Aplicación del paradigma Cell-DEVS binario con tiempo para modelado y simulación de sistemas complejos

Gabriel A. Wainer [1,2], Norbert Giambiasi [2]

[1]Departamento de Computación. Facultad de Ciencias Exactas y Naturales.
Universidad de Buenos Aires.
Pabellón I - Ciudad Universