

Chapter 5. Petri Nets (PN)

Mary Ann Lundteigen Marvin Rausand

RAMS Group
Department of Mechanical and Industrial Engineering
NTNU

(Version 0.1)



NTNU – Trondheim
Norwegian University of
Science and Technology

Learning Objectives

The main learning objectives associated with these slides are to:

1. Explain the main attributes and concepts associated with Petri Nets
2. Explain how Petri Nets may be used for system reliability analysis
3. Discuss some pros and cons for the application of Petri Nets

The slides include topics from Chapter 5 in **Reliability of Safety-Critical Systems: Theory and Applications**. DOI:10.1002/9781118776353.

Outline of Presentation

- 1 Introduction
- 2 Basic concepts
- 3 Modeling
- 4 Analysis

About Petri nets

Petri Nets is a well known approach in many application areas, but has not until recently got a high status within the field of system reliability.

- ▶ Petri Nets was introduced by *Carl Adam Petri* in the 1960s
- ▶ It has been widely used for analysis of telecommunication, software engineering, and transportation
- ▶ Petri Nets are now referenced by IEC 61508 as a suitable approach for reliability analysis
- ▶ IEC 62551 defines terminology and gives requirements for the use of Petri Nets

We will use the abbreviation PN for Petri nets in the following slides.

Places, transitions, and arcs

A PN consists of two basic nodes: *places*, drawn as circles, and *transitions*, drawn as bars.

- ☞ **Places:** Circles used to model *local states* or conditions (e.g., failed or functioning).
- ☞ **Transitions:** Bars used to model *local events* (e.g., failure or restoration).
- ☞ **Arcs:** Directed arcs that link places and transitions.

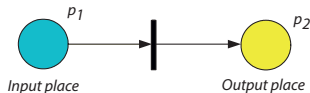


Figure: Input places vs output places

Tokens and firing

Tokens (black circles) are dynamic elements used to illustrate the system state at a certain point in time.

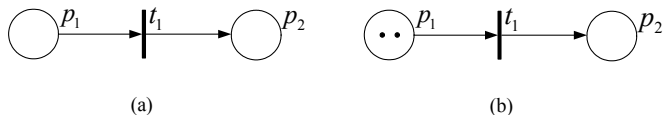


Figure: Simple PN, without (a) and with (b) tokens

Note that:

- ▶ A PN with tokens is called a *marked PN*
- ▶ Firing is the event where one or more tokens are moved from one place to another

Multiplicity and enabling

- **Multiplicity** (or weight): A digit (e.g., 1, 2 etc) assigned to an arc, and which represents the number of tokens the arc delivers at a time.

It may be remarked that:

- ▶ When multiplicity is 1, it is not specified in the PN
- ▶ A transition is enabled (ready for firing) if the number of tokens in each of its input places is equal to or greater than the multiplicity of the associated arcs.

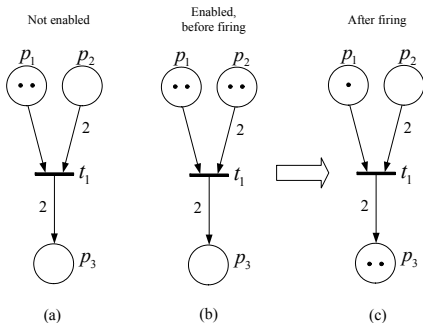


Figure: PN before and after firing

Inhibitor arc

- **Inhibitor arc:** A directed arc with by a small circle, and whose main purpose is to block the output transition if the number of tokens is in the input places are equal to or higher than the multiplicity (weight) of the arc.

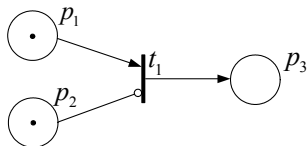






Figure: PN with inhibitor arc

Figure above: Transition t_1 is inhibited as long as there is a token in place p_2 .

Transition types

Transitions may be immediate (when enabling conditions are fulfilled), or timed with a delay > 0 . The delays may be *deterministic* or *stochastic*, as shown in the table below:

Type of transition in PN				
	<i>Deterministic</i>		<i>Stochastic</i>	
Parameter	<i>Delay is zero</i>	<i>Delay is d</i>	<i>Exponentially or geometrically distributed</i>	<i>Arbitrary distributed</i>
Symbol		 d	 λ	

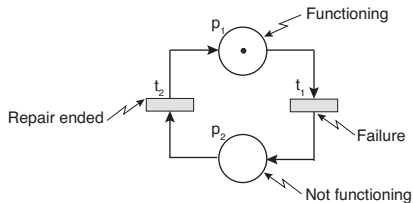
Stochastic PNs

- **Stochastic PN (SPN):** A PN with stochastic transitions only.

