University of Castilla-La Mancha

A publication of the

Department of Computer Science

Modelos Temporizados y probabilísticos. Situación actual.

by

Juan José Pardo Mateo, Fernando Cuartero Gómez.

Technical Report **#DIAB-99-01-04** September 1999

DEPARTAMENTO DE INFORMÁTICA
ESCUELA POLITÉCNICA SUPERIOR
UNIVERSIDAD DE CASTILLA-LA MANCHA
Campus Universitario s/n
Albacete - 02071 - Spain
Phone +34.967.599200, Fax +34.967.599224

Índice General

1 Introducción **1**

1.1 Propósito del presente trabajo 1

1.2 Otros Trabajos 1

2 Modelos Temporizados **3**

2.1 Técnicas Basadas en LOTOS 3

2.1.1 Quemada-Fernandez 3

2.1.2 LOTCAL 4

2.1.3 Lotos Temporal 4

2.1.4 PADS 5

2.1.5 CELOTOS. Lotos de Reloj Extendido 5

2.1.6 C. Calvelli LOTOS 6

2.1.7 Timed-Accion LOTOS, Timed-Interaccion LOTOS . . 6

2.2 Técnicas basadas en CCS 7

2.2.1 SCCS 7

2.2.2 Contributions Analysis with CCS 7

2.2.3 CCS con prioridades CCS-Pri 8

2.2.4 CCS Temporal TCCS 8

2.2.5 L. Chen Timed CCS 9

2.2.6 W. Yi Timed CCS 10

2.3 Técnicas basadas en CSP 10

2.3.1 CSP temporizado 10

2.3.2 Timed Observations Model. Timed TCSP 11

2.3.3 Yolanda Ortega 11

2.4 Otras Técnicas 12

2.4.1 ATP 12

2.4.2 $ACP^t_{\tau\epsilon}$ 12

2.4.3 ACP_ρ 13

2.4.4 TPL Temporal process Algebra 13

3 Modelos Probabilísticos **15**

3.1 Técnicas basadas en CCS 15

3.1.1 PCCS 15

3.1.2 $PCCS_\zeta$ 15

3.1.3 WSCCS Weighted SCCS 17

3.2 Técnicas basadas en CSP 17

3.2.1 CSP probabilistico CSP-Pro 17

3.2.2 PCSP 18

4 Modelos Temporizados y Probabilísticos **19**
4.1 Técnicas basadas en LOTOS 19
4.1.1 Bochmann-Vaucher 19
4.1.2 TPAL 19
4.2 Técnicas basadas en CCS 20
4.2.1 TPCCS 20

5 Resumen y conclusiones **21**
5.1 Resumen 21
5.2 Conclusiones 24

Capítulo 1

Introducción

1.1 Propósito del presente trabajo

Durante mucho tiempo cuando se diseñaba y creaba un lenguaje de especificación, se hacia de modo que permitían describir el comportamiento como una secuencia de eventos ordenados en el tiempo.

Debido a la estrecha relación con el tiempo de los sistemas especificados, pronto se vió la necesidad de poder expresar requerimientos y restricciones temporales, así como la caracterización probabilística de los mismos. Esto llevo al planteamiento de la extensión de los lenguajes básicos(CCS, CSP, LOTOS) para que incorporasen estos requisitos.

En los años 80 los trabajos se centraron principalmente en la extensión temporal, para después en los 90 centrarse en las extensiones probabilísticas. En estos momentos los trabajos se centran fundamentalmente en las extensiones que combinen tanto aspectos temporales como probabilísticos.

Debido a la gran diversidad de modelos que estas extensiones han producido, hemos creido conveniente dar, con el presente trabajo, una visión general sobre los modelos temporizados y probabilísticos utilizados para la descripción de sistemas concurrentes, centrandonos especialmente en aquellos lenguajes basados en álgebras de procesos.

Dentro del trabajo los modelos han sido clasificados en tres grandes grupos según el caracter de la extensión y dentro de estos grupos subclasificados según el lenguaje base de ellos. Así en el capítulo 2 se realizará la descripción de los modelos temporizados. Posteriormente en el capítulo 3 se realizará la descripción de los probabilísticos, para terminar en el capítulo 4 describiendo los modelos que combinan la extensíon temporal y probabilística. Para finalizar el trabajo en el último capítulo presentaremos un breve resumen y conclusiones sobre el trabajo.

1.2 Otros Trabajos

Con anterioridad a este trabajo se han realizado varias revisiones sobre los modelos existentes, siendo tres de los trabajos más importantes los siguientes:

El presentado en [FP90] donde se resumen las principales características de los lenguajes de especificación apropiados para la descripcion formal de sistemas de tiempo real clasificandolos en tres grupos como son los lenguajes diseñados especificamente para la especificación de tiempo real, los lenguajes que son una extensión de tiempo de los lenguajes ya existentes y lenguajes que han mostrado su idoneidad para especificar sistemas en tiempo real.

El trabajo presentado por Rudin, en [Rud85, Rud86], en el que analiza la inclusión del tiempo en técnicas basadas principalmente en redes de Petri y Máquinas de Estados Finitos.

El último ha sido realizado por el D.I.T. de la U.P.M.[MF91],en el cual presentan una revision de los modelos basados en álgebras de procesos. Este ultimo trabajo ha servido de base y punto de partida para el trabajo que en estos momentos esta leyendo.

Capítulo 2

Modelos Temporizados

2.1 Técnicas Basadas en LOTOS

2.1.1 Quemada-Fernandez

Quemada y Fernandez en [QF87] realizan una extensión de LOTOS con tiempo cuantitativo y discreto, donde además consideran la ocurrencia de eventos instantáneas.

Esta extensión consiste en asociar a cada evento un nuevo atributo como es un instante o intervalo de tiempo relativo al principio de su creación, el cual indicará cuando puede tener lugar. Si el atributo es un instante estaremos representando comportamientos síncronos, mientras que si es un intervalo estaremos representando comportamientos asíncronos. Este nuevo atributo impone dos restricciones sobre la ejecución de las acciones:

- La sincronización solo puede tener lugar entre acciones con el mismo atributo temporal, es decir, que pueden ocurrir al mismo tiempo.

- El tiempo debe avanzar al mismo ritmo para todos los procesos del sistema, de modo que la evolución temporal en un proceso implica la actualización temporal del resto de los procesos en paralelo. Este hecho puede causar que algunas acciones no estén disponibles,en un momento dado. Igualmente esta restricción puede llevar al sistema a representar comportamientos anómalos, como por ejemplo la interpretación de la expresión
$$(i\{3\}; ...)|||(i\{5\}; ...)$$
permitiría la ocurrencia de $i\{5\}$ bloqueando así la ocurrencia de $i\{3\}$.

Para resolver este problema posteriormente, los autores en su artículo [QAF89], presentan el modelo TIC en el cual realizan una interpretación del operador paralelo con el criterio que llaman "*must timing*",consistente en que el operador paralelo realiza un intercalado de los eventos en el orden temporal correcto. Por ejemplo en la interpretación de la expresión

$$(a\{3\}; \ldots)|||(b\{5\}; \ldots)$$

no se permitirá la ocurrencia de "$b\{5\}$"(ocurrencia de b en el instante 5) porque "a" debe ocurrir antes(en el instante 3).

Posteriormente, en su artículo [QAF90], realizan otra extensión del modelo consistente en la introducción de un evento especial ϵ, que representa el paso del tiempo y permite la modelización a un nivel semántico del cambio de estado del comportamiento de un sistema por el paso del tiempo. Esta nueva extensión permite definir una equivalencia observacional basada en bisimulación débil que es congruente respecto a todos los operadores excepto el operador elección.

