

# Redes de Petri tiempo real



**MODELADO E IMPLEMENTACIÓN  
DE SISTEMAS DE TIEMPO REAL  
MEDIANTE REDES DE PETRI CON TIEMPO  
FRANCISCO JOSÉ GARCÍA IZQUIERDO**

# Justificación



- Las redes de Petri cuentan con una gran base matemática que permite la verificación de propiedades
  - vivacidad (las cosas pasan)
  - corrección (las cosas pasan de manera adecuada).
- El estudio de estas propiedades permite la validación de los requisitos funcionales y temporales del sistema modelado en etapas tempranas.
  - Se debe tener en cuenta la asignación de prioridades de la etapa de planeación
- De esta manera es posible la detección temprana de comportamientos erróneos, facilitando su corrección antes de la puesta a punto del sistema
- Las redes de Petri facilitan la inclusión en el modelo de un conjunto de elementos redundantes que permitan la comprobación, en ejecución, del comportamiento del sistema
  - Lo que facilitara la etapa de mantenimiento y prueba.

# Formalismos de redes de Petri



- Existe una familia de formalismos, con distintos grados de descripción, desde alto a bajo nivel, cada uno apropiado para distintos propósitos
- Un modelo de red de Petri *autónomo*, *consiste de*:
  - Una *estructura de red*
  - Un *marcado*
- El mero formalismo matemático subyacente tras una RdP no asocia ningún significado a los objetos que la integran; por ejemplo
  - los lugares no tienen por que representar datos o estados,
- La conexión del formalismo con la realidad es propiciada por la *interpretación*.
- Las RdP autónomas son semi-interpretadas, pues aportan algún significado a los elementos de la red:
  - los lugares describen variables de estado
  - las transiciones, los transformadores de los estados y
  - la lógica de la sensibilización/disparo aporta algunas reglas para la descripción del comportamiento dinámico.

# Implicancia de las interpretaciones



- Si una transición modela la finalización de una determinada actividad, puede haber restricciones temporales para su disparo una vez que se encuentra sensibilizada
- Si dos o mas transiciones se encuentran en conflicto, su sindicado (política) puede aportar información sobre como debe resolverse este conflicto.
- Si las restricciones aportadas por la interpretación son tenidas en cuenta, el comportamiento del modelo se altera, se restringe, en realidad.
- Hay diversas extensiones interpretadas de redes de Petri, algunas de las cuales consideran el tiempo

# Semánticas relacionadas con el grado de sensibilización y la prioridad



- Grado de sensibilización de una transición ( $E(t)$ )  
El numero de veces que una transición puede ser disparada concurrentemente  
Servidor sencillo o múltiple
- Redes con prioridad  
la asociación de prioridades  
un numero entero positivo  $\pi(t)$  a las transiciones de la red
- La asociación de prioridades tiene influencia en la resolución de conflictos en RdP, de modo que se sensibilizara
  - será disparada únicamente la transición con mayor prioridad de entre las que cumplan la condición de sensibilización en la RdP

# Interpretaciones con tiempo



- Indeterminismo

No se especifica cuando se disparara una transición que esta sensibilizada (incluso si se disparara realmente)

No se especifica cual de entre un grupo de transiciones en conflicto es la disparada.

- Las interpretaciones con tiempo de las RdP han tratado de reducir el indeterminismo de distintas maneras:

- Las redes de Petri estocásticas, Stochastic PN lo hacen introduciendo una estimación estocástica del instante de disparo de las transiciones
- las redes de Petri temporizadas, Timed PN con un tiempo
- Las fuzzy-timing PN acotan los instantes en los que la transición puede o debe ser disparada



- Un tratamiento estocástico no es admisible en sistemas de tiempo real

# Semántica de tiempo de disparo o de tiempo de sensibilización



- Es posible encontrar dos formas de interpretar (semántica) el parámetro temporal asociado a una transición, lo que puede afectar a la regla de evolución del marcado en dos formas, según se mantenga o no la atomicidad del disparo:
  - **Cuando el parámetro temporal determina el tiempo que ha de transcurrir desde que una transición queda sensibilizada hasta que se dispara**
    - ✦ La retirada y colocación de marcas se produce de forma atómica, se habla de tiempo de sensibilización (enabling time)
  - **El parámetro temporal puede determinar también el tiempo que debe transcurrir entre la retirada (instantánea) de marcas de los lugares de entrada, y la colocación (instantánea) de marcas en los lugares de salida; en este caso se habla de tiempo de disparo (ring time).**
    - ✦ Esto es, el disparo de la transición tiene tres fases (retirada de marcas de entrada, disparo, colocación de marcas de salida) y no es atómico, sino que tienen una "duración", esta interpretación es conocida como semántica de duración.
- La segunda interpretación es un caso particular de la primera, que puede ser simulado por una secuencia:
  - 'disparo inmediato de comienzo de transición' + 'actividad en curso' + 'disparo inmediato de transición de después de n intervalos de tiempo'.
- Entre otras cosas, la interpretación de tiempo de disparo no permite el modelado de actividades interrumpibles, lo que la incapacitara para su uso en el modelado de sistemas de tiempo real. Por ello, elegiremos la semántica interpretativa de tiempo de sensibilización para el disparo de una transición con tiempo

# Formas de especificar el tiempo asociado a las transiciones



- *Redes de Petri con Delay*
  - El tiempo asociado al disparo de una transición
  - Cada transición modela el final de una actividad (representará la duración de la actividad)
  - Se puede hacer explícita esta dependencia especificando los delay de las transiciones mediante expresiones en funciones de timestamps o colores asociados a las marca
- *Redes de Petri con tiempo*
  - Se especifica el tiempo de disparo de una transición mediante la asociación de un intervalo que abarque todas las posibilidades de duración de la actividad.
    - ✦ Esto permite le análisis del peor caso, o el calculo de limites de tiempo de ejecución.



