Two places have 0 token, the number of tokens in other places are greater than 0: There are $\binom{2}{4} \cdot 3^2$ states and $\binom{1}{2} + \binom{2}{2} = 3$ ways to fire so that we have $\binom{2}{4} \cdot 3^2 \cdot 3 = 162$ (transitions).
- Three places have 0 tokens and the number of tokens in other places are greater than 0: There are $\binom{3}{4} \cdot 3^1$ states and 1 ways to fire so that we have $\binom{3}{4} \cdot 3^1 \cdot 1 = 12$ (transitions).

Thus, there are $1215 + 756 + 162 + 12 = 2145$ (transitions)

3 Application: Consulting Medical Specialists

SCENARIO: Under a SARS pandemic where a huge lack of ICU beds occurs in city H, patients should consult specialists in the outpatient clinic of a hospital, we describe the course of business around a specialist in this outpatient clinic of hospital X as a process model, formally, we use Petri Net.

ASSUMPTION and DATA

- **Specialist**: Each patient has an appointment with a certain specialist. The specialist receives patients. At each moment, the specialist is in one of the following three states:
  1. the specialist is free and waits for the next patient (state free),
  2. the specialist is busy treating a patient (state busy), or
  3. the specialist is documenting the result of the treatment (state docu).
- **Every patient** who visits a specialist is in one of the following three states:
  1. the patient is waiting (state wait, gets value n if there are n patients waiting),
  2. the patient is treated by the specialist (state inside), or
  3. the patient has been treated by the specialist (state done)

  The specialist goes through the three states in an iterative way. A patient goes through these states only once (per visit).
- Event data: three events are important for this business process. First, the specialist starts with the treatment of a patient (event "start").

  Second, the specialist finishes the treatment of a patient and starts documenting the results of the treatment (event "change").

  Third, the specialist stops documenting (event "end").

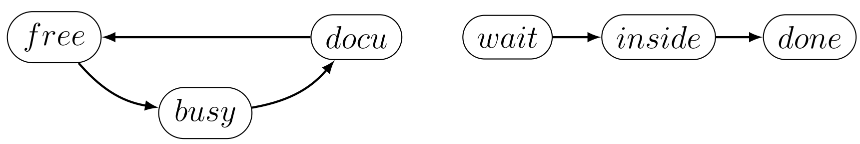


Figure 22: *The transition systems of a specialist (left) and a patient (right)*

Problem 1: (Given the Petri net N_S modeling the state of the specialist, as in Fig. 23)

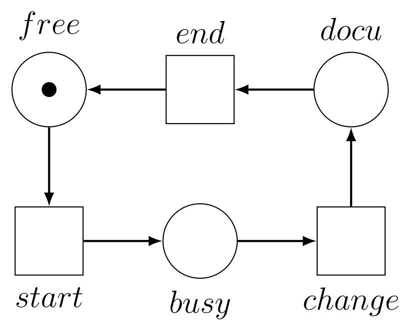


Figure 23: *A Petri net modeling the state of the specialist.*

In the displayed marking, the specialist is in state "free".

(a) Write down states and transitions of the Petri net

(b) Could you represent it as a transition system assuming that:

  (i) Each place **cannot** contain more than one token in any marking.

  (ii) Each place may contain any natural number of tokens in any marking.

Solution:

(a) Observe a transition system in Figure 23 we can see:

- States $S = \{free, busy, docu\}$.
- Transition $T = \{start, change, end\}$.

(b) Let the triplet (a_0, b_0, c_0) represent the transition system equivalent to the above net where a_0, b_0, c_0 respectively the number of token of each state free,busy,docu and the transition system be represented as a triplet (S, TR, s_0) in which:

  (i) We will consider about all possible initial state:

  - $M_0 = [0, 0, 0]$. Because it does not have token, there are 0 transition relation and 1 state.

- $M_0 = [1,0,0]$, $M_0 = [0,1,0]$, $M_0 = [0,1,0]$. There are 3 states and 3 transition relations. For example, we consider $M_0 = [1,0,0]$:

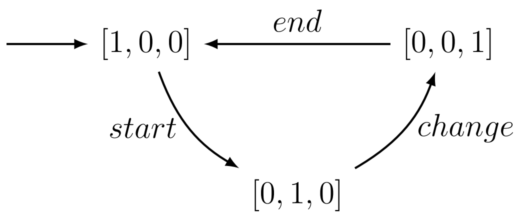


Figure 24: *A transition system when* $M_0 = [1,0,0]$

- $M_0 = [1,1,0]$, $M_0 = [1,0,1]$, $s_0 = [0,1,1]$. There are 3 states and 3 transition relations.For example, we consider $M_0 = [1,1,0]$:

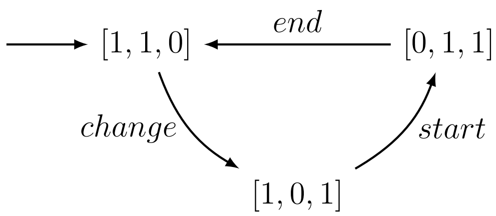


Figure 25: *A transition system when* $M_0 = [1,1,0]$

- $M_0 = [1,1,1]$. There are 0 transition relation and 1 state because each place can not contain more than 1 token.

(ii) Each place may contain any natural number of tokens in any marking:

We assume that $M_0 = [a_0, b_0, c_0]$ and the next state is $M_i = [a_i, b_i, c_i]$. We can observe that $a_0 + b_0 + c_0 = a_i + b_i + c_i$. So we can use the formula for Euler's candy division.

Let n is total tokens of 3 places. So the number of state is $\binom{2}{n+2}$.

About the number of transition relation. We have 3 cases:

- Case $n = 1$, we have that it only has 3 transition relations. We proof this in the previous question.
- Case $n = 2$, because $n < 3$ so there are only 6 states including 3 states can fire 2 transitions and 3 states can fire 1 transitions. Therefore, we have 9 transition relations.

- Case $n \geqslant 3$. We have:
  i. Two of 3 place will be empty tokens: so there is only 1 transition firing and we have 3 cases. Therefore, we have 3 transition relations.
  ii. One of 3 place will be empty tokens: we have $\binom{2}{3}$ cases and there are $n-1$ case so that the sum is equal to n so we have $(n-1)\binom{2}{3}$ states. Moreover, we have 2 ways to fire the transition so we have $2(n-1)\binom{2}{3}$ transition relations.
  iii. The number of states which do not have empty place is $\binom{2}{n+2}-(n-1)\binom{2}{3}-3$. Moreover, we have 3 ways to fire the transition. Therefore, we have $(\binom{2}{n+2}-(n-1)\binom{2}{3}-3)\cdot 3$ transition relations.

  The total of transition relations is :
  $((\binom{2}{n+2}-(n-1)\binom{2}{3}-3)\cdot 3+2(n-1)\binom{2}{3}+3 = 3\binom{2}{n+2}-3(n-1)-6$

In short, the number of states is $\binom{2}{n+2}$ and the number of transition relations is

$$\begin{cases} 3 & \text{if } n = 1 \\ 9 & \text{if } n = 2 \\ 3\binom{2}{n+2}-3(n-1)-6 & \text{if } n \geqslant 3 \end{cases}$$

Problem 2: Figure 23 of net N_S is made by information of **Specialist** and **Event data** above. Define N_{Pa} as the Petri net modeling the state of patients. By the similar ideas,

(a) Explain the possible meaning of a token in state **inside** of the net N_{Pa};

(b) Construct the Petri net N_{Pa}, assuming that there are five patients in state wait, no patient in state **inside**, and one patient is in state **done**.

Solution:

(a) When we fire the transition start, it means that specialist will start to treat the patient and patient will go to inside. Therefore, a token in state **inside** represents that the patients is being treated by the specialist.

(b) Construct the Petri net N_{Pa}:

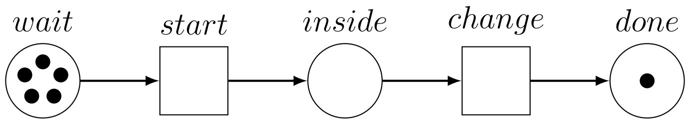


Figure 26: *A Petri net N_{Pa} that has five patients at place **wait**, no patient at place **inside** and one patient at place **done**.*

With Petri net N_{Pa}, we have that:

- $P = \{wait, inside, done\}$
- $T = \{start, change\}$
- $F = \{(wait, start), (start, inside), (inside, change), (change, done)\}$

With the initial marking : $M_0 = [5.wait, done]$, transition **start** is enable to fire to $M = [4.wait, inside, done]$. We can do step by step to have marking $M = [6.done]$ in the final.

Problem 3: Determine the superimposed (merged) **Petri Net** model N $= N_S \oplus N_{Pa}$ allowing a specialist treating patients, assuming there are four patients are waiting to see the specialist/ doctor, one patient is in state done, and the doctor is in state free. (The model then describes the whole course of business around the specialist).

