



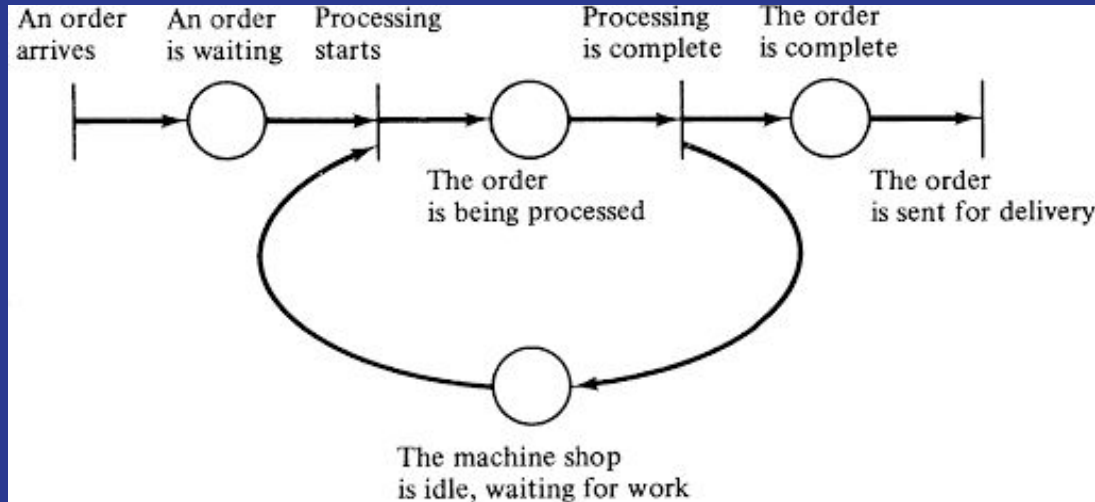
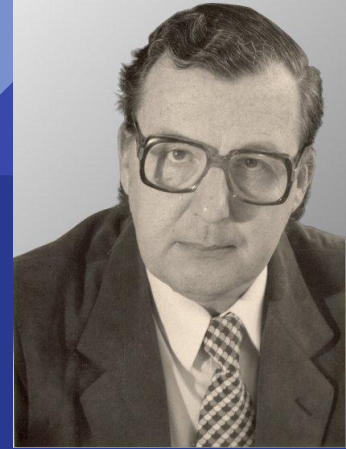
Redes de Petri

Presentación TP 1 - Técnicas de Programación Concurrente

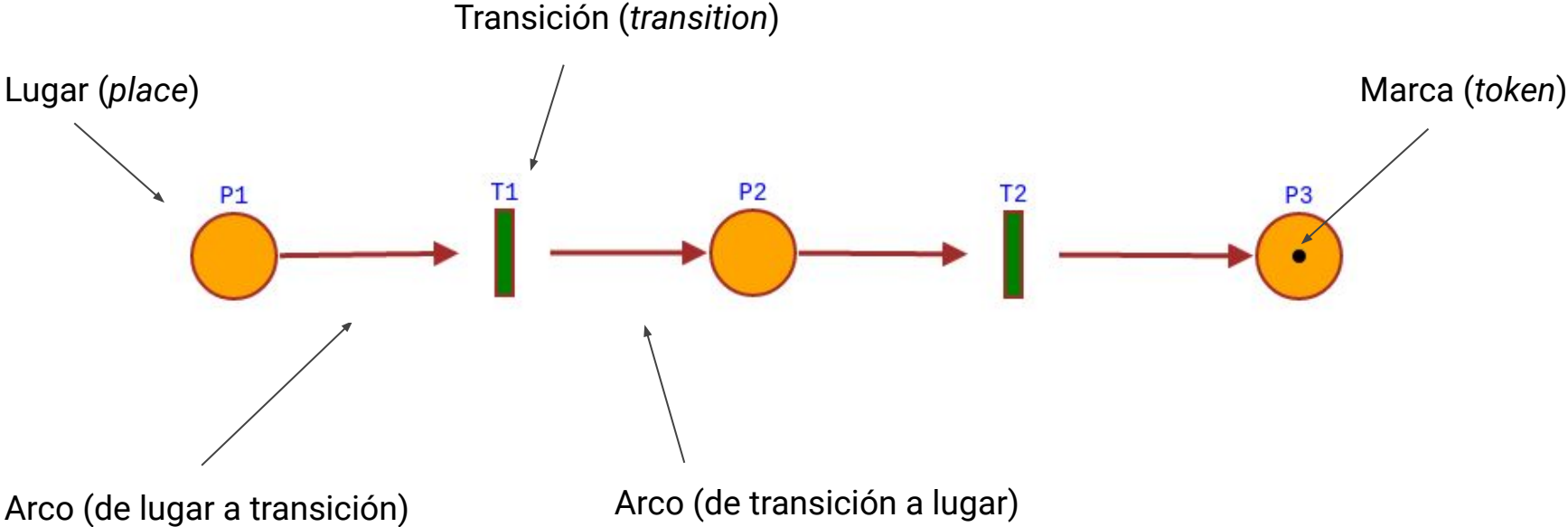
Integrantes: Marco Rodrigo Albanesi, Horacio Lisdero Scaffino

Introducción

- Introducidas por Carl Adam Petri en 1962.
- Una herramienta para modelar concurrencia y sincronización en sistemas distribuidos.
- Usadas como una ayuda visual para modelar el comportamiento del sistema.
- Construidas sobre una sólida base matemática.

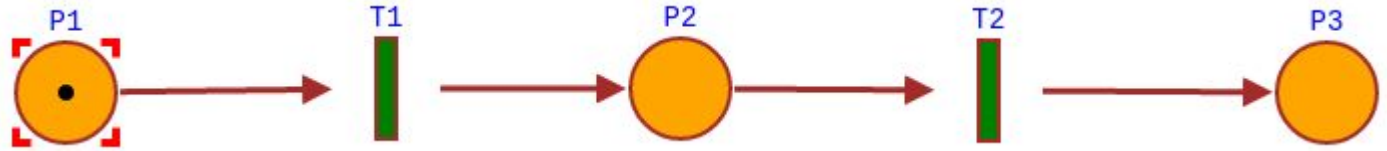


Componentes

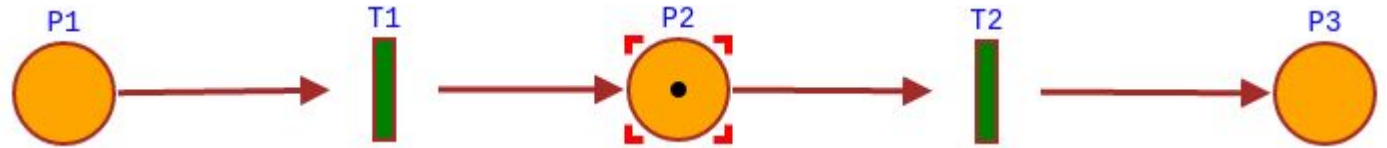


¿Y cómo funcionan?

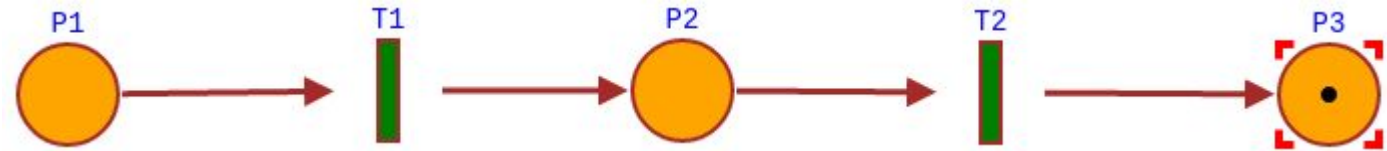
Paso 1: Ponemos un token al inicio



Paso 2:

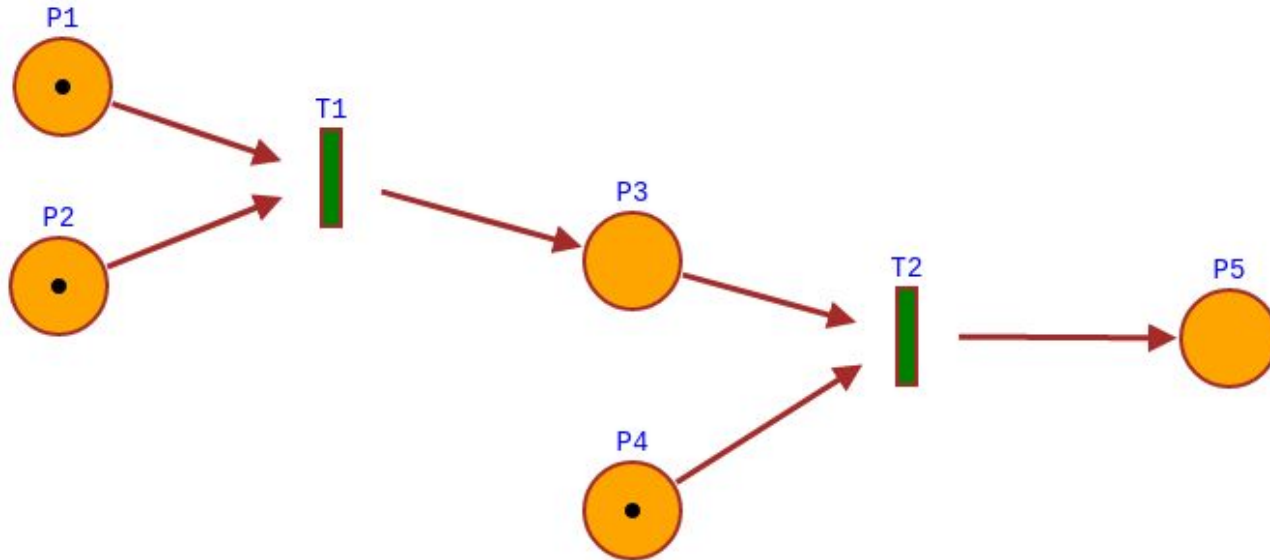


Paso 3:



