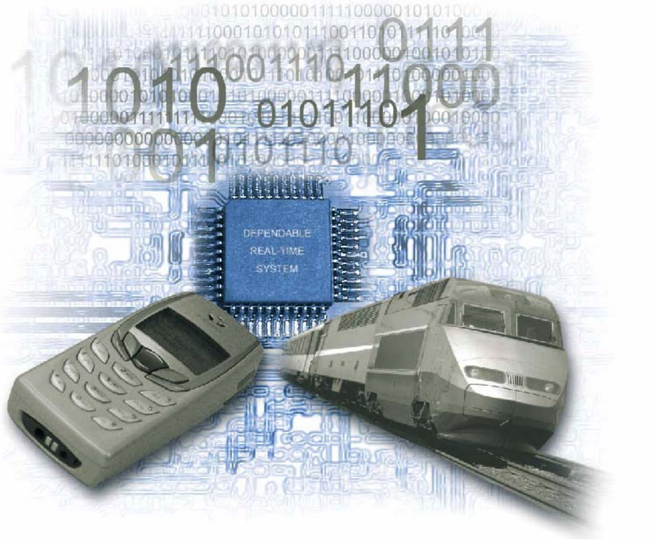
Embedded Systems 6

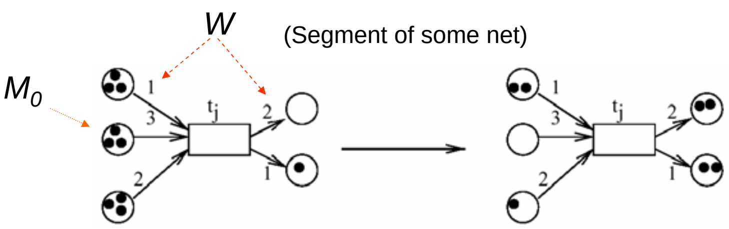


Place/transition nets REVIEW

Def.: (P, T, F, K, W, M_0) is called a **place/transition net (P/T net)** iff

1. $N=(P,T,F)$ is a **net** with places $p \in P$ and transitions $t \in T$
2. $K: P \rightarrow (\mathbf{N_0} \cup \{\omega\}) \setminus\{0\}$ denotes the **capacity** of places (ω symbolizes infinite capacity)
3. $W: F \rightarrow (\mathbf{N_0} \setminus\{0\})$ denotes the **weight of graph edges**
4. $M_0: P \rightarrow \mathbf{N_0} \cup\{\omega\}$ represents the **initial marking** of places

(Segment of some net)

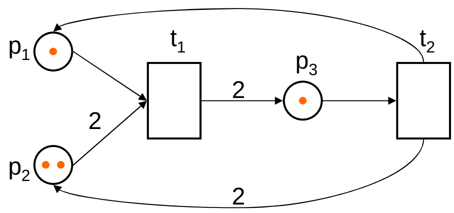


defaults:
K = ω
W = 1

In the following: assume initial marking is finite, capacity ω.

Unbounded Petri net

REVIEW



Boundedness

REVIEW

Theorem 1: A P/T net (with finite initial marking) is bounded iff its reachability set is finite.

Theorem 2: A P/T net is unbounded iff there exist two reachable markings M, M', such that M[*>M' and M' > M.

Algorithm for deciding boundedness

REVIEW

- Explore RG(M_0) depth-first:
  - If there exists a marking M‘ on the stack such that M‘<M, stop with result UNBOUNDED;
- If entire graph explored, return BOUNDED.