2.1.2 LOTCAL

Brinksma, en su tesis doctoral [Bri88], realiza una extensión de LOTOS tanto a nivel sintáctico como semántico, orientada principalmente a incrementar la aplicabilidad de LOTOS y a mejorar el estilo de especificación.

Las nuevas características introducidas son :

- Define acciones compuestas formadas por un conjunto finito de eventos. Esto permitirá la modelización de la concurrencia real, aunque Brinksma no lo usa con ese proposito sino con el de modelar la ocurrencia atómica de un conjunto de eventos.
- Asocia la terminación exitosa de un proceso con la ocurrencia de un evento observable. Esto es consecuencia de la definición de acciones compuestas.
- Soporta la descripción de módulos LOTOS que facilitan la especificación jerárquica de sistemas grandes y complejos. De este modo una especificación completa consta de un conjunto de modulos que pueden ser manejados independientemente.

Además de introducir estas características sugiere nuevas ideas para extensiones posteriores como:

- Introducción de tiempo cuantitativo. Para ello usa **timers**, que pueden ser inicializados a un valor de modo que expiran una vez transcurrido ese tiempo pero que pueden ser cancelados antes de su expiración. Cuando se produce la expiración de uno de estos timers el sistema se bloqueará hasta que el proceso asociado lo atienda, por lo que se obligará al proceso especificador de un timer a que lo cancele si no puede atender a su expiación. Todos los timers especificados sincronizan en una puerta especial (evento de reloj explicito "w"), que representa el paso de una unidad de tiempo discreto, y que formará parte de las acciones.

  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 permitirá modelar comportamientos síncronos basados en el evento de reloj w.
- sincronización en puertos indexados con datos.
- Terminación negativa en la composición paralela como la generalización del operador de deshabilitación como es propuesto en PADS.

2.1.3 Lotos Temporal

En el artículo [SR88], Smith y Rudkin realizan una extensión de LOTOS introduciendo tiempo cuantitativo y sustituyendo el lenguaje de especificación de TAD [1] ACT-ONE por ASN-1.[ISO].

[1] Tipos Abastactos de Datos

Introducen un nuevo operador que llaman **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 temporal ASAP de modo que los eventos ocurrirán tan pronto como sea posible y una acción solamente se producirá si no hay posibilidad de que ocurra una acción interna de antemano. Aunque el especificador maneja tiempo real discreto, el modelo semántico acumula el tiempo pasado desde un estado inicial, de modo que todas las transiciones (cambios de estado) llevan asociado un atributo temporal que indica el tiempo total pasado.

El lenguaje es conceptualmente correcto, pero 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.

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.

2.1.4 PADS

Azcorra, en su tesis doctoral [Azc90], introduce las siguientes mejoras sobre LOTOS para adaptarlo al diseño de sistemas distribuidos tanto síncronos como asíncronos:

- Define eventos compuestos con una estructura de bolsa, como en SCCS[Mil83], que permitirá modelar la concurrencia real.

- Modifica y generaliza el operador composición secuencial, lo que permite asociar la terminación de un proceso con la ocurrencia de un evento observable. Esta generalización permite un manejo más flexible de excepciones y la secuenciación.

- Para la modelización de tiempo real, define un modelo básico con un evento especial "t" que es interpretado como el paso de una unidad de tiempo discreto, mientras que el resto de acciones no consumen tiempo. Sobre este modelo básico define un lenguaje de eventos con atributos temporales donde el evento especial "t" aparece solamente en el modelo semántico.

- 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", por lo que para evitar este problema 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.

2.1.5 CELOTOS. Lotos de Reloj Extendido

Hulzen, Tilanus y Zuidweg en [vHTZ89] realizan una extensión de LOTOS que permitirá modelizar tiempo discreto cuantitativo. Para ello introducen en el lenguaje tanto a nivel semántico como sintáctico, relojes que pueden inicializarse o consultarse como valores de una clase particular. Esta introducción de los relojes dentro del lenguaje incrementa considerablemente la complejidad del modelo semántico, pero por contra tiene la ventaja de facilitar la descripción de las restricciones temporales de los eventos no consecutivos en el comportamiento.

El tiempo es indicado por un reloj global al sistema que genera ticks (a nivel semántico) y el cual nunca podrá ser bloqueado. Cada uno de los ticks de este reloj hará que se incrementen los valores de los relojes. No existe ningún mecanismo que permita que una acción se realice antes de un instante especificado debido a que el tiempo pasa siempre.

El sistema puede ser considerado como asíncrono pues permite que en un instante dado no se ejecute ninguna acción.

2.1.6 C. Calvelli LOTOS

La extesión que realiza en [Cal89] consiste en la introducción de un nuevo proceso *wait(t)* cuyo significado es similar al del operador del mismo nombre definido en CSP-temporizado,es decir, representa un proceso que termina de forma exitosa después de un tiempo "t". El modelo denotacional definido es basicamente igual al de CSP, considerando los criterios:

- El tiempo es continuo
- Todas las acciones, incluidas las llamadas recursivas de un proceso, requieren un tiempo τ para ser ejecutadas.
- El criterio ASAP no es aplicado, al menos inicialmente.

Los eventos ocurren asíncronamente, por lo que sólo es posible retrasar su ocurrencia.

Define una relación de equivalencia cuando aplica el criterio ASAP, que por el hecho de eliminar las acciones internas es incongruente, y cuya consecuencia final es que el criterio ASAP no es aplicado cuando define la función semántica. Tampoco puede aplicarse el teorema de expansión para expandir comportamientos con componentes paralelas, debido al hecho de que la acciones consumen tiempo.

2.1.7 Timed-Accion LOTOS, Timed-Interaccion LOTOS

Bolognesi, Lucidi y Trigila en su artículo [BLT90], proponen dos extensiones a LOTOS inspirados en la expresividad de los dos modelos de Redes de Petri temporizados que aparecen en [Wal83, MF76].

Timed-Action LOTOS

Es el primer modelo que proponen y es similar al propuesto por Quemada y Fernandez.

Termed Time-Interaction LOTOS

En este nuevo modelo introduce dos nuevos conceptos como son:

Urgencia .- Implica que el evento envuelto debe ocurrir con absoluta prioridad. Aunque los autores lo denominan "*as soon as possible*" realmente no es así porque se suprimen los eventos internos que es donde el criterio ASAP podría ser aplicado.

Resticciones temporales no locales .- Las restricciones temporales son introducidas a nivel de proceso y no a nivel de acción.

Además de en el operador paralelo, el criterio *must timing* es aplicado en el operador elección de modo que una acción solo es posible en un tiempo t si existen otras acciones que se ejecutarán posteriormente o no existen más acciones posteriores.

Considera que todas las acciones pueden ocurrir en un instante de tiempo, aunque permite la imposición de restricciones temporales para la ocurrencia de acciones en los procesos, expresandolas de la siguiente forma "$timer\ a < x, y > in\ P$" , lo que indica que la acción "a" ocurrirá urgentemente entre los instantes "x" e "y" contados desde el inicio del ofrecimiento de las acción "a" en el proceso "P". Si no ocurre la acción el sistema se bloqueará.

2.2 Técnicas basadas en CCS

2.2.1 SCCS

En [Mil83] se presenta una generalización de CCS que permite modelar sistemas síncronos. Para ello define los operadores prefijo,alternativa (elección), restricción, renombrado y operador paralelo que soporta solamente rendezvous binario. Dentro del modelo no se definen eventos internos, pero define una acción llamada "idle" 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 modelado del tiempo está basado en el concepto de retraso en el ofrecimiento de una acción, cuya duración consideramos que es de una unidad de tiempo discreto. Esto hace que el modelado de la concurrencia, que será real y no de interleaving, se realice mediante un evento compuesto de tipo bolsa. La introducción de estos eventos compuestos introduce otro nuevo concepto como es el de bloqueo en el tiempo, el cual se producirá cuando un sistema en paralelo con otro se comporta como $STOP$.

Además de esta generalización introduce un subcalculo asíncrono AS-CCS, donde un sistema asíncrono es modelado como un sistena en el que cada nodo puede aceptar el paso del tiempo y permanecer en el mismo estado. En este caso para evitar que dos sistemas con cadenas de eventos "idle" 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.

2.2.2 Contributions Analysis with CCS

Nounou y Yemini, en [NY85], proponen un lenguaje (simplificación de CCS) que permitirá la evaluación de rendimientos de los protocolos de comunicación. Las variaciones con respecto a CCS son:

- Permite una sincronización punto por punto. Por ejemplo el comportamiento de "$a.A$" y "$\bar{a}.B$" produce un evento interno(τ), y después se comporta como la composición paralela de "A" y "B".

$$(a.A \times \bar{a}.B = \tau(A \times B))$$

.

- Los eventos internos son etiquetados con el nombre de la puerta que produce la sincronización. Esta distinción entre acciones internas que se realiza se vuelve necesaria para poder aplicar “a posteriori” diferentes comportamientos temporales para cada uno de ellos.

- El modelo propuesto es limitado a variables aleatorias distribuidas exponencialmente, lo que evita el problema de la actualización temporal del sistema completo cuando ocurra una acción en una de las dos componentes paralelas.

La especificación y evaluación de rendimientos tendrá una primera fase en la que se realizará una especificación funcional(sin atributos temporales) a traves de la cual se realizará un análisis funcional y tras la cual obtendremos un modelo semántico, simplificación del de CCS y asíncrono. Posteriormente, y sobre este primer modelo, se asume que el tiempo en el que cada acción puede ocurrir es una variable real aleatoria distribuida exponencialmente.

2.2.3 CCS con prioridades CCS-Pri

Cleaveland y Hennessy, en [CH88], proponen una extensión de CCS que permite modelizar prioridades en casos donde existe no-determinismo. En este artículo, ellos llegan a la conclusión de que una decisión de prioridad no puede realizarse cuando el entorno es capaz de influenciarla, es decir, solamente cuando el evento es interno, la decisión podrá basarse en información adicional introducida para expresar prioridad.

La extensión realizada permite expresar dos niveles de prioridad de modo que cada acción (a) tiene una versión de alta prioridad denotada como $\bar{a}$. Esta dualidad de acciones influye en la sincronización de modo que la sincronización entre eventos de la misma prioridad produce una acción interna con esa prioridad mientras que si los eventos poseen diferente prioridad no será posible la sincronización.

A la hora de definir el sistema de transiciones inducido por la semántica operacional se realiza una primera fase en la que se obtiene el sistema de transiciones aplicando las reglas de derivación como en CCS, indicando información de prioridades para después en una segunda fase eliminar las transiciones imposibles debido a las prioridades.

Además definen una relación de bisimulación fuerte congruente respecto a los operadores CCS, así como con los otros operadores definidos por incremento o reducción de la prioridad de los eventos ofrecidos por un proceso (priorización y despriorización).

2.2.4 CCS Temporal TCCS

Tofts en su artículo [Tof88], realiza una extensión de CCS que permitirá modelar tiempo absoluto, consistente en la introducción de un nuevo operador llamado “acción prefijada temporizada” cuya sintaxis es $\lambda[t].P$ y cuyo significado es un comportamiento que realiza la acción λ y después de t unidades de tiempo pasa a comportarse como P.

La definición que realiza del operador paralelo causa una perdida de la concurrencia ya que el paso del tiempo en una de las partes no es tomada en consideración en la otra.

Introducen también el concepto de orden temporal entre procesos de modo que definen una relación de orden entre procesos que determina si un proceso es más rápido que otro en lugar de la relación de igualdad o equivalencia usada normalmente.

Debido a las críticas que recibe este modelo por la escasa aplicabilidad de la nueva operación prefijo y otras consideraciones, este modelo es modificado por el autor junto con Moller en su artículo [MT90] introduciendo cuatro nuevos operadores como son :

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

$\delta.P$.- Puede esperar un tiempo hasta que el proceso "P" empieza (no ASAP).

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

$P \oplus Q$ **(elección debil)** .- El tiempo pasa si al menos una de las partes permite que pase y además el tempo puede causar que una alternativa que llegue a stop este disponible.

Otras características introducidas en el nuevo modelo son:

- Las acciones no consumen tiempo.
- La semántica operacional viene divida en dos partes ortogonales, donde una representa el comportamiento temporal mediante transiciones etiquetadas por la cantidad de tiempo pasado y una parte funcional donde la ocurrencia de una acción es representada por una transición etiquetada como el nombre de la acción.
- Asume tiempo discreto.
- Permite expresar comportamientos síncronos forzando a que las acciones ocurran en un instante específico de tiempo "t(a)P" y asíncronos expresados como "$\delta.P$".
- Define una relación de equivalencia basada en bisimulación fuerte además de una teoría ecuacional completa consistente en la semántica de la relación antes definida para procesos finitos.

2.2.5 L. Chen Timed CCS

Chen, Anderson y Moller, en su artículo [CAM90] realizan una extensión de CCS con tiempo cuantitativo modificando la acción prefijada de modo que se puedan imponer restricciones de tiempo general, así como que se puedan medir el instante de tiempo en que ocurre una acción.

La sintaxis de esta nueva acción prefijada es $a(t)|_{e_1}^{e_2}.P$, donde t es una variable libre de P, y representa un comportamiento que puede realizar la acción a en un tiempo entre e_1 y e_2 pero tan pronto como sea posible y que posteriormente se comportará como $P[d/t]$ donde d es el momento en que se produce la acción a .

La ocurrencia de acciones no consume tiempo, el dominio de tiempos es el de los números reales positivos y es aplicado el criterio ASAP.

2.2.6 W. Yi Timed CCS

W. Yi, en su artículo [Yi91], define una extensión de CCS con tiempo cuantitativo, en el cual extiende el operador de acción prefijada de modo que permita medir el instante de tiempo en el que ocurre una acción . Esta nueva operación se representa como $a@t.P$ donde "t" es una variable libre de P, y modela un proceso que está esperando la sincronización del entorno sobre a para posteriormente llegar a ser $P[d/t]$ donde d es el retraso antes de que se produzca la sincronización.

También permite definir retrasos con la operación $\varepsilon(d).P$ que modela un proceso que está "idle" durante "d" unidades de tiempo y después se comporta como P.

Al igual que el modelo anterior de L.Chen, las ocurrencias de acciones no consumen tiempo, el dominio utilizado es el de los reales positivos y el criterio ASAP es aplicado.

2.3 Técnicas basadas en CSP

2.3.1 CSP temporizado

Con este nombre se hace referencia a varias propuestas sucesivas realizadas por Reed y Roscoe [RR86, RR87, RR88]. Con el fin de introducir el tiempo en las especificaciones CSP, introducen un nuevo constructor denotado por **WAIT t**, que representa un proceso cuya terminación satisfactoria se producirá después de transcurrir un periodo de tiempo t.

