

# ESTADO DEL ARTE EN MODELOS TEMPORIZADOS Y PROBABILÍSTICOS

Juan José Pardo Mateo

*Licenciado en Informática y actualmente becario en el  
Departamento de Informática dentro del Grupo de Modelos  
Formales de Concurrencia*

## RESUMEN

Con el presente trabajo se pretende dar una visión general sobre los modelos temporizados y probabilísticos utilizados para la descripción de sistemas concurrentes, centrándonos especialmente en aquellos lenguajes basados en álgebras de procesos. Realizaremos una breve descripción de las características de cada uno de los modelos.

## 1. INTRODUCCIÓN

CUANDO una persona se enfrenta a la especificación de un proceso se le plantea el gran problema de seleccionar que lenguaje utilizar.

Muchas veces, a la hora de elegir un lenguaje para realizar una especificación, se cae en el error de pensar que un lenguaje de especificación es lo mismo que un lenguaje de programación, pero nada más lejos de la realidad ya que existe una diferencia fundamental entre ambos tipos de lenguaje como es que el de especificación se limita a indicar qué realizará el sistema mientras que el de programación indica cómo realizará las cosas el sistema. Esta diferencia se reduce cada día más, debido a la proliferación de compiladores que permiten la ejecución de especificaciones.

Aunque casi todos los lenguajes de especificación han sido diseñados para describir el comportamiento como una secuencia de eventos ordenados en el tiempo, es extremadamente importante poder especificar restricciones de tiempo de una forma cuantitativa ya que cada día es más necesario poder expresar el comportamiento de sistemas que dependan estrechamente del tiempo.

Además de la especificación del tiempo cuantitativo, la caracterización probabilística es otro tipo importante de requerimientos no funcionales impuestos a un sistema.

Estas necesidades de especificación han hecho necesario la extensión del poder expresivo de los lenguajes de especificación para modelar restricciones de tiempo y de probabilidades. Lo que ha incrementado de una forma significativa la utilidad de los FDT (Formal Description Techniques) y decrementado significativamente el coste inducido por la evaluación de rendimientos.

## 2. TÉCNICAS BASADAS EN LOTOS

### Quemada-Fernández

J. Quemada y A. Fernández en [QF87] presentan una extensión de LOTOS con tiempo cuantitativo, consistente en asociar a cada evento un instante de tiempo relativo al principio de su creación que indica cuando puede tener lugar. Este nuevo atributo implica que la sincronización puede tener lugar sólo entre acciones que tengan el mismo atributo temporal y que el tiempo debe avanzar al mismo ritmo en todos los procesos del sistema, por lo que la evolución temporal en un proceso provocará la actualización temporal del resto de los procesos en paralelo, causando así que algunas acciones no estén disponibles, representando así restricciones de tiempo general.

Permitirá modelar comportamientos síncronos y asíncronos dependiendo de si se fuerza a que un evento ocurra en un único instante de tiempo o dentro de un intervalo.

El modelo considera el tiempo discreto y las ocurrencias de eventos instantáneas.

Esta extensión con tiempo absoluto fue criticada porque no permitía el correcto modelado de la descripción de timers, así como los resultados de su expiación, permitiendo a los sistemas desarrollarse inapropiadamente, obteniendo como resultado un comportamiento anómalo. La solución fue dada en [QAF89] donde se planteaba el modelo TIC consistente en una interpretación del operador paralelo con el criterio que llaman *must timing*, que hace que el proceso paralelo intercale los eventos en el orden temporal correcto, en contraposición al criterio llamado *may timing* que se utilizaba anteriormente.

Este nuevo modelo permite modelar restricciones generales de tiempo, así como comportamientos síncronos y asíncronos.

Posteriormente en su artículo [QAF90], los autores extienden el modelo de modo que la definición de la equivalencia observacional basada en bisimulación débil es congruente respecto a todos los operadores excepto el operador elección. Esta extensión consiste en la introducción de un evento especial que representa el paso del tiempo y permite modelar, a un nivel semántico, el cambio de estado de un sistema por el paso del tiempo.

## Bochmann-Vaucher

En [BV88] Bochmann y Vaucher hacen una extensión informal de la técnica de Estelle, SDL y LOTOS, lo que permite utilizar esta técnica para la evaluación de rendimientos.

La extensión es realizada de la siguiente forma:

- **Tiempo.** A cada evento interno se le asocia un atributo que indica el instante de tiempo en el cual puede ocurrir. Un evento interno sin atributo temporal tendrá lugar inmediatamente, mientras que uno observable tendrá lugar cuando el entorno lo determine. El criterio ASAP es aplicado de forma implícita. Asume que la ocurrencia de un evento no consume tiempo, por lo que esto podría ser expresado mediante retrasos o comportamientos asíncronos.
- **Probabilidad.** A cada evento interno se le asocia un atributo que indica el peso (probabilidad) de cada evento dentro de una alternativa.

Esta extensión no ha sido formalizada en un modelo semántico.

## LOTICAL

Brinksma, en su tesis [BRI88] extiende LOTOS, sintáctica y semánticamente, principalmente para incrementar la aplicabilidad de LOTOS, así como mejorar el estilo de especificación.

Las extensiones más significativas son las siguientes:

- Las acciones, más que un solo evento, son un conjunto finito de eventos; lo que permite la asociación de la terminación exitosa de un proceso con la ocurrencia de un evento observable.
- Soporta la descripción de módulos LOTOS, que facilitan la especificación jerárquica de sistemas grandes y complejos.
- La especificación completa consistirá en un conjunto de módulos que pueden ser manejados independientemente.
- Sugiere ideas para futuras extensiones, entre las que se encuentra el tiempo cuantitativo, sincronización en puertos indexados con datos, así como terminación negativa en la composición paralela como la generalización del operador de deshabilitación como es propuesto en PADS.

Aunque la existencia de acciones compuestas de diferentes eventos permitirá la modelización de la concurrencia real, Brinksma la usa para modelizar la ocurrencia atómica de un conjunto de eventos entre los cuales se puede encontrar una terminación satisfactoria de un proceso.

Sugiere la idea de introducir tiempo cuantitativo usando *timers*, que pueden ser inicializados de modo que expiren después del paso de  $n$  instantes de tiempo y puedan ser cancelado antes de que expiren.

Todos los timers especificados sincronizan en una puerta especial (evento de reloj explícito  $\omega$ ), que representa el paso de una unidad de tiempo (tiempo discreto).

La expiración de un timer hace que el paso del tiempo sea imposible hasta que el proceso que use éste atienda a la expiración. Esto significa que el especificador de los timers debe cancelar todos los inicializados en el caso de que no pueda atender a su expiración. Esta idea se basó en la dependencia del tiempo de gran parte de la existencia de acciones compuestas de diferentes eventos, entre los cuales debe estar el evento  $\omega$ .

Las acciones que no sincronizan con el «time-out» de un timer se producirán asíncronamente, pero se podría forzar a un evento a que ocurriese antes de que pase un tiempo dado; lo que permitiría modelar comportamientos síncronos basados en el evento de reloj  $\omega$ .

## Lotos Temporal

En el artículo [SR88], C. Smith y S. Rudkin extienden LOTOS con tiempo cuantitativo y sustituyen el lenguaje de especificación de TAD ACT-ONE por ASN-1.

Ellos introducen el operador llamado *temporal process constructor*  $[@t]P$ , que impone un retraso « $t$ » en la ocurrencia de las acciones disparadas por el proceso  $P$ .

La semántica operacional atiende al modelo del criterio ASAP que causa que los eventos ocurran tan pronto como sea posible y además una expresión producirá una acción sólo si no hay posibilidad de que ocurra una acción interna de antemano.

El especificador maneja tiempo real discreto, pero el modelo semántico acumula el tiempo pasado desde un estado inicial, y todos los cambios de estado (transición) llevan como atributo el tiempo total pasado.

Aunque el lenguaje es conceptualmente correcto, la semántica operacional del operador paralelo no tiene en cuenta la actualización temporal de todo el sistema cuando una de sus partes evoluciona en el tiempo. Además el criterio temporal ASAP en el operador de ocultamiento es aplicado de forma incompleta.

Este modelo es asíncrono en el sentido que un evento ocurre en un instante después de que esté disponible, pero tan pronto como sea posible. Se asume que las ocurrencias de los eventos son instantáneas.

## PADS

A. Azcorra en su tesis doctoral [AZC90] propone varias mejoras sobre LOTOS para adaptarlo al diseño de sistemas distribuidos tanto síncronos como asíncronos.

Las mejoras incluidas son:

- Introduce eventos compuestos con una estructura de bolsa como en SCCS, lo que permitirá modelar concurrencia real.
- El operador *composición secuencial* es modificado y generalizado de modo que permitirá asociar la ocurrencia de un evento observable a la terminación de un proceso. Esto es considerado como una terminación perjudicial como se sugiere en LOTCAL. La generalización permite el manejo de excepciones y una secuenciación más flexible.
- Modelización de tiempo real. Se define un modelo básico que usa un evento especial  $t$  interpretado como el paso de una unidad de tiempo, mientras el resto de acciones no consumen tiempo. Después, basándose en el modelo básico anterior, se define un lenguaje de eventos con atributos temporales donde el evento especial  $t$  aparece solamente en el modelo semántico. Éste no está basado en el concepto de retraso; un evento puede ser deshabilitado debido al paso del tiempo.

El bloqueo temporal de uno de los componentes del sistema implica el bloqueo del sistema completo ya que todos los componentes sincronizan en  $t$  y para evitar esto se define un proceso *idle* que solo permite el paso del tiempo.

El modelo temporal permite la especificación de sistemas síncronos (usando el evento  $t$  dentro de un evento compuesto), así como el de asíncronos.

## CELOTOS. Lotos de reloj extendido

Hulzen, Tilanus y Zuidweg en su artículo [VHT89] introducen en el lenguaje, tanto a nivel semántico como sintáctico, unos relojes que pueden inicializarse o consultarse como valores de una clase particular. El tiempo pasa igual para todo el sistema siendo indicado por un reloj global que genera ticks (a nivel semántico) y el cual nunca podrá ser bloqueado. Los valores de los relojes se incrementan en una unidad cada vez que se produce un tick de reloj. No existe ningún mecanismo que permita que una acción se realice antes de un instante especificado debido a que el tiempo pasa siempre.