# Semántica de tiempo débil y fuerte



- Semántica de tiempo fuerte, (strong time semantic)
  - Se debe indicar específicamente en que momento y con que condiciones se debe realizar el disparo
  - Es apropiada para el modelado de sistemas de tiempo Real
- Semántica de tiempo débil (*weak time semantic*)
  - La transición no esta obligada a disparar, pero si lo hace, debe ser en el intervalo especiado
  - Esta semántica puede ser útil cuando se trata de modelar una acciones que puede ocurrir únicamente durante un periodo de tiempo, pero que no necesariamente debe ocurrir.
  -
- Sin embargo, esta ultima puede ser modelada, de acuerdo a la semántica fuerte, con una transición en conflicto que aborte la posibilidad de ejecución. Esta ultima alternativa, además, es mas clara.

# Redes de Petri con Tiempo (RdPT)



- Definición formal

Una Red de Petri con Tiempo (RdPT),  $N$ , es una tupla  
 $N \{P, T; Pre, Post; Mo; CIS\}$

- donde:

- ✦  $P$  es un conjunto finito y no vaco de lugares
- ✦  $T$  es un conjunto finito y no vaco de transiciones
- ✦  $P \cap T = \emptyset$
- ✦  $Pre : P \times T \rightarrow \mathbb{N}$  es la función de incidencia previa, donde  $\mathbb{N}$  es el conjunto de números naturales
- ✦  $Post : P \times T \rightarrow \mathbb{N}$  es la función de incidencia post, donde  $\mathbb{N}$  es el conjunto de números naturales
- ✦  $Mo$  es la función de marcado inicial  $Mo : P \rightarrow \mathbb{N}$
- ✦  $CIS$  es una correspondencia de intervalos estáticos,  $CIS : T \rightarrow Q^+ \times (Q^+ \cup \infty)$ , donde  $Q^+$  es el conjunto de los números racionales positivos junto con el cero

- La ultima funcion asocia a cada transicion un par  $(CIS(t_i) = (\alpha_i, \beta_i))$ , que define un intervalo temporal, por lo que se debe verificar:

$$0 \leq \alpha_i < \infty, 0 \leq \beta_i \leq \infty \text{ y } \alpha_i \leq \beta_i \text{ si } \beta_i \neq \infty \text{ o } \alpha_i < \beta_i \text{ si } \beta_i = \infty$$

# Regla de disparo



- Este intervalo permite enunciar la regla :
  - Suponiendo que la transición  $ti$  comienza a estar sensibilizada en el instante  $wi$ , y que continua sensibilizada,
  - el disparo de la transición se producirá no
    - ✦ antes del instante  $wi+\alpha i$ ,
    - ✦ y no mas tarde del instante  $wi+\beta i$ .
  - El intervalo de tiempos de disparo validos para  $ti$  será, por tanto
    - ✦  $[wi+\alpha i, wi+\beta i]$ .
- ❖ La semántica de este disparo es del tipo de tiempo débil, el disparo se puede producir durante el intervalo de tiempo
- ❖ Las marcas permanecen en los lugares de entrada durante el tiempo necesario, y una vez que se produce el disparo, este no consume tiempo: es instantáneo.

# Casos Particulares de Disparo



- Intervalo estático de disparo  $[\alpha_i, \beta_i]$ 
  - - Al valor  $\alpha_i$  se le llama instante de disparo más cercano (EFT) earliest firing time
    - Al valor  $\beta_i$  se le llama instante de disparo más lejano (LFT) latest firing time
- Intervalo puntual  $[\alpha_i, \alpha_i]$ 
  - transiciones de tiempo de sensibilización fijo
- Intervalo sin restricción temporal  $[0, \infty]$ 
  - Formalismo autónomo de redes de Petri
  - Por conveniencia no suele representarse explícitamente

# Estado de una Rede de Petri Temporal



- Para decidir sobre las posibilidades de disparo debemos tener en cuenta dos aspectos:
  - La sensibilización de una transición
  - el tiempo que ha transcurrido de que ha sido sensibilizada
- Por lo cual se define al estado como
  - ✦  $S = (M, I)$ , donde
  - ✦  $M$  es el marcado e
  - ✦  $I$  es un vector de todos los posibles intervalos de disparo de todas las transiciones sensibilizadas por el marcado  $M$ .

# Subclases de RdPT y otras interpretaciones temporales

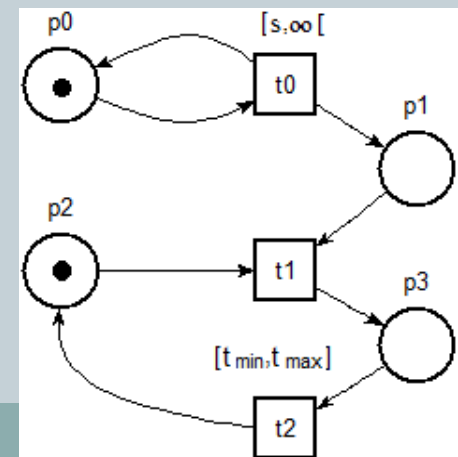


- Time Environment/Relationship nets (TER)
  - Un formalismo de redes de alto nivel similar a las CPN
  - Un formalismo más general, que incluye las RdPT como caso particular
- Los token llevan información temporal en forma de "timestamp", una variable llamada *chronos*
- El disparo de una transición esta condicionado por el valor de *chronos* de las marcas de los lugares de entrada, y produce el valor de *chronos* de las marcas de salida, de acuerdo a la expresión asociada a la transición
  - el valor de *chronos* de las marcas producidas en un disparo debe ser idéntico, y no puede ser inferior al mayor de los valores de *chronos* de las marcas de entrada
- Otras
  - Las redes TB (*Time Basic nets*), caso particular de las TER en las que el unico dato que pueden almacenar las marcas es la variable *chronos*.

# Formalismo a utilizar



- Es el definido como **Redes de Petri con Tiempo**
  - Es simple, es adecuado para modelar sistemas de tiempo real y otros formalismos que consideran el tiempo son modelables por este
  - No es suficiente para modelar sistemas que requieren de CPN
- La semántica de tiempo de sensibilización es la descrita en la filmina **Regla de disparo**
- Con el fin de reducir aun mas el indeterminismo de este modelo consideraremos las siguientes extensiones:
  - Eventualmente asociaremos predicados a las transiciones, que condicionarán el disparo de las mismas. Los predicados dependerán de datos del sistema que no serán explícitamente representados en la red
  - Podremos asignar prioridades a las transiciones, que serán utilizadas para resolver conflictos
  - Para las transiciones que no tengan asociado un intervalo de disparo explicito supondremos un intervalo  $[0,0]$  (hay casos que se considera  $[0, \infty]$  )



# Modelado de sistemas de tiempo real mediante RdP con Tiempo



- En sistemas de RT se debe representar de manera no ambigua aspectos como :
  - Eventos, tanto internos como externos, a los que debe responder el sistema
  - Los patrones temporales que rigen estos eventos, es decir, su periodicidad o aperiodicidad
  - Las acciones activadas por estos eventos y sus características temporales, por ejemplo, su tiempo de computo en el mejor y el peor de los caso
  - Las interacciones entre la distintas acciones, que pueden consistir en sincronizaciones, comunicaciones, relaciones de precedencia u otras



# Tipos de transiciones



- En un sistema de RT existen diferentes situaciones susceptibles de ser modeladas mediante una transición. Con el fin de mostrar en nuestros modelos los diferentes papeles de una transición, y para hacer mas claro y reusable el código, distinguimos entre tres tipos de transiciones:
  - Transiciones CODE
    - ✦ Representan la ejecución de un código o actividad que comienza a ejecutarse en el momento que es sensibilizada, todos los lugares marcados, la etiqueta  $[\alpha_i, \beta_i]$  representa el menor tiempo y el mayor tiempo
    - ✦ Modela las partes operativas del sistema, la prioridad de esta transición esta relacionada con la planificación del sistema
  - Transiciones TIME
    - ✦ Son transiciones asociadas a alguna actividad temporal, como un timeout o la activación periódica de un proceso, la etiqueta  $[\alpha_i, \alpha_i]$  representa el intervalo en que se producirá el ventó después de ser sensibilizado, provocará que se desarrollen acciones de control en el sistema
    - ✦ Modela las partes de control y de supervisión temporal, la prioridad de estas transiciones puede ser usada para resolver conflictos
  - Transiciones SYCO
    - ✦ Son el resto de las transiciones, y no tienen significado temporal explicito asociado. Su disparo será inmediato, lo que supone un intervalo temporal implícito  $[0; 0]$ . Son usadas para modelar sincronizaciones. El disparo de este tipo de transiciones conduce a simples cambios de estado del sistema o es utilizado para sincronizar actividades, modelar eventos no temporales, excepciones, etc
    - ✦ Modela la parte de control y de supervisión temporal, la prioridad de estas transiciones puede ser usada para resolver conflictos

# Modelado de situaciones habituales



- *Ejecución de un código secuencial.*
  - Se representada por una transición CODE junto con sus lugares de entrada y se caracteriza por los tiempos de ejecución máximo y mínimo.
- *Eventos de activación*
  - eventos de entorno , provenientes del exterior
  - eventos internos , ocurrencia de un cambio de estado dentro del sistema
  - eventos temporales, causados por el transcurso de una cantidad de tiempo
- Caracterizan de eventos temporales por el patrón de llegada
  - **Periódico** (periodo T), **irregular**, **acotado** (una separación mínima entre dos ocurrencias), en **ráfaga** y **no acotado**
- *Eventos externos*, situaciones dependientes del entorno que puedan alterar el comportamiento del sistema
  - Por ejemplo interrupciones, son parte del entorno y no del control del sistema, se considera la parte que debe responder

# Modelado de situaciones habituales



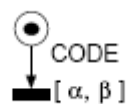
- **Activación Periódica**
  - Todas las interpretaciones que añaden tiempo a las RdPT permiten su modelado
- ***Timeout en ejecución***
  - En la semántica de tiempo de disparo las transiciones interrumpibles no son modelables
  - Además, sin intervalos de tiempo, es imposible modelar actividades de duración variable que puedan elevar el timeout.
- ***Timeout en comunicación.***
  - En redes con semántica de tiempo de sensibilización, el modelado de esta situación se realiza como se indico en la figura, pero no en redes con semántica de tiempo de disparo
- **Transferencia asincrónica de control.**
  - Por la misma razón que en los puntos anteriores, esta situación puede ser modelada por una red con semántica de tiempo de sensibilización, pero no en redes con semántica de tiempo de disparo.