Weak Petri net computers

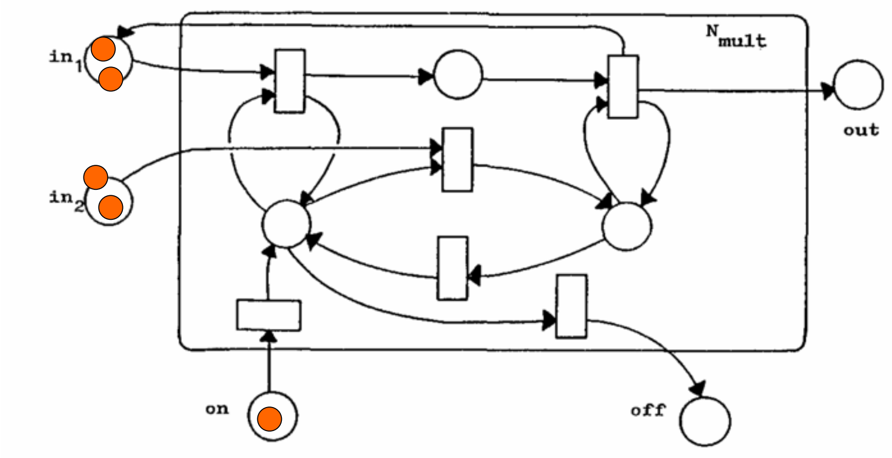
REVIEW

A P/T net with
r distinguished input places (in_i),
a finite number of internal places s_i,
one extra output place (out),
one extra start place (on), and
one extra stop place (off)
is called a **weak Petri net computer** for the function f: $N^r \rightarrow N$
iff there exists for each $x \in N^r$ an initial marking M_x
such that

- M_x(on)=1 and $M_{\mathbf{x}}(in_i)=\mathbf{x_i}$ for $1 \leq x_i \leq r$;
- M_x(out)=M_x(off)=0;
- $M_x(s_i)=0$;
- For all reachable markings $M \neq M_x$, M(on)=0 and $1 \leq$ M(off) $\leq$ 1 and M(out) $\leq$ f(x);
- For all reachable markings $M \neq M_x$, if M(off)=1 then M is dead;
- For all $0 \leq k \leq f(x)$, there exists a reachable marking M such that M(out)=k and M(off)=1.

Multiplication

REVIEW



Source: Matthias Jantzen, Complexity of Place/Transition Nets (1986)

Computation of Invariants

REVIEW

We are interested in subsets *R* of places whose number of labels remain invariant under transitions,

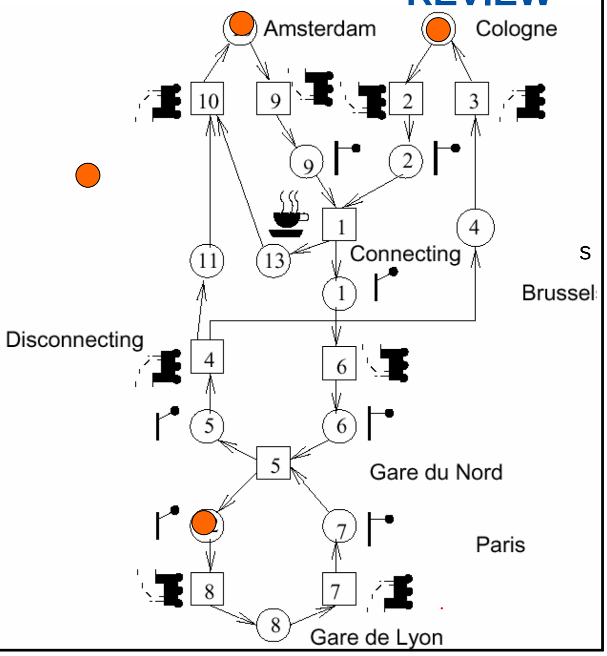
•e.g. the number of trains commuting between Amsterdam and Paris (Cologne and Paris) remains constant

Important for correctness proofs, e.g. the proof of liveness

Example Thalys trains: more complex

REVIEW

- Thalys trains between Cologne, Amsterdam, Brussels and Paris.
- Synchronization at Brussels and Paris



Application to Thalys example

$\underline{N}^T \underline{c}_R = 0$, with $\underline{N}^T$=

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}	p_{12}	p_{13}
t_1	1	-1							-1				1
t_2		1	-1										
t_3			1	-1									
t_4				1	-1						1		
t_5					1	-1	-1					1	
t_6	-1					1							
t_7							1	-1					
t_8								1				-1	
t_9									1	-1			
t_{10}										1	-1		-1