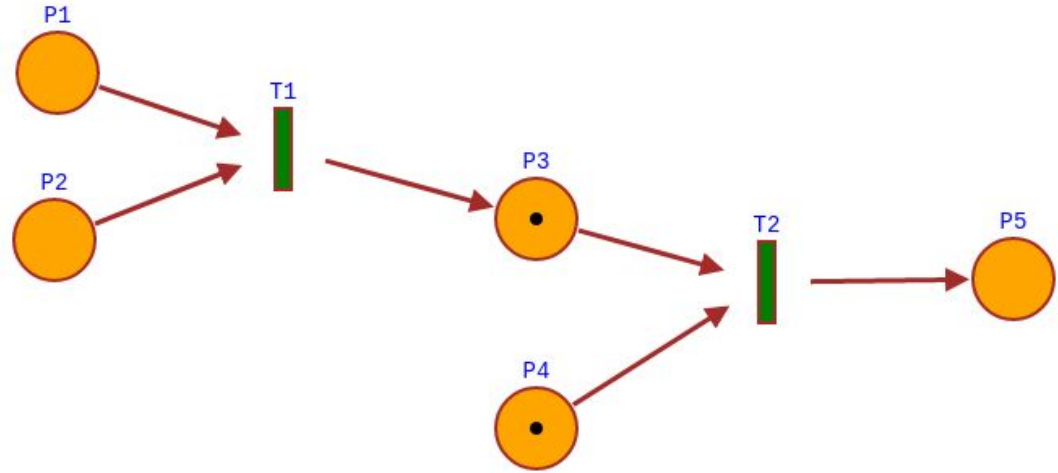
¿Y si hay más arcos y más lugares?

Decimos que una transición está **activa** si todos los lugares con arcos entrantes a la transición tienen un token. Las transiciones activas pueden ocurrir en cualquier orden. En otras palabras, ocurren de forma no determinística.



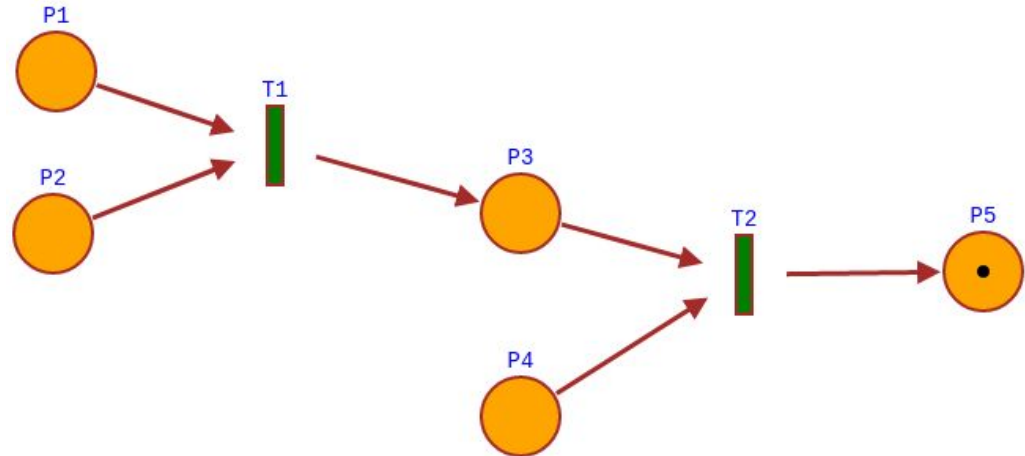
Paso siguiente:

Por cada flecha, un token.
Entran dos, sale uno.

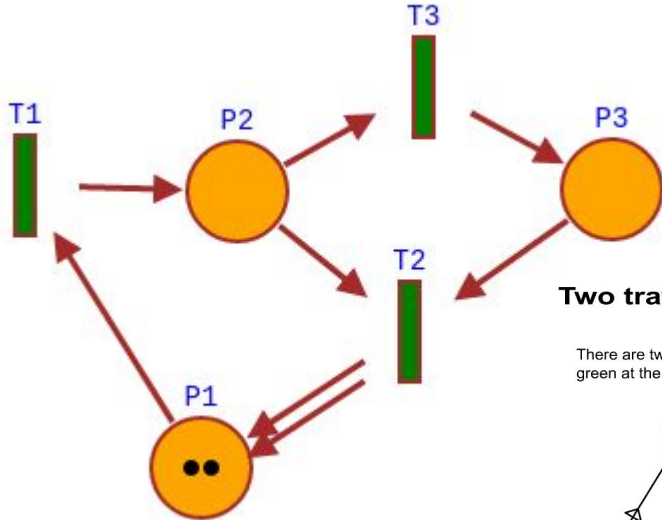


Paso final:

Si en P4 no hubiera un token, la transición T2 no podría ocurrir.

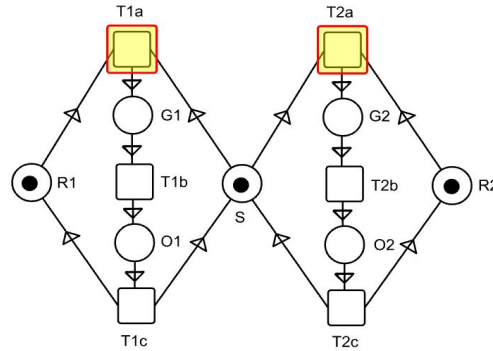


Jugando con simuladores online



Two traffic lights

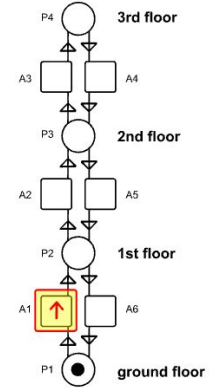
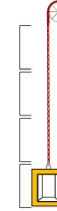
There are two traffic lights, which are not allowed to signal green at the same time.



Press: to fire the activity

Elevator (2)

The token represents the elevator.

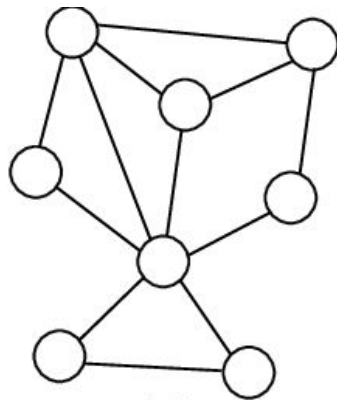
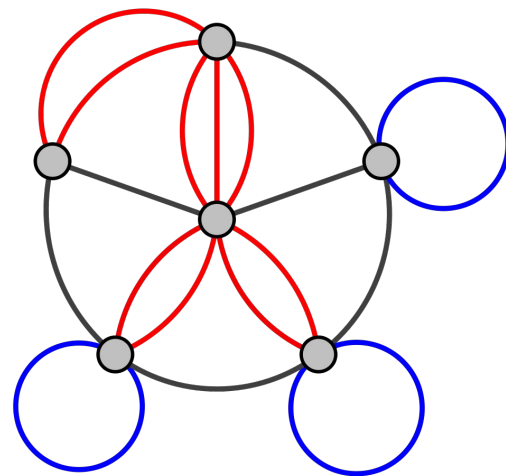
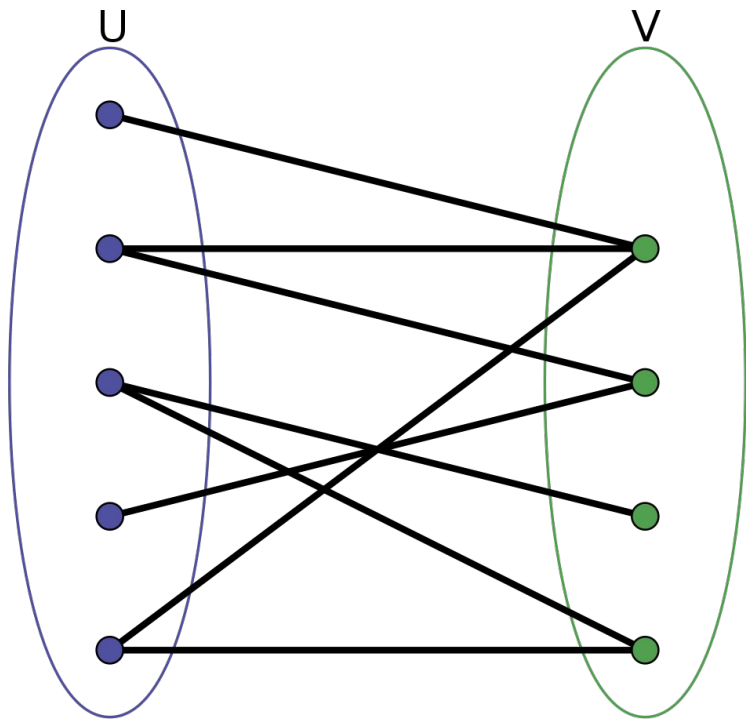


Press: To fire the activity

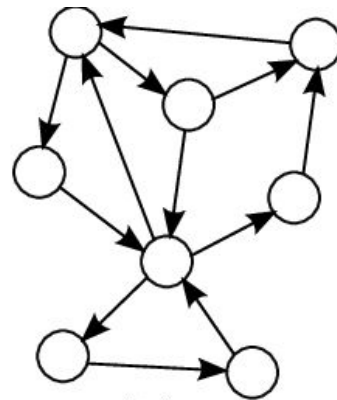


Teoría matemática de las Redes de Petri

Recordando algunos conceptos



(a)



(b)

Definición

Una red de Petri es un grafo bipartito, dirigido, compuesto por un conjunto de nodos y de arcos. Para una red ordinaria de Petri (es decir, sin varios arcos entre dos nodos), usaremos la siguiente definición de una 3-tupla:

$$PN = (T, P, A)$$

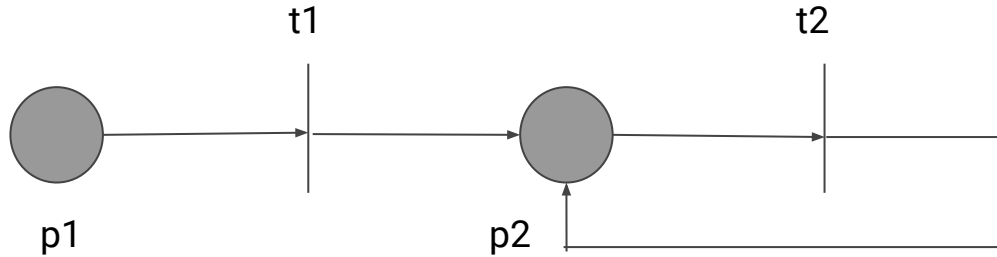
donde $T = \{t_1, t_2, \dots, t_n\}$ es el conjunto de nodos llamados *transiciones*.

$P = \{p_1, p_2, \dots, p_m\}$ es el conjunto de nodos llamados *lugares*.

$A \subseteq (T \times P) \cup (P \times T)$, un conjunto de arcos dirigidos.



Ejemplo



Si la red no tiene esta clase de aristas “en loop”, la red se denomina “pura”.

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

$$P = \{p_1, p_2\}$$

$$A = \{(p_1, t_1), (t_1, p_2), (p_2, t_2), (t_2, p_2)\}$$

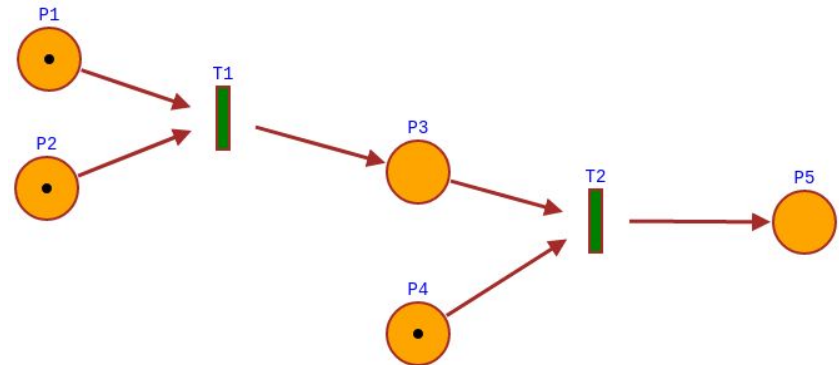
Concepto de Marcado (*marking*)

Llamamos función de marcado (*marking function*) $M: P \rightarrow \mathbb{N}_0$ a la función que nos devuelve un número entero no negativo para cada lugar p_i que nos indica la cantidad de tokens en ese lugar.

A partir de esa función se puede definir un vector M_t de longitud m , donde m es la cantidad de lugares. Este vector representa el estado actual del sistema.

Ejemplo: $M_0 = (1, 1, 0, 1, 0)$

M_0 es el marcado inicial (*initial marking*)



Funciones de input y output

$$I(t) = \{ p \mid (p, t) \in A \}$$

$$O(t) = \{ p \mid (t, p) \in A \}$$

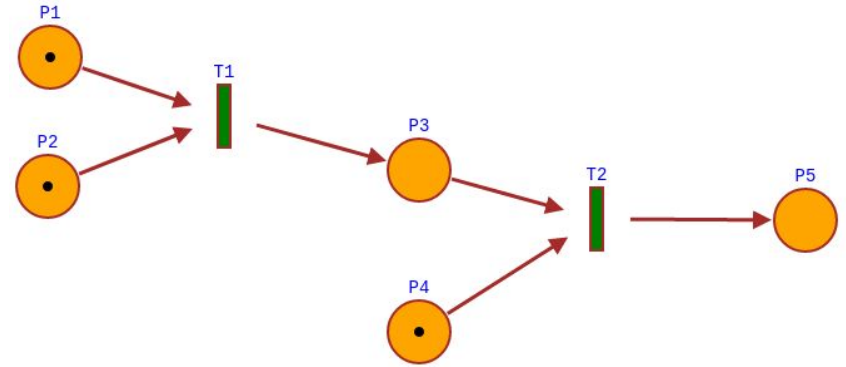
Ejemplo:

$$I(t_1) = \{ p_1, p_2 \}$$

$$O(t_1) = \{ p_3 \}$$

$$I(t_2) = \{ p_3, p_4 \}$$

$$O(t_2) = \{ p_5 \}$$



Definición de transición activa:

La transición t está *activa* (esto es, *puede disparar*) si $M(p) > 0$ para todo $p \in I(t)$

Cambio de estado

Si la transición t está activa en un marcado M , entonces el disparo (*firing*) de esa transición crea un nuevo marcado M' , removiendo un token de cada lugar $p \in I(t)$ y depositando un token en cada lugar $p \in O(t)$

Cada nuevo marcado define un nuevo estado del sistema. Si varias transiciones están activas, se habla de que están *en conflicto*. La transición que se dispara se elige de forma no determinística. La competencia por los tokens es entonces el método para representar el modelo de ejecución paralela intercalada (*interleaved model of parallelism*).






Análisis de alcance
Reachability analysis

Conjunto alcanzable

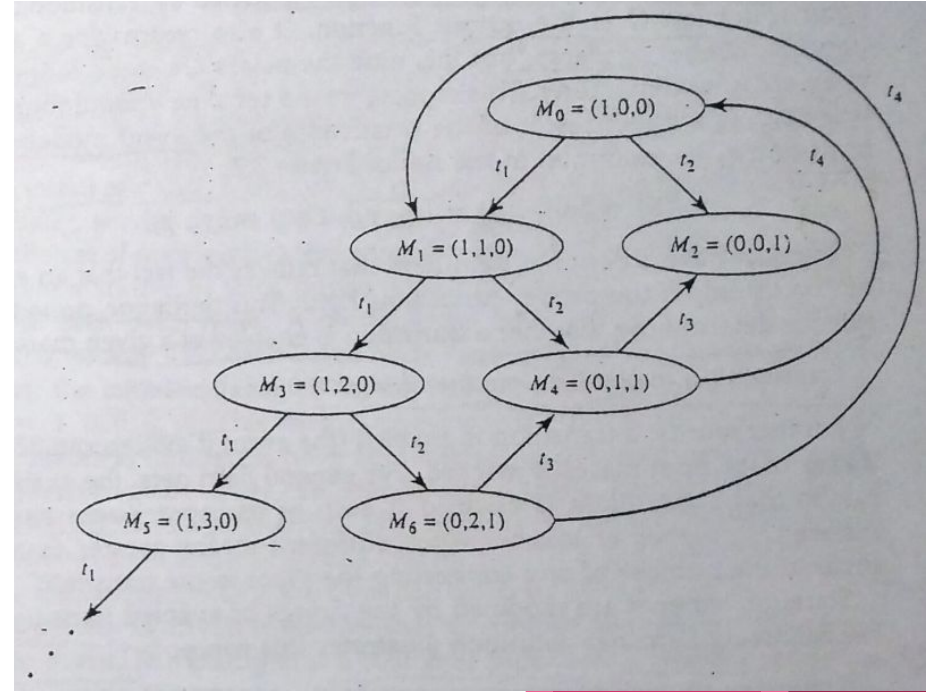
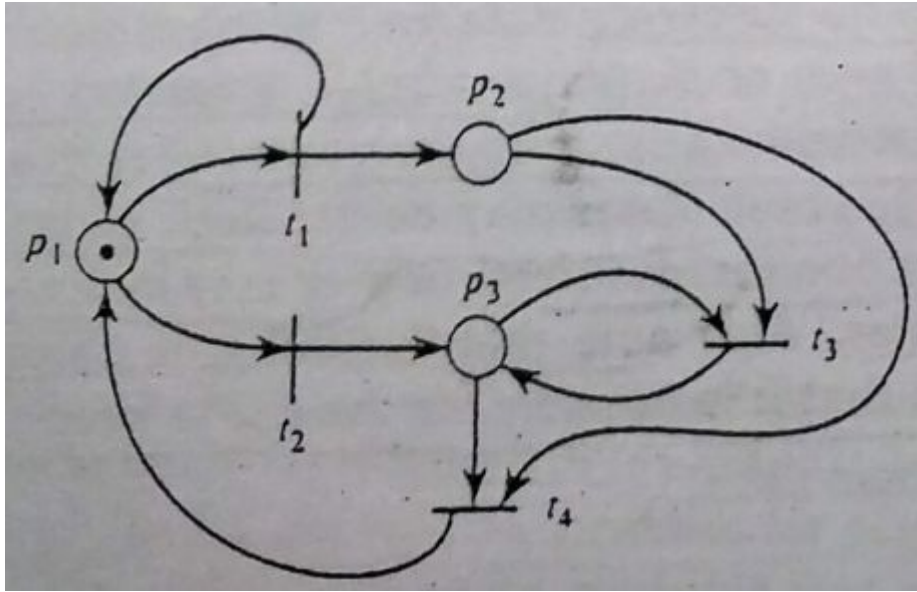
El conjunto alcanzable $R(M)$ para un marcado M es el conjunto de todos los marcados alcanzables partiendo desde M . Usualmente estamos interesados en $R(M_0)$.

Se construye a partir de $R(M_0)$ el grafo de alcance que tiene como nodos a los elementos de $R(M_0)$ y por arcos a las transiciones que llevan de un estado al otro. En otras palabras, el grafo de alcance muestra los estados posibles y cómo se llega a ellos.

Nos permite entre otras cosas descubrir si la cantidad de estados posibles es finita o infinita y si hay deadlocks.



Ejemplo interesante



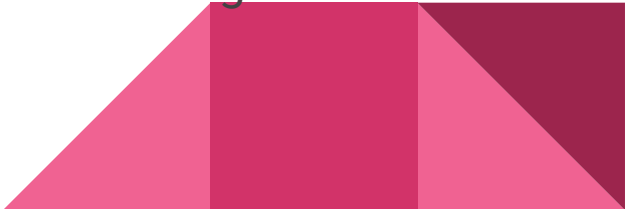
Lugares acotados (*bounded*) y seguros (*safe*)

k -acotado (*k-bounded*): si el lugar tiene a lo sumo k tokens para cualquier marcado M alcanzable.

Acotado: si existe un k entero, tal que el lugar es k -acotado.

Seguro (*safe*): si el lugar es 1-acotado, es decir, tiene a lo sumo un token para cualquier marcado M alcanzable.

Estas definiciones se extienden a la red en general si todos los elementos cumplen la condición. Por ejemplo una red es segura si todos los lugares son seguros y así sucesivamente.



Ventajas y desventajas del análisis de alcance

- + El análisis de alcance nos permite descubrir deadlocks.
- + Considera todos los estados posibles. Es la técnica más poderosa.
- El número de estados podría ser muy grande. Esto se debe a la naturaleza combinatoria al decidir los disparos de las transiciones.
- La generación del grafo de alcance es en el peor caso de orden exponencial.
- Se requieren aproximaciones para redes con un número de estados no acotado.

Existen otras técnicas llamadas estructurales que no requieren trabajar con el espacio de estados.

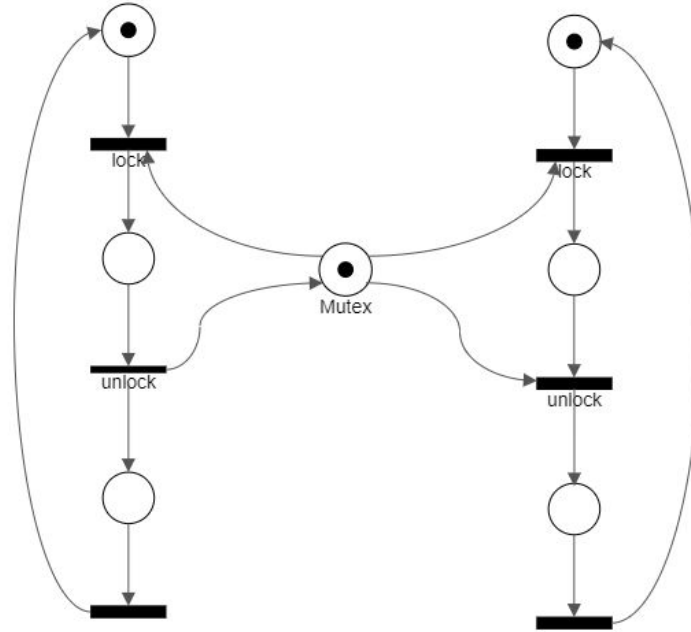


Extensiones a las Redes de Petri

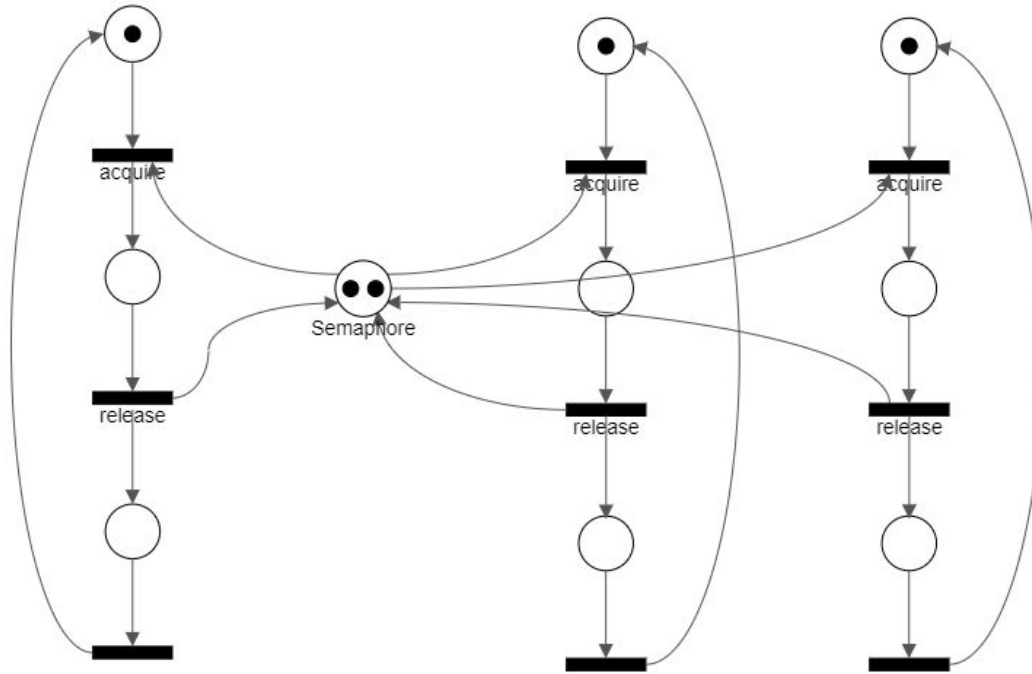
- Se puede agregar una noción de tiempo para indicar que ciertas transiciones tardan más que otras.
- Se puede agregar un arco inhibidor que permite activar una transición si un lugar está vacío. Este cambio le agrega poder al modelo. Está probado que una red de arco inhibidor alcanza el mismo poder de modelado que una máquina de Turing.
- Si los retrasos de las transiciones están especificados por una distribución de probabilidad, tenemos lo que se conoce como una red de Petri estocástica.
- Se pueden definir subredes como módulos con entradas y salidas y así encapsular complejidad para modelar sistemas más grandes, similar a lo que se hace con circuitos en electrónica.