A generalized SPN (GSPN) allows timed as well as immediate transitions, but only exponentially distributed firing times are accepted for the timed transitions. This means that GSPN is not so general, in the meaning that it allows any type of stochastic distributions for the times transitions.

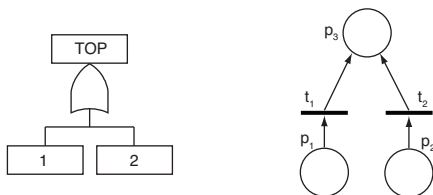
State of an item

A PN may be used to model the functioning and failure of a single item, as shown below:



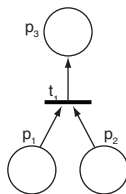
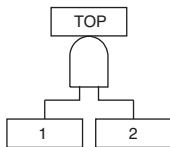
Fault tree

A PN may be used to model an OR gate in a fault tree, as shown below:



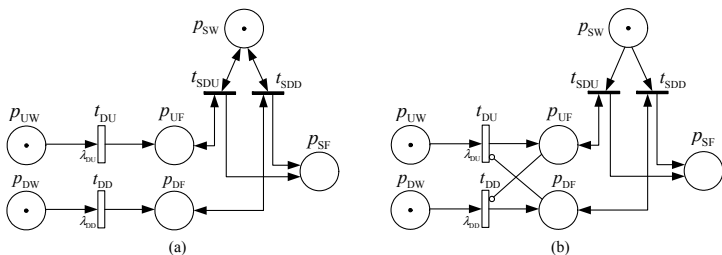
Fault tree

A PN may be used to model an AND gate in a fault tree, as shown below:



Having different failure modes

A PN may be used to model an item that can experience a dangerous detected (DD) or dangerous undetected (DU), as shown below:



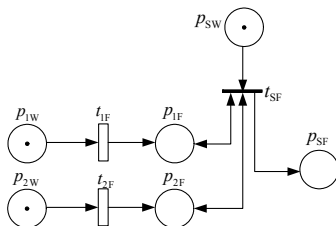
Note: Notations are explained in the next slide slide.

Notations (previous slide)

Parameter	Working states (when token is present)
p_{UW}	State where no DU failure is present
p_{DW}	State where no DD failure is present
p_{SW}	State where system is working (no DU or DD failure present)
Parameter	Failed states (when token is present)
p_{UF}	State where a DU failure is present
p_{DF}	State where a DD failure is present
p_{SF}	State where the system has failed (DU or DD failure is present)
Parameter	Transitions
t_{DU}	Firing of a DU failure (exponentially distributed) with rate λ_{DU}
t_{DD}	Firing of a DD failure (exponentially distributed) with rate λ_{DD}
t_{SDU}	(Immediate) firing of a <i>system</i> failure caused by a DU failure
t_{SDD}	(Immediate) firing of a <i>system</i> failure caused by a DD failure

1oo2 voted system (non-repairable)

A PN may be used to model a *non-repairable* 1oo2 voted system, as seen below.



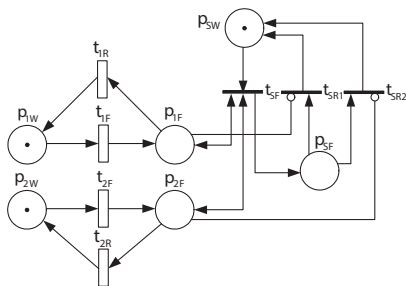
Note: Notations are provided on the next slide.

Notations (previous slide)

Parameter	Working states (when token is present)
p_{SW}	State where system is working
p_{1W}	State where channel 1 is working
p_{2W}	State where channel 2 is working
Parameter	Failed states (when token is present)
p_{1F}	State where channel 1 has failed
p_{2F}	State where channel 2 has failed
p_{SF}	State where system has failed
Parameter	Transitions
t_{1F}	Firing of a failure (exponentially distributed) with rate λ_{1F}
t_{2F}	Firing of a DU failure (exponentially distributed) with rate λ_{2F}
t_{SF}	(Immediate) firing of system failure if both channels have a failed

1002 voted system (repairable)

A PN may be used to model a *repairable* 1002 voted system, as seen below.



Note: Notations are provided on the next slide.

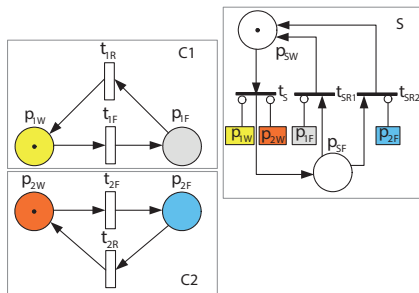
Notations (previous slide)

Parameter	Working states (when token is present)
	See notations for non-repairable 1oo2 system
Parameter	Failed states (when token is present)
	See notations for non-repairable 1oo2 system
Parameter	Transitions (failure)
t_{1F}	Firing of a failure (exponentially distributed) with rate λ_{1F}
t_{2F}	Firing of a failure (exponentially distributed) with rate λ_{2F}
t_{SF}	(Immediate) firing of system failure when both channels have failed
Parameter	Transitions (repair)
t_{1R}	Firing of a repair (exponentially distributed) with rate μ_{1R}
t_{2R}	Firing of a repair (exponentially distributed) with rate μ_{2R}
t_{SR1}	(Immediate) firing of a return to functioning state after repair of channel 1
t_{SR1}	(Immediate) firing of a return to functioning state after repair of channel 1

Modularization of PNs

It may be feasible to modularize the PN, by modeling each functional block (in a reliability block diagram). A module can include two parts:

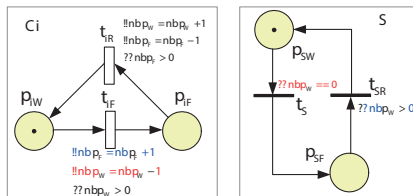
- ▶ *Intrinsic part*: Modules that describes the behavior of the items in the module
- ▶ *Extrinsic part*: Module that links the intrinsic part to system properties (e.g., failure, repair, working)



Note: Notations are as defined on previous slides. Note that i is the channel

Predicates and Assertions

By *Predicates* (“??”) and *Assertions* (“!!”), the modeling of the 1oo2 system may be simplified as follows:



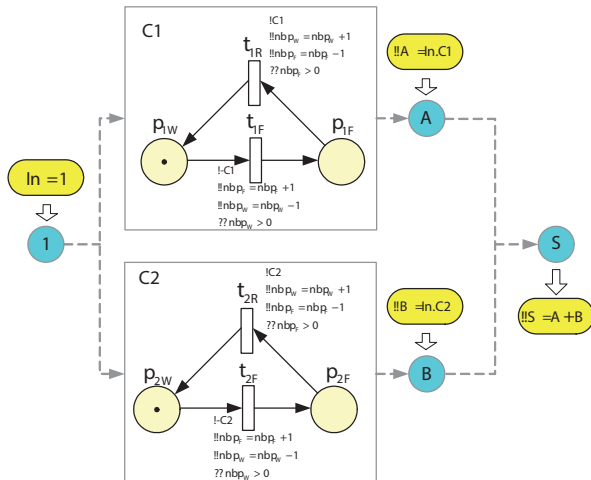
Note: Notations are as defined on previous slides. Explanation of “??” and “!!” are found in the textbook.

Reliability block diagram driven PNs

Reliability block diagram (RBD) driven PNs may be constructed with basis in modules of item/channel states, as seen in the illustration on the next slide. The following features are added to the model:

- ▶ States variables:
 - Local item or channel level, here denoted C_i
 - Global/system level, here denoted S
 - Temporary states (local or global), here global and denoted A and B
- ▶ Global assertions, which are statements that are related to global state variables (by pointing at global states)
- ▶ $!!A = In.C_i$ means that the temporary state A is assigned the value of the local state C_i

Reliability block diagram driven PN



Analysis (of reliability)

Software tool is more or less required in order to use PNs for reliability analysis. Some alternatives are:

- ▶ SHARPE developed by Duke University
- ▶ GRIF developed by Total
- ▶ Matlab (?)

For repairable systems, availability (or unavailability) may be found by calculating the (average) time spent in places p_{SW} and p_{SF} .