# Modelado de situaciones habituales

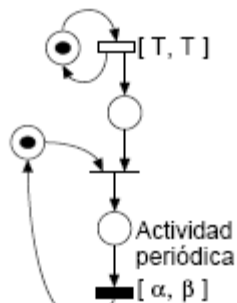


- *Suspensión o aborto de una actividad.*
  - *Timeout en ejecución*
    - ✦ La ejecución de una actividad se abandona si no es finalizada antes de un instante especificado
  - *Transferencia asíncrona de control*
    - ✦ Permite abandonar la actividad si se produce determinado evento
  - *Timeout en una comunicación*
    - ✦ Para evitar una espera infinita

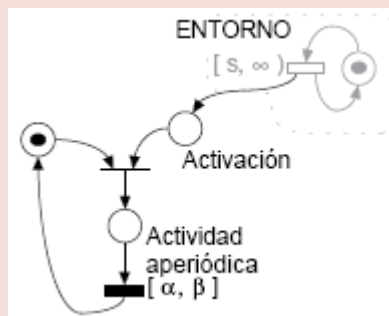
# Modelado de situaciones habituales



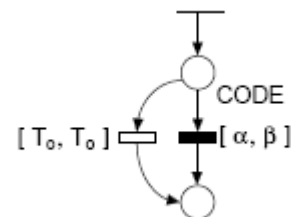
Ejecución de un código secuencial



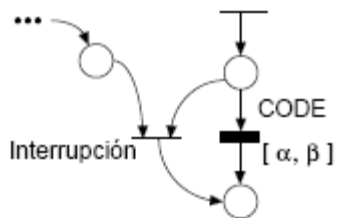
Activación Periódica



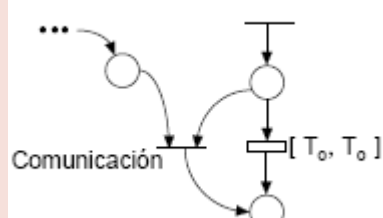
Activación Aperiódica



Time-out en Ejecución de Código



Transferencia Asíncrona De control



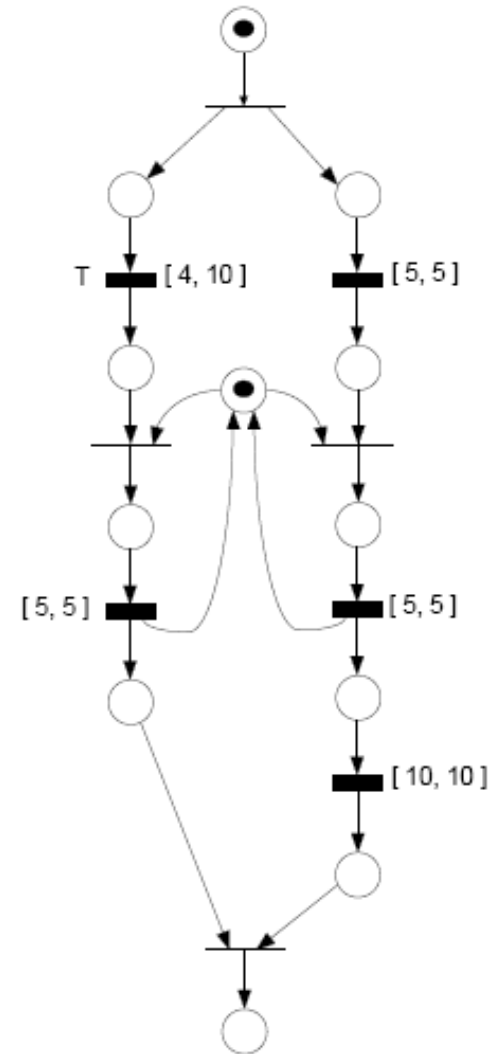
Time-out en Comunicación

# Modelado de situaciones habituales.



## *Ejecución de código secuencial*

- El uso de intervalos temporales asociados a las transiciones permite la representación de código cuyo tiempo de ejecución dependa del estado o de entradas del sistema.
- La ejecución de todos los elementos en su peor caso no equivale al peor caso de ejecución del sistema completo. Si el tiempo de ejecución de la transición T es el de peor caso (10 u.t.) la red se ejecutara en 20 u.t.; sin embargo, si el tiempo es el de mejor caso (4 u.t.), la red tardará 24 u.t.



# Restricciones del Modelo



- Peso de los arcos: Con peso unitario o entero
  -
- Arco Inhibidor (si o no)
  -
- Plaza limitada (si o no)
  -
- Que las transiciones puedan ser disparada al menos una vez
  - Permite que todas las partes del sistema sean ejecutables y
  - permite modelar las condiciones iniciales (con plaza iniciada a uno) y de finalización (con plaza limitada en uno)
- Grado de sensibilidad de la transición
  - Uno, sistemas secuenciales sin código reentrante
  - Dos, sistema paralelo con código reentrante
  - Mas de dos ...
- Transiciones con y sin lugares de entrada
  - Transiciones que pueden ser disparadas para introducir marcas en el sistema (si o no)
- Cada lugar puede tener
  - Una transición CODE de salida, esto modela que un sistema no ejecute simultáneamente dos códigos distintos
  - Mas de un TIME donde  $[\alpha, \alpha]$  y se cumple que  $\alpha > 0$
  - Mas de un SYSCO  $[0, 0]$

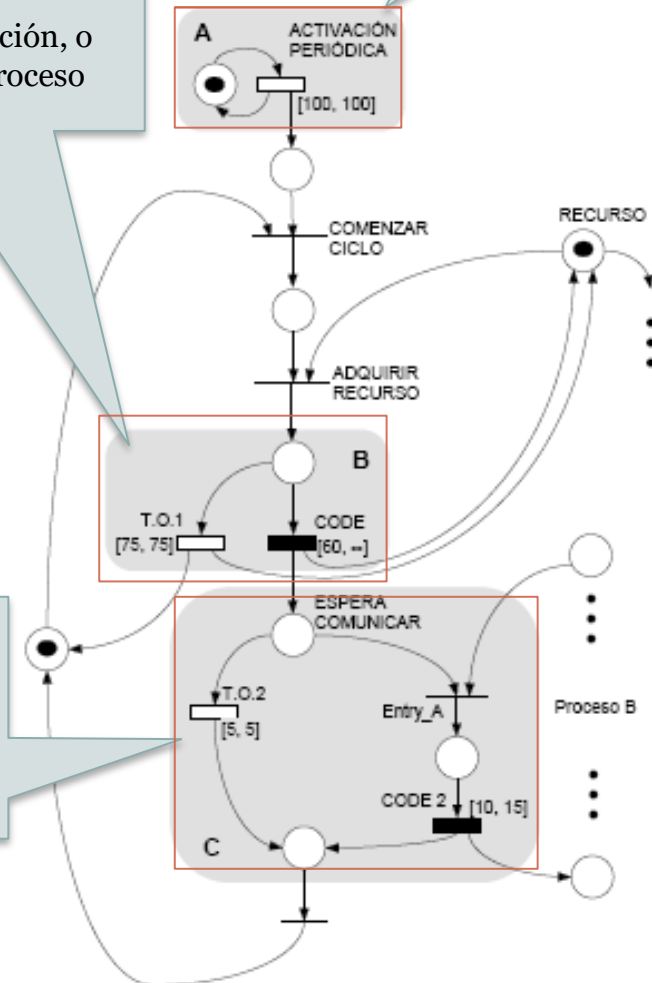
# Ejemplos de modelado

La caja A modela la activación periódica del proceso

- La figura modela un fragmento de un sistema de RT compuesto por:
  - Un proceso periódico ( $T=100$  u.t.)
  - El proceso adquiere un recurso para su ejecución (código, CODE) que tiene un timeout  $T.O.1=75$  u.t. Tras la ejecución del código, el recurso es liberado
  - Luego se establece una comunicación con un segundo proceso sincronizado en entry A, con un timeout asociado  $T.O.2=5$  u.t)

La caja B (CODE) una acción, o código, que ejecuta el proceso

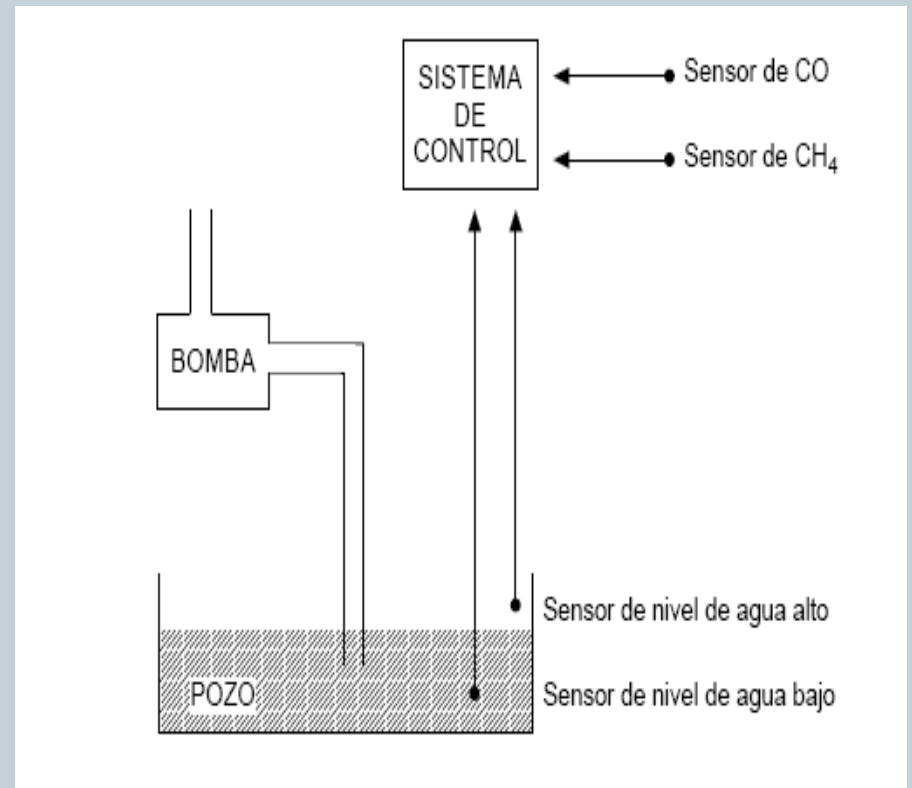
La caja C comunicación síncrona con el proceso B que tiene un timeout





# Caso de estudio

- Este caso es muy estudiado, en la bibliografía sobre sistemas de tiempo real, porque muestra de forma simple las características más típicas de sistemas de RT.
- Sistema simplificado para controlar una bomba que trabaja en una mina.
- - El objetivo de este sistema es bombear a la superficie agua extraída del sumidero de un pozo de una mina. El principal requisito de seguridad es que la bomba no debe funcionar cuando el nivel de gas metano en la mina es alto, debido al riesgo de explosión, en la figura se presenta el diagrama esquemático del sistema.



# Caso de estudio



- **Requerimientos Funcionales**

- Podemos dividirlos en cuatro componentes, las que son: accionamiento de la bomba, monitorización del entorno, interacción con el operador y monitorización del sistema.
- **Accionamiento de la Bomba**
  - ✦ Se monitorean los niveles de agua en el sumidero, y las acciones son las siguientes:
  - ✦ Si el agua alcanza el nivel superior o lo indica el operador se enciende la bomba para drenar el pozo
  - ✦ La bomba puede funcionar si el nivel de metano está por debajo del valor crítico
  - ✦ Si el agua alcanza el nivel inferior o lo indica el operador (se apaga la bomba)
- **Monitorización del entorno**
  - ✦ Se detecta el nivel de metano en el aire ya que hay un nivel por sobre el cual no es seguro extraer carbón o drenar agua. También se mide el nivel de monóxido de carbono y el flujo de aire y si esto tienen valores críticos los debemos señalar con una alarma.
- **Interacción con el operador**
  - ✦ El sistema es controlado por el operador por medio de una consola instalada en la superficie, dicha consola informa al operador de todos los eventos críticos.
- **Monitorización del Sistema**
  - ✦ Todos los eventos del sistema son guardados para luego poder visualizarlos.

# Caso de estudio



- ***Requerimientos no Funcionales***

- Podemos dividirlos en tres componentes, las que son: tiempo, confiabilidad y seguridad. En este caso de estudio nos centramos en el tiempo, para una consideración completa se puede encontrar en [\[31\]](#)

- **Periodo de Monitorización**

- ✦ Asumimos que todos los sensores son leídos cada 100ms. También asumimos que el tiempo de conversión para los sensores de CH<sub>4</sub> y CO es de 40ms, por lo que el tiempo límite es de 60ms.
- ✦ El sensor de flujo de agua tiene como objetivo corroborar que el agua fluye mientras la bomba está encendida y que ha dejado de fluir cuando está apagada, para realizar esta última comprobación se debe esperar un segundo cuestión que el agua deje de fluir y el estado real de la bomba se determina con dos lecturas consecutivas, estas lecturas deben ser divisibles por 40ms por lo que se realiza una a los 960ms y la siguiente a los 1040ms después de haber apagado la bomba.
- ✦ Suponemos que los detectores de agua están dirigidos por eventos y el sistema debe responder en 200ms. El modelo de control (función de transferencia) del sistema muestra que debe haber al menos 6 segundos entre las interrupciones de las indicaciones y los niveles del agua.

# Caso de estudio



## *Requerimientos no Funcionales*

- **Tiempo límite de parada**

- Para evitar explosiones se debe apagar la bomba en un tiempo límite a partir de que el nivel de metano pasa el tiempo límite. Esto está relacionado con el periodo de muestreo, con la velocidad que se acumula el metano en la mina y con los márgenes de seguridad de la reglamentación y normativa vigente. Haciendo una lectura directa del sensor, la relación es expresada por la inecuación
  - ✦  $R(T + D) < M$
  - ✦ Donde
    - ✦  $R$  es la tasa de acumulacion del Metano
    - ✦  $T$  es el periodo de muestreo
    - ✦  $D$  es el tiempo limite de parada
    - ✦  $M$  es el margen de seguridad
- Si se usa desplazamiento de periodo se requiere un periodo de tiempo adicional, y la inecuación resulta
  - ✦  $R(2T + D) < M$
- Para este estudio suponemos que la presencia de bolsas de metano puede producir rápidos incrementos del nivel, por lo que asumimos un tiempo límite conservador (desde que el metano sobrepasa el nivel hasta que la bomba se detiene) de 200ms. Esto se obtiene con una configuración de sensor menos de 80ms con un tiempo límite de 30 ms. Este nivel nos asegura que se obtienen lecturas correctas del sensor (el desplazamiento entre lecturas es de al menos 50ms).