Las principales características de este modelo son:

- El operador paralelo soporta multiple rendezvous, e incluye una lista de eventos que deben sincronizar.
- Carece de eventos internos.
- Existe un operador de elección no determinista.
- La concurrencia es modelada como el interleaving de todas las acciones que ocurren en el mismo instante de tiempo.
- La acción prefijo siempre tiene un mínimo retraso.
- Se introduce el concepto de estabilidad como una generalizacíon de la divergencia de CSP.
- Soporta el concepto de ASAP, de modo que en el instante en que un evento es ocultado o un proceso termina satisfactoriamente se produce la *transición expontanea* resultante.
- Los eventos ocurren asíncronamente pudiendo solamente retrasar su ocurrencia.

Los autores definen también una métrica completa sobre el dominio semántico de CSP temporizado (TM_{FS})la cual permite determinar la distancia entre dos expresiones de modo 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.

Posteriormente Schnider en su tesis doctoral [Sch90] mejora CSP temporizado introduciendo nuevas operaciones que son definidas en términos de las ya existentes.Estas nuevas operaciones son:

Timeout .- Expresa el paso del control de un proceso a otro si el primero no realiza una acción externa en un periodo 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).

2.3.2 Timed Observations Model. Timed TCSP

Yolanda Ortega y David de Frutos, en [OMFE90], presentan un modelo semántico para representar sistemas concurrentes en tiempo real, basado en un conjunto de observaciones temporales, las cuales son definidas como una tupla (*tf,rt,div*) que representa que después de observar la traza tf durante un periodo finito de tiempo, la máquina es capaz de rechazar la *bolsa de acciones* de rechazos rt y además tiene la capacidad para diverger dependiendo del valor de div. Además este modelo asume las siguientes ideas:

- El tiempo es discreto.
- Las acciones no son instantáneas, sino que sus ocurrencias tienen una duración $d : a \rightarrow N^+$
- La concurrencia real es modelada mediante acciones compuestas de tipo bolsa con un número finito de eventos.
- La ejecución de una acción puede empezar en cualquier momento.

Este modelo es usado como modelo denotacional de una versión restringida de TCSP [Old87], la cual extiende TCSP con Timeouts y añade la construcción $a[t] \rightarrow$ que significa que la ejecución de la acción "a" puede ser retrasada hasta t unidades de tiempo.

Al final del artículo se propone un nuevo modelo en el que el criterio ASAP es aplicado, y que hace que sea necesario mantener los rechazos en todas las trazas observables, y no solo en el final de cada traza.

2.3.3 Yolanda Ortega

Yolanda Ortega, en su tesis [OM90], realiza una extensión de CSP para contemplar la caracterización temporal de los sistemas concurrentes. Las principales características de este modelo son:

- El tiempo viene marcado por un reloj global que coordina todos los procesos por lo que nos permitirá modelar sistemas síncronos.
- El dominio de tiempos es el conjunto de los números naturales.
- Las acciones son indivisibles pero no instantáneas, lo que hace que estas acciones aunque sean ocultadas no desaparecen completamente sino que dejarán el rastro de su duración.
- En las trazas, cada acción llevará asociada un instante de tiempo que indica el momento en que se produjo.

- En el primer modelo propuesto no se aplica el criterio ASAP, pero posteriormente plantea dos versiones reducidas de este criterio que son aplicadas al modelo propuesto. La primera es introducir $time - outs$ para la espera de los procesos, mientras que el segundo es aplicar el criterio pero solamente a las acciones internas.

2.4 Otras Técnicas

2.4.1 ATP

En los artículos [NRSV90a, NS90, NRSV90b], los autores presentan un álgebra para la descripción de procesos temporizados en la que introducen un evento semántico χ que representa el paso del tiempo, mientras que el resto de los eventos no consumen tiempo (discreto).

En este nuevo álgebra no existe la posibilidad de bloqueo temporal (excepto para una cadena infinita de acciones que no consume tiempo) ya que el sistema puede desarrollarse siempre en el tiempo.

A nivel de lenguaje se definen las siguientes operaciones:

- Operador de terminación δ que solo permite el paso del tiempo.

- Acción prefijada

- Elección no-determinista con respecto a acciones asíncronas, es decir, que el tiempo no decide sobre las alternativas.

- Operadores que permiten expresar la evolución temporal de un sistema.

  - Start-Delay(d,T,E).- Si el proceso T no empieza antes de que pasen d unidades de tiempo el sistema se comporta como E.
  - Execution-Delay (d,T,E).- El sistema se comporta como T antes del paso de d unidades de tiempo y como E después de d unidades de tiempo.

  Pese a los nombres de los operadores, el modelo temporal no está basado en el concepto de retraso sino que se basa en el concepto de *time-out*:

- El operador paralelo que representa interleaving y sincronización al mismo tiempo. De este modo podemos describir el tipo de sincronización mediante la definición de una función de comunicación binaria y mediante un operador de restricción podemos prohibir la ocurrencia de acciones. Esta característica permite expresar la concurrencia real representando la ocurrencia simultanea de dos acciones como una única acción que puede sincronizar con el entorno.

2.4.2 $ACP^t_{\tau\epsilon}$

Groote, en su artículo [Gro90] propone una extensión del lenguaje ACP [BK85] que permite modelar sistemas de tiempo real discreto.

Para representar el paso del tiempo utiliza un evento especial t que será interpretado como el paso de una unidad de tiempo, mientras que las acciones se considera que no consumen tiempo.

Todos los componentes del sistema deben sincronizar con t. El operador elección es no-determinista en el tiempo, es decir, el paso del tiempo puede decidir sobre una alternativa u otra.

2.4.3 ACP_ρ

Baeten y Bergstra, en [BB90], proponen una generalización del lenguaje ACP [BK85] para modelar sistemas concurrentes en tiempo real. Dentro del artículo describen un amplio conjunto de lenguajes, realizando en cada uno de ellos varias extensiones, que son caracterizados semánticamente mediante una relación de bisimulación y ecuacionalmente. Las principales características de la extensión son:

- Asume que las acciones no consumen tiempo.
- La existencia de un dominio $R^{\geq 0}$ de números reales positivos que representan el dominio del tiempo.
- Las acciones son etiquetadas con el tiempo absoluto en el que ocurren aunque es posible expresar tiempo relativo.
- La semántica operacional es presentada en términos de un sistema de transiciones cuyos estados son de la forma $< e, t >$ donde e es una expresión de comportamiento y t es el tiempo absoluto transcurrido, y cuyas transiciones pueden ser de tres tipos (ocurrencia de una acción, paso del tiempo y terminación de un proceso).
- Introduce el operador integración ($\int$) como alternativa en tiempo continuo.
- Para que el tiempo pueda transcurrir es necesario que todos los componentes del sistema permitan que transcurra.
- Dos acciones no pueden ocurrir en el mismo instante de tiempo.
- Para evitar el bloqueo resultante de la característica anterior, se introducen acciones combinadas formadas por conjuntos de acciones simples.

2.4.4 TPL Temporal process Algebra

Hennessy y Regan , en [HR90], proponen un algebra de procesos con tiempo cuantitativo. De esta propuesta surge un lenguaje claro e intuitivo como es TPL, que básicamente coincide con CCS extendido con una nueva acción sintáctica y semántica σ que representa el paso de una unidad de tiempo discreto, mientras que el resto de acciones no consumirán tiempo.

El modelo se encuentra basado en el ofrecimiento de una acción para que ocurrirá tan pronto como sea posible, de modo que la expresión $a.P$ representa un comportamiento que puede permitir el paso del tiempo (acción σ), pero de modo que la acción a **debe** ocurrir tan pronto como el entorno lo permita.