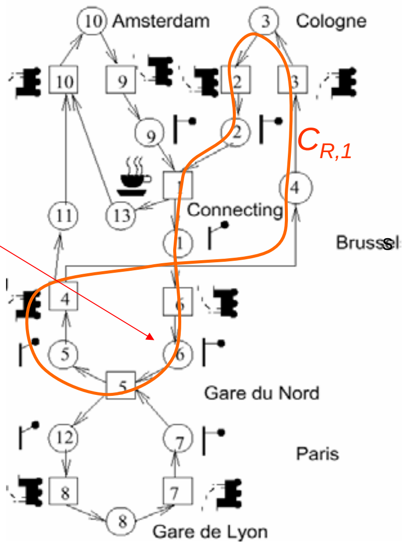
$$c_{R,1} = (1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0)$$

Interpretation of the 1st invariant

$$c_{R,1} = (1\,1\,1\,1\,1\,1\,0\,0\,0\,0\,0\,0\,0)$$

Characteristic vector describes places for Cologne train.

We proved that: the number of trains along the path remains constant.



Application to Thalys example

$\underline{N}^T \underline{c}_R = 0$, with $\underline{N}^T$=

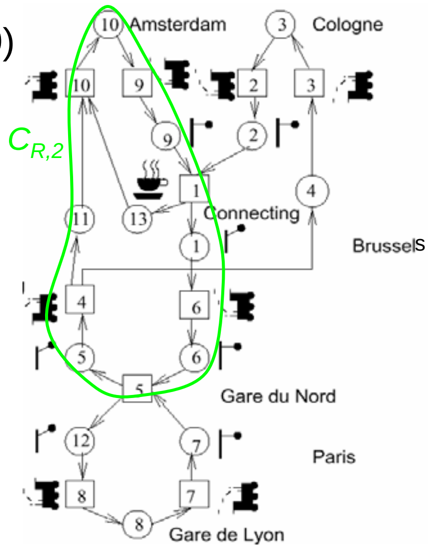
	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}	p_{12}	p_{13}
t_1	1	-1							-1				1
t_2		1	-1										
t_3			1	-1									
t_4				1	-1						1		
t_5					1	-1	-1					1	
t_6	-1					1							
t_7							1	-1					
t_8								1				-1	
t_9									1	-1			
t_{10}										1	-1		-1

$$c_{R,2} = (1,0,0,0,1,1,0,0,1,1,1,0,0)$$

Interpretation of the 2nd invariant

$$c_{R,2} = (1,0,0,0,1,1,0,0,1,1,1,0,0)$$

We proved that:
None of the Amsterdam trains gets lost.



Application to Thalys example

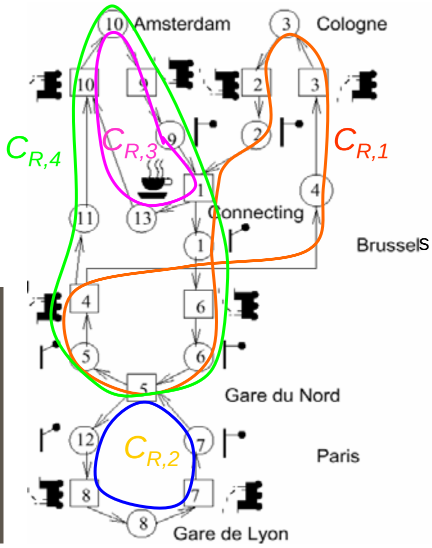
$\underline{N}^T \underline{c}_R = 0$, with $\underline{N}^T$=

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}	p_{12}	p_{13}
t_1	1	-1							-1				1
t_2		1	-1										
t_3			1	-1									
t_4				1	-1						1		
t_5					1	-1	-1					1	
t_6	-1					1							
t_7							1	-1					
t_8								1				-1	
t_9									1	-1			
t_{10}										1	-1		-1

$$c_{R,2} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Solution vectors for Thalys example

$$c_{R,1} = (1\,1\,1\,1\,1\,1\,0\,0\,0\,0\,0\,0\,0)$$
$$c_{R,2} = (0\,0\,0\,0\,0\,0\,1\,1\,0\,0\,0\,1\,0)$$
$$c_{R,3} = (0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,0\,0\,1)$$
$$c_{R,4} = (1\,0\,0\,0\,1\,1\,0\,0\,1\,1\,1\,0\,0)$$



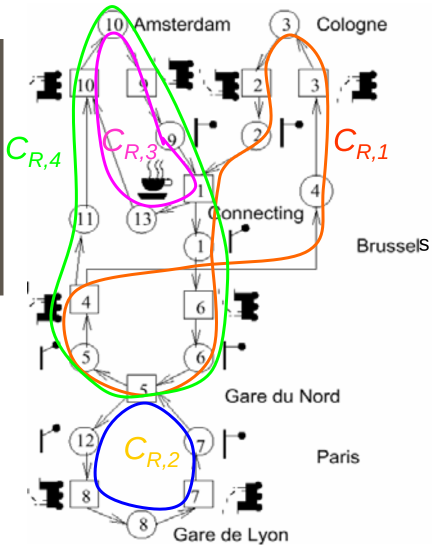
We proved that:

- the number of trains serving Amsterdam, Cologne and Paris remains constant.
- the number of train drivers remains constant.

Solution vectors for Thalys example

It follows:

- each place invariant must have at least one label at the beginning, otherwise "dead"
- at least three labels are necessary in the example



Invariants & boundedness

- A net is **covered** by place invariants
  iff every place is contained in some invariant.
- **Theorem 4:**
  a) If R is a place invariant and $p \in R$, then p is bounded.
  b) If a net is covered by place invariants then it is bounded.

a) $c_R M_0^T = c_R M^T$ for every reachable marking M

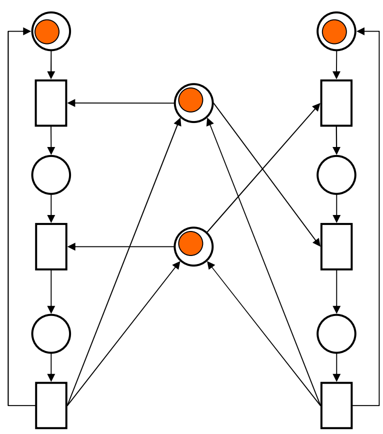
$c_R M^T \geq c_R(p) \cdot M(p) \geq M(p)$

a) $\Rightarrow$ b) .

Deadlock

REVIEW

- A **dead marking (deadlock)** is a marking where no transition can fire.
- A Petri net is **deadlock-free** if no dead marking is reachable.



Liveness

REVIEW

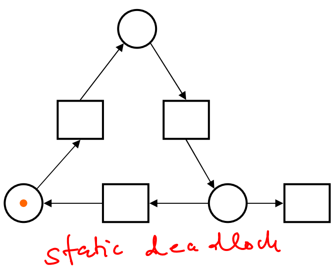
- A **transition** t is **dead** at M if no marking M‘ is reachable from M such that t can fire in M‘.
- A **transition** t is **live** at M if there is no marking M‘ reachable from M where t is dead.
- A **marking** is **live** if all transitions are live.
- A **P/T net** is **live** if the initial marking is live.

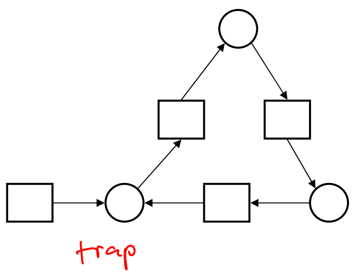
Observations:

- A live net is deadlock-free.
- No transition is live if the net is not deadlock-free.

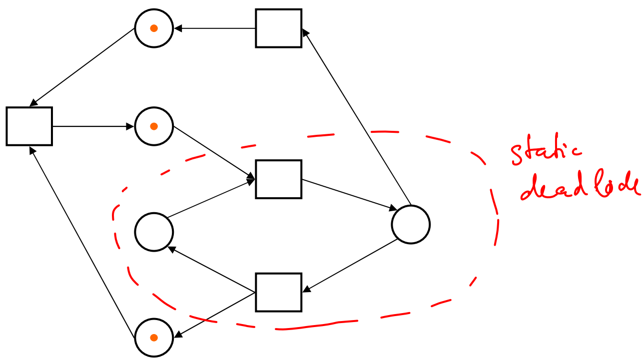
Structural properties: deadlock-traps

- A place set *S* is a **(static) deadlock** if every transition that adds token ~~from~~ to S also removes token from *S*.
- A place set *S* is a **trap** if every transition that removes token from S also adds token to *S*.





Empty structural deadlocks and marked traps

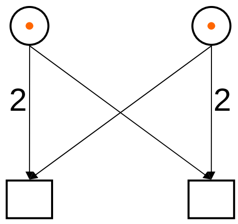


- Empty structural deadlocks are never re-marked;
- Marked traps are never emptied.

Sufficiently marked places

A place is called sufficiently marked if there are enough token for one of the outgoing transitions.

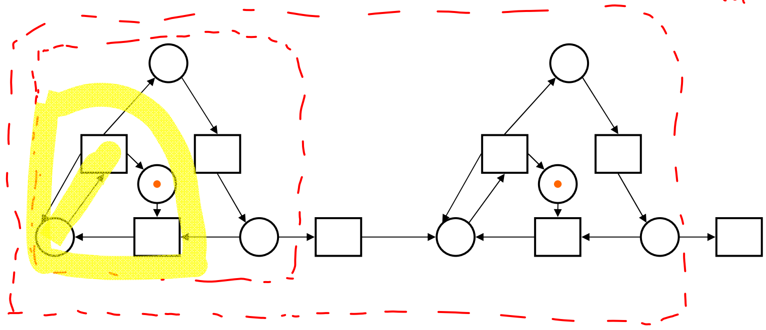
- Define
  $W^-(p) = \min \{ W(p,t) \mid (p,t) \in F \}$ if there exists a $(p,t) \in F$
  and 0 otherwise
- Place p is **sufficiently marked** in marking M, if $M(p) \geq W^-(p)$
- A set of places is sufficiently marked if it contains a sufficiently marked place.



Deadlock-Trap Property

▪ A P/T has the **deadlock-trap property**, if every (static) deadlock contains a trap that is sufficiently marked in M_0.

⟺ if the maximal trap of every minimal (static) deadlock is sufficiently marked in M_0.



Deadlock-Trap Property

Theorem 5:
Every homogeneous P/T net with non-blocking weights that has the deadlock-trap property is deadlock-free.

Homogeneous: For each place, all outgoing edges have the same weight.

Non-blocking weights: $W^+(p) \geq W^-(p)$

- $W^-(p)$= min { $W(p,t)$ | $(p,t) \in F$ } if there exists a $(p,t) \in F$ and 0 otherwise
- $W^+(p)$= min { $W(t,p)$ | $(t,p) \in F$ } if there exists a $(t,p) \in F$ and 0 otherwise

Proof of Theorem 5

Theorem 5: Every homogeneous P/T net with non-blocking weights that has the deadlock-trap property is deadlock-free.

- Suppose the net has a deadlock; i.e. there is a reachable dead marking M.
- Let $D := \{ p \in P \mid M(p) < V^-(p) \}$ be the set of places that are insufficiently marked in M
- $D \neq \emptyset$ (otherwise all transitions would be enabled)
- D is a static deadlock.