Igualmente es posible que el sistema no ejecute ninguna acción por lo que el sistema debería ser clasificado como asíncrono.

La introducción de relojes dentro del lenguaje incrementa considerablemente la complejidad del modelo semántico, aunque tiene la ventaja de facilitar la descripción de las restricciones temporales de los eventos no consecutivos en el comportamiento.

## C. Calvelli LOTOS

Calvelli en su artículo [CAL89] extiende LOTOS con un nuevo proceso *wait(t)* que representa la terminación exitosa después de un tiempo  $t$ .

El modelo denotacional está basado en el modelo CSP temporizado, pero sin la aplicación (inicialmente) del criterio ASAP.

Los criterios temporales modelados son básicamente los de CSP Temporizado:

- El tiempo es continuo
- Todas las acciones requieren un tiempo  $\tau$  para ser ejecutadas.
- Las llamadas recursivas de un proceso consumen un tiempo  $\tau$ .
- Los eventos ocurren asíncronamente, por lo que solo es posible retrasar su ocurrencia.

Define una interesante relación de equivalencia al aplica el criterio ASAP. El criterio ASAP no es aplicado cuando define la función semántica porque al eliminar las acciones internas hace que sea incongruente.

Además el consumo de tiempo por las acciones significa que no se puede aplicar el teorema de expansión para expandir el comportamiento con componentes paralelos.

## Timed-Accion LOTOS, Timed-Interacción LOTOS

En el artículo [BLT90], Bolognesi, Lucidi y Trigila proponen dos extensiones a LOTOS inspirados en la expresividad de los dos modelos temporizados de Redes de Petri.[WAL83],[MF76].

El primer modelo, llamado Timed-Action LOTOS posee las mismas características del modelo de Quemada y Fernández.

El segundo denominado Time-Interaction LOTOS, introduce dos nuevos conceptos:

- Urgencia. Expresa realmente una absoluta prioridad en la ocurrencia de un evento. No considera eventos internos que es donde podría aplicarse ASAP.
- Restricciones temporales no locales. Impone una restricción temporal a nivel de proceso, pero no a nivel de acción, como en el modelo anterior.

La acción prefijo y el operador paralelo son manejados de la misma forma que en el modelo TIC [QAF89], aplicando el criterio *must timing* en el operador de elección, además de en el operador paralelo. Esto es, una acción de la elección solamente es posible en un tiempo  $t$  si existe alguna acción en la elección esperando para un instante  $t' \geq t$  o no existen más acciones.

En principio, todas las acciones pueden ocurrir en un instante de tiempo, aunque pueden imponerse restricciones temporales para la ocurrencia de acciones, que son expresadas como «*timer a <x,y> in P*» cuya interpretación es que *contando desde el inicio del ofrecimiento de la acción a en el proceso P y solo durante el tiempo en el cual es ofrecida a ocurrirá urgentemente entre los instantes x e y, o el sistema se bloqueará.*

### 3. TÉCNICAS BASADAS EN CCS

#### SCCS

Generaliza CCS para modelar sistemas síncronos que fue propuesta en [MIL83].

Define las operaciones: prefijo, alternativa, restricción, renombrado. Y el operador paralelo que solo soporta rendezvous binario. Además siempre produce productos de eventos (conurrencia real) y nunca interleaving.

Carece de eventos internos, pero introduce la acción «*idle*» conocida como «*1*» que es interpretada como un evento interno de CCS, de modo que el no-determinismo producido por la alternativa entre dos eventos internos se pierde.

El tiempo está basado en el concepto de retraso en el ofrecimiento de una acción, la cual consume una unidad de tiempo discreto, lo que significa que para modelar la concurrencia es necesario un evento compuesto de tipo bolsa.

Introduce el concepto de bloqueo en el tiempo, el cual se produce cuando un sistema en paralelo con otro se comporta como *STOP*.

También introduce un subcálculo asíncrono ASCCS. Un sistema asíncrono es modelado, como un sistema en el que cada nodo puede aceptar el paso del tiempo y permanecer en el mismo estado. Para que dos sistemas que tengan cadenas de eventos «*1*» no provoquen no-determinismo sobre la alternativa se introduce una acción prefijada de modo que el paso del tiempo no decide entre los dos subsistemas de una alternativa.

## Contributions Analysis with CCS

En su artículo [NY85], Nounou y Yemini proponen un CCS simplificado, que permitirá la evaluación de rendimientos de los protocolos de comunicación y cuyos principales cambios son:

- Permite una sincronización punto por punto.
- Los eventos internos son etiquetados con el nombre de la puerta que produce la sincronización. Esta distinción entre acciones internas es necesaria para poder aplicar diferentes comportamientos temporales para cada uno de ellos.

La especificación y evaluación de rendimientos son llevadas a cabo en dos pasos:

En primer lugar se realiza una especificación funcional con la que realizar un análisis funcional, obteniendo un modelo semántico simplificado y asíncrono.

A continuación, sobre el modelo funcional, se asume que el tiempo en el que cada acción puede ocurrir es una variable real aleatoria distribuida exponencialmente, de modo que se evita el problema de tener que actualizar el tiempo del sistema entero cuando una acción ocurra en una de los componentes paralelos del sistema.

Además es posible determinar la probabilidad de que ocurra una cierta restricción temporal o general así como evaluar el rendimiento del sistema.

## CCS con prioridades CCS-Pri

En su artículo [CH88] Cleaveland y Hennessy proponen un modelo que incluye la posibilidad de modelizar prioridades en casos donde existe no-determinismo.

Los autores llegan a la conclusión de que una decisión de prioridad no puede realizarse cuando el entorno es capaz de influenciarla, es decir, solo cuando el evento es interno puede basarse una decisión en información adicional que será introducida para expresar prioridad.

El modelo extiende CCS para expresar dos niveles de prioridad (normal y alto) de modo que cada acción ( $a$ ) tiene una versión de alta prioridad ( $\bar{a}$ ). La sincronización solamente puede producirse entre eventos de la misma prioridad, de modo que si es entre eventos de prioridad normal produce una acción interna de prioridad normal, si es entre eventos de alta prioridad se produce una acción interna de alta prioridad y entre uno de prioridad alta y otro de normal no es posible.

La semántica operacional consta de una primera fase en la que se obtiene un sistema de transiciones aplicando las reglas de derivación incluyendo la información de prioridad en las acciones que etiquetan



las transiciones, y de una segunda en la que se define un nuevo sistema de transiciones eliminando las transiciones imposibles por tener una acción interna de alta prioridad como alternativa. De esta forma los casos de no-determinismo donde haya acciones internas priorizadas estarán resueltos.

Se define una relación de bisimulación fuerte congruente respecto a los operadores CCS, así como a los otros operadores definidos por incremento o reducción de la prioridad de los eventos ofrecidos por un proceso.

## CCS Temporal TCCS

Tofts en su artículo [TOF88], define una extensión de CCS para modelar tiempo absoluto, en la que define un nuevo operador llamado acción prefijada temporizada  $\lambda[t]P$  que representa el comportamiento que lleva a cabo la acción  $\lambda$  y pasa al estado  $P$  después del paso del tiempo  $t$ .

La semántica del operador paralelo causa la pérdida de la concurrencia ya que el tiempo pasado en una parte del sistema no se toma en consideración en la otra parte.

Se introduce el concepto de orden temporal entre procesos TCCS introduciendo una relación de orden que determina si un proceso particular es más rápido que otro en vez de la relación de igualdad o equivalencia normalmente usada.

Posteriormente, Moller y Tofts en sus artículos [MT89] y [MT90] modificaban el modelo propuesto. Continuaban asumiendo que las acciones no consumen tiempo e incluyen dos operadores temporales:

$(t).P$ . Se comporta como  $P$  después de  $t$  unidades de tiempo.

$\delta.P$ . Puede esperar un tiempo hasta que el proceso  $P$  empiece.

Además introducen dos operaciones de elección cuyo comportamiento en el tiempo es diferente.

Elección fuerte ( $P+Q$ ). El tiempo pasa solamente si ambas partes permiten que pase. La alternativa debe ser resuelta por la ocurrencia de una acción.

Elección débil ( $P\oplus Q$ ). El tiempo pasa si al menos una de las partes permite que pase y además el tiempo puede causar que una alternativa que llegue a STOP este disponible.

Un proceso puede evolucionar en el tiempo si todos los otros procesos en paralelo también pueden hacerlo.

Ellos proponen una semántica operacional formada por dos partes ortogonales, comportamiento temporal donde el paso del tiempo se representa por una transición etiquetada con la cantidad de tiempo que ha pasado y el comportamiento funcional donde la ocurrencia de una

acción es modelada con una transición etiquetada con el nombre de la acción.

Se asume que el tiempo es discreto.

El lenguaje puede especificar restricciones generales de tiempo, y comportamientos síncronos forzando a las acciones a que ocurran en un instante específico de tiempo ( $(t)a.P$ ) y asíncronos expresados con  $\delta.P$ .

Se define una relación de equivalencia basada en una bisimulación fuerte, así como una teoría ecuacional completa que consiste en la semántica de la relación antes definida para procesos finitos.

## PCCS<sub>ξ</sub>

En [GSST90] Glabbeek, Smolka, Steffen y Tofts proponen tres modelos de procesos probabilísticos basados en SCCS, llamados respectivamente Reactivo, Generativo y Estratificado. Al igual que SCCS soportan tiempo discreto cuantitativo para modelar comportamientos síncronos y usan eventos compuestos de tipo bolsa para modelar la concurrencia real.

El lenguaje PCCS consta de las siguientes operaciones:

- iniciación (0);
- acción prefijada ( $\alpha.E$ ),
- alternativa probabilística  $\sum_{i \in I} [P_i] E_i$
- producto sincronizado  $E \times F$
- restricción
- renombrado
- procesos recursivos.

### Modelo reactivo

Se considera un subconjunto de PCCS en el que no existe renombramiento y donde la alternativa probabilística está en la forma  $\sum_{i \in I} [P_i] \alpha_i E_i$  como un sistema de transiciones etiquetadas, es decir las transiciones son etiquetadas por una acción más su probabilidad asociada.

Este modelo no permite conocer la probabilidad de elegir una acción u otra en caso donde ambas son posibles y además la información se pierde.

## *Modelo generativo*

Interpreta PCCS como un sistema de transiciones etiquetado y probabilístico donde para cada estado la suma de las probabilidades de todas las transiciones es 1 si no hay probabilidad de bloque.

Además las probabilidades relativas de cada acción son conocidas.

## *Modelo estratificado*

Interpreta PCCS como un sistema de transiciones etiquetado donde las etiquetas pueden ser de dos diferentes tipos; probabilidades o acciones.

El modelo estratificado conserva la información de las probabilidades estratificadas que se pierde en el modelo generativo.

Los autores definen una relación de bisimulación para todos los modelos y funciones para la conversión de un modelo a otro más abstracto (estratificado a generativo y generativo a reactivo).

Posteriormente en [SS90], extienden el modelo estratificado para expresar prioridades y para gestionar transiciones cuya probabilidad a priori es 0.

Esas transiciones pueden tener lugar solo si todas las otras alternativas, con probabilidades diferentes de 0 son preexcluidas por las restricciones de contexto. Además se pueden modelar procesos con baja prioridad.

## **WSCCS Weighted SCCS**

Tofts, en sus artículos [TOF90a] y [TOF90b] propone una modificación de SCCS para modelar comportamientos probabilísticos y prioridades, en la que introduce un operador alternativa  $n$ -ario donde cada una de las alternativas tiene un peso asignado en vez de una probabilidad.

El modelo semántico es un sistema de transiciones multicolorado, es decir, hay dos mecanismos para el cambio de estados, uno es la ocurrencia de eventos y la otra la elección de una alternativa con un cierto peso, que dividido por la suma de los pesos de todas las alternativas determina la probabilidad de esa opción.

Este modelo básicamente coincide con el PCCS estratificado, con la única diferencia de que usa pesos en vez de probabilidades. Este cambio permite la introducción de prioridades usando un peso infinito así como una semántica ligeramente menos complicada.

Para introducir prioridades se define un peso infinito especial que además tiene prioridad sobre otras alternativas.

Esta forma de incluir prioridades es dual (probabilidad 0 contra peso  $\infty$ ) para la extensión de PCCS.

## L. Chen Timed CCS

Chen, Anderson, Moller, en su artículo [CAM90] definen una extensión de CCS con tiempo cuantitativo. Para lo que extienden el operador acción prefijada para poder imponer restricciones de tiempo general y poder medir el instante de tiempo en el que ocurre una acción. El nuevo operador tendrá la forma  $a(t) \Big|_e^{e_1}.P$  donde  $t$  es una variable libre de  $P$ , representa un proceso el cual puede realizar la acción  $a$  en un tiempo entre  $e$  y  $e_1$  pero tan pronto como sea posible y en hacer esto tardará  $P[d/t]$  donde  $d$  es el tiempo actual en el que la acción  $a$  es realizada.

La ocurrencia de acciones no consume tiempo y el criterio de máximo progreso (ASAP) es aplicado. El dominio de tiempo usado es el de números reales positivos.

## W. Yi Timed CCS

W. Yi en su artículo [YI91] extiende el operador de CCS acción prefijada para poder medir el instante de tiempo en el cual ocurre una acción y define retrasos:

- $a@t.P$  donde « $t$ » es una variable libre de  $P$ , modela un proceso que está esperando la sincronización del entorno sobre  $a$  y entonces llega a ser  $P[d/t]$  donde  $d$  es el retraso antes de que se produzca la sincronización.
- $\epsilon(d).P$  modela un proceso que está *idle* durante  $d$  unidades de tiempo y se comporta como  $P$ .

La ocurrencia de acciones no consume nada de tiempo y el criterio de máximo progreso (ASAP) es aplicado. El dominio de tiempos son los números reales positivos.

El modelo propuesto es aplicado a TPL (lenguaje de procesos temporales) para ejecutar un teorema de expansión para álgebras de procesos.

## 4. TÉCNICAS BASADAS EN CSP

### CSP temporizado

Este termino fue utilizado para referirse a sucesivas propuestas realizadas por Reed y Roscoe para la introducción de tiempo en CSP. Para ello introducen un nuevo constructor **WAIT t**, que representa un

proceso que termina satisfactoriamente después de un período de tiempo  $t$ . El operador paralelo soporta múltiple rendezvous, y el operador incluye una lista de eventos que deben sincronizar.

Carece de eventos internos pero tiene un operador de elección no determinista.

Modela la concurrencia como el interleaving de todas las acciones que ocurren en el mismo instante de tiempo.

La acción prefijo siempre contiene un mínimo retraso, de modo que el especificador puede obtener solamente eventos simultáneos a través del operador paralelo.

Los autores introducen la noción de estabilidad como una generalización de la divergencia (dual) introducida en CSP.

Su propuesta soporta el concepto de ASAP, de modo que cuando un evento es ocultado o un proceso termina satisfactoriamente la transición espontánea resultante se produce en el instante en el que se produce el evento oculto o la terminación.

Los eventos ocurren asíncronamente pero sólo es posible retrasar su ocurrencia.

Se define una métrica completa sobre el dominio semántico de CSP temporizado ( $TM_{FS}$ ) la cual permite determinar la distancia entre dos expresiones tales que si  $d(\epsilon(B_1), \epsilon(B_2)) = 2^t$  significa que  $B_1$  y  $B_2$  representa el mismo comportamiento hasta el instante  $t$ .

S. Schnider en su tesis [SCH90] mejora CSP temporizado con nuevas operaciones definidas en términos de las ya existentes.

Estos nuevos operadores son:

- **Timeout.** Expresa el paso del control de un proceso a otro si el primero no realiza una acción externa en un período de tiempo dado.
- **Tireout.** Expresa que se puede ejecutar un proceso durante un tiempo dado después del cual el control pasa a un segundo proceso.
- **Interrupts.** Expresa el paso del control de un proceso a otro por la ocurrencia de un evento especial de interrupción ( $i$ ).

## CSP probabilístico CSP-Pro

J.J. Zic en su artículo [ZIC90] propone la posibilidad de extender CSP en términos de probabilidad y tiempo.

Propone un operador elección probabilístico n-ario  $\nabla_{\epsilon, i=1}^n (P_i, \pi_i)$  donde  $\epsilon$  es el entorno al que se refiere y  $\pi_i$  es la probabilidad de obtener el comportamiento  $P_i$ .

El modelo semántico propuesto (sin dar la función semántica) es un proceso estocástico dado por la tupla  $(A, F, \phi_{F, \epsilon}, D, \phi_{D, \epsilon})$  donde  $A$  es el alfabeto,  $F$  es el conjunto de fallos,  $\phi_{F, \epsilon}$  es una función que asigna probabilidades a los elementos de  $F$ ,  $D$  es el conjunto de divergencias y  $\phi_{D, \epsilon}$  es la función que asigna probabilidades a los elementos de  $D$ .

En las conclusiones del artículo el autor expone, como un factor que complica la obtención de un modelo semántico, el hecho de que uno debe estar familiarizado con el entorno de un proceso para poder determinar su espacio probabilístico.

Este artículo además hace referencias a la introducción de tiempo cuantitativo, pero sin avanzar en el desarrollo de esta idea.

## PCSP

Fernando Cuartero en su Tesis Doctoral [CUA] propone una extensión de CSP asíncrono.

En este modelo sustituye el no-determinismo por una cuantificación probabilística y proporciona la semántica de forma operacional, denotacional, algebraica y mediante pruebas.

Las prioridades se definen como probabilidades extremas en la elección externa.

## 5. RESUMEN

A continuación se presenta una tabla resumen comparativa de las características de los modelos descritos en este artículo basada en los siguientes criterios:

### Modelización del tiempo

#### *Tiempo cuantitativo y no cuantitativo*

Se diferencian las técnicas que permiten la modelización cuantitativa del tiempo de aquellas que solo permiten la ordenación de los eventos en el tiempo.

#### *Tiempo discreto y continuo*

En los modelos de tiempo discreto, el tiempo es tratado como un dominio contable, mientras que en modelos de tiempo continuo el tiempo es un dominio continuo.

## *Síncrono y/o asíncrono*

En los sistemas síncronos los eventos ocurren solo en unos instantes específicos de tiempo indicados generalmente por un reloj global, mientras que en los asíncronos los eventos pueden ocurrir en cualquier instante.

## *Con o sin evento de reloj explícito*

En las técnicas en las que se incluye el uso del reloj hay un evento especial que representa un tic de reloj y el cual indica el paso de un intervalo de tiempo.

## *Basado exclusivamente en retrasos o no*

En los sistemas basados en retrasos la única restricción temporal que se puede imponer es un retraso antes de que un evento pueda ser ofrecido. Una vez pasado ese tiempo, el evento puede ocurrir en cualquier instante.

Existen otros modelos en los que las restricciones temporales pueden ser más generales; por ejemplo un evento puede ser ofrecido durante un período de tiempo, un conjunto de instantes o un único instante.

## *Soporte o no al concepto de «Tan pronto como sea posible (ASAP)»*

Los eventos que no dependen del entorno ocurren tan pronto como sea posible si el modelo soporta ASAP.

El criterio ASAP significa que si el sistema puede evolucionar en un momento dado este lo hace en ese momento. Además, puede ser especificado una espera hasta que todo los componentes del sistema envueltos en una sincronización estén listos para sincronizar.

Este criterio no debe ser confundido con la posibilidad de forzar a una acción para que ocurra en un determinado instante de tiempo.