Todo el sistema se ve forzado a sincronizar con la acción σ, mientras que el paso del tiempo solo puede ocurrir si el sistema no puede desarrollarse internamente, es decir si no existen acciones τ.

Se propone un preorden $\sqsubseteq$ que relaciona los comportamientos desde el punto de vista de testing [Hen88]; de modo que $B_1 \sqsubseteq B_2$ si todo los test que pasa B_1 son además pasados por B_2. Además proponen una caracterización equivalente del preorden anterior, en términos de "barbas" que teóricamente hace más fácil manejar esta relación de orden y la cual puede servir como base para obtener un modelo denotaciona TPL completo consistente con la equivalencia de testing.

El artículo además propone una caracterización ecuacional consistente con el pre-orden definido (contiene la precongruencia más grande) y completo en términos del subconjunto del lenguaje (STPL), que no incluye ni operador paralelo, ni restricción ni renombrado.

Capítulo 3

Modelos Probabilísticos

3.1 Técnicas basadas en CCS

3.1.1 PCCS

Giacalone, Jou y Smolka, en su articulo [GJS90], realizan una extensión de SCCS (CCS Síncrono), consistente en la sustitución del operador de elección no-determinista ($\sum_{i \in I} E_i$), por un operador de elección probabilística representado como $\sum_{i \in I}[p_i]E_i$ en la que se especifica explicitamente la probabilidad de que ocurra un comportamiento.

La semántica operacional está basada en la noción de derivación probabilística y viene dada estructuralmente mediante un conjunto de de reglas de inferencia. Igualmente presentan una teoría ecuacional basada en bisimulación probabilística cuyos resultados representan la primera axiomatización de la bisimulación probabilística.

3.1.2 $PCCS_{\zeta}$

Glabbeek, Smolka, Steffen, y Tofts, en su artículo [GSST90] proponen tres modelos que llaman respectivamente Reactivo, Generativo y Estratificado, basados en SCCS y que permitirán modelar procesos probabilísticos. Al igual que SCCS modelan comportamientos síncronos mediante tiempo discreto cuantitativo y usan eventos compuestos de tipo bolsa para modelar la concurrencia real. Dentro de PCCS, propuesto ya en [GJS90] se definen las operaciones: inicicación (0); acción prefijada($\alpha.E$); producto sincronizado ($E \times F$); restricción, renombrado, recursión y alternativa probabilística ($\sum_{i \in I}[P_i]E_i$), cuya interpretación depende del modelo especificado.

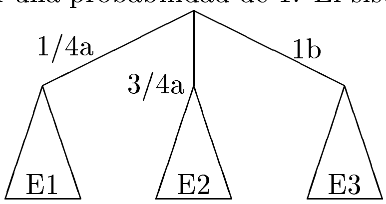
Modelo reactivo

Fue introducido por Larsen y Skou en [LS89], pero en este caso es considerado un sublenguaje de PCCS en el que no existe renombramiento y donde la alternativa probabilística tiene la forma ($\sum_{i \in I}[P_i]\alpha_i.E_i$), donde $\alpha_i \in Act$, $p_i \in (0, 1]$ y $\sum_{i \in I} = 1$ generando un sistema de transiciones etiquetadas, es decir las transiciones son etiquetadas por una acción más su probabilidad asociada. Aplicando el punto de vista reactivo a esta alternativa tendremos que la suma de las probabilidades de las transiciones etiquetadas con la misma acción será 1, de modo que la probabilidad de que un proceso E realice la transición i-esima será p_i condicionada a la asunción de que se produce α_i

Por ejemplo, la expresión :

$$\frac{1}{6}a.E_1 + \frac{3}{6}a.E_2\frac{2}{6}b.E_3$$

puede ser interpretada como que si ocurre la acción a el sistema pasa a comportarse como E_1 con una probabilidad de $1/4$ y como E_2 con probabilidad de $3/4$, mientras que si ocurre la acción b el sistema se comportará como E_3 con una probabilidad de 1. El sistema de transiciones representado sería:



Reactivo

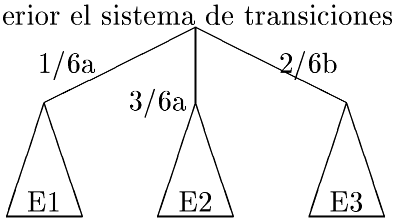
Este modelo no permite conocer la probabilidad de elegir una acción u otra en caso donde ambas sean posibles, perdiendose ademas esa información.

Modelo generativo

En este caso el modelo es definido sobre PCCS completo y no sobre un sublenguaje como ocurría en el modelo anterior. Fue analizado y descrito perviamente en [GJS90, JS90].

En este caso la semántica operacional 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 son conocidas las probabilidades relativas de cada acción.

Si tomamos como ejemplo la misma expresión utilizada en el modelo anterior el sistema de transiciones sería:



Generativo

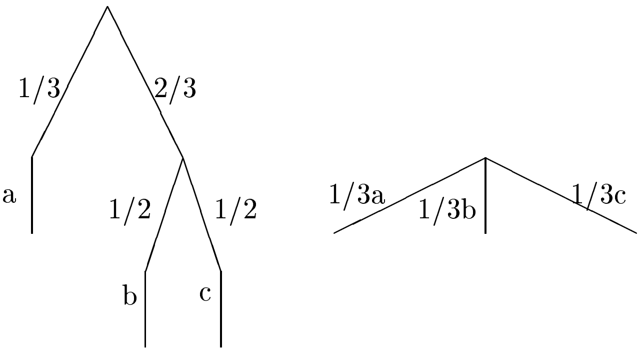
Modelo estratificado

En este modelo, el sistema de transiciones de la semántica operacional posee dos tipos de transiciones diferentes, dependiendo de que su etiqueta sea una probabilidad o una acción. En este modelo se conserva información que en el modelo generativo se perdía como es la probabilidad estratificada

Por ejemplo: Si tomamos la expresión

$$\frac{1}{3}a.0 + \frac{2}{3}(\frac{1}{2}b.0 + \frac{1}{2}c.0)$$

tendremos que en las interpretaciones (sistemas de transiciones) segun los modelos generativo y estratificado serán:



Estratificado generativo

Segun estos sistemas tenemos que en el modelo estratificado la no ocurrencia de "c" significa que "a" continuará con una probabilidad de 1/3, mientras que en el modelo generativo hace que "a" ocurra con una probabilidad de 1/2 que no es el resultado deseado.

Dentro del artículo, los autores definen una relación de bisimulación para cada uno de los modelos, así como unas funciones que permiten la conversión de un modelo a otro más abstracto, es decir de estratificado a generativo y de generativo a reactivo.

Posteriormente Smolka y Steffen, en [SS90], extienden el modelo estratificado para expresar prioridades, de modo que permitirá gestionar transiciones cuya probabilidad a priori es 0. Esas transiciones pueden tener lugar solo si todas las otras alternativas que poseen probabilidades diferentes de 0 son excluidas por las restricciones de contexto.

3.1.3 WSCCS Weighted SCCS

Tofts, en sus artículos [Tof90a, Tof90b] realiza una modificación de SCCS que permitirá modelar comportamientos probabilísticos y prioridades. Para ello introduce un operador alternativa n-ario donde cada una de las alternativas tiene un peso asignado (número natural) en vez de una probabilidad.

El modelo semántico es un sistema de transiciones multicoloreado, que posee dos mecanismos para el cambio de estado como son la ocurrencia de eventos (simple o compuesto) y la elección de una alternativa con un cierto peso.

La probabilidad de que una alternativa sea seleccionada viene dada por el cociente entre el peso de esa alternativa y la suma de los pesos de todas las alternativas.

La asociación de pesos en vez de probabilidades permite la introducción de prioridades utilizando un peso especial infinito que tiene prioridad sobre el resto de alternativas. Esta es la única diferencia de este modelo con el estratificado.

3.2 Técnicas basadas en CSP

3.2.1 CSP probabilistico CSP-Pro

J.J. Zic, en [Zic90] propone la posibilidad de extender CSP en términos de probabilidad y tiempo, aunque a la hora de la verdad realiza la extensión probabilística mientras que sobre la temporal solamente realiza una ligera introducción.

De cara a la extensión probabilística, propone un operador elección probabilístico n-ario $\bigtriangledown_{\epsilon i=1}^{n}(P_i, \pi_i)$ donde ϵ es el entorno al que se refiere y π_i es la probabilidad de obtener el comportamiento P_i.

El modelo semático que propone es un proceso estocástico definido por la tupla $(A, F, \phi_{F,\epsilon}, D, \phi_{D,\epsilon})$ donde A es el alfabeto, F es el conjunto de fallos, D es el conjunto de divergencias y $\phi_{F,\epsilon}$ y $\phi_{D,\epsilon}$ son dos funciones que asignan probabilidades a los elementos de F y D respectivamente.

Expone también la dificultad que supone, a la hora de obtener el modelo semántico, el hecho de que se debe estar familiarizado con el entorno de un proceso para poder determinar su espacio probabilístico.

3.2.2 PCSP

Fernando Cuartero , en su tesis doctoral [CG93], realiza una extensión de CSP que permite modelar sistemas asíncoronos introduciendo una caracterización probabilística de los sistemas. Esta extensión mantiene dentro de lo posible las propiedades fundamentales de CSP para lo que se han incorporado de forma coherente los conceptos probabilísticos básicos.

La variación realizada ha consistido en la introducción de una distribución probabilística en aquellas operaciones que de forma implicita o explicita suponen una elección , como son la elección externa, interna y el operador paralelo, de modo que sustituye el no-determinismo por una cuantificación probabilística, que permitirá también la definición de prioridades utilziando probabilidades extremas.

A la hora de definir la semántica del lenguaje no se ha limitado a definirla de una única forma sino que la ha definido en su forma operacional, denotacional, algebraica y mediante test, estableciendo posteriormente la equivalencia entre éstas.

Capítulo 4

Modelos Temporizados y Probabilísticos

4.1 Técnicas basadas en LOTOS

4.1.1 Bochmann-Vaucher

En su artículo [BV88] realizan una extensión informal de la técnica de Estelle, SDL y LOTOS, haciendo posible la utilización de esta técnica para la evaluación de rendimientos. La extensión es realizada tanto en la vertiente temporal como en la probabilística:

Extensión temporal .- Asocia a cada uno de los eventos internos un atributo que indica el instante de tiempo (discreto) en el cual puede ocurrir. Si un evento interno no posee este atributo, entonces tendrá lugar inmediatamente. Los eventos observables (externos) ocurrirán cuando el entorno lo determina. Aunque no lo hace de forma explicita con estas condiciones se está aplicando el criterio ASAP. Asume que la ocurrencia de un evento es instantánea por lo que para expresar la duración de estos habría que hacerlo mediante retrasos.

Extensión probabilística .- Asocia a cada evento interno un atributo que indica el peso (probabilidad) de cada evento dentro de una alternativa.

Esta extensión no fue formalizada en un modelo semántico.

4.1.2 TPAL

Bueno, Valero y Cuartero, en [BVC97], proponen una extensión de LOTOS básico (sin tipos de datos), en la que combinan los factores tiempo y probabilidad. Para ello introducen nuevos operadores como son:

- Elección Probabilística.- Indica que será el sistema el que decida el comportamiento que se seguirá según una distribución de probabilidad.

- Prefijo Temporizado.- Fija el tiempo máximo para la ejecución de una accion, después del cual no podrá ser ejecutada.

- Urgencia.- Establece la priorización de las acciones urgentes, deberán ser ejecutadas inmediatamente cuando las pida el entorno.

En una primera versión del modelo, se ha excluido la operación de ocultamiento debido a la complicación que introduce en la semántica , y se obliga también a que en la sintaxis exista una alternancia entre los dos operadores de elección (probabilístico y no-determinista).

Para la especificación del tiempo, se ha incluido una acción especial "i" que representa un tic de un reloj global, que coordinará todos los procesos del sistema y que desde el punto de vista de la sintaxis tiene una gran importancia para imponer la alternancia entre terminos probabilísticos y no-deterministas.

La semántica operacional dada intenta cumplir con las indicaciones de urgencia expresadas por el usuario para lo que se define el grado de urgencia de un proceso no-determinista como el numero máximo de acciones urgentes que ese proceso puede ejecutar.

Actualmente se esta desarrollando este modelo mediante la definición de la semántica de redes de Petri, y construyendo una herramienta que permita ejecutar las especificaciones realizadas, así como la comprobación de las propiedades de estas.

4.2 Técnicas basadas en CCS

4.2.1 TPCCS

Hansson y Jonsson, en su artículo [HJ90], proponen un álgebra de procesos que denominan Timed Probabilistic Calculus of Communicating System, como extensión de CCS que permitirá modelar sistemas asíncronos con probabilidades y tiempo.

La extensión probabilística consiste en la introducción de un nuevo operador de elección probabilística que sera resuelta internamente sin influencia del entrorno. Este nuevo operador, hará que un proceso que se encuentre en un determinado estado pueda realizarse una elección no-determinista influenciada por las acciones del entorno o una probabilística, debiendo existir una alternancia estricta entre ambos tipos de elecciones debido a razones técnicas.

Para la extensión temporal consideran un dominio de tiempo discreto de modo que el paso del tiempo viene proporcionado por un reloj global que emite ticks.

Un sistema dejará pasar el tiempo cuando este paso haya sido especificado explicitamente meidante un retraso o cuando no exxistan acciones posibles para ser ejecutadas. Este paso del tiempo podemos considerarlo caracterizado como "*Espera Arbitraria y mínimo retraso*" de modo que no existe una cota superior para el tiempo que esperará un proceso para ejecutar una acción, pero en el momento que una acción sea posible, el proceso deberá ejecutarla sin retraso.

Para representar este paso del tiempo definen dos operadores, uno a nivel sintáctico como es el operador "timeout" representado por $N_1 \rhd_i N_2$ cuyo significado es un proceso que pasará a comportarse como N_2 después de "i" unidades de tiempo a no ser que N_1 realice antes una acción, y otro a nivel semántico como es una acción especial χ que indica el paso del tiempo.

Al final del artículo define una relación de bisimulación fuerte y una axiomatización completa para expresiones de procesos finitos no recursivos que permitiran transformar cualquier expresión en su forma normal.

Capítulo 5

Resumen y conclusiones

5.1 Resumen

A lo largo del presente trabajo se han ido describiendo diferentes modelos utilizados para especificar formalemente sistemas cumcurrentes, de una forma superficial y sin realizar ninguna comparación entre ellos

Llega ahora el momento de resumir las características de estos modelos en un cuadro que nos puede permitir comparar fácilmente las diferencias en las características fundamentales de éstos.

Los aspectos recogidos en este cuadro resumen son:

- 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.

- Sistemas Síncrono y/o asíncrono. El término síncrono significa que los eventos ocurren sólo en unos instantes específicos de tiempo indicados generalmente por un reloj global, mientras los sistemas asíncronos no poseen esta restricción permitiendo además que los eventos puedan ocurrir en cualquier instante.

- Con o sin evento de reloj explicito. En las técnicas en las que se incluye el uso del reloj hay un evento especial que representa un tick de reloj y el cual indica el paso del tiempo. Cada tick representa 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 periodo 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 los eventos. Se puede considerar que los eventos consumen tiempo o por el contrario se consideran instantáneos.

- 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. Solo unos pocos modelos permiten la caracterización probabilística del comportamiento de un sistema, mientras que los demás describen los sistemas de forma determinista o no-determinista, pero sin caracterización probabilística. Algo estrechamente relacionados con el concepto de caracterización probabilística es 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 entrelazado 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).

Todos los modelos presentados en esta sección tienen en común la característica de que la sincronización entre los procesos es síncrona, es decir, que la sincronización solo puede ocurrir si todas las partes envueltas en ella permiten que 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.

La siguiente tabla muestra el resumen comparativo de las características de los modelos descritos de acuerdo con los criterios expuestos anteriormente.

	T.Cuantitativo	T. Discreto	T. Continuo	Evento de Reloj	S. Síncrono	S. Asíncrono	ASAP	Eventos Instantaneos	C. Probabilística	Concurrencia	Ev. Compuestos	Semántica	Prioridades
Quemada-Fernandez (LOTOS)	X	X	*		X	X		X		I		O	
TIC	X	X	*		X	X		X		I		O	
LOTCAL	X	X		X	X	X		?		I	S	O	
LOTOS temporal	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.Calvelli (LOTOS)	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-Pri		?	?			X		?		I		O	X
TCCS	X	X	*		X	X		X		I		O	
CCS temp. (L. Chen)	X	*	X		X	X	X	X		I		O	
CCS temp.(W. Yi)	X	*	X			X	X	X		I		O	
CSP Temp	X	*	X			X	X			I*		D	
TCSP Temp	X	X	*			X	X			R	B	D	
Ortega Mallen	X	X		X	X		X			I		D	
ATP	X	X		X	X	X		X		R	B	O	
$ACP^t_{\tau\epsilon}$	X	X		X	X	X		X		I		O	
ACP_ρ	X		X		X	X		X		R	S	O	
TPL	X	X		X		X	X	X		I		O	
PCCS	X	X			X	X			X	I		O	
$PCCS_\zeta$	X	X			X				X	R	B	O	X
WSCCS	X	X			X				X	R	B	O	X
CSP Prob									X	I*		D	
PCSP	X					X		X	X	I		O/D	X
Bochmann-Vaucher (LOTOS)	X	X	*			X	X	X	X	I		O	
TPAL	X		X	X			X	X		I		O	X
TPCCS	X	X		X		X	X	X	X	I		O	

Donde tenemos que

X= Característica soportada

blanco= Característica no soportada

*= Podría soportarla pero no lo hace explicitamente.

?= Propiedad no aplicable

B= Eventos compuestos tipo bolas

S= Evento compuesto tipo conjunto.

O= Semántica operacional

D= Semántica denotacional

I= Interleaving

R= Concurrencia Real

I*= Eventos no instantáneos y no soportan eventos compuestos. Además la ocurrencia simultánea de eventos no puede ser expresada por el lenguaje como el interleaving de eventos. Sin embargo la semántica de estas técnicas modela la concurrencia copmo el interleaving de los eventos simultáneos.

5.2 Conclusiones

Observando la tabla de la sección 5.1 y como resumen podemos indicar las siguientes conclusiones:

- La mayor parte de los modelos con extension prbabilística definen un dominio de tiempo discreto, siendo la mayoria de ellos facilmente modificados para la utilización de tiempo continuo.
- En cuanto la restriccion sobre la ejecución de las acciones, la mayor parte de los modelos son asíncronos de modo que las acciones pueden ser ejecutadas en cualquier instante yu no solamente en unos instantes determinados.
- Podemos considerar dos grandes grupos de modelos:
  - Esos modelos que consideran que un evento debe ocurrir **tan pronto como sea posible**, los cuales son muy homogéneos permitiendo la generación de modelos intuitivos. Estos modelos se basan en la idea de que "La sincronización entre varias partes solamente se producirá cuando todas las partes estén listas para sincronizar.
  - Los modelos que no aplican el criterio anterior y que ordenan los eventos de los procesos concurrentes en el tiempo , pero consideran que un evento puede ocurrir en cualquier momento en que este disponible. Estos modelos son más heterogéneos y a veces dificiles 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.

Mirando la extensa variedad de técnicas existentes para la especificación formal de requerimientos, la conclusión general que se puede obtener es que el campo de la especificación formal de los requerimientos de tiempo y probabilidad son un campo activo de investigación que todavía no ha sido completamente desarrollado.

Bibliografía

[Azc90] A. Azcorra. Formal Modeling of Synchronous Systems. PhD thesis, ETSI Telecomunicacion, UPM, Madrid, Ciudad Universitaria s/n, September 1990.

[BB90] J.C.M. Baeten and J.A. Bergstra. *Real time process algebra.* Technical Report P8916b, Programing Research Group, University of Amsterdam, March 1990.

[BK85] J.A. Bergstra and J.W. Klop. *Algebra of comunicating Processes with abstraction. Theoretical Computer Science*, 37(1):77–121, 1985.

[BLT90] T. Bolognesi, F. Lucidi, and S. Trigila. *From timed petri nets to timed LOTOS.* In L. Logrippo, R. Probert, and H. Ural, editors, *Tenth Internacional IFIP Symposium on Protocol Specification, Testin and Verification*, pages 377–406. Elsevier Science, june 1990.

[Bri88] Ed. Brinksma. On The Design of Extended LOTOS. PhD thesis, Univerisdad de Twente, 1988.

[BV88] G.V. Bochmann and J. Vaucher. *Adding Perfonmance Aspects to Specification Languajes.* In North Holland, editor, *Workshop on Protocol Specification, Testing an Verification: VIII*, 1988.

[BVC97] Antonio Bueno, Valentin Valero, and Fernando Cuartero. *TPAL. Un modelo algebraico con probabilidades y tiempos basado en LOTOS.* In J. Llamas, J.J. Pazos, and M.J. Fernandez, editors, *V Jornadas de Concurrencia*, pages 223–233, Vigo, Junio 1997. Servicio de Publicacións. Universidade de Vigo.

[Cal89] C. Calveli. *A timed model for basic LOTOS.* Technical report, Hewlett-Packard LAboratories, Bristol, UK, November 1989.

[CAM90] L. Chen, S. Anderson, and F. Moller. *A timed calculus of communicating systems.* Technical Report ECS-LFCS-89-104, Laboratory for Foundations of Compueter Science LFCS, University of Edinburg, December 1990.

[CG93] Fernando Cuartero Gomez. CSP PRobabilístico. PCSP. PhD thesis, Facultad de Matemáticas. Universidad Complutense de Madrid, 1993.

[CH88] R. Cleaveland and M. Hennessy. *Priorities in Process Algebras.* In *Proceedings Symposium on Logic in Computer Science LICS 88*, 1988.

[FP90] C. Fidge and M. Pilling. *Specification languajes for distributed real-time software.* Technical Report 156, University of Queensland, Key Center for Software Technology Department of Computer Science, St Lucia, Queensland,4067, Australia, Marzo 1990.

[GJS90] A. Giacalone, C. Jou, and S.A. Smolka. *Algebraic reasoning for probabilistic concurrent systems.* In *Proceedings of working conference on programming concepts and methods IFIP TC2*, 1990.

[Gro90] J.F. Groote. *Specification and verification of real time systems in ACP.* In L. Logrippo, R. Probert, and H. Ural, editors, *Protocol Specification, Testing and Verification X*, The Netherlands, 1990. Elsevier Science.