Redes de Petri para estructuras de sincronización

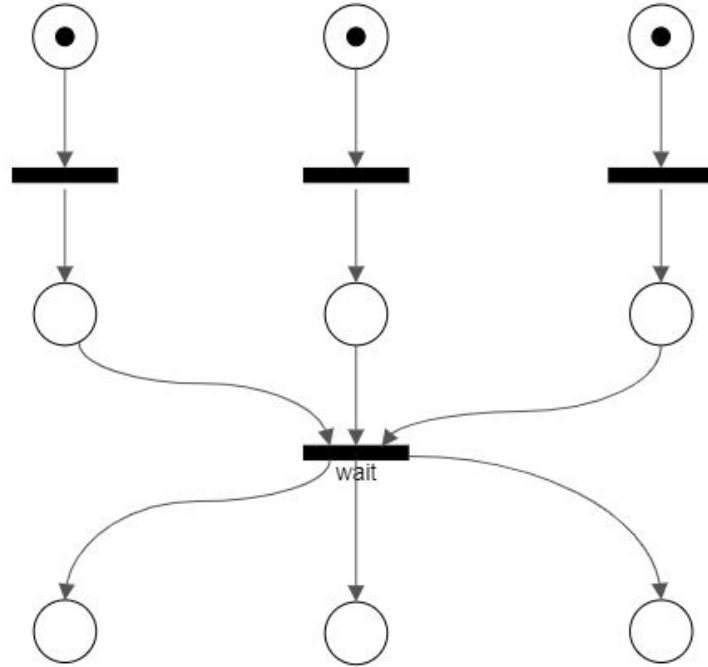
Exclusión mutua



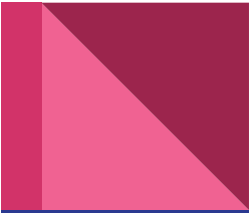
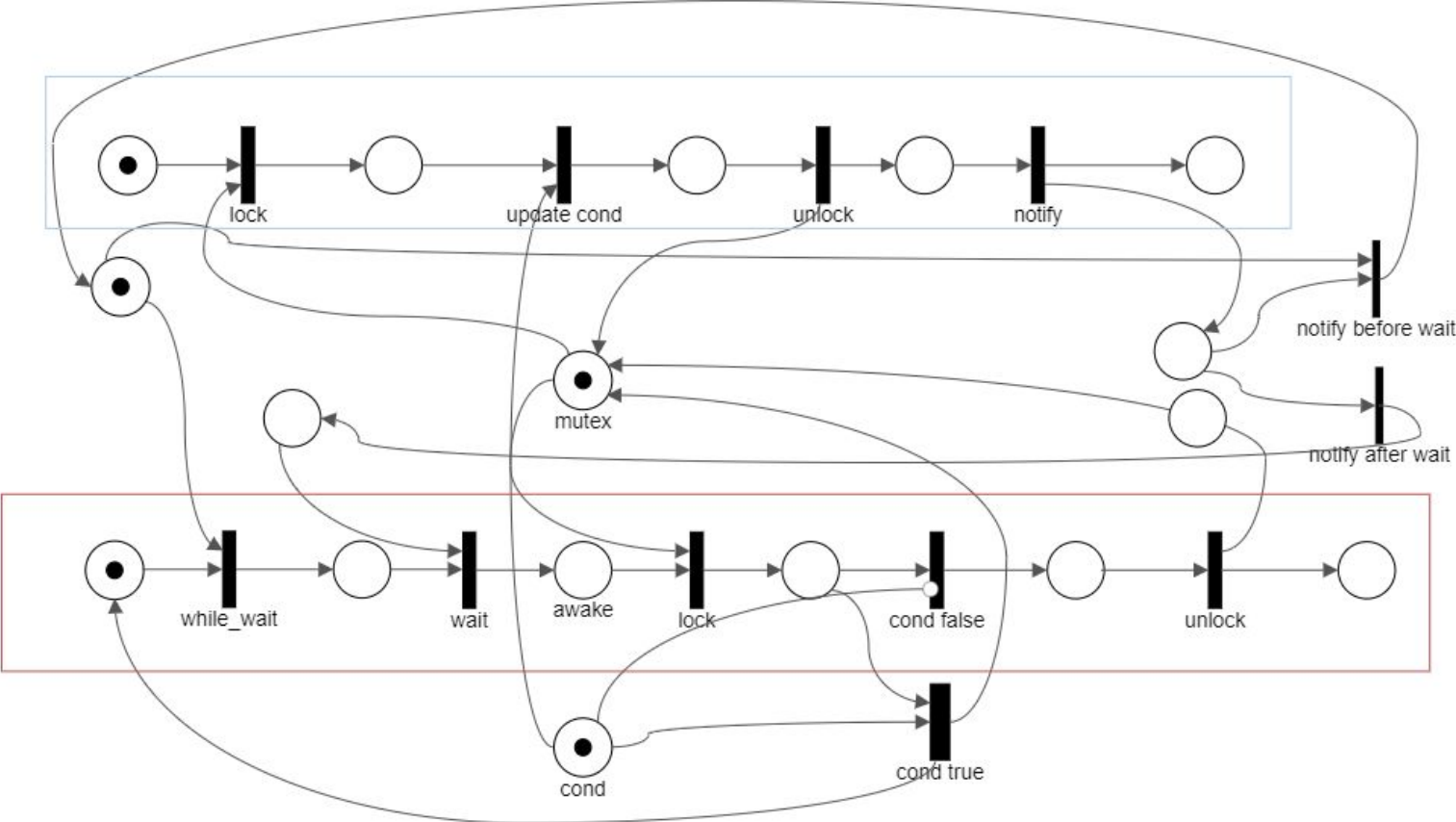
Semáforos