## *Ocurrencia instantánea de eventos o con consumo de tiempo*

## *Soporte o no de eventos compuestos*

La combinación de varios eventos en una sola acción puede ser de dos tipos:

- Tipo conjunto. Los eventos no pueden repetirse dentro de una acción
- Tipo bolsa. Los eventos pueden repetirse dentro de una acción

## **Caracterización de probabilidades y prioridades**

Existen pocos modelos que permiten la caracterización probabilística del comportamiento de un sistema, mientras que la mayoría lo hacen de forma determinista o no-determinista, pero sin caracterización probabilística. Estrechamente relacionado con el concepto de caracterización probabilística está la posibilidad de especificar prioridades donde existe no-determinismo dentro de la ocurrencia de varias acciones.

## **Concurrencia**

El lenguaje puede modelar la concurrencia como el interleaving de todos los eventos que pueden ocurrir en un mismo instante (I), o pueden modelar concurrencia real permitiendo que varios eventos ocurran juntos (R).

## **Modelo semántico**

El modelo semántico puede ser denotacional (D) u operacional (O).

## **Sincronización**

Todos los modelos tienen en común la característica de que la sincronización entre los procesos es síncrona, es decir, solo ocurrirá si todas las partes envueltas en ella permiten que esta ocurra (rendezvous). Este tipo de sincronización contrasta con la de tipo asíncrono de los lenguajes basados en máquinas de estados donde los componentes se comunican mediante colas.

## **6. OTRAS TÉCNICAS DE DESCRIPCIÓN FORMALES**

Las Redes de Petri fueron diseñadas hace mucho tiempo y existen una gran variedad de extensiones para cubrir los requerimientos de tiempo. Algunas de estas técnicas ya han sido clasificadas y analizadas por H. Rudin en [RUD85].



	1	2	3	4	5	6	7	8	9	10	11	12	13
LOTOS		?	?			X		?		I		O	
CCS		?	?			X		?		I		O	
CSP		?	?			X		?		I		O	
QUEMADA-FERNÁNDEZ	X	X	*		X	X		X		I		O	
BOCHMAN-VAUCHER	X	X	*			X	X	X	X	I		O	
LOTICAL	X	X		X	X	X		?		I	S	O	
LOTOS TEMPORAL	X	X	*			X	X	X		I		O	
TIC	X	X	*		X	X		X		I		O	
PADS	X	X		X	X	X		X		R	B	O	
CELOTOS	X	X		X		X		X		I		O	
C. CALVELI	X		X			X	X			I*		D	
TI-LOTOS	X	X	*		X	X		X		I		O	
SCCS	X	X			X	X				R	B	O	
ANALYSIS CCS	X		X			X			X	I		O	
CCS-PRIOR		?	?			X		?		I		O	X
TCCS	X	X	*		X	X		X		I		O	
PCCS	X	X			X				X	R	B	O	X
WSCCS	X	X			X				X	R	B	O	X
L CHEN	X	*	X		X	X	X	X		I		O	
W. YI	X	*	X			X	X	X		I		O	
CSP TEMPORA	X	*	X			X	X			I*		D	
CSP PROB									X	I*		D	
PCSP	X					X		X	X	I		O/D	X

1. Tiempo cuantitativo.
2. Tiempo Discreto.
3. Tiempo Continuo.
4. Reloj.
5. Sistema Síncrono.
6. Sistema Asíncrono.
7. Criterio ASAP.

8. Eventos Instantáneos.
9. Caracterización Probabilística.
10. Concurrencia.
11. Eventos Compuestos.
12. Semántica.
13. Prioridades.

El modelo FSM expresa restricciones de tiempo añadiendo parámetros de tiempo a las transiciones y/o los estados.

## 7. CONCLUSIONES

Con la extensa variedad de técnicas, la conclusión general es que la especificación formal de los requerimientos de tiempo y probabilidad

son un campo activo de investigación que todavía no ha sido completamente desarrollado.

Los modelos descritos pueden clasificarse en dos grupos bien definidos:

- Los modelos que consideran que un evento debe ocurrir tan pronto como sea posible. Este tipo de modelos son muy homogéneos, lo que permite generar modelos intuitivos. Están basados en la idea de que cuando todas las partes envueltas en una sincronización están listas para sincronizar, en ese momento tiene lugar la sincronización.
- Los modelos que no aplican el criterio anterior. Estos ordenan en el tiempo los eventos de procesos concurrentes, pero consideran que un evento puede ocurrir en cualquier momento en que este disponible.  
Estos modelos son más heterogéneos y a veces difíciles de entender intuitivamente. Sin embargo algunos de ellos tienen la ventaja de poder generar especificaciones más abstractas debido a la ocurrencia no determinista de eventos en el tiempo

Ninguno de ellos puede especificar el concepto de ASAP.