[GSST90] R. Glabbeek, S.A. Smolka, B. Steffen, and C.M.N. Tofts. *Reactive,Generative, and Stratified Models of Probabilistic Processes.* In *Proceedings 5th Annual Symposium on Logic in Computer Science*, Philadelphia, USA, 1990. LICS 90.

[Hen88] M. Hennessy. Algebraic Theory of Processes. The MIT Press, Cambridge, 1988.

[HJ90] Hans Hansson and Bergt Jonsson. *A Calculus for Comunicating Systems with Time and Probabilities.* In *Real-Time Systems Symposium.* IEEE Computer Society, Tc Real-Time Systems, December 1990.

[HR90] M. Hennessy and R. Regan. *A temporal process algebra.* In J. Quemada, J. Maas, and E. Vazquez, editors, *Third International Conference on formal Description Techniques*, Madrid, Spain, November 1990. FORTE 90.

[ISO] ISO. Specification of Abstract Syntax Notation One (ASN.1). ISO.

[JS90] C. Jou and S.A. Smolka. *Equivalences, Congruences, and Complete Axiomatizations for Probabilistic Processes.* In *Lecture Notes in Computer Science*, number LNCS-458, ISBN 3-540-53048-7, pages 367–383. Springer-Verlag, Berlin Heidelberg, New York, 1990.

[LS89] K.G. Larsen and A. Skou. *Bisimulation through probabilistic testing.* In *Proceedings 16th Annual ACM Symposium on Principles of Programming Languages*, 1989.

[MF76] P. Merlin and D.J. Farder. *Recoverability od Communication PRotocols-Implications of a Theoretical Study. IEEE Trans. Communications*, 24:1036–1043, September 1976.

[MF91] C. Miguel and A. Fernandez. *State of the Art on Timed Probabilistic Models (Based on process algebras).* Technical Report OSI95/90-7, Departamento de Ingeniería Telemática. Universidad Politécnica de Madrid, 1991.

[Mil83] R. Milner. *Calculi for Synchrony and Asynchrony. Theoretical Computer Science*, 25:267–310, 1983.

[MT90] F. Moller and C. Tofts. *A temporal calculus of communicating systems.* In *Lecture Notes in Computer Sciences*, number LNCS-458 ISBN 3-540-53048-7, pages 401–415. Springer-Verlag, Berlin Heidelberg, New York, 1990.

[NRSV90a] X.Ñicollin, J.L. Richier, J. Sifakis, and J. Voiron. *ATP: an algebra for timed processes.* Technical Report RT-C16, Project SPECTRE, LAboratoire de Génie Informatique de Grenoble, B.P. 68, 38402 Saint MArtin D'Hares Cedex, January 1990.

[NRSV90b] X.Ñicollin, J.L. Richier, J. Sifakis, and J. Voiron. *ATP: an algebra for timed processes.* Technical Report RT-C16, Project SPECTRE, LAboratoire de Génie Informatique de Grenoble, B.P. 68, 38402 Saint MArtin D'Hares Cedex, January 1990.

[NS90] X.Ñicollin and J. Sifakis. *ATP: an algebra for timed processes.* Technical report, The Netherlands, 1990.

[NY85] N.ÑouNou and Y. Yemini. *Algebraic specification-based performance analysis of communication protocols.* In *Workshop on Protocol Specification, Testing and Verification: IV. IFIP*, 1985.

[Old87] E.R. Olderog. *TCSP: Theory of Communicating Sequential Processes. Advances in Petri Nets LNCS 255*, 1987.

[OM90] Yolanda Ortega Mallen. En Busca del Tiempo Perdido. PhD thesis, Facultad de Matemáticas. Universidad Complutense de Madrid, Mayo 1990.

[OMFE90] Yolanda Ortega-Mallen and David de Frutos-Escrig. *Timed observations: A semantic model for real-time concurrency.* In M. Broy and C.B. Jones, editors, *TC2-Working Conference on Programming Concepts and Methods*, North-Holand, 1990.

[QAF89] J. Quemada, A. Azcorra, and David Frutos. *A Timed calculus for LOTOS.* In S. Vuong, editor, *Formal Description Technique*, Vancouver, CANADA, December 1989. FORTE 89.

[QAF90] J. Quemada, A. Azcorra, and David Frutos. *TIC: A Timed Calculus .* Technical report, Departamento de Ingenieria Telemática, ETSI. Telecomunicación, UPM, Madrid, Spain, 1990.

[QF87] J. Quemada and A. Fernandez. *Introduction of quantitative relative time into LOTOS.* In *Workshop on Protocol Specification, Testing an Verification: VII*, Zurich, May 1987. IFIP.

[RR86] G.M. Reed and A.W. Roscoe. A timed Model for Communicating Sequential Processes, volume 226. Springer-Verlag, 1986.

[RR87] G.M. Reed and A.W. Roscoe. *Metric Spaces as Models for Real-Time Concurrency.* In *Third Workshop on the MAthematical Foundations of Programming Language Semantics*, Lecture

Notes in Computer Science, pages 331–343. Springer-Verlag, 1987.

[RR88] G.M. Reed and A.W. Roscoe. A timed Model for Communicating Sequential Processes, volume 58, pages 249–261. North-Holland, 1988.

[Rud85] H. Rudin. *Time in formal protocol specification.* In *GI/NTG conference on Communication in Distributed Systems*, Karlsruhe, West Germany, 1985.

[Rud86] H. Rudin. *The decision of time in protocol specification.* In *OSI IBM-Networkshop Oberlech*, August 1986.

[Sch90] S. Schneider. Correctness and Communication in Real-Time Systems. PhD thesis, Oxford University Computing Laboratory. Programming Research Group, Oxford OX1 3QD , England, March 1990.

[SR88] C. Smith and S. Rudkin. *Time Guards and ASN.1 in LOTOS.* Technical Report Ref. 40, Formal Methods Group, British Telecom, St. Vincen House, Ipswich, UK, 1988.

[SS90] S.A. Smolka and B. Steffen. *Priority as Extremal Probability. Lecture Notes in Computer Science*, (LNCS-458, ISBN 3-540-53048-7):456–466, 1990.

[Tof88] C. Tofts. *A temporal calculus of communicating systems.* Technical report, University of Edinburgh, April 1988.

[Tof90a] C. Tofts. *A synchronous Calculus of Relative Frequency.* In *Lecture Notes in Computer Science*, number LNCS-458, ISBN 3-540-53048-7, pages 467–480. Springer-Verlag, Berlin Heidelberg, New York, 1990.

[Tof90b] C. Tofts. *Relative frequency in a synchronous calculus.* Technical Report ECS-LFCS-90-108, Laboratory for Foundations of Computer Science LFCS, University of Edinburg,the King's Buildings, February 1990.

[vHTZ89] W.H.P. van Hulzen, P.A.J. Tilanus, and H. Zuidweg. *LOTOS Extended with Clocks.* In S. Vuong, editor, *Formal Description Techniques*, Vancouver, CANADA, December 1989. FORTE 89.

[Wal83] B. Walter. *Timed petri nets for modeling and analysing protocols with real time characteristics.* In North-Holland, editor, *Protocol Specification, Testing and Verification III*, pages 149–159, 1983.

[Yi91] Wang Yi. *CCS+Time= an Interleaving Model for Real Time Systems.* In *ICALP'91.* 1991.

[Zic90] J.J. Zic. *Some thoughts on communication system performance evaluation.* In *Proceedings of the Open Distributed Processing Workshop*, Sydney, January 1990.