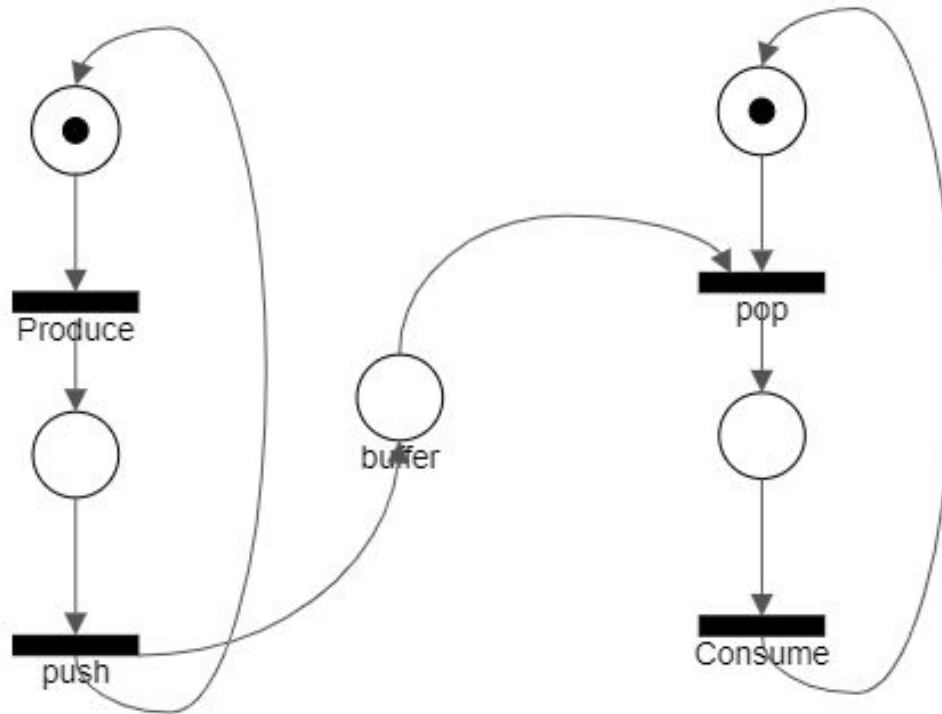
Barreras



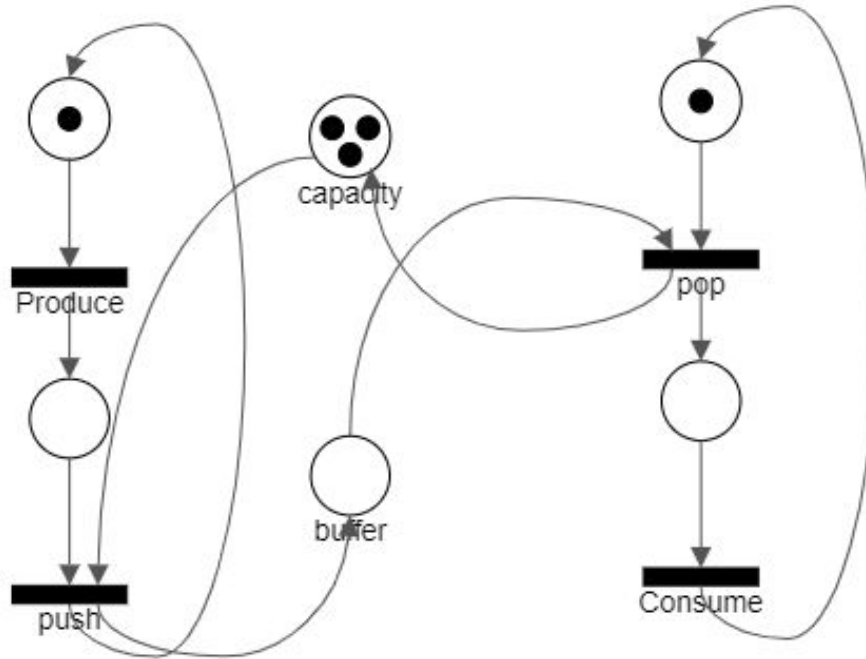
Cond vars



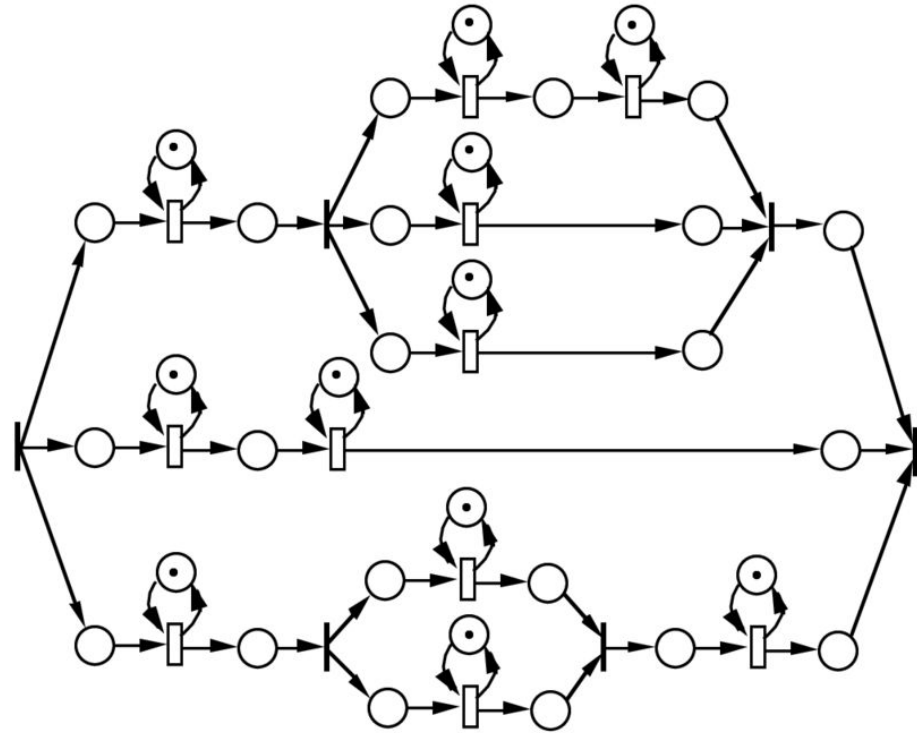
Productor - Consumidor con buffer infinito



Productor - Consumidor con buffer acotado

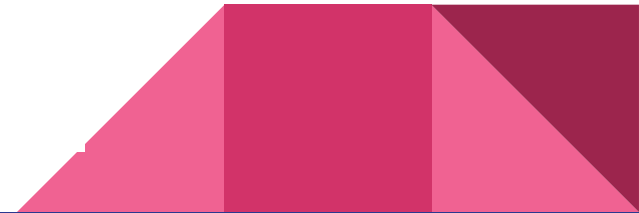
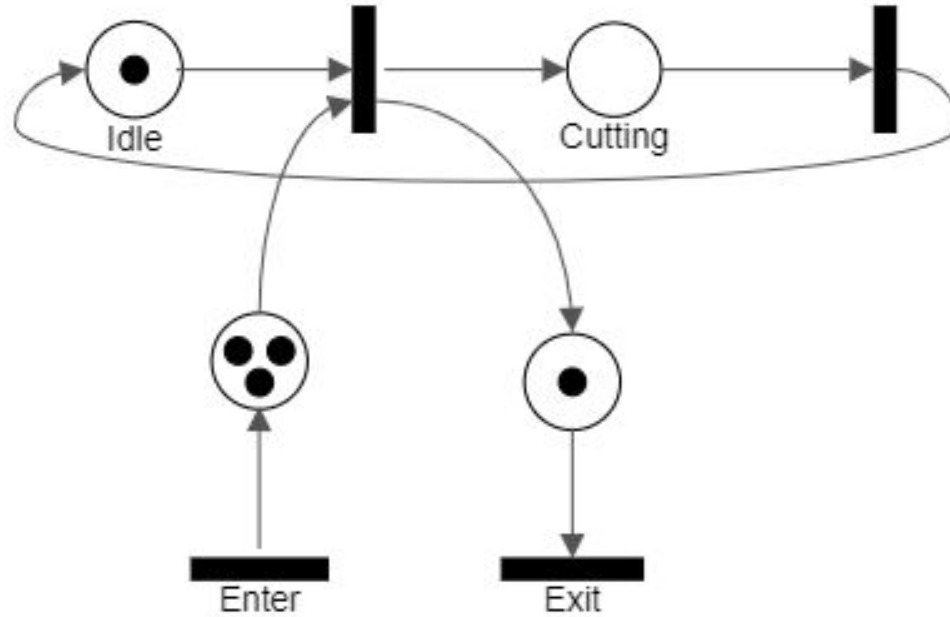