## 8. BIBLIOGRAFÍA

- [AZC90] AZCORRA, A. 1990: «*Formal Modeling of Synchronous Systems. PhD Thesis*», ETSI Telecomunicación, UPM, Madrid.
- [BV88] BOCHMANN, G. V.; VAUCHER, J. 1988: «*Adding Performance Aspects to Specification Languages*». In Proc. Workshop on protocol Spec. Test. and Ver. VIII.
- [BLT90] BOLOGNESI, T.; LUCIDI, F.; TRIGILA, S. 1990: «*From timed petri nets to timed LOTOS*». 10<sup>o</sup> Internacional IFIP Symposium on Protocol Specification, Testin and Verification, pages 377-406.
- [BRI88] BRINKSMA, E. 1988: «*On The Design of Extended LOTOS PhD. Thesis*». Universidad de Twente.
- [CAL89] CALVELI, C. 1989: «*A timed model for basic LOTOS*» Technical Report. Hewlett-Packard Laboratories, Bristol, UK.
- [CAM90] CHEN, L.; ANDERSON, S.; MOLLER, F. 1990: «*A timed calculus of communicating systems*» Technical Report ECS-LFCS-89-104». Laboratory for Foundations of Computer Science LFCS, University of Edinburg.
- [CH88] CLEAVELAND, R.; HENNESSY, M. 1988: «*Priorities in Process Algebras*». In Proceedings Symposium on Logic in Computer Science LICS 88.
- [CUA] CUARTERO, F.: «*PCSP Un modelo probabilístico de procesos concurrentes*» Tesis Doctoral. Universidad Autónoma de Madrid.
- [GSST90] GLABBEEK, R.; SMOLKA, S. A.; STEFFEN, B.; TOFTS, C. 1990: «*Reactive, Generative, and Stratified Models of Probabilistic Processes*». In Proceedings 5th Annual Symposium on Logic in Computer Science, Philadelphia, USA, LICS 90.

- [MF76] MERLIN, P.; FARDER, D. J. 1976: «*Recoverability of Communication. Protocols-Implications of a Theoretical Study*». IEEE Trans. Communications,(24) pág: 1036-1043.
- [MIL83] MILNER, R. 1983: «*Calculi for Synchrony and Asynchrony*». Theoretical Computer Science,( 25) pág: 267-310
- [MT89] MOLLER, F.; TOFTS, C. 1989: «*A temporal calculus of communicating systems*». Technical Report ECS-LFCS-89-104».Laboratory for Foundations of Computer Science LFCS, University of Edinburgh.
- [MT90] MOLLER, F.; TOFTS, C. 1990: «*A temporal calculus of communicating systems*». Lecture Notes in Computer Sciences, pág. 401-415.
- [NY85] NOU NOU, N.; YEMINI, Y. 1985: «*Algebraic specification-based performance analysis of communication protocols*». In Workshop on Protocol Specification, Testing and Verification: IV. IFIP.
- [QAF89] QUEMADA, J.; AZCORRA, A.; FRUTOS, D. 1989: «*A Timed calculus for LOTOS*». Formal Description Techniques, Vancouver, CANADÁ, FORTE 89.
- [QAF90] QUEMADA, J.; AZCORRA, A.; FRUTOS, D. 1990: «*TIC: A Timed Calculus*». Technical report. Departamento de Ingeniería Telemática, ETSI. Telecomunicación, UPM, Madrid.
- [QF87] QUEMADA, J.; FERNÁNDEZ, A. 1987: «*Introduction of quantitative relative time into LOTOS*». In Workshop on Protocol Specification, Testing and Verification: VII, Zurich.
- [RUD85] RUDIN, H.: «*Time in formal protocol specification*» 1985. In GI/NTG conference on Communication in Distributed Systems, Karlsruhe.
- [SCH90] SCHNEIDER, S. 1990: «*Correctness and Communication in Real-Time Systems PhD thesis*», Oxford University Computing Laboratory. Programming Research Group.
- [SR88] SMITH, C.; RUDKIN, S. 1988: «*Time Guards and ASN.1 in LOTOS*». Technical Report Ref.40, Formal Methods Group, British Telecom.
- [SS90] SMOLKA, S. A.; STEFFEN, B. 1990: «*Priority as Extremal Probability*».Lecture Notes in Computer Science. Pág. 456-466.
- [TOF88] TOFTS, C. 1988: «*Temporal Ordering for Concurrency*» Technical Reports». University of Edinburgh.
- [TOF90a] TOFTS, C. 1990: «*A synchronous Calculus of Relative Frequency*». Lecture Notes in Computer Science. pág. 467-480.
- [TOF90b] TOFTS, C. 1990: «*Relative frequency in a synchronous calculus*» Technical Report ECS-LFCS-90-108»: Laboratory for Foundations of Computer Science, University of Edinburgh, the Kings Buildings.
- [VHT89] VAN HULZEN, W. H. P.; TILANUS, P. A. J.; ZUIDWEG, H. 1989: «*LOTOS Extended with Clocks*». In Formal Description Techniques, Vancouver, CANADÁ.
- [WAL83] WALTER, B. 1983: «*Timed petri nets for modeling and analysing protocols with real time characteristics*». Protocol Specification, Testing and Verification III, pages 149-159.
- [YI91] YI, W. 1991: «*CCS+Time= an Interleaving Model for Real Time Systems*» In [ICALP91].
- [ZIC90] ZIC, J. J. 1990: «*Some thoughts on communication system performance evaluation*» In Proceedings of the Open Distributed Processing Workshop, Sydney.