# Caso de estudio



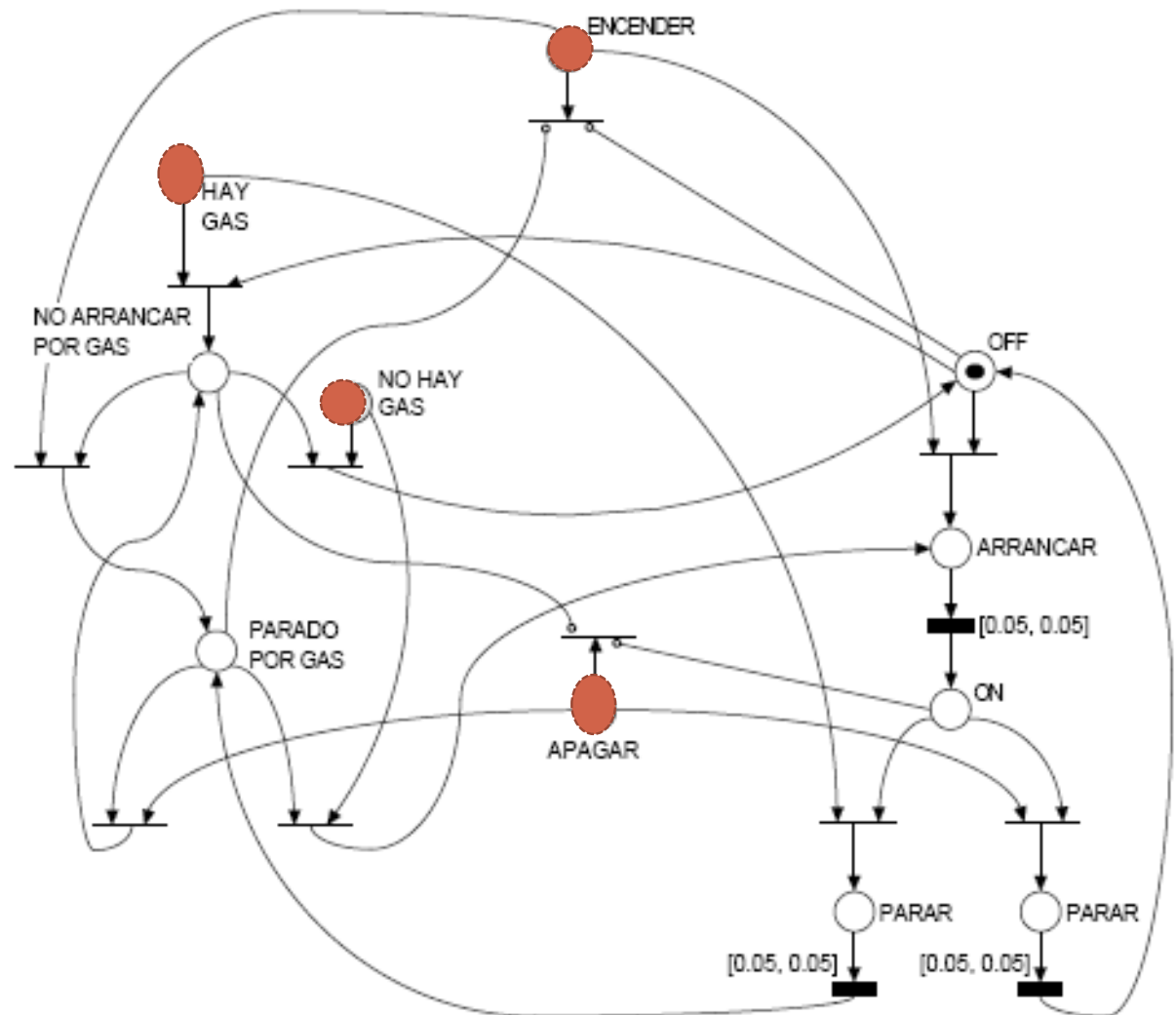
- **Tiempo límite de información al operador**
- El operador debe ser informado:
  - Cada un segundo
    - ✦ Lecturas por sobre el límite de monóxido de carbono y metano
  - Cada dos segundos
    - ✦ Lectura por debajo del valor critico de flujo de aire

	Periódico/Espon taneo	Periodo	Tiempo limite
Sensor CH <sub>4</sub>	P	80	30
Sensor CO	P	100	60
Flujo de aire	P	100	100
Flujo de agua	P	1000	40
Detector de nivel de agua	E	6000	200

# *RdPT que modela el sistema de control de la bomba de extracción de agua de la mina*

Modelo del controlador de la bomba en normal

- encendida (ON) o
- apagada (OFF)
- no arrancar por gas
- deshabilitada si el nivel de metano es muy alto y volver al estado anterior si desciende
- o parada por gas

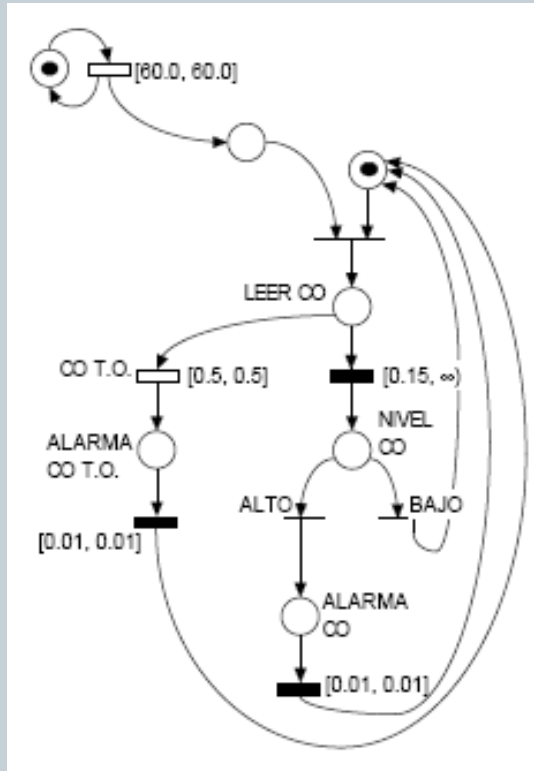


# RdPT que modela la lectura de los sensores del sistema

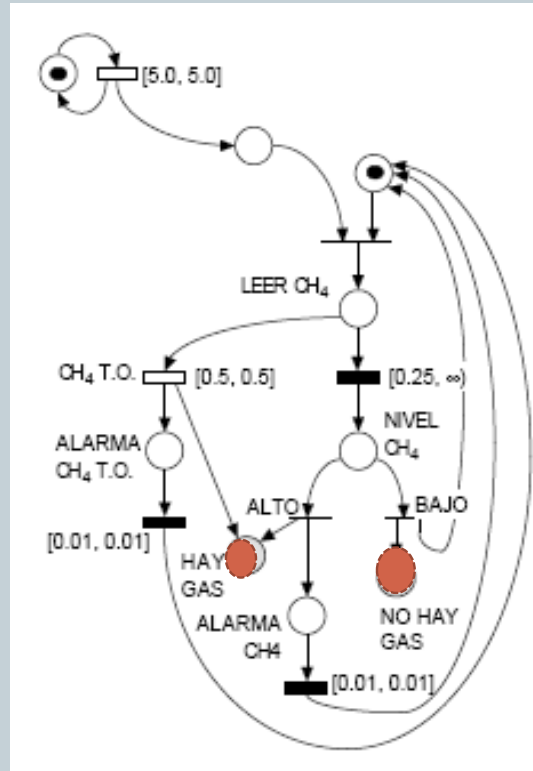


muestreo de los niveles de metano y monóxido de carbono

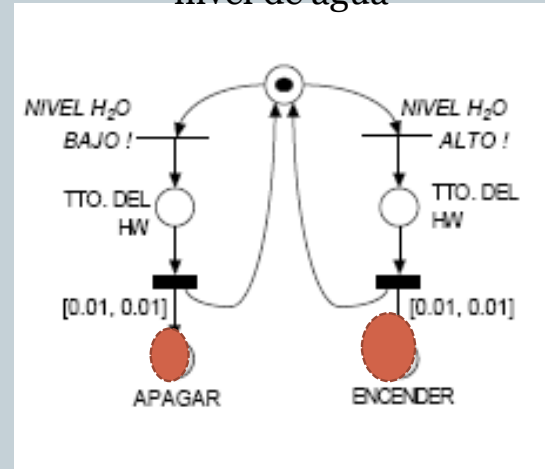
Predicado para indicar el nivel de agua



60 segundos para el sensor de CO



metano 5 segundos



*Plazo de apagado* en presencia de un nivel de metano superior al umbral máximo de 1 segundo

*Plazo de información al operador* de 1 segundo, ante niveles de metano y/o monóxido de carbono superiores a los permitidos.

se ha protegido el acceso mediante un timeout: si el dispositivo no responde, la lectura es suspendida y el operador es informado

# Planificación



- El modelo teórico de concurrencia de las redes de Petri supone la existencia de infinitos recursos
  - Lo que implica que cualquier unidad de ejecución que pueda ser ejecutada lo será en paralelo con las demás en la misma situación
- La implementación cuenta con un conjunto de restricciones impuestas por la plataforma
  - *restricciones de hardware*, recurso finitos (procesadores, memoria, etc.)
  - *restricciones de software*, sistema operativo (scheduling), lenguaje de codificación (primitivas de concurrencia y tiempo real)
  - prioridades de los procesos, hay que garantizar, mediante la adecuada asignación de prioridades u otros medios, que todos los requisitos de tiempo real sean cumplidos
    - ✦ No se conoce trabajos de TPN con prioridades dinamicas



# Análisis



- **Se cumplen todos los requisitos temporales?**
  - Los métodos de análisis de redes de Petri extendidas con tiempo están basados principalmente en técnicas enumerativas relacionadas con el calculo del grafo de alcanzabilidad
  - La construcción de un grafo de estados en redes de Petri con tiempo es complicado, ya que los intervalos temporales hacen que el numero de estados alcanzables (estado e instante en el que se alcanza) sea muy grande, e incluso infinito en tiempo continuo.
  - El problema se ha solucionado mediante discretización temporal o por medio de la definicio de clases de estados y la construcción de un grafo de clases de estados, State Class Graph (ver la bibliografía)

# La implementación. Aspectos generales



- El software debe:
  - ✦
  - Simular el disparo de las transiciones
  - Respetar las reglas de evolución del marcado del modelo de la red (el estado del sistema modelado)
    1. Estableciendo la regla de sensibilización de una transición
    2. Disparo de las transiciones sensibilizadas
    3. Actualización del marcado
- La evolución del estado de la red, y en consecuencia las interpretaciones de esta influyen muy directamente en los algoritmos de implementación reduciendo el indeterminismo implícito en el segundo punto de la anterior regla de disparo
- Simplificaciones:
  - Semántica de servidor único
  - Disparo inmediato de transiciones SYCO
  - Comienzo inmediato de la ejecución del código asociado a las transiciones CODE
  - Semántica de tiempo de sensibilización
  - **Política de resolución de conflictos**
    - ✦ Primero SYCO, que son instantáneos, si hay varios se implementa prioridad
    - ✦ Segundo CODE y TIME, se dispara antes el que termina antes (tiempo de sensibilización menor). En caso de igualdad TIME más prioritaria

# Implementaciones Secuencial/Concurrente



- **Concurrencia ejecución simultanea**
  - la concurrencia en el disparo puede interpretarse como entrelazado
  - puede pensarse en implementaciones secuenciales construidas con procedimientos.
  - una implementación secuencial el orden de disparo de las transiciones concurrentes tiene gran importancia desde el punto de vista del comportamiento del modelo