Fork Join

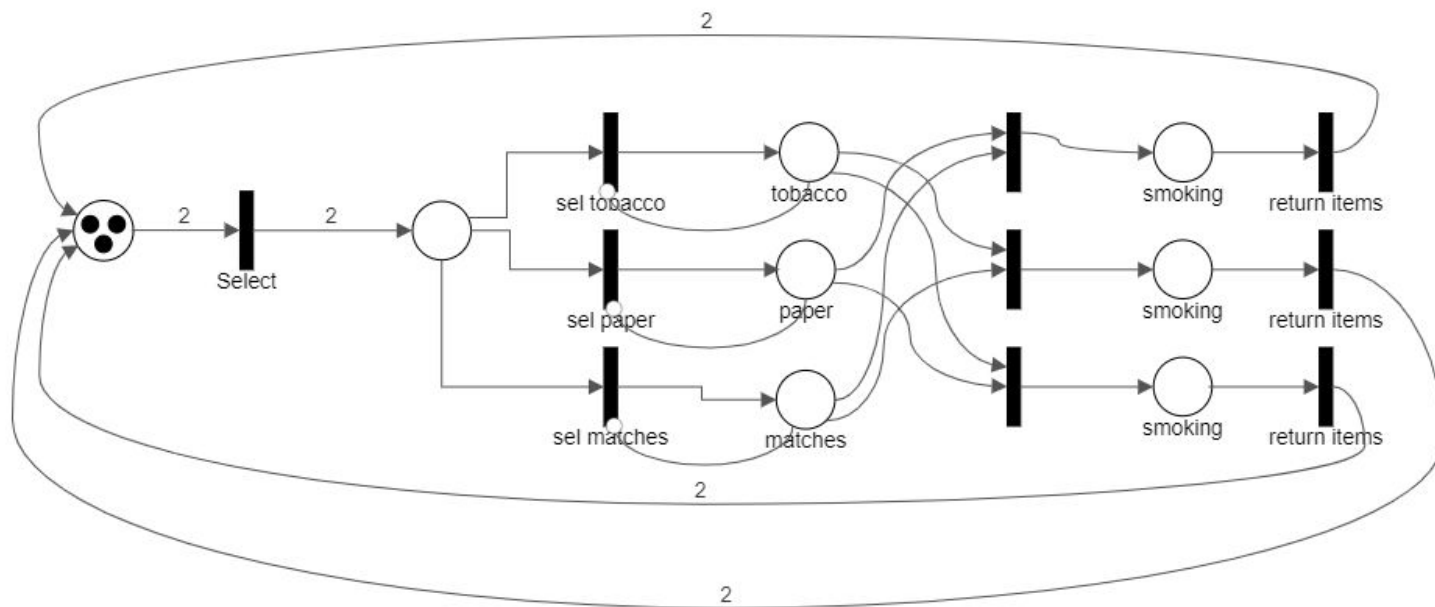


Modelado de problemas

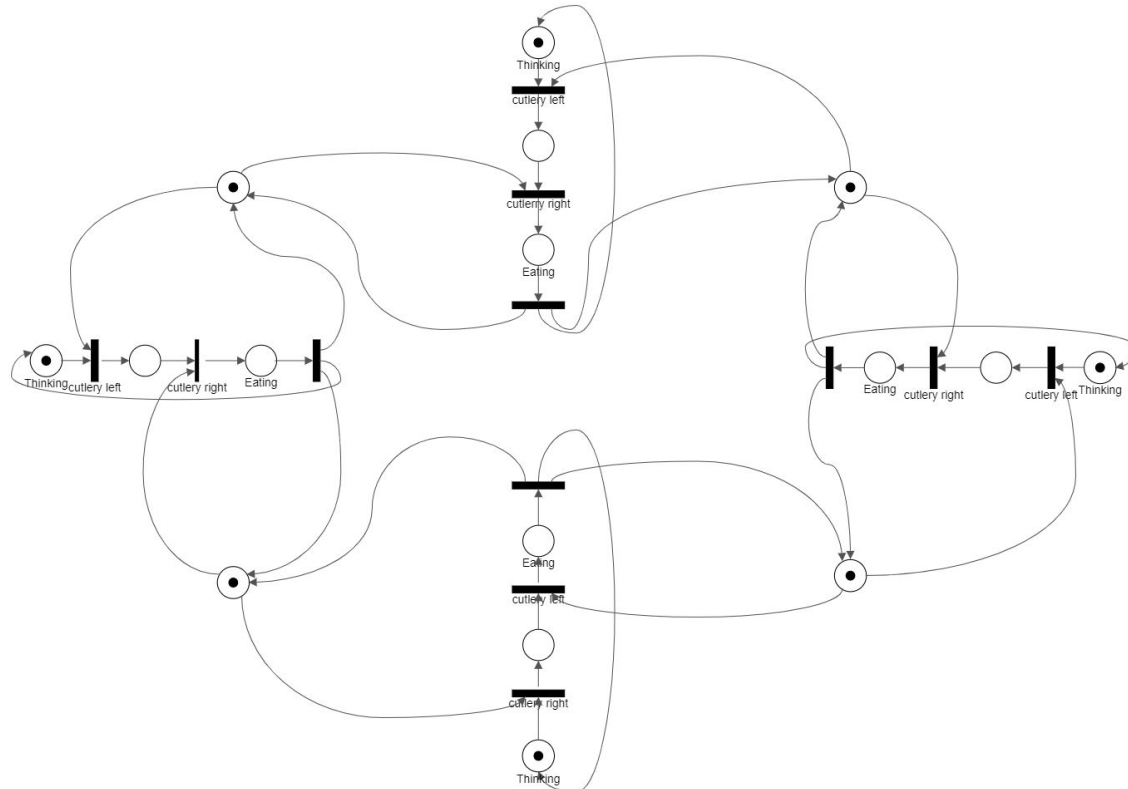
Problema del barbero



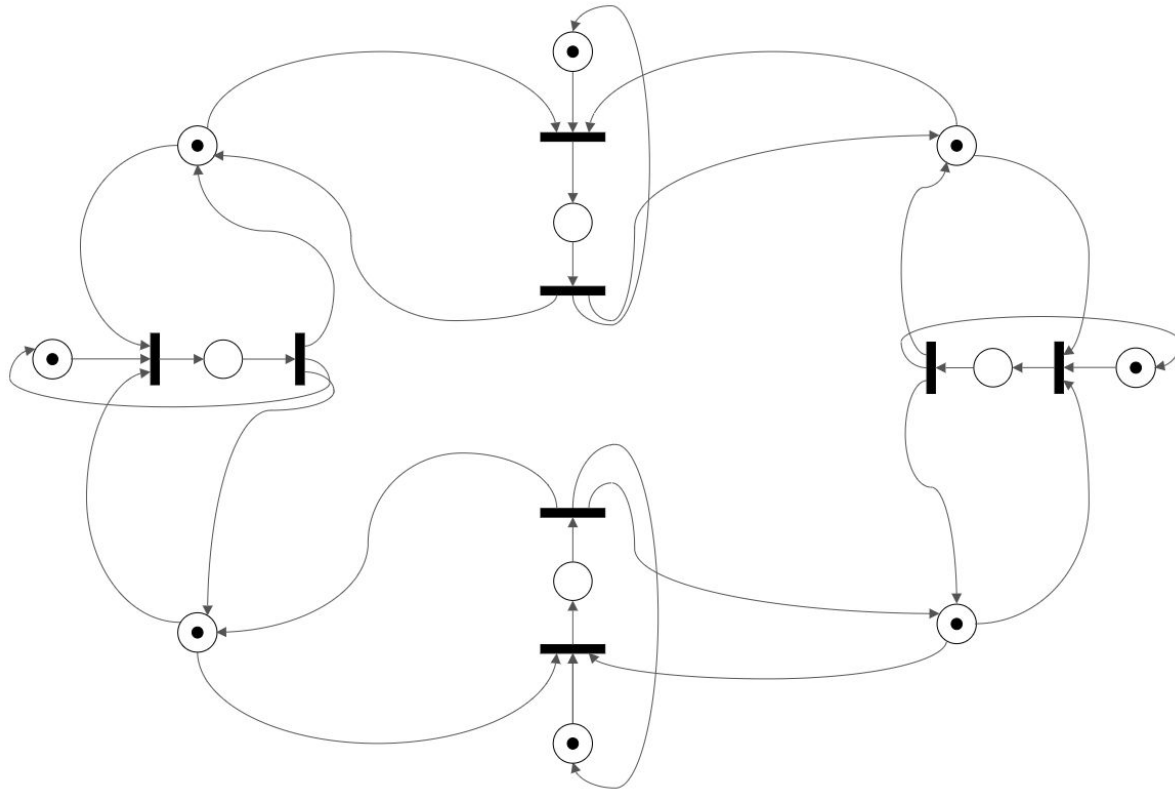
Fumadores



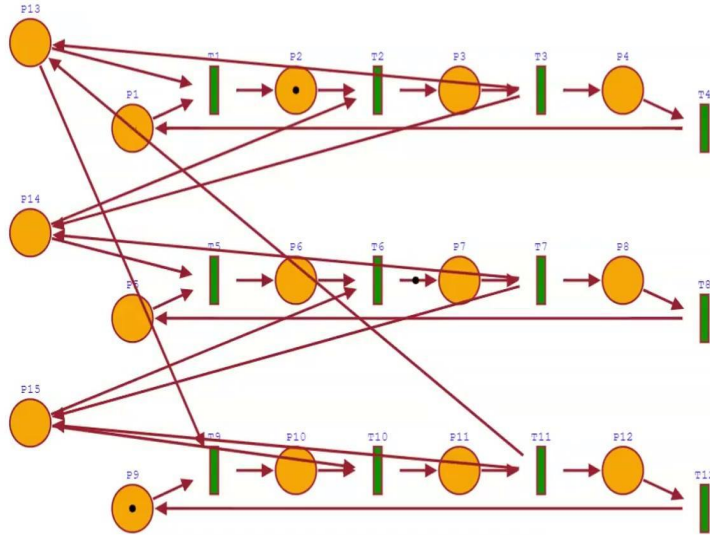
Filósofos I



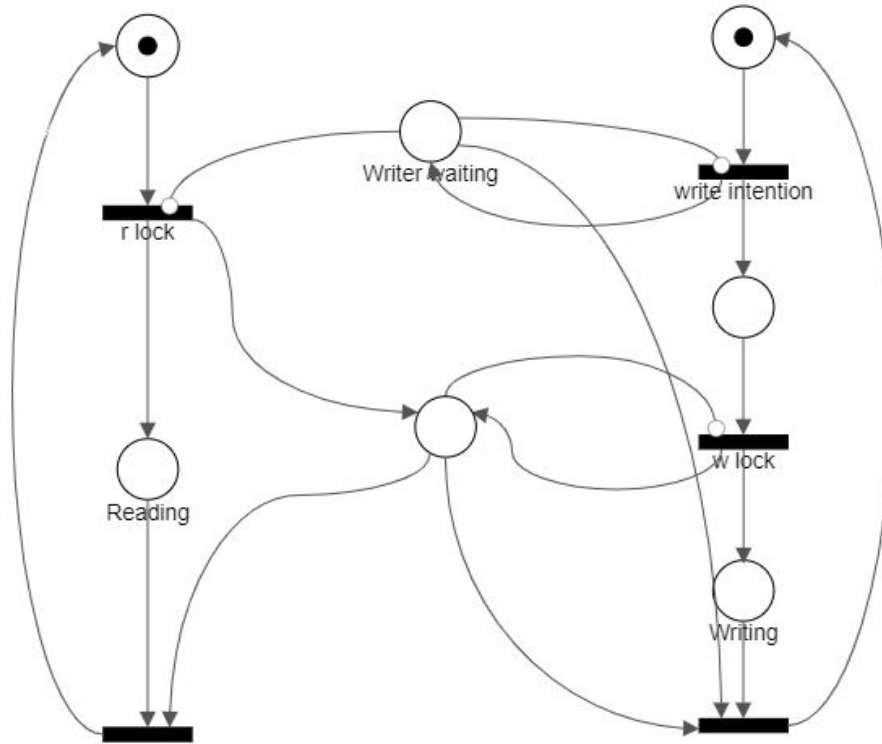
Filósofos II



Filósofos - animación

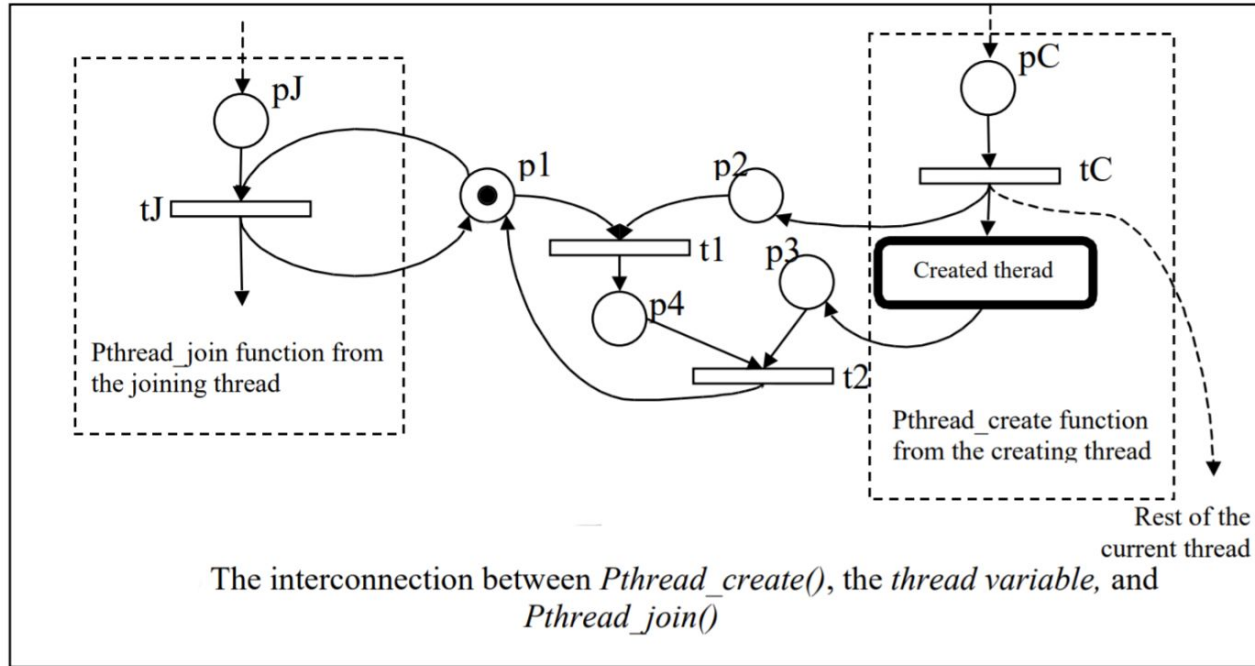


Lector/Escritor

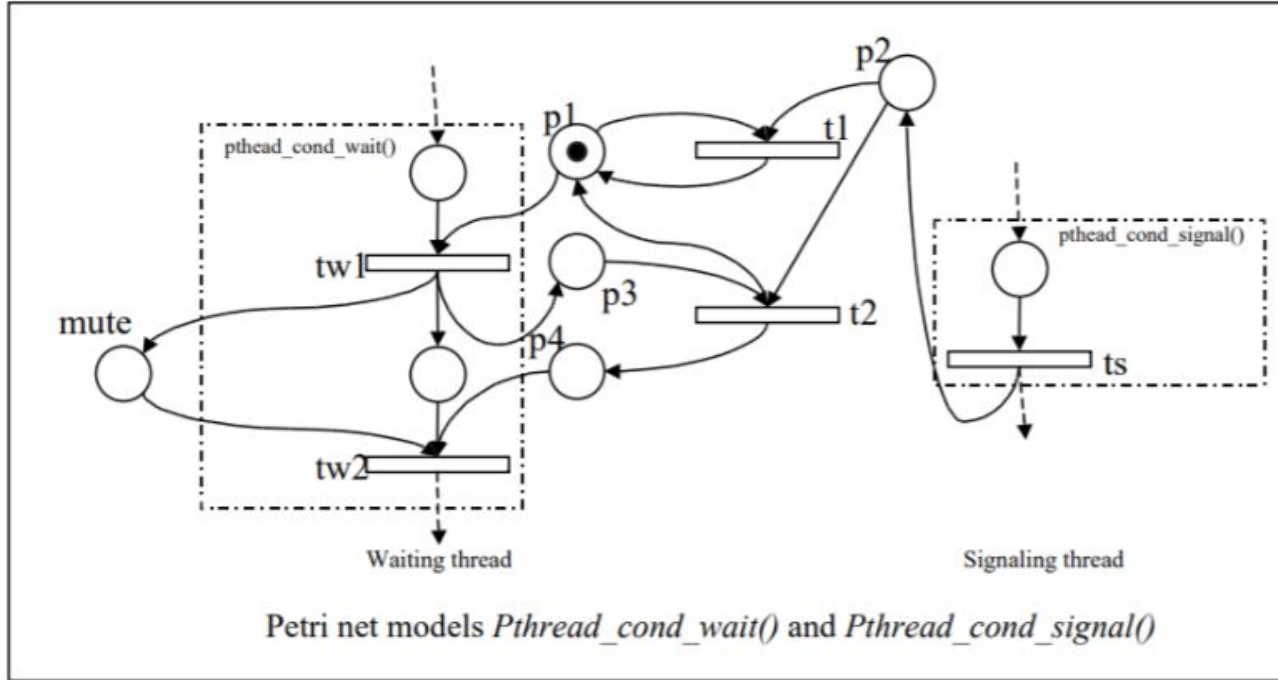


Pthread & c2petri

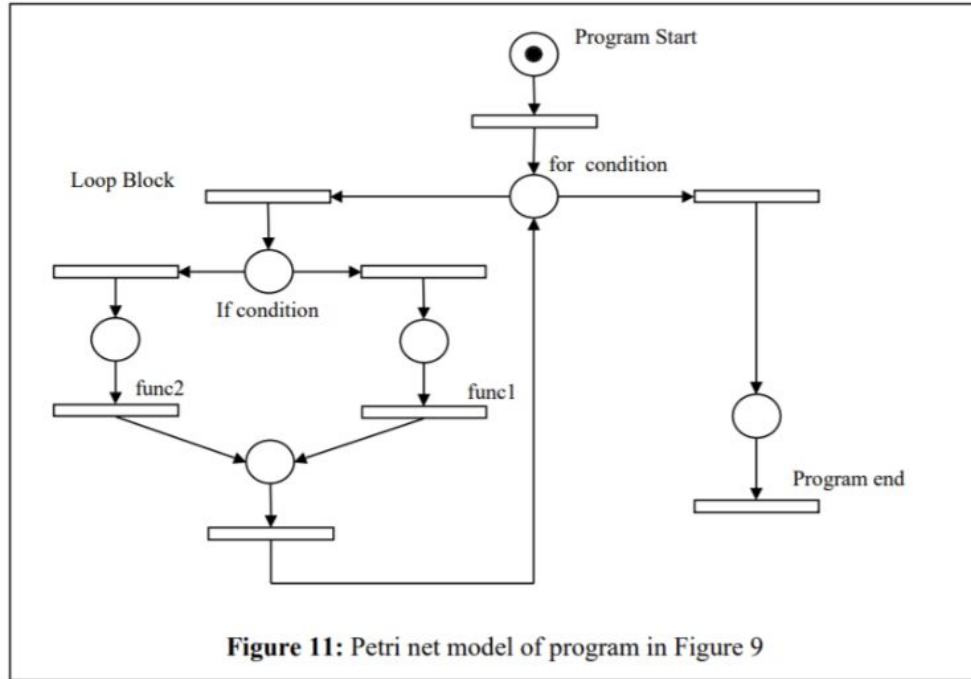
pthread_join



signal & wait



c2petri



Para qué sirve?

- Detectar señales perdidas
- Ausencia de deadlocks
(transiciones salientes para un marking M)
- Ausencia de starvation
(alcance de la transición T_n)

Cerca de 13.200 resultados (0,51 segundos)

Quizás quisiste decir: **c2 petri**

[https://link.springer.com > article](https://link.springer.com/article) · Traducir esta página

Modeling Multithreaded Applications Using Petri Nets ...

por KM Kavi · 2002 · Mencionado por 50 — A tool called **C2Petri** has been developed which converts C-Pthreads programs to the equivalent Petri net model. This tool helps verification of Pthread-based ...

[https://www.researchgate.net > 220...](https://www.researchgate.net/220...) · Traducir esta página

(PDF) Modeling Multithreaded Applications Using Petri Nets

A tool called **C2Petri** has been developed which converts C-Pthreads programs to the equivalent Petri net model. This tool helps verification of Pthread-based ...

[https://www.researchgate.net > 266...](https://www.researchgate.net/266...) · Traducir esta página

ConEE: An Exhaustive Testing Tool to Support Learning ...

... Pthread Concurrir [9], **C2Petri** [48], RegressionMaple [110] Model .Net Gambit [20] Based C#, Java, D DemonL [115] testing Message C Magic [14] passing C, ...

[https://search.proquest.com > openv...](https://search.proquest.com/openv...) · Traducir esta página

Modeling multithreaded applications using Petri nets

por KM Kavi · 2002 · Mencionado por 50 — A tool called **C2Petri** has been developed which converts C-Pthreads programs to the equivalent Petri net model. This tool helps verification of Pthread-based ...

[https://www.tuto.fi > uutiset > petri-...](https://www.tuto.fi/uutiset/petri-...) · Traducir esta página

Jääkiekko - C2 - Petri Niukkanen kanssa sovittu ... - TuTo ry

15 feb. 2019 — Niukkanen vastaa tulevaisuudessa TuTon C1- ja C2- ikäisten pelaajien valmennuksesta yhdessä Mika Seppäsen kanssa. Sopimus on ...

Referencias

- Apunte sobre Redes de Petri disponible en Discord
- Kavi, Moshtaghi, Chen. "Modeling Multithreaded Applications Using Petri Nets". *International Journal of Parallel Programming* volume 30, pages 353–371 (2002).
- Simulador de Redes de Petri por Igor Kim, Novosibirsk University.
<http://petri.hp102.ru/pnet.html>
- Interactive Tutorials on Petri Nets by Wil van der Aalst et al., TU Eindhoven, the Netherlands.
<http://www.informatik.uni-hamburg.de/TGI/PetriNets/introductions/aalst/>
- Flash Emulator: http://flashplayer.fullstacks.net/?kind=Flash_Emulator
- Einar Smith: Carl Adam Petri. Eine Biographie. Springer-Vieweg, 2014, ISBN 978-3-642-40222-7.
https://books.google.de/books?id=WiUjBAAAQBAJ&printsec=frontcover&hl=de&source=gbs_atb#v=onepage&q&f=false

Referencias

- Apuntes de la materia
- Andrea BOBBIO. "System Modelling with Petri Nets". (1990).
<https://www.cse.iitk.ac.in/users/cs698g/papers/miopetrinet.pdf>
- Esparza, Nielsen. Decidability Issues for Petri Nets. (1994).
https://www.researchgate.net/publication/220287072_Decidability_Issues_for_Petri_Nets_-_a_survey