Solution:

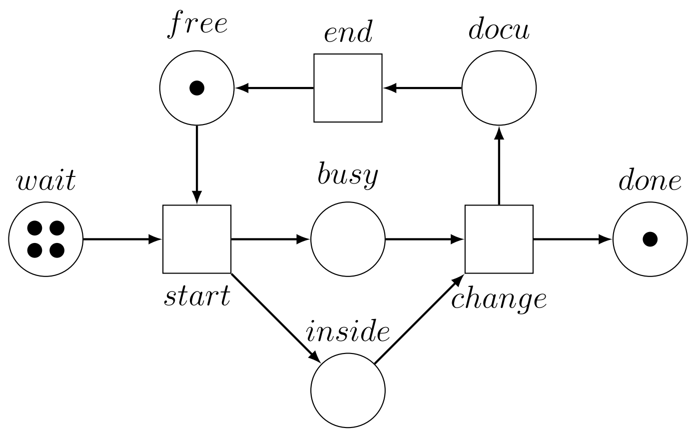


Figure 27: *The superimposed (merged)* ***Petri Net*** *model* $N = N_S \oplus N_{Pa}$ *that has four patients* ***waiting****, one patient at* ***done*** *and one specialist at* ***free****.*

We can observe that Petri net N_S and N_{Pa} have transition **start** and **change**, so we can merge them to have the Petri net $N = N_S \oplus N_{Pa}$.

With the initial marking: $M_0 = [4.wait, free, done]$, transition **start** is enable to fire to $M = [4.wait, busy, inside, done]$. We can do step by step to have marking $M = [free, 5.done]$ in the final.

Problem 4: Consider an initial marking $M_0 = [3.\text{wait},\text{done},\text{free}]$ in the grand net $\text{N} = N_S \oplus N_{Pa}$, Which markings are reachable from M_0 by firing one transition once?

Solution:

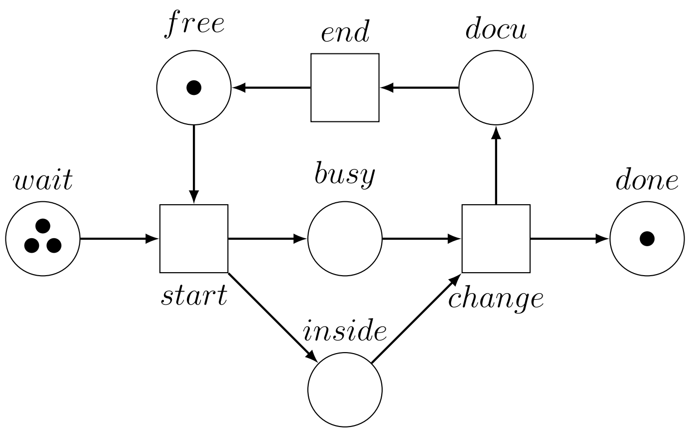


Figure 28: *The superimposed (merged)* ***Petri Net*** *model $N = N_S \oplus N_{Pa}$ that has three patients* ***waiting****, one patient at* ***done*** *and one specialist at* ***free****.*

There are total 10 reachable markings from M_0 by firing one transition once.

1. Firing sequence: ()

   Marking: $M = [0.busy, 0.docu, 1.done, 1.free, 0.inside, 3.wait]$

2. Firing sequence: (start)

   Marking: $M = [1.busy, 0.docu, 1.done, 0.free, 1.inside, 2.wait]$

3. Firing sequence: (start, change)

   Marking:$M = [0.busy, 1.docu, 2.done, 0.free, 0.inside, 2.wait]$

4. Firing sequence: (start, change, end)

   Marking: $M = [0.busy, 0.docu, 2.done, 1.free, 0.inside, 2.wait]$

5. Firing sequence: (start, change, end, start)

   Marking: $M = [1.busy, 0.docu, 2.done, 0.free, 1.inside, 1.wait]$

6. Firing sequence: (start, change, end, start, change)

   Marking: $M = [0.busy, 1.docu, 3.done, 0.free, 0.inside, 1.wait]$

7. Firing sequence: (start, change, end, start, change, end)

   Marking: $M = [0.busy, 0.docu, 3.done, 1.free, 0.inside, 1.wait]$

8. Firing sequence: (start, change, end, start, change, end, start)

   Marking: $M = [1.busy, 0.docu, 3.done, 0.free, 1.inside, 0.wait]$

9. Firing sequence: (start, change, end, start, change, end, start, change)

   Marking: $M = [0.busy, 1.docu, 4.done, 0.free, 0.inside, 0.wait]$

10. Firing sequence: (start, change, end, start, change, end, start, change, end)

    Marking: $M = [0.busy, 0.docu, 4.done, 1.free, 0.inside, 0.wait]$

Problem 5: Is the superimposed Petri net N deadlock free? Explain.