$\longrightarrow$

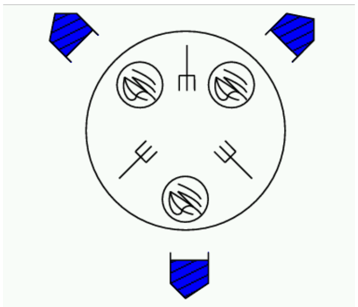
- D is a static deadlock.
  - Suppose t is a transition that adds token to D
    $\Rightarrow$ t is not enabled $\rightarrow$ one of its precond. is insufficiently labeled $\rightarrow$ t takes token from D.
- Now suppose that D contains a trap that is sufficiently marked in M_0
  $\Rightarrow$ some place in trap is sufficiently marked in M

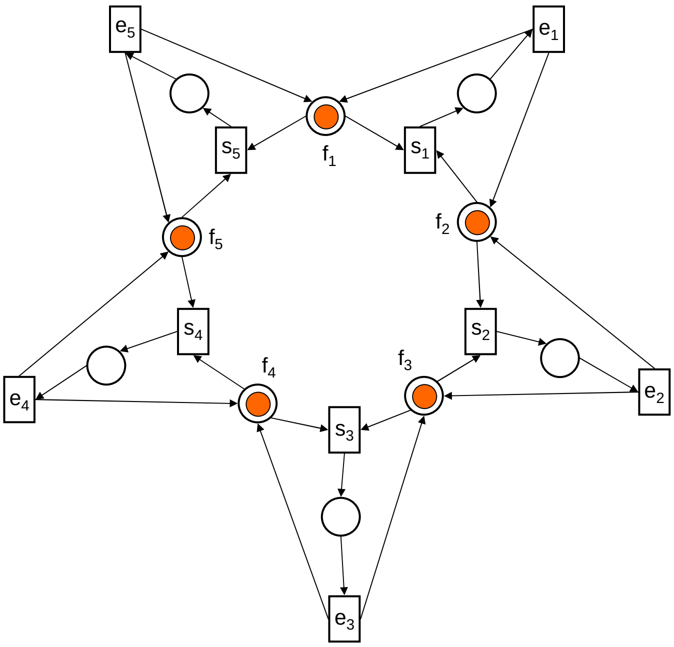
  ↯

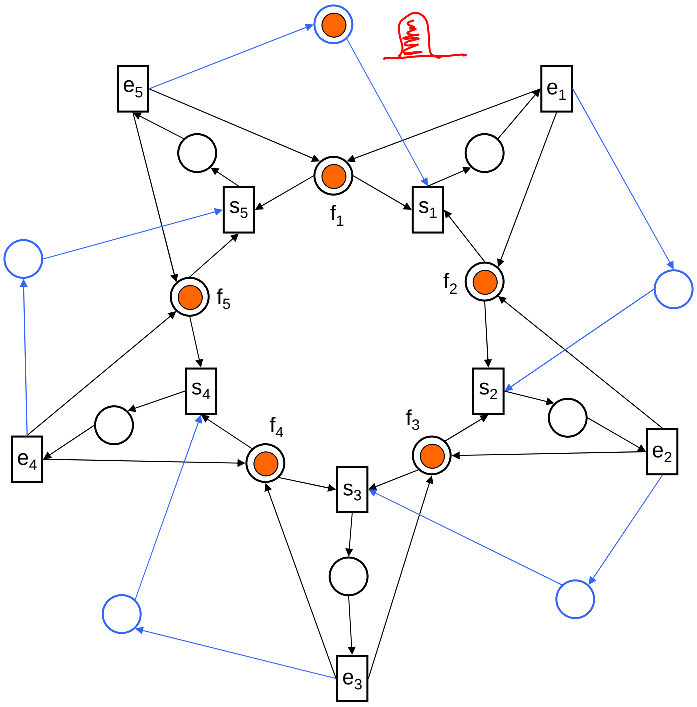
Fairness

Dining philosophers problem

- *n>1* philosophers sitting at a round table;
- *n* forks,
- *n* plates with spaghetti;
- philosophers either thinking or eating spaghetti (using left and right fork).
- 2 forks needed!







Executions

- Let w = (t_i), i ≥0, an infinite sequence of transitions.
- We call w an **execution** of the Petri net if there exists an infinite sequence of markings (M_i), i ≥0, starting with the initial marking M_0, such that
  M_0 [t_0> M_1 [t_1> M_2 [t_2> …
- Set of all exections of *N*: L(*N*)

Emptiness

- **Theorem 6:** Emptiness of L(*N*) is decidable.

· Is N unbounded? Yes → Nonempty

No → Does RG(N) contain a loop?

Yes → Nonempty

No → Empty.

Fairness

Let *N* be a Petri net and *w* an execution of *N*.

- *w* is **impartial** with respect to a set of transitions *T* iff every transition in *T* occurs infinitely often in *w*.
- *w* is **just** with respect to a set of transitions *T* iff every transition in *T* that is enabled in all except finitely many markings occurs infinitely often in *w*.
- *w* is **fair** with respect to a set of transitions *T* iff every transition in *T* that is enabled in infinitely many markings occurs infinitely often in *w*.

- *w* is **impartial** ⇒ *w* is **fair**
- *w* is **fair** ⇒ *w* is **just**

Persistent nets

- A pair of transitions t_1 and t_2 are **in conflict** at marking M iff t_1 and t_2 are enabled in M, but M is too small to satisfy both preconditions.
- A net is **statically conflict-free** if there is no marking where two transitions are in conflict;
- A net is **dynamically conflict-free (persistent)** if there is no reachable marking where two transitions are in conflict.

persistent

Persistent nets

Theorem 7: If the net is persistent, then every just execution is fair.

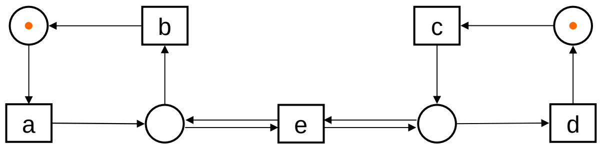
- Since the net is persistent, a transition can only be disabled by firing
- Let w be a just execution and let t be a transition that is enabled inf. often

Case 1: t is enabled continuously after some point ⇒ inf. often taken (justice)

Case 2: t becomes disabled inf. often ⇒ must be taken inf. often.

⇒ w is fair.

State Fairness



$(ab\,cd)^\omega$ is fair (not impartial)

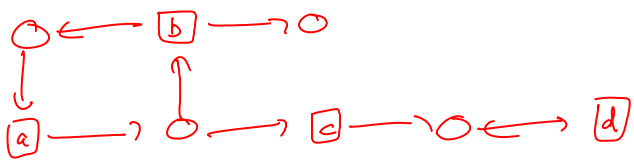
An execution $w=(t_i)$ is **state-fair** if, for all markings M and all transitions t that are enabled in M, the following holds:

If M is visited infinitely often, then *t* is taken infinitely often at M.

State Fairness

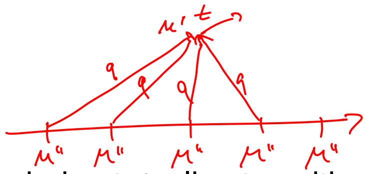
Theorem 8: Let *N* be a bounded net, *t* a live transition, and *w* a state-fair execution of *N*. Then *t* occurs infinitely often in *w*.

Does not hold for unbounded nets!



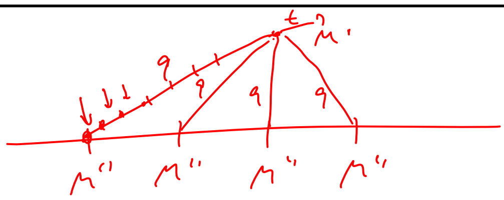
$(ab)^\omega$ state-fair, c never taken

Proof of Theorem 8



Theorem 8: Let *N* be a bounded net, *t* a live transition, and *w* a state-fair execution of *N*. Then *t* occurs infinitely often in *w*.

- t is live : from every reachable marking M $\exists M'$, $M\,[*\rangle\,M'$, s.t. t is enabled in M'.
- N is bounded: $\exists\, M'$ s.t. t is enabled in M' and for inf. many i, $M_i\,[*\rangle\,M'$
- This implies that M' is visited inf. often
  - Let M'' be some marking s.t. $M''\,[q\rangle\,M'$ and that is visited inf. often.



