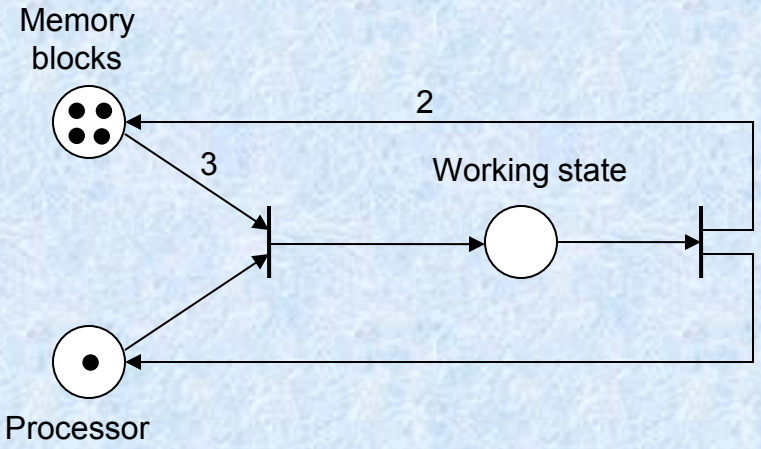
Faculty of Transportation Sciences
Department of Transport
Telematics

Systems analysis
6th lecture
Petri nets,
Decision tables

Zuzana Bělinová, Zdeněk Votruba

Petri nets

- Used for system modelling
- Mathematical construct based on graph theory
- Easily managed by computers
- Useful for description of processes



Petri nets – summary

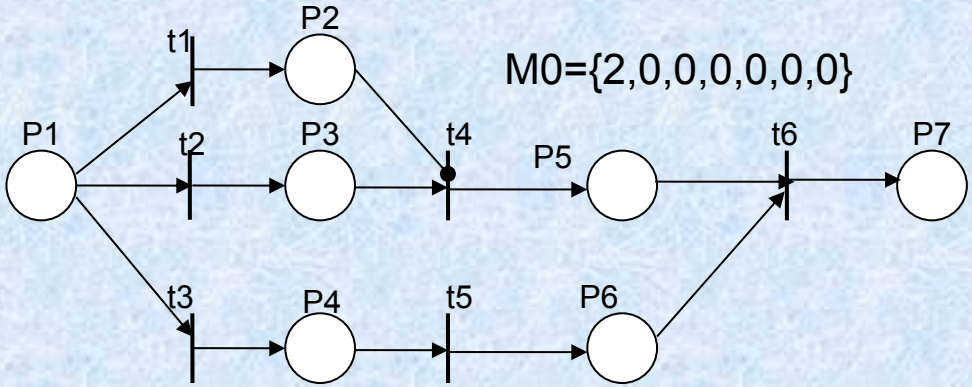
Definition: PN := (P,T,F,B,M^0)

- **P** – set of all places in the PN (they represent possible states of the system)
- **T** –set of all transitions in the PN (they represent possible events)
- **F** – forward matrix
- **B** – backward matrix **M** = $\{m_1, m_2, \ldots, m_n\}$ is the marking vector (contains list of places that are active in the current state of PN
- **M^0** – is the vector of initial marking

Rules

- 1. A transition is said to be enabled if each input place has at least as many tokens as the weight of the arc connecting them.
- 2. Enabled transition may be fired by removing from each input place the number of tokens equal to the weight of the arc connecting them.
- 3. When the transition is fired, tokens will be added to the output places connected to the transition.The number of tokens to be added to each output place is equal to the weight of the arc joining them.

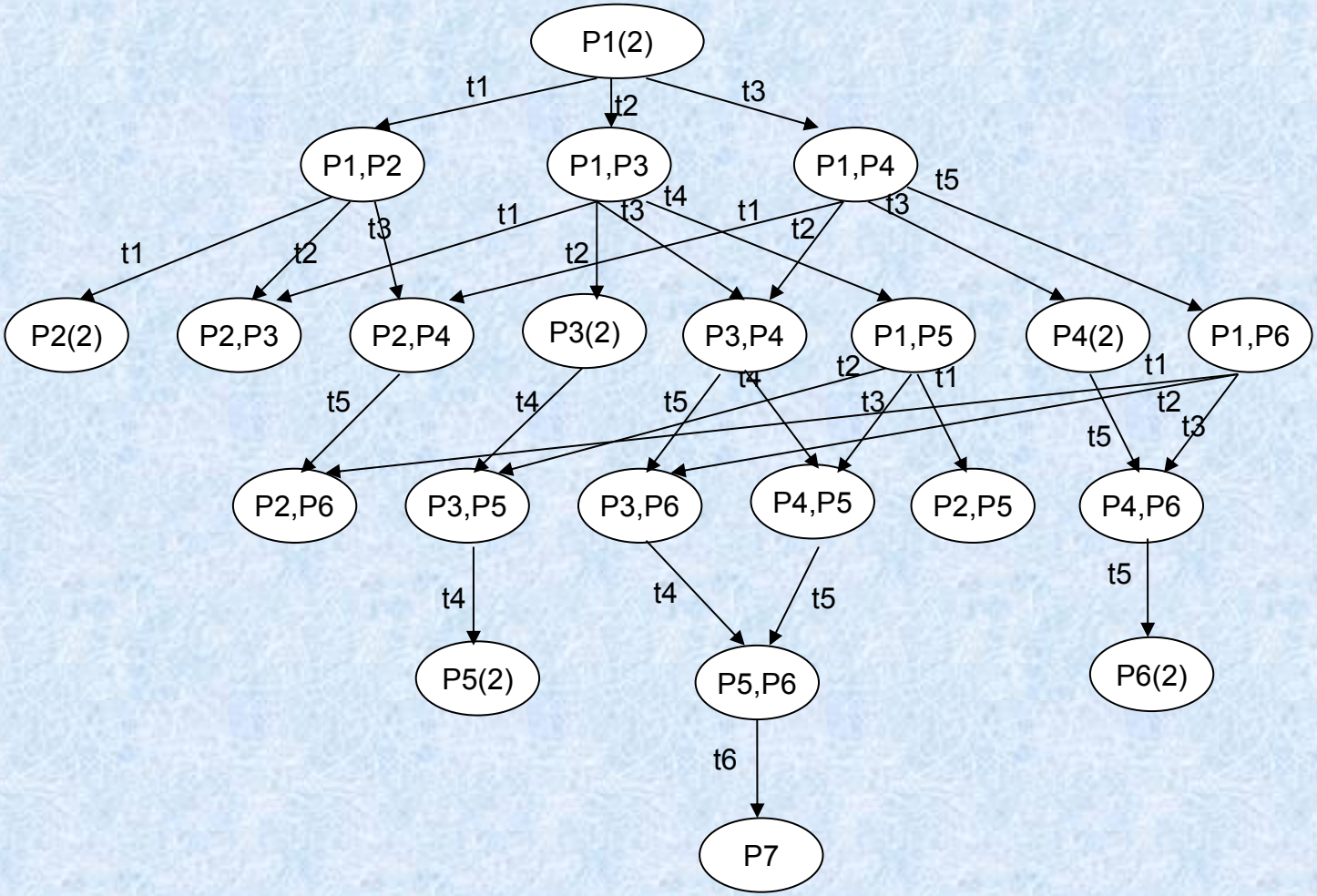
PN State Transition Diagram - example



F	T1	T2	T3	T4	T5	T6
P1						
P2						
P3						
P4						
P5						
P6						
P7						

B	T1	T2	T3	T4	T5	T6
P1						
P2						
P3						
P4						
P5						
P6						
P7						

PN State Transition Diagram – example - transition state diagram



Petri nets – development

- Introducing capacity of places – maximal namber of tokens in place
- Introducing conditions for enabling transitions
- Coloured Petri nets – instead of black tokens tokens of different colours

  - possible to use data types and complex data manipulation

  - Each token has attached a data value called the token colour.
  - The token colours can be investigated and modified by the occurring transitions.
- Adding other elements, such as variables, type declarations, etc.

Decision tables

Decision tables

- Tool for modeling sets of logical terms
- Similar to if-then rules
- Work with several conditions at the same time
- Advantages
  - Possibility of modelling complex alternative processes
  - Suitable for algorithms with many decisions
  - Easily computerized

Decision tables - structure

- 4 quadrants

List of conditions	Combinations of conditions
List of actions	Combination of actions

- DTs describe which actions are to be done under particular combination of conditions
- In the „Combination of conditions“ quadrant different **possible** combinations of conditions are stated and to each combination there are in the appropriate column in the quadrant „combination of actions“ - markes actions to be done

Decision tables - creating

- 1. Identify conditions and their values
- 2. Identify possible actions
- 3. Enter combinations of conditions
- 4. Define actions for each combination of conditions
- 5. Verify the table and if possible, simplify the table

Types of decision tables

- According the inputs
  - With limited-entry (binary input)
  - With extended entry (each condition has more possible results)
  - With mixed inputs
- According cohesion with other DT
  - Open – there are more DT connected (using links from one to another)
  - Closed – one self-sustaining table
- Full tables – with limited binary entry, number of rules is 2^n
  - More often there are less rules because of redundancy or contradictory

Decision tables - examples

Limited entry table - Buying car

Have lots of money	0	0	1	1
Like luxury cars	0	1	0	1
Buy e.g. Škoda	1			
Save money		1		
Buy e.g. BMW			1	
e.g. Bentley				1

Multiple-state entry

Average study grade A	1	0	0
B	0	1	0
other	0	0	1
Scholarship 100€	0	1	0
200 €	1	0	0
-	0	0	1

Simplified:

Grade	A	B	C-E
Schol. 100€		1	
200 €	1		
-			1

Decision tables – examples

- Contradictory entry

  e.g. choosing signal plan for traffic control

Day: public holiday	1	0	0	0	….	0
Monday	-	1				
Tuesday	-		1			
Wednesday	-			1		
….	-					
Sunday	-					
Use signal plan A	1					
B		1				
C			1	1		

Decision table – example – open tables

Assessment

DT 1

Presence at lectures	0	0	0	0	1	1	1	1
Correct homeworks delivered	0	0	1	1	0	0	1	1
Quality seminary work delivered	0	1	0	1	0	1	0	1
Additional work assigned	1	1	1	1	0	0	0	0
Go to DT 2	1	1	1	1	0	0	0	0
Repeat DT 1	0	0	0	0	1	1	1	0
Give assessment	0	0	0	0	0	0	0	1
Go to DT 3	0	0	0	0	0	0	0	1

DT 2

Delivered additional work	0	1	1	1
Correct homeworks delivered	-	0	1	1
Quality seminary work delivered	-	-	0	1
Repeat DT 2	1	1	1	0
Give assessment	0	0	0	1
Go to DT 3	0	0	0	1

Exam

DT 3

Perfect knowledge	1	0	0	0	0
…					
…					
Unsufficient knowledge	0	0	0	0	1
grade A	1	0	0	0	0
….					
….					
Repeat DT 3	0	0	0	0	1

Thank you for your attention

References

- Petri nets: http://www.informatik.uni-hamburg.de/TGI/PetriNets/