Solution:

The superimposed Petri net in problem 3 is not deadlock free.
To prove this, we have to consider about 3 possible initial marking:

1. $M_0 = \{n.wait, free, m.done\}$ with $n, m \in \mathbb{N}$, the transition **start** enable to fire to $M = \{(n-1).wait, busy, inside, m.done\}$. Finally, we have $M = \{free, (m+n).done\}$ by firing transition step-by-step.

2. $M_0 = \{n.wait, busy, inside, m.done\}$ with $n, m \in \mathbb{N}$, the transition **change** enable to fire to $M = \{n.wait, docu, (m+1).done\}$. Finally, we have $M = \{free, (m+n+1).done\}$ by firing transition step-by-step.

3. $M_0 = \{n.wait, docu, m.done\}$ with $n, m \in \mathbb{N}$ the transition **end** enable to fire to $M = \{n.wait, free, m.done\}$. Finally, we have $M = \{free, (m+n).done\}$ by firing transition step-by-step.

From 3 possible case, we observe that we will have the marking $M = \{free, p.done\}$ with $p \in \mathbb{N}$. This is a terminal marking so that the superimposed Petri net is not deadlock free.

Problem 6: Propose a similar **Petri Net** with two specialists already for treating patients, with explicitly explained construction.

Solution:

In order to solve this problem, we will build the colored Petri net.

Explain about the step of building this Petri Net:

1. We will mark 2 specialists as 2 colors, so we have $NP = \{n_1, n_2\}$ to set showing the coloring for specialists.
2. We will mark the patients in the order $d_1, d_2, \ldots, d_n$ where n is the order of patients and $PT = \{p(d_1), p(d_2), \ldots, p(d_n)\}$ is the set representing the coloring for the patients.
3. We will assume that the collection of information about patients is treated by specialist and also name of them is $NT \subseteq NP \times PT$.

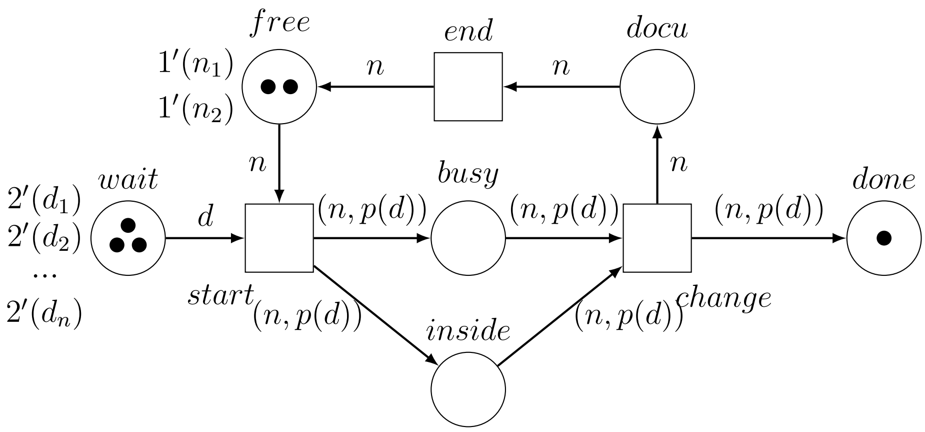


Figure 29: *A* ***Petri net*** *for medical examination and treatment of two specialists*

With the Petri net $N = (P, T, F)$, we will keep the information about P, T, F to be the same with the previous question about superimposed (merged) **Petri Net** of patients and specialist. And we will add some special thing in this Petri net:

- The number of tokens in place **wait, done** is depend on the number of patients who want to be treated.
- Place **free, docu, busy** can hold up to 2 tokens for 2 specialist. Place **inside** can hold up to 2 tokens for 2 patients.

Remarkable: There is no case that both of specialists takes part in treating the same patient.

If the transition **start** is enabled, an information $t = (n, p(d)) \in NT$ will be generated by using the order of patient and specialist who will treat this patient. And then, a token representing for the patient will go to place **inside**, and a token representing for the specialist will go to place **busy** for medical examination and treatment.

When the patient completes the examination and treatment, we will have:

- The information t will be saved helping us to know about specialists have treated and the order in examination of specialists.
- The data n will be transmitted showing the order of the specialists when completing the examination and treatment for each patient.

References

[1] Man VM. Nguyen - Mathematical modeling and Risk analysis - Faculty of Science Mahidol University (2021)

[2] Kurt Jensen, Lars M. Kristensen - Coloured Petri Nets_Modelling and Validation of Concurrent Systems-Springer (2009)