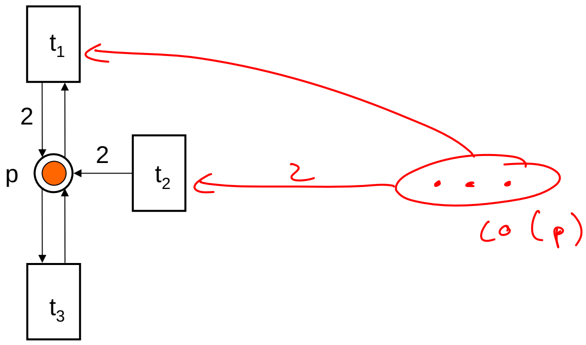
We prove by induction on q that every marking on the path from M'' to M' is visited inf. often.

- $q = \varepsilon$ ✓
- $q \to qt$. Let M''' be the marking reached after q. Since M''' is visited inf. often and the net is state-fair $\Rightarrow$ t is taken inf. often.

Since t is enabled in M' and net is state-fair $\Rightarrow$ t is taken inf. often.

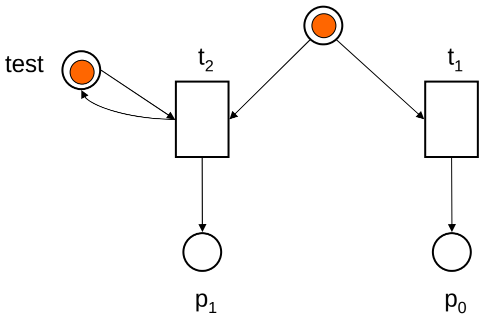
Extensions: finite capacities

- K(p)=4



Extensions: Petri nets with priorities

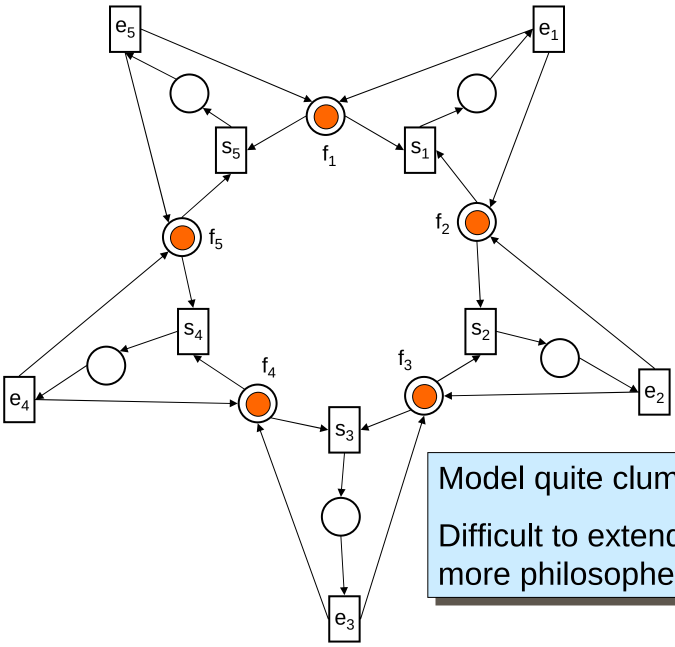
- $t_1 \langle t_2$: t_2 has higher priority than t_1.



- Petri nets with priorities are Turing-complete.

Extensions: Predicate/transition nets

- Goal: compact representation of complex systems.
- Key changes:
  - Tokens are becoming individuals;
  - Transitions enabled if functions at incoming edges true;
  - Individuals generated by firing transitions defined through functions
- Changes can be explained by folding and unfolding C/E nets,
  - ☞ semantics can be defined by C/E nets.



Model quite clumsy.

Difficult to extend to more philosophers.

Predicate/transition model of the dining philosophers problem

- Let x be one of the philosophers,
- let l(x) be the left fork of x,
- let r(x) be the right fork of x.

Token: individuals.

Semantics can be defined by replacing net by equivalent condition/event net.

Model can be extended to arbitrary numbers.