ETH zürich

dynamo.ethz.ch

Petri Nets (2)

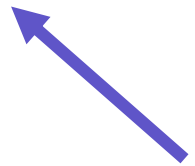
Jiahui Xu
DYNAMO group

Computer Engineering and
Networks Laboratory

Schedule

We have four exercise sessions:

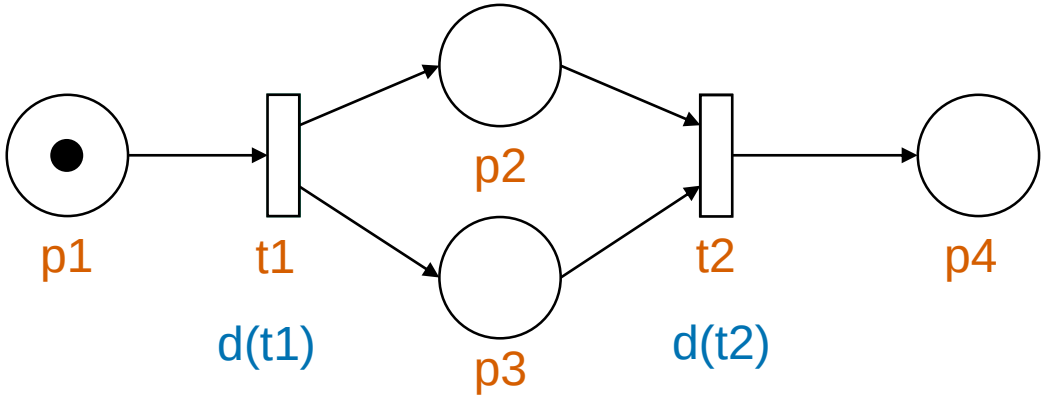
- 30.11.2023: set operations, characteristic functions, BDDs
- 07.12.2023: reachability analysis and temporal logic
- 14.12.2023: Petri nets
- 21.12.2023: time Petri nets



Today's plan:

- **Simulating time Petri nets**
- **Modeling arithmetic using Petri nets**
- **Q & A**
- 🎄

Adding time-dependent behaviors to Petri nets



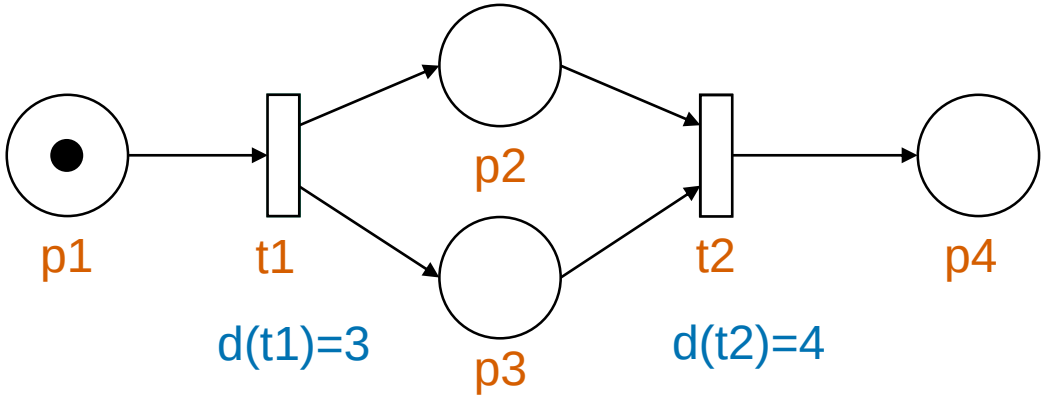
Set of places
Set of transitions
Set of flow relations
Initial marking

+ Delay functions

Marking + Event list

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net

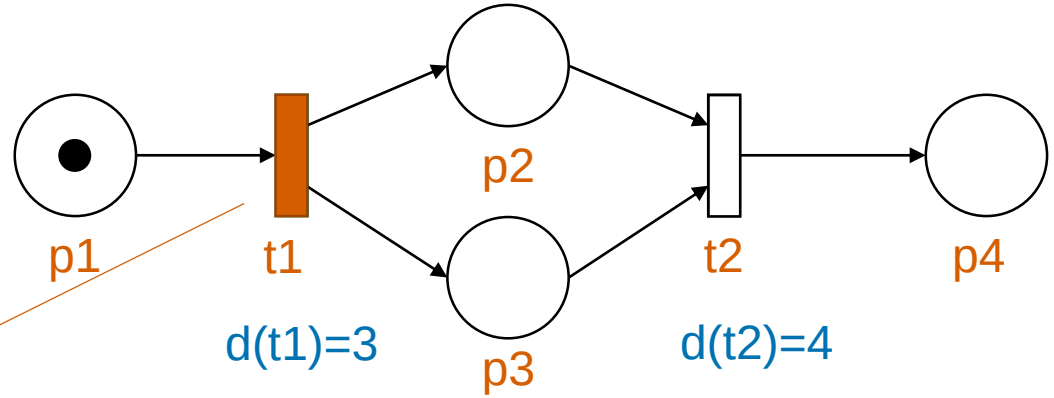


- **Step**: event index
- **Tau**: simulation time
- **Fired transition**: the fired transition
- **Event list**: a list of enabled transitions and their firing time

Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0, 0]	

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



@ tau = 0, t1 is activated:
Place a firing event on the event list at tau = 0 + d(t1).

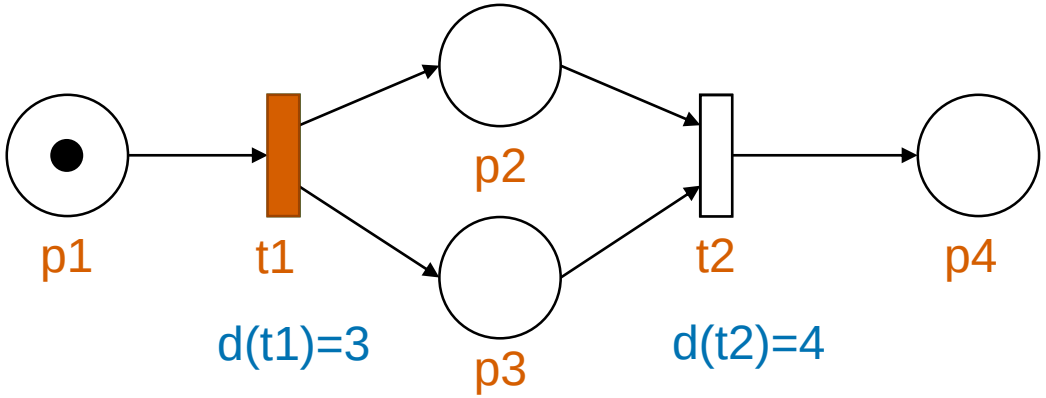
- **Step**: event index
- **Tau**: simulation time
- **Fired transition**: the fired transition
- **Event list**: a list of enabled transitions and their firing time

Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0, 0]	(t1, 3)

No transition can be fired @ tau = 0
Go to the next event @ tau = 3

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



- **Step**: event index
- **Tau**: simulation time
- **Fired transition**: the fired transition
- **Event list**: a list of enabled transitions and their firing time

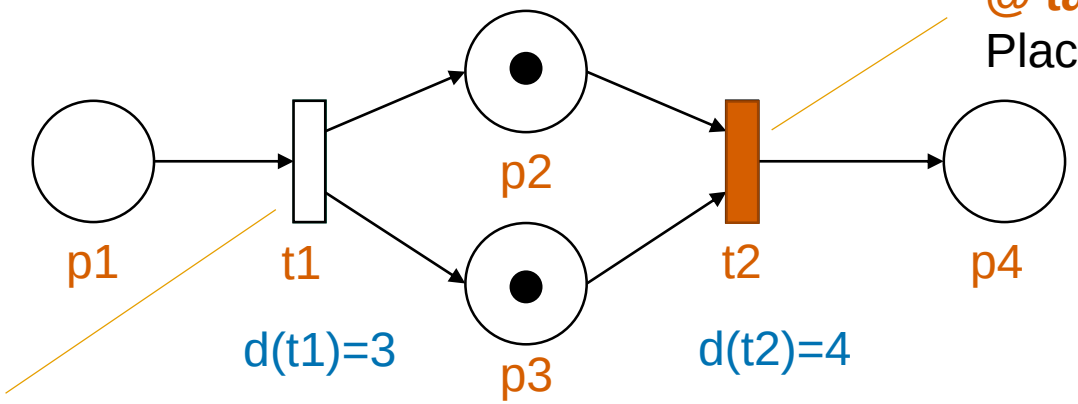
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0, 0]	(t1, 3)
1	3	t1		

No transition can be fired @ tau = 0
Go to the next event @ tau = 3

Determining which transition will be fired, update the marking vector, update the event list

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



@ tau = 3, t2 is activated:
Place an event @ tau = 3 + d(t2) = 7

@ tau = 3, t1 is fired, loses activation, and not activated again

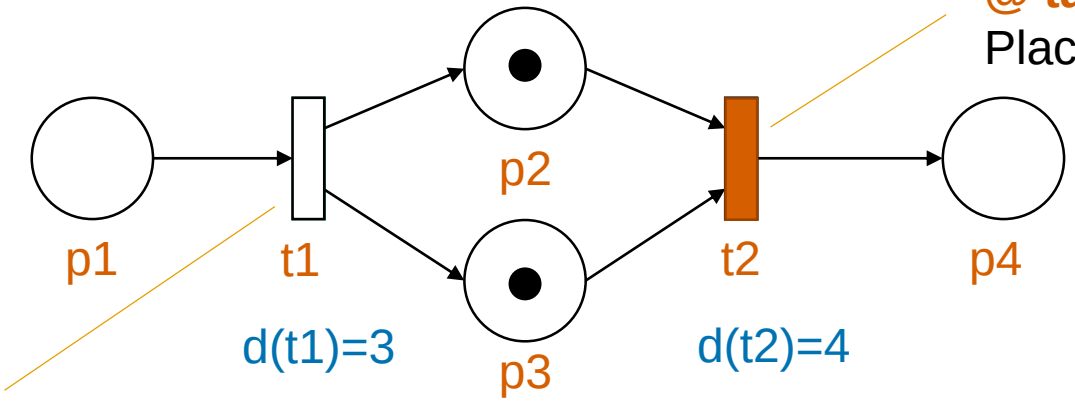
- **Step**: event index
- **Tau**: simulation time
- **Fired transition**: the fired transition
- **Event list**: a list of enabled transitions and their firing time

Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0, 0]	(t1, 3)
1	3	t1	[0, 1, 1, 0]	

Determining which transition will be fired, update the marking vector, update the event list

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



@ tau = 3, t2 is activated:
Place an event @ tau = 3 + d(t2) = 7

- **Step**: event index
- **Tau**: simulation time
- **Fired transition**: the fired transition
- **Event list**: a list of enabled transitions and their firing time

@ tau = 3, t1 is fired, loses activation, and not activated again

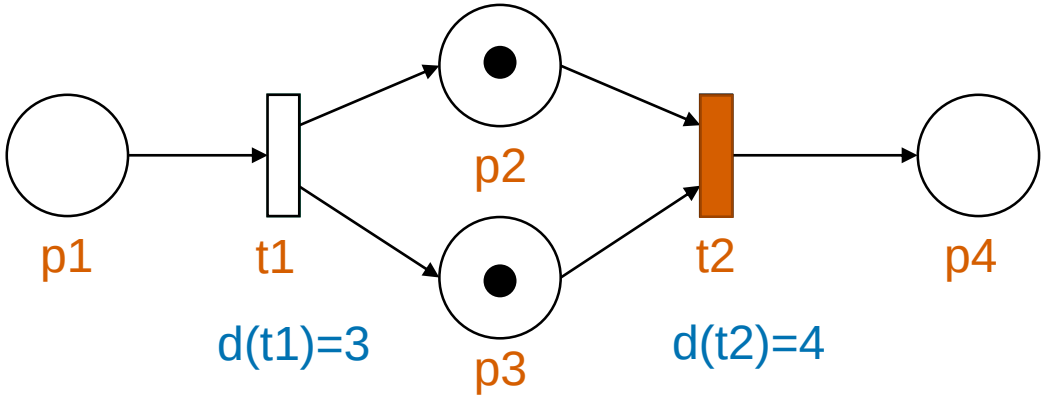
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0, 0]	(t1, 3)
1	3	t1	[0, 1, 1, 0]	(t2, 7)

No transition can be fired @ tau = 3
Go to the next event @ tau = 7

Determining which transition will be fired, update the marking vector, update the event list

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



- **Step**: event index
- **Tau**: simulation time
- **Fired transition**: the fired transition
- **Event list**: a list of enabled transitions and their firing time

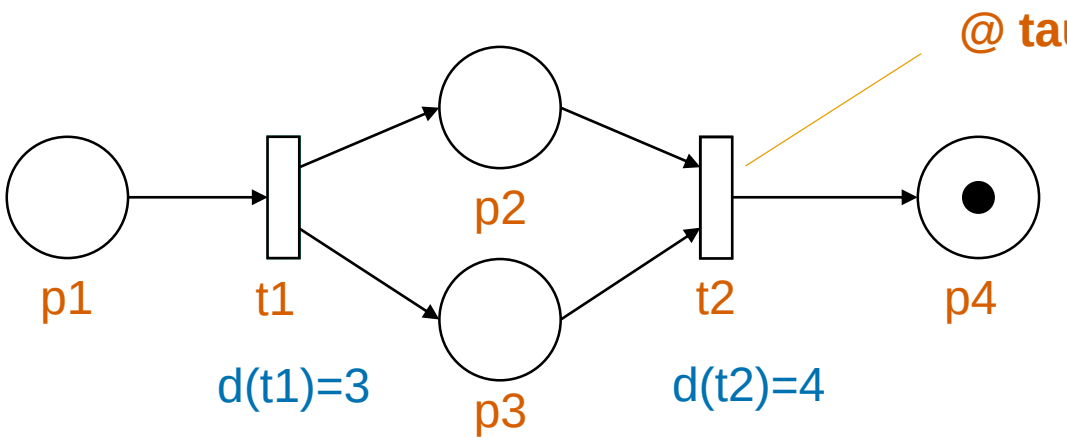
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0, 0]	(t1, 3)
1	3	t1	[0, 1, 1, 0]	(t2, 7)
2	7	t2		

No transition can be fired @ tau = 3
Go to the next event @ tau = 7

Determining which transition will be fired, update the marking vector, update the event list

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



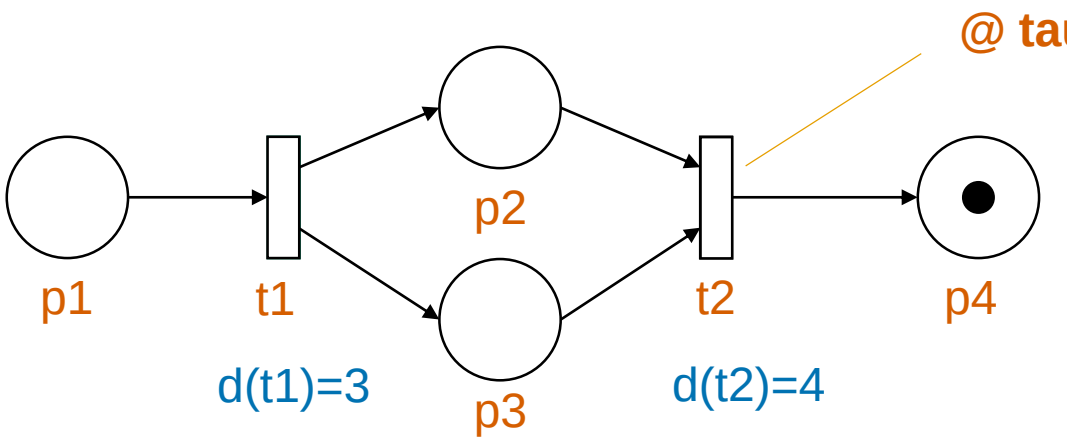
- **Step**: event index
- **Tau**: simulation time
- **Fired transition**: the fired transition
- **Event list**: a list of enabled transitions and their firing time

Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0, 0]	(t1, 3)
1	3	t1	[0, 1, 1, 0]	(t2, 7)
2	7	t2		

Determining which transition will be fired, update the marking vector, update the event list

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



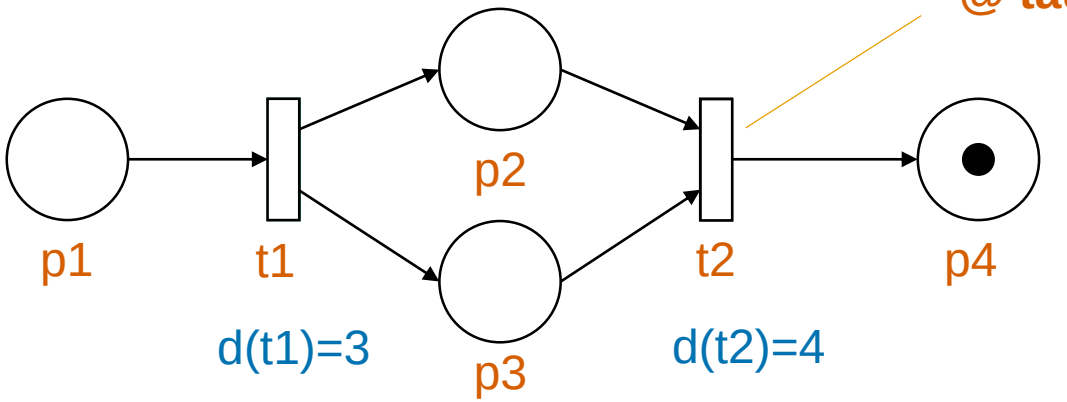
- **Step**: event index
- **Tau**: simulation time
- **Fired transition**: the fired transition
- **Event list**: a list of enabled transitions and their firing time

Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0, 0]	(t1, 3)
1	3	t1	[0, 1, 1, 0]	(t2, 7)
2	7	t2	[0, 0, 0, 1]	

Determining which transition will be fired, update the marking vector, update the event list

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



@ tau = 7, t2 is fired

- **Step**: event index
- **Tau**: simulation time
- **Fired transition**: the fired transition
- **Event list**: a list of enabled transitions and their firing time

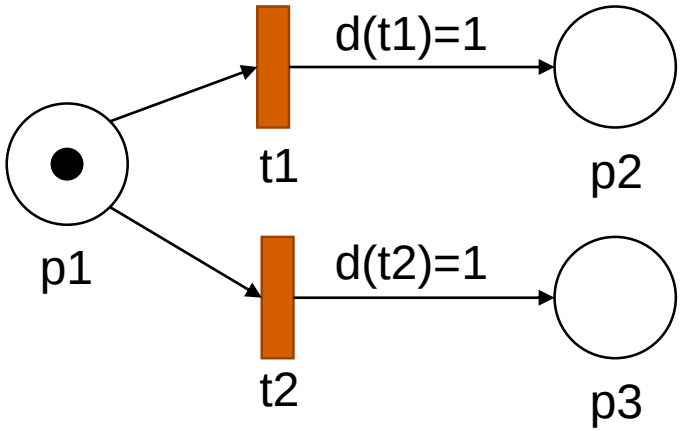
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0, 0]	(t1, 3)
1	3	t1	[0, 1, 1, 0]	(t2, 7)
2	7	t2	[0, 0, 0, 1]	-

@ tau = 7,
No event on the list

Determining which transition will be fired, update the marking vector, update the event list

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net

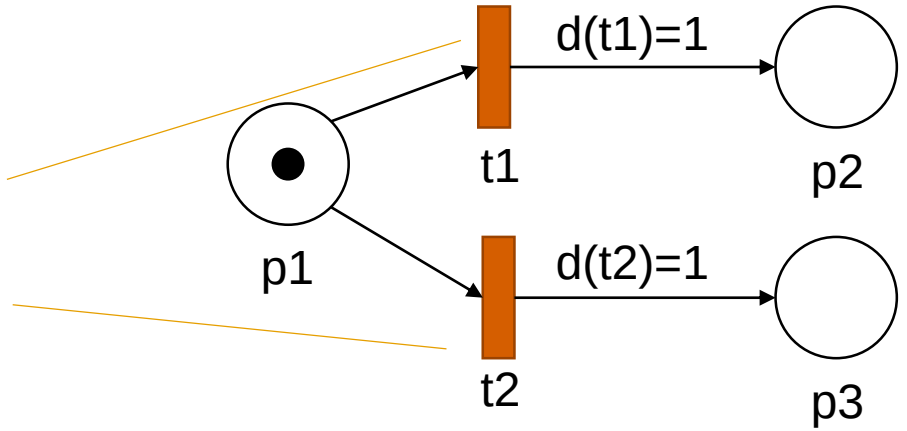


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0]	(t1, 1), (t2, 1)

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net

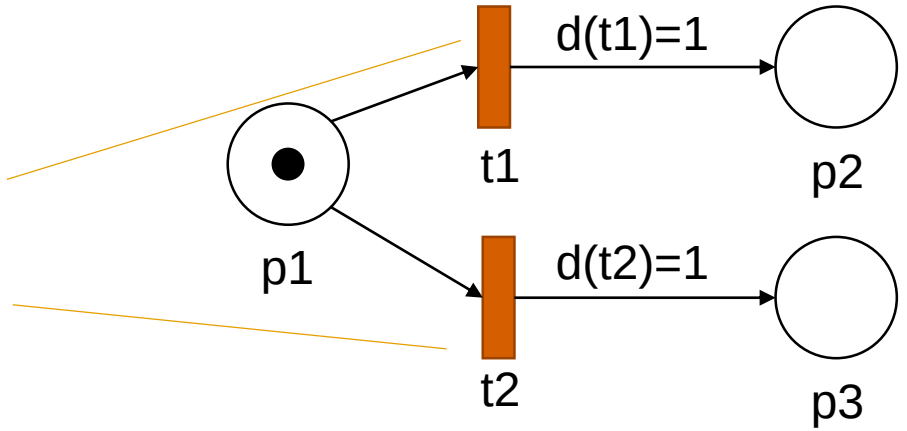
@ tau = 0: both t1 & t2 are activated, and schedule two firing events @ tau = 1



Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0]	(t1, 1), (t2, 1)

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



@ tau = 0: both t1 & t2 are activated, and schedule two firing events @ tau = 1

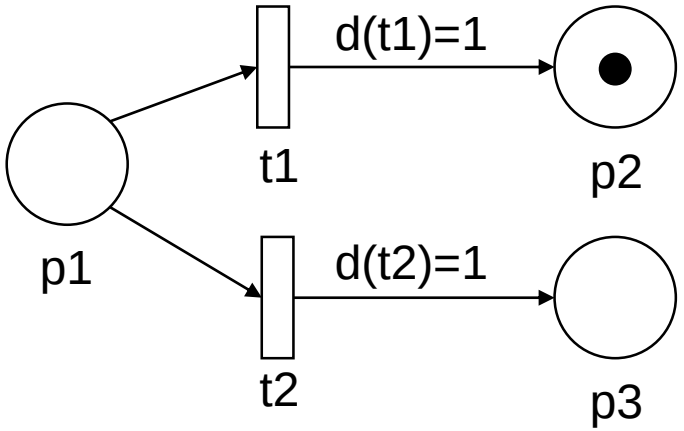
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0]	(t1, 1), (t2, 1)
1	1	?		

Both t1 and t2 are activated

Choose exactly one of them

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



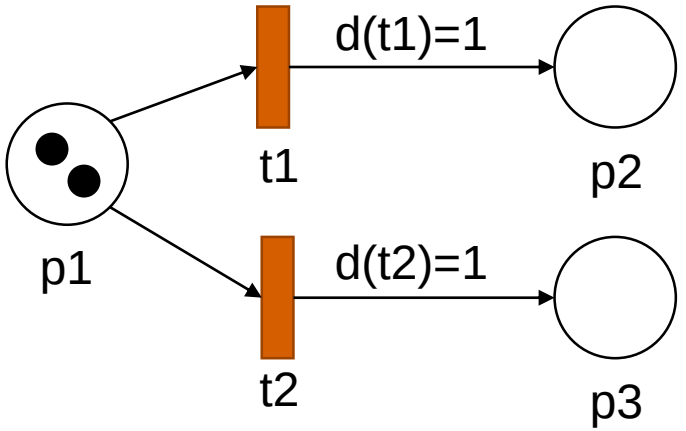
@ tau = 1: Fire t1; both t1 & t2 lose activation

Step	tau	Fired transition	Marking vector	Event list
0	0	-	[1, 0, 0]	(t1, 1), (t2, 1)
1	1	t1	[0, 1, 0]	-

A transition loses its activation whenever a token is removed from any of its input places!

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



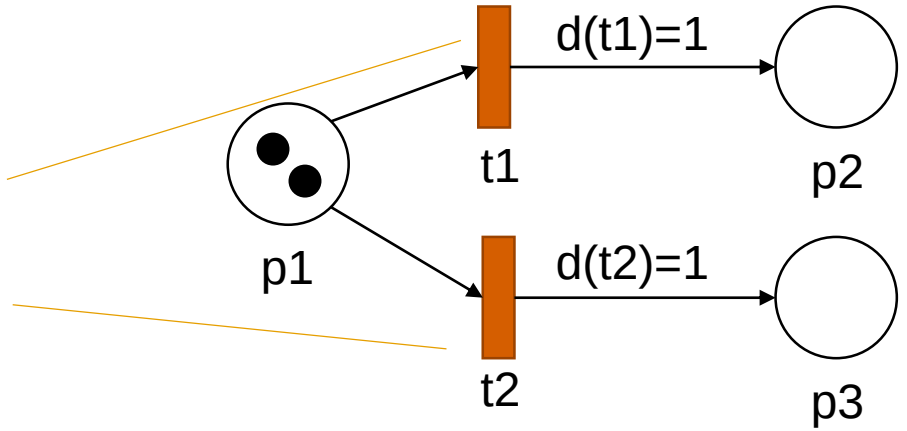
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[2, 0, 0]	

A transition loses its activation whenever a token is removed from any of its input places!

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net

@ tau = 0: both t1 & t2 are activated, and schedule two firing events @ tau = 1

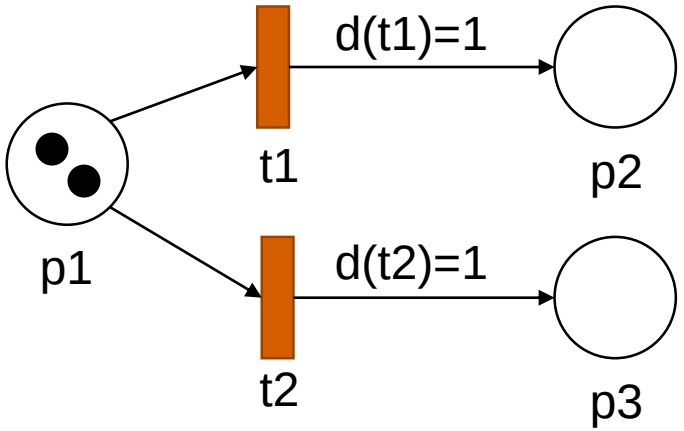


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[2, 0, 0]	(t1, 1), (t2, 1)

A transition loses its activation whenever a token is removed from any of its input places!

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net

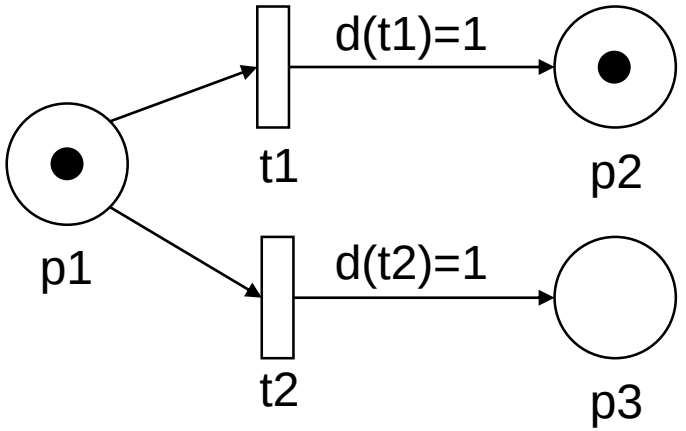


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[2, 0, 0]	(t1, 1), (t2, 1)
1	1	Choose t1		

A transition loses its activation whenever a token is removed from any of its input places!

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



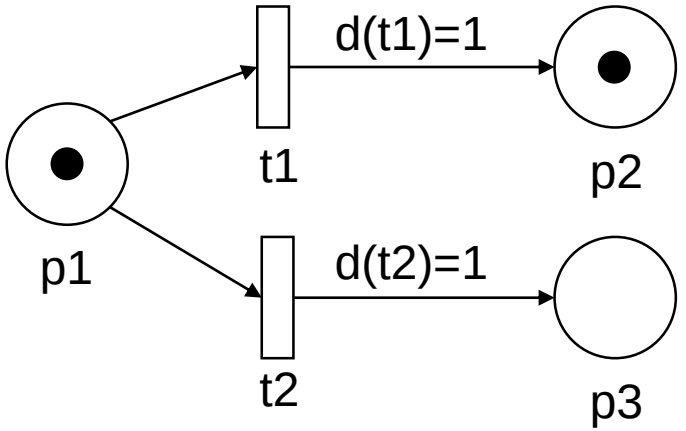
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[2, 0, 0]	(t1, 1), (t2, 1)
1	1	Choose t1	[1, 1, 0]	

A transition loses its activation whenever a token is removed from any of its input places!

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net

@ tau = 1: both t1 & t2 are **deactivated**

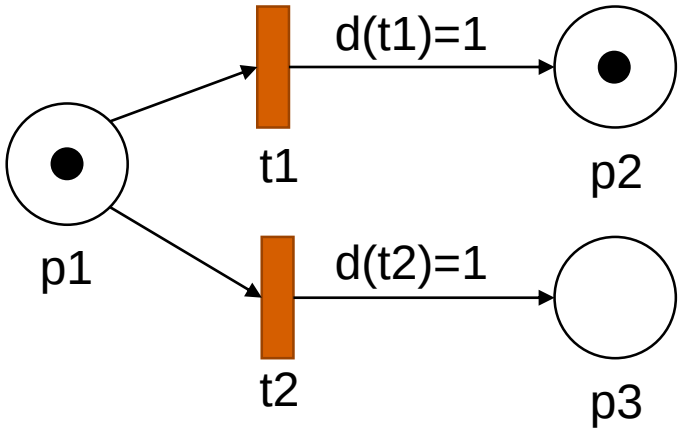


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[2, 0, 0]	(t1, 1), (t2, 1)
1	1	Choose t1	[1, 1, 0]	

A transition loses its activation whenever a token is removed from any of its input places!

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



@ **tau = 1**: both t1 & t2 are **deactivated**

@ **tau = 1**: both t1 & t2 are **reactivated**

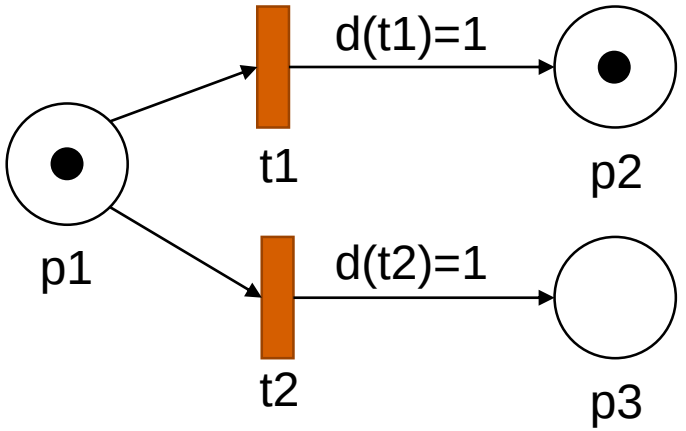
schedule two firing events @ tau = 2

Step	tau	Fired transition	Marking vector	Event list
0	0	-	[2, 0, 0]	(t1, 1), (t2, 1)
1	1	Choose t1	[1, 1, 0]	(t1, 2), (t2, 2)

A transition loses its activation whenever a token is removed from any of its input places!

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net

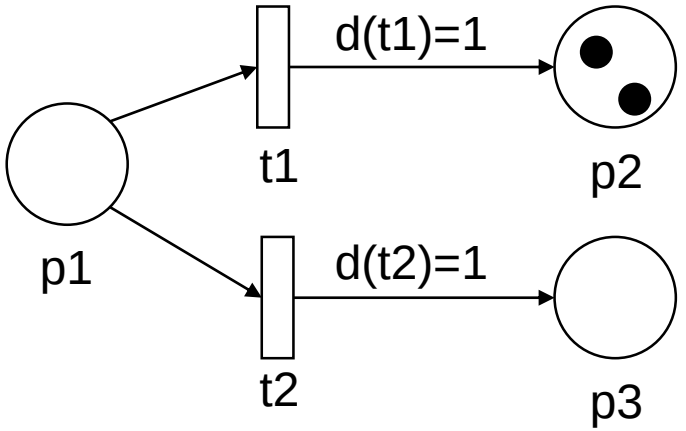


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[2, 0, 0]	(t1, 1), (t2, 1)
1	1	Choose t1	[1, 1, 0]	(t1, 2), (t2, 2)
2	2	Choose t1		

A transition loses its activation whenever a token is removed from any of its input places!

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net

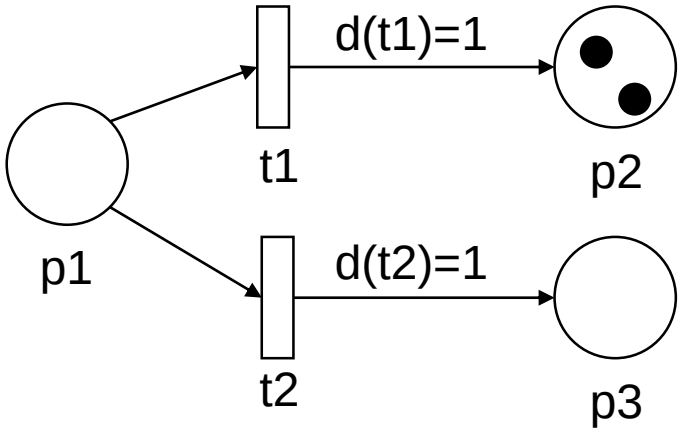


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[2, 0, 0]	(t1, 1), (t2, 1)
1	1	Choose t1	[1, 1, 0]	(t1, 2), (t2, 2)
2	2	Choose t1	[0, 2, 0]	

A transition loses its activation whenever a token is removed from any of its input places!

Adding time-dependent behaviors to Petri nets

Simulating a time Petri net



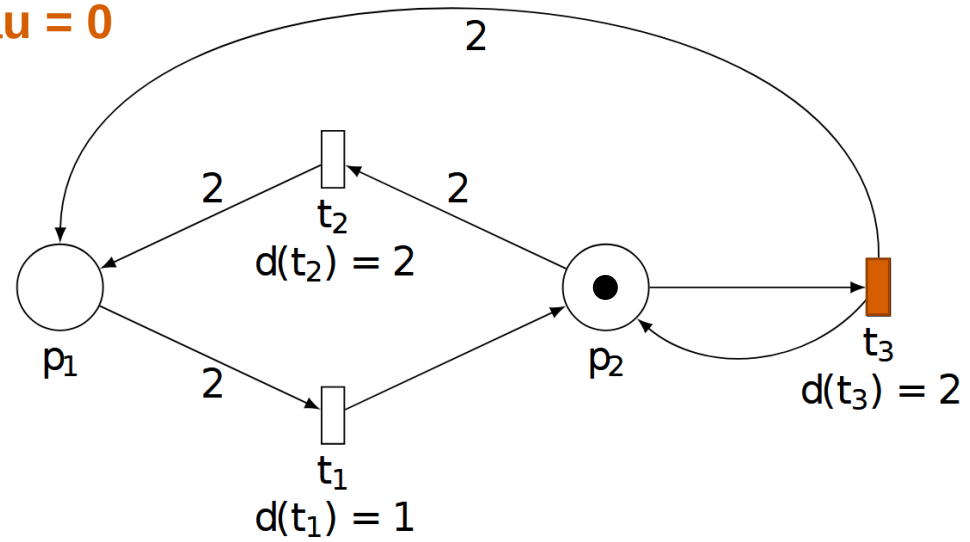
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[2, 0, 0]	(t1, 1), (t2, 1)
1	1	Choose t1	[1, 1, 0]	(t1, 2), (t2, 2)
2	2	Choose t1	[0, 2, 0]	-

A transition loses its activation whenever a token is removed from any of its input places!

Adding time-dependent behaviors to Petri nets

* When several transitions are enabled at the same time, choose the one with the smallest index first

@ tau = 0

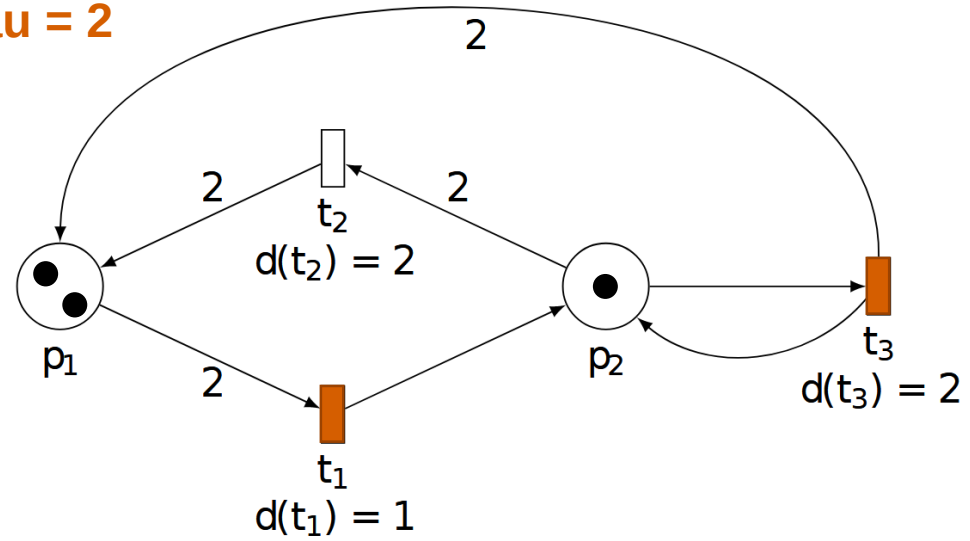


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[0, 1]	(t3, 2)

Adding time-dependent behaviors to Petri nets

* When several transitions are enabled at the same time, choose the one with the smallest index first

@ tau = 2

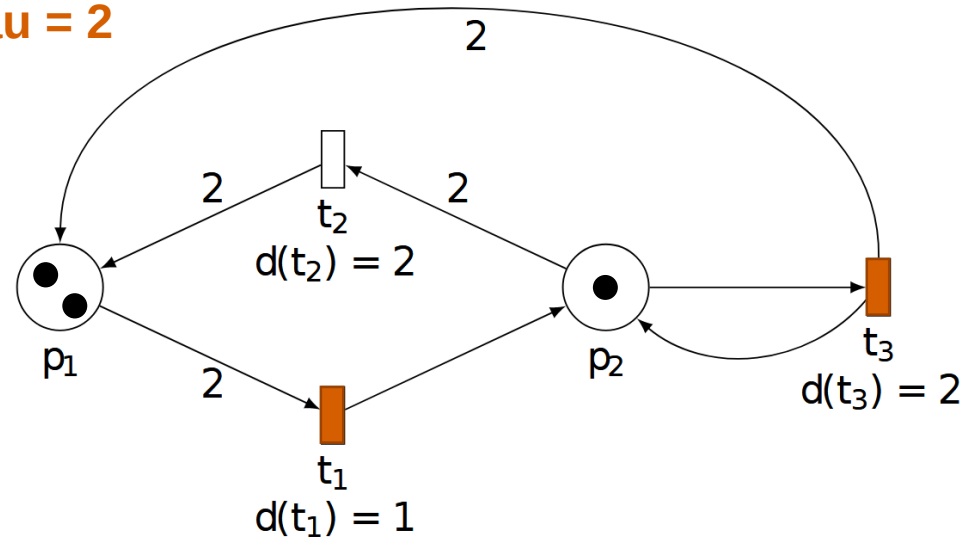


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[0, 1]	(t3, 2)
1	2	t3	[2, 1]	

Adding time-dependent behaviors to Petri nets

* When several transitions are enabled at the same time, choose the one with the smallest index first

@ tau = 2

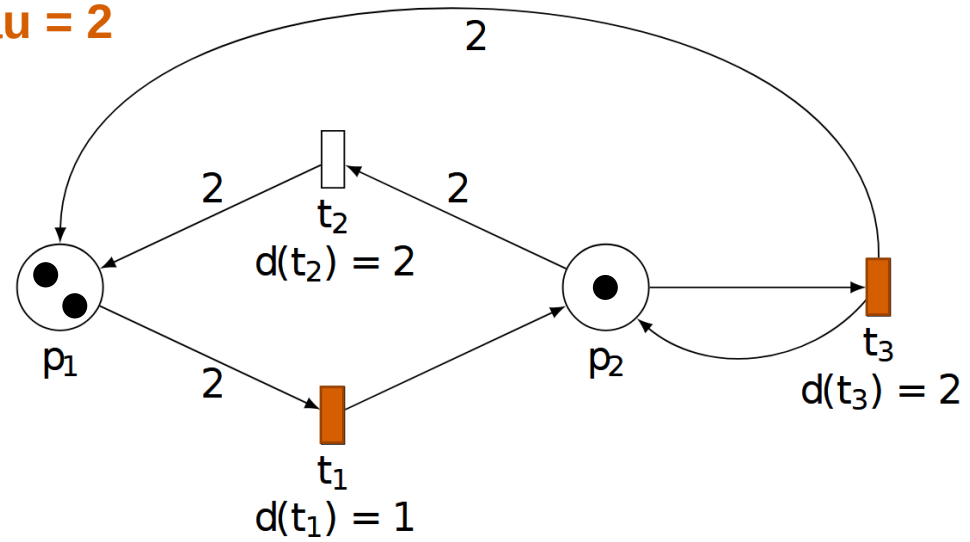


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[0, 1]	(t3, 2)
1	2	t3	[2, 1]	(t1, 3), (t3, 4)

Adding time-dependent behaviors to Petri nets

* When several transitions are enabled at the same time, choose the one with the smallest index first

@ tau = 2



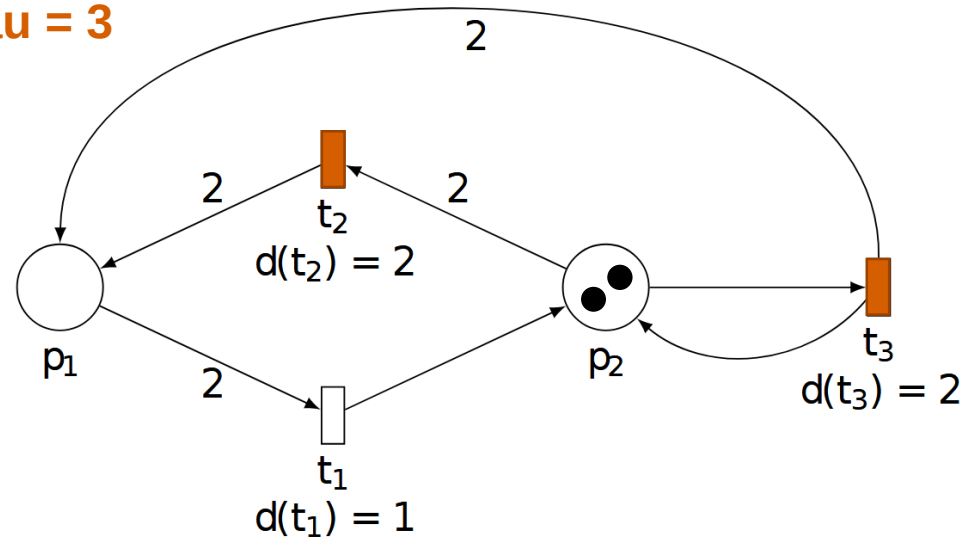
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[0, 1]	(t3, 2)
1	2	t3	[2, 1]	(t1, 3), (t3, 4)

Your turn! Please determine the simulation outcome for the next 4 steps (i.e., until step = 5)!

Adding time-dependent behaviors to Petri nets

* When several transitions are enabled at the same time, choose the one with the smallest index first

@ tau = 3

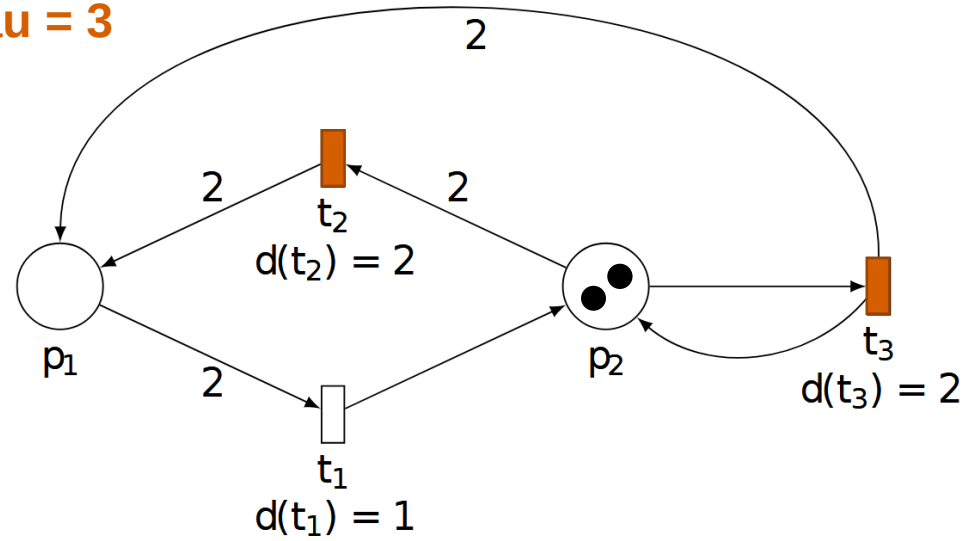


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[0, 1]	(t3, 2)
1	2	t3	[2, 1]	(t1, 3), (t3, 4)
2	3	t1	[0, 2]	(t2, 5), (t3, 4)

Adding time-dependent behaviors to Petri nets

* When several transitions are enabled at the same time, choose the one with the smallest index first

@ tau = 3



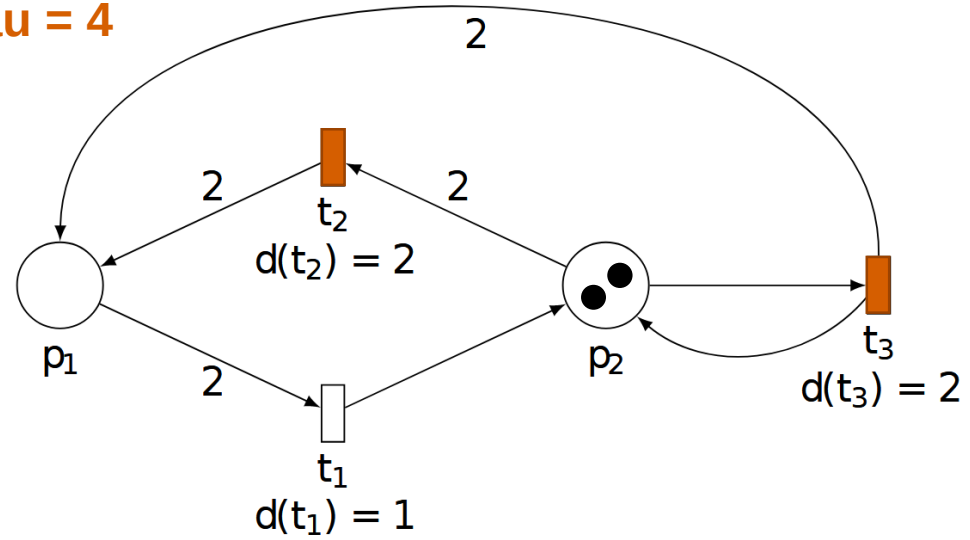
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[0, 1]	(t3, 2)
1	2	t3	[2, 1]	(t1, 3), (t3, 4)
2	3	t1	[0, 2]	(t2, 5), (t3, 4)

@ tau = 3:
t3 is not deactivated

Adding time-dependent behaviors to Petri nets

* When several transitions are enabled at the same time, choose the one with the smallest index first

@ tau = 4



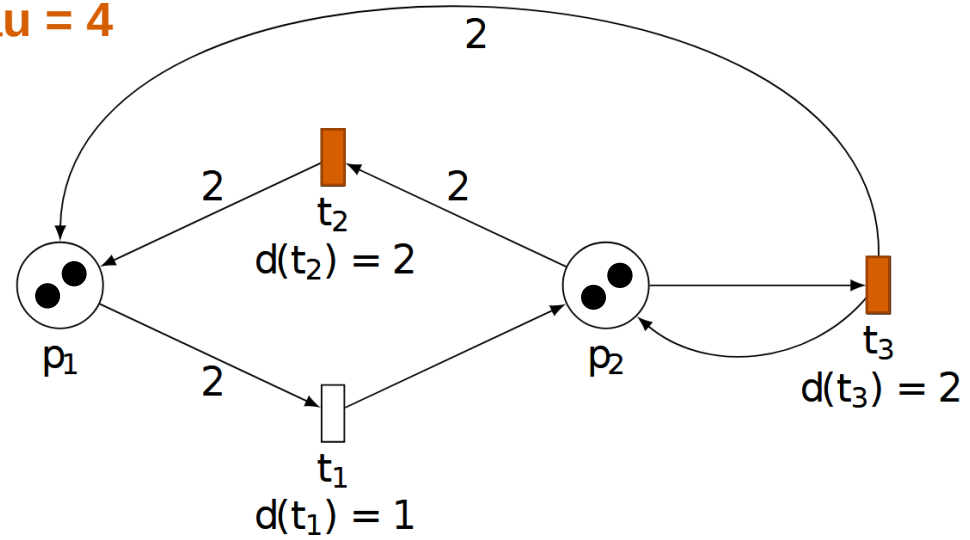
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[0, 1]	(t3, 2)
1	2	t3	[2, 1]	(t1, 3), (t3, 4)
2	3	t1	[0, 2]	(t2, 5), (t3, 4)
3	4	t3	[2, 2]	

@ tau = 4, token is consumed from p2:
t2 and t3 both lose activation, and immediately reactivated.

Adding time-dependent behaviors to Petri nets

* When several transitions are enabled at the same time, choose the one with the smallest index first

@ tau = 4



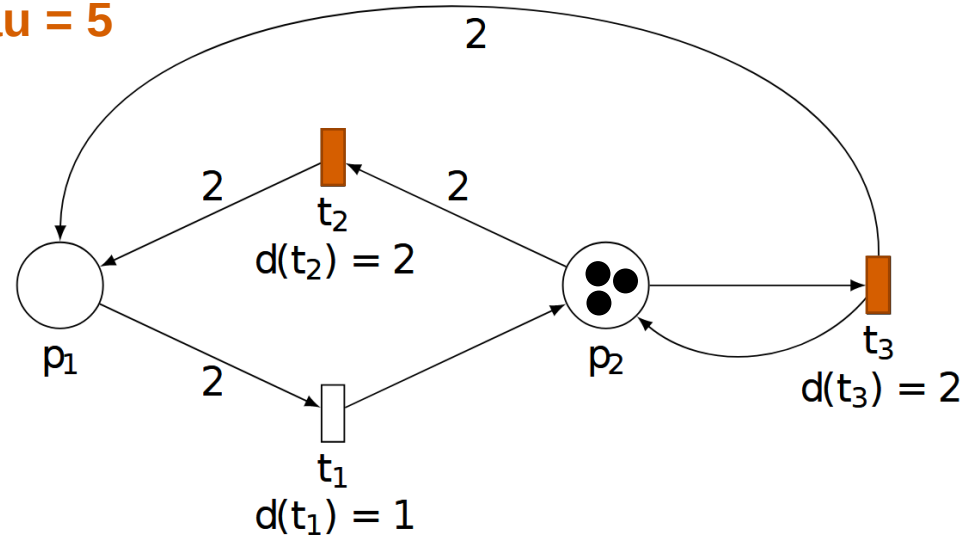
Step	tau	Fired transition	Marking vector	Event list
0	0	-	[0, 1]	(t3, 2)
1	2	t3	[2, 1]	(t1, 3), (t3, 4)
2	3	t1	[0, 2]	(t2, 5), (t3, 4)
3	4	t3	[2, 2]	(t1, 5), (t2, 6), (t3, 6)

@ tau = 4, token is consumed from p2:
t2 and t3 both lose activation, and immediately reactivated.

Adding time-dependent behaviors to Petri nets

* When several transitions are enabled at the same time, choose the one with the smallest index first

@ tau = 5



Step	tau	Fired transition	Marking vector	Event list
0	0	-	[0, 1]	(t3, 2)
1	2	t3	[2, 1]	(t1, 3), (t3, 4)
2	3	t1	[0, 2]	(t2, 5), (t3, 4)
3	4	t3	[2, 2]	(t1, 5), (t2, 6), (t3, 6)
4	5	t1	[0, 3]	(t2, 6), (t3, 6)

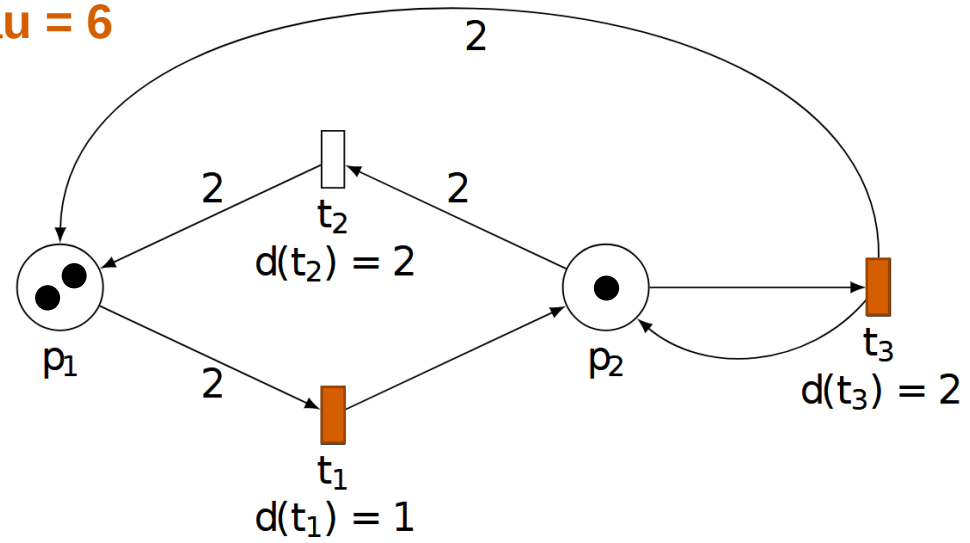
@ tau = 3:

t3 is not deactivated when firing t1

Adding time-dependent behaviors to Petri nets

* When several transitions are enabled at the same time, choose the one with the smallest index first

@ tau = 6

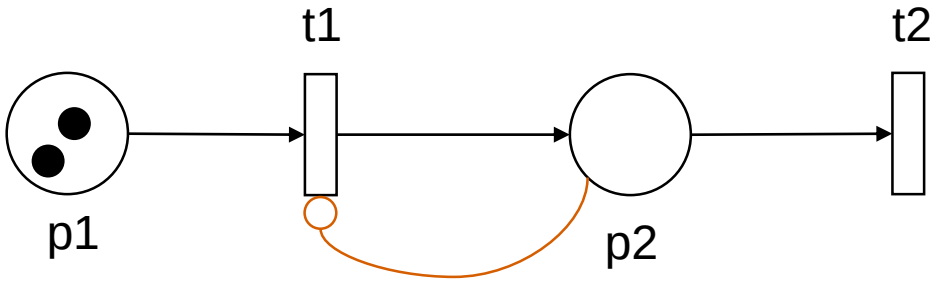


Step	tau	Fired transition	Marking vector	Event list
0	0	-	[0, 1]	(t3, 2)
1	2	t3	[2, 1]	(t1, 3), (t3, 4)
2	3	t1	[0, 2]	(t2, 5), (t3, 4)
3	4	t3	[2, 2]	(t1, 5), (t2, 6), (t3, 6)
4	5	t1	[0, 3]	(t2, 6), (t3, 6)
5	6	t2	[2, 1]	(t1, 7), (t3, 8)

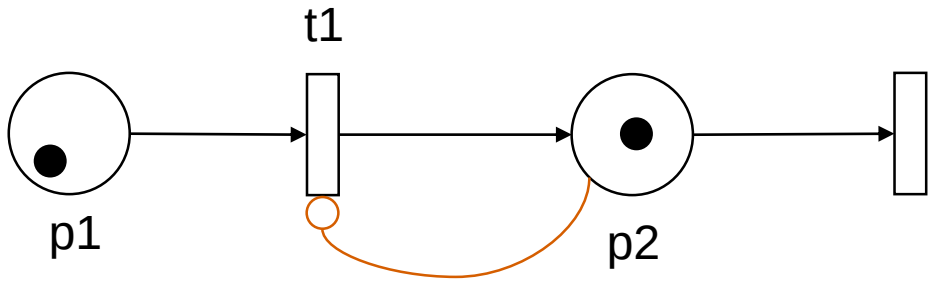
@ tau = 6, token is consumed from p2:
t2 and t3 both lose activation, t3 is immediately reactivated

Inhibitor Arc

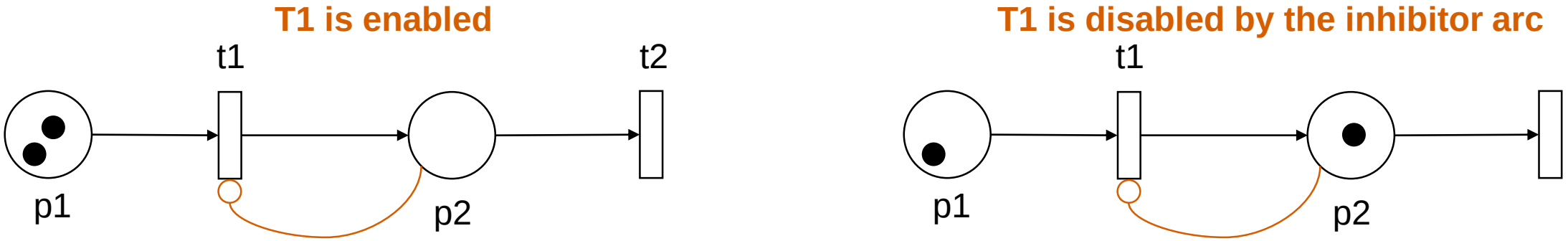
T1 is enabled



T1 is disabled by the inhibitor arc



Calculation with Petri nets



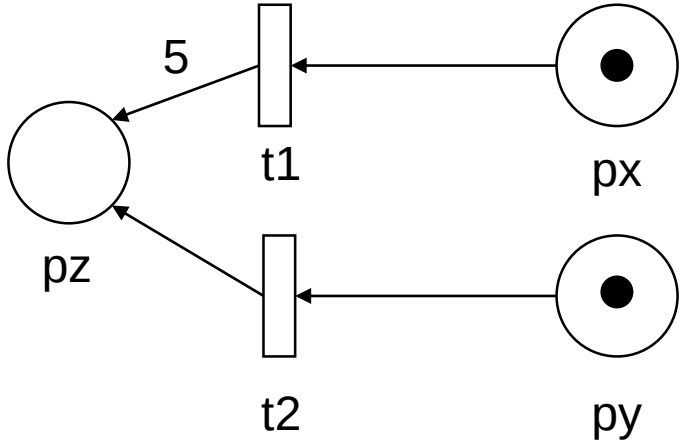
Goal of the exercise: model a function $f_i(x, y)$ using a Petri net.

- The Petri net must contain two places P_x and P_y that hold x and y tokens respectively in the beginning.
- The net must contain a place P_z which holds $f_i(x, y)$ tokens when the net is dead
- The Petri nets are supposed to work for arbitrary numbers of tokens in P_x and P_y.

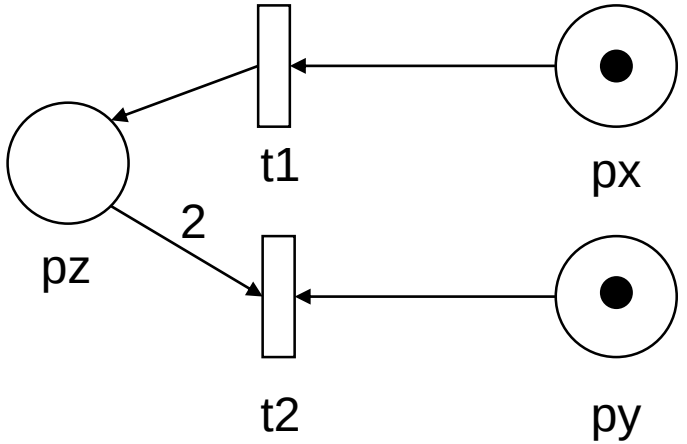
1. $f_1(x, y) \coloneqq 5x + y, \forall x, y \geq 0$
2. $f_2(x, y) \coloneqq x - 2y, \forall y \geq 0, x > 2y$
3. $f_3(x, y) \coloneqq xy, \forall x, y \geq 0$

For f3, we need to first create a token duplicator that duplicates the tokens from P_x to P_z (this maybe requires the use of one or more inhibitor arcs).

$f_1(x, y) \coloneqq 5x + y, \forall x, y \geq 0$

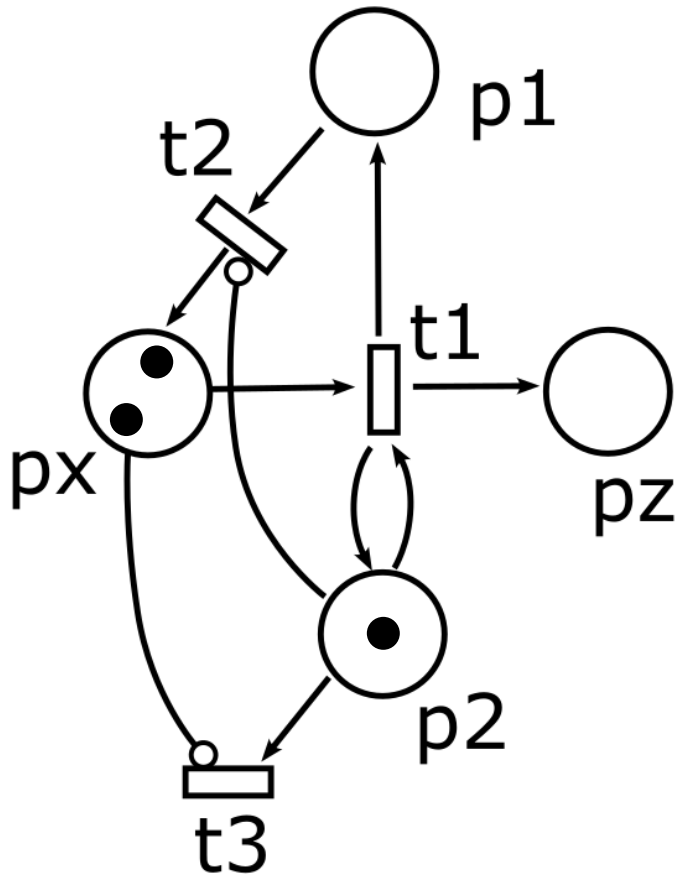


$f_2(x, y) \coloneqq x - 2y, \forall y \geq 0, x > 2y$



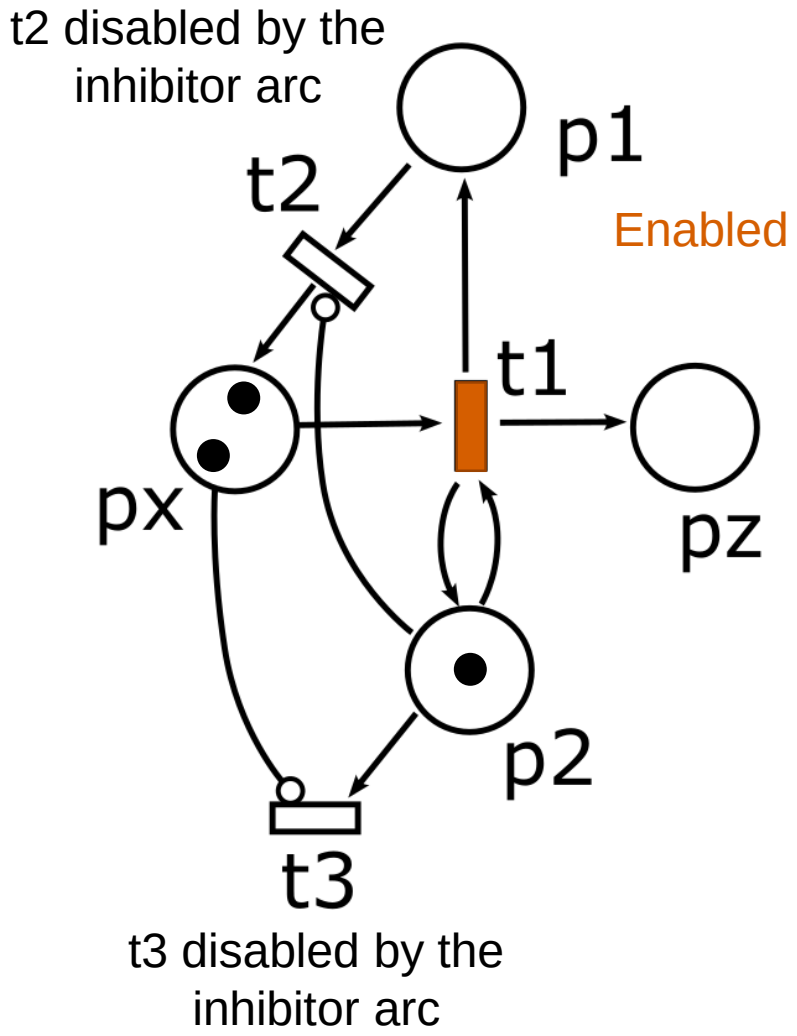
Token duplicator

$f_3(x, y) \coloneqq xy, \forall x, y \geq 0$



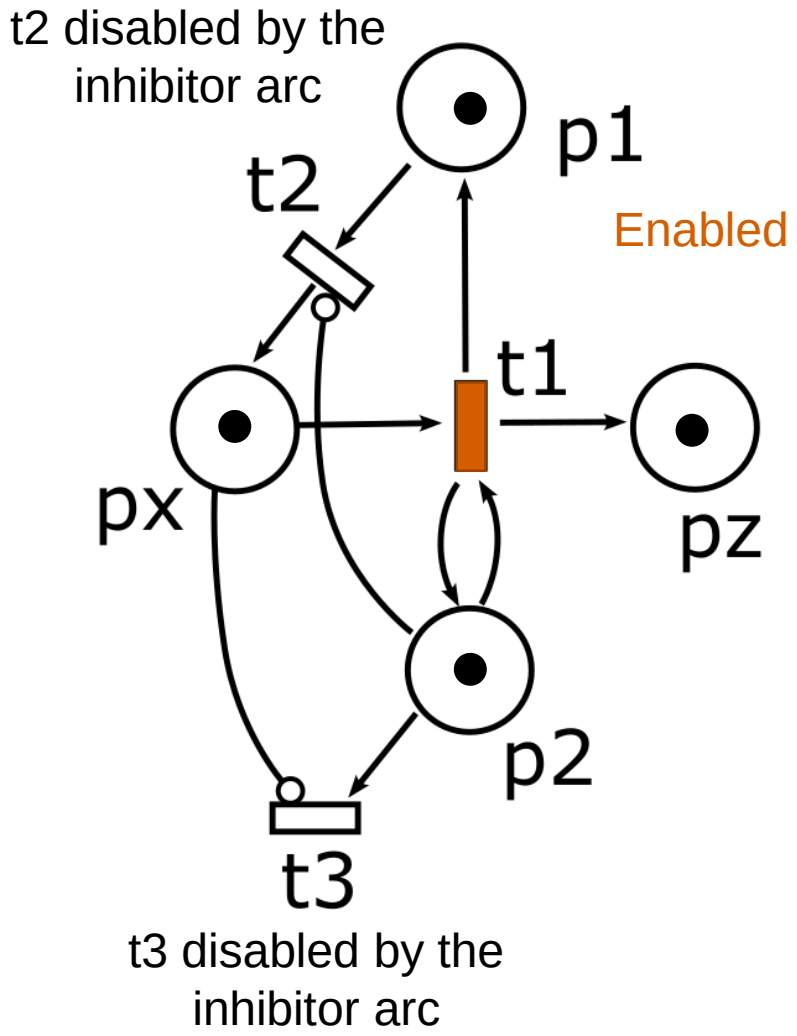
Token duplicator

$f_3(x, y) \coloneqq xy, \forall x, y \geq 0$



Token duplicator

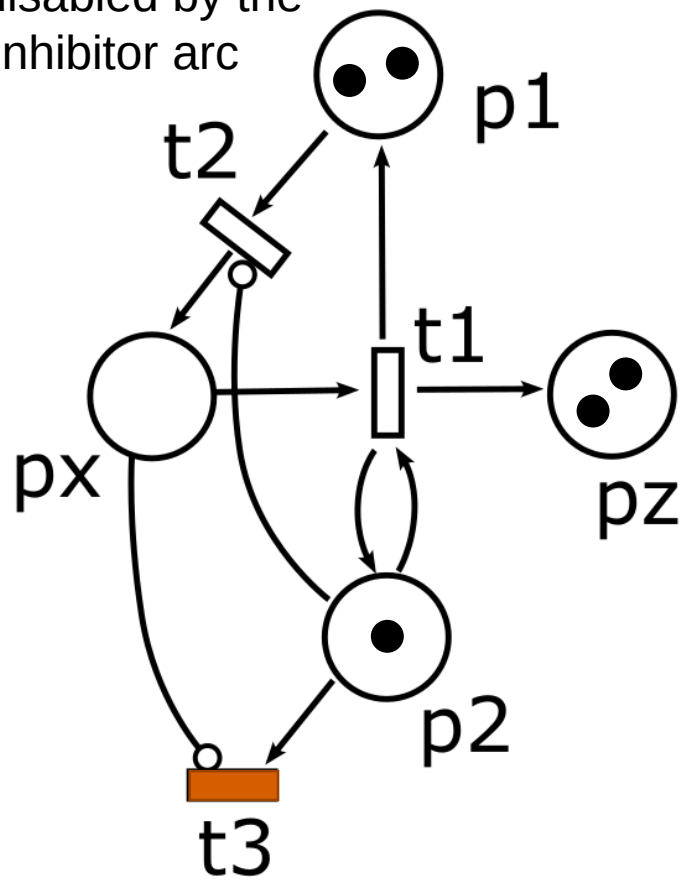
$f_3(x, y) \coloneqq xy, \forall x, y \geq 0$



Token duplicator

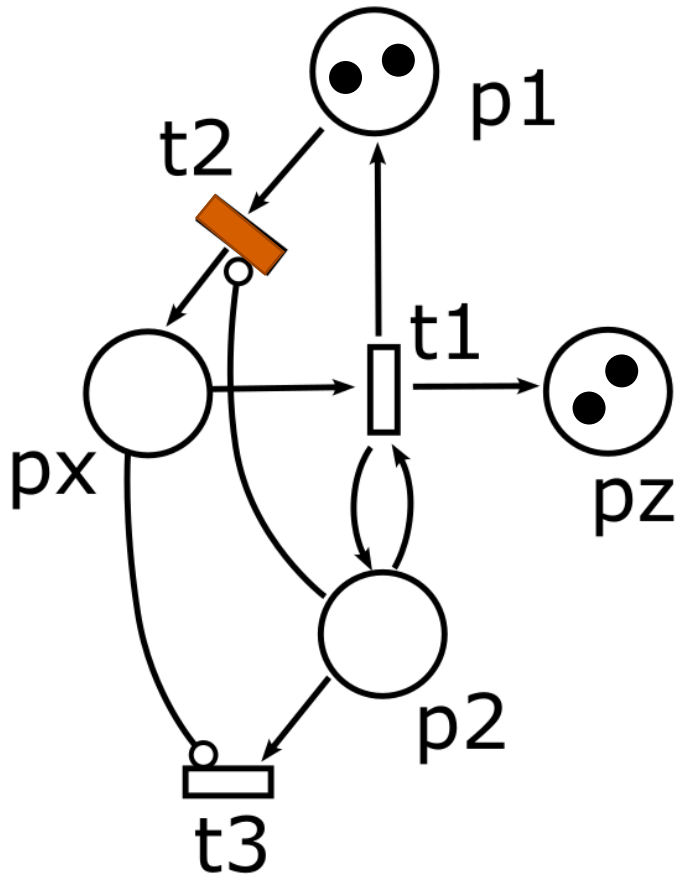
$f_3(x, y) \coloneqq xy, \forall x, y \geq 0$

t2 disabled by the inhibitor arc



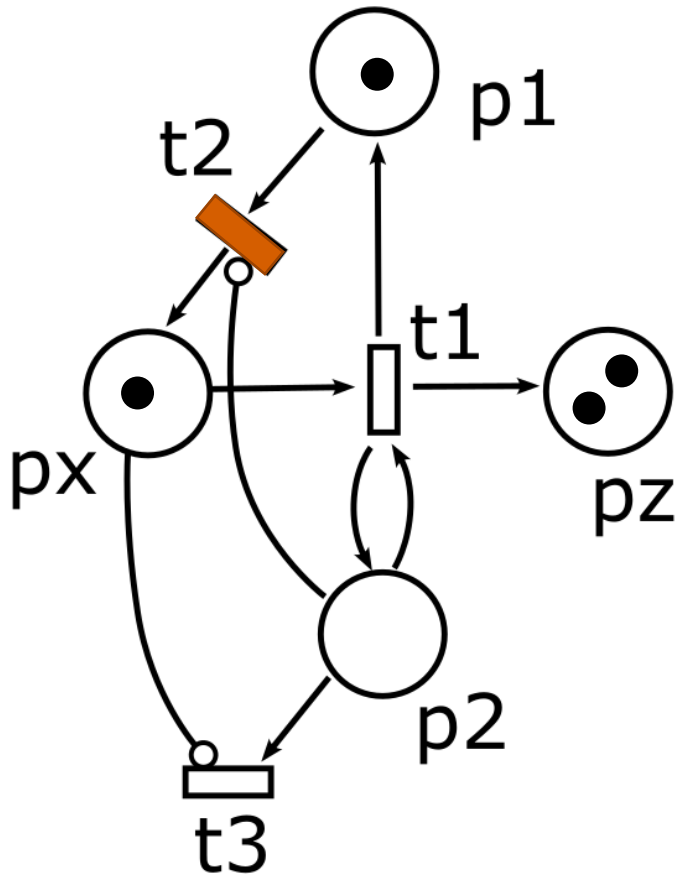
Token duplicator

$f_3(x, y) \coloneqq xy, \forall x, y \geq 0$



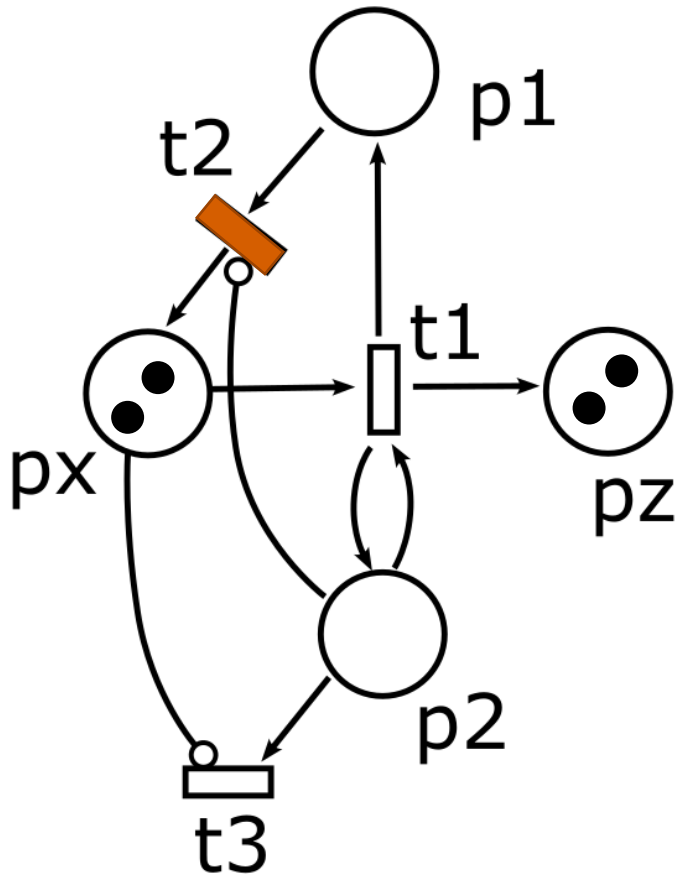
Token duplicator

$f_3(x, y) \coloneqq xy, \forall x, y \geq 0$



Token duplicator

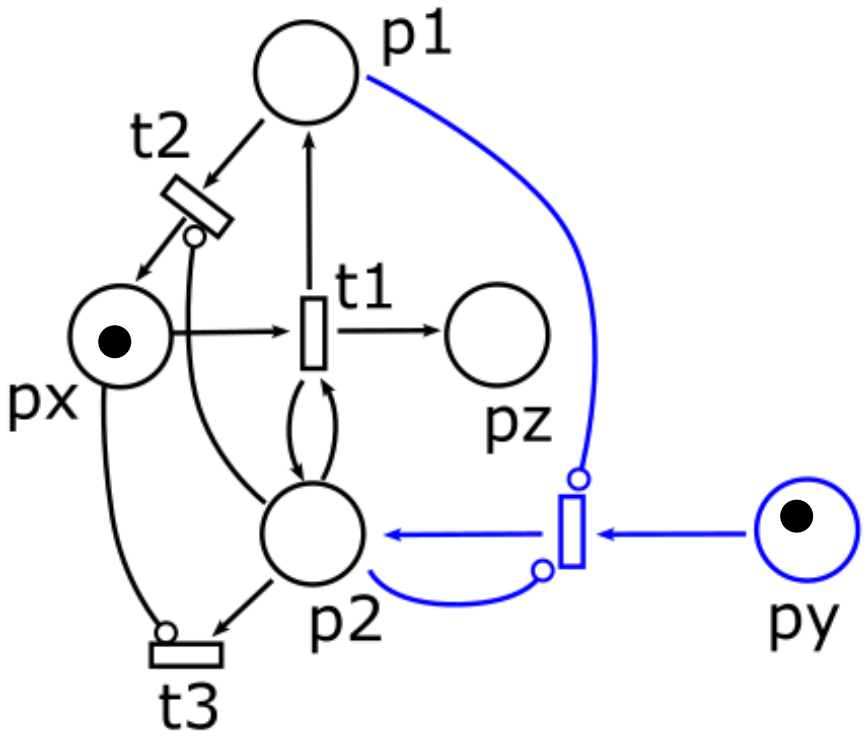
$f_3(x, y) \coloneqq xy, \forall x, y \geq 0$



Token duplicated from px to pz

Token duplicator

$f_3(x, y) \coloneqq xy, \forall x, y \geq 0$



Idea: supply p2 with exactly py tokens
(duplicate tokens from px to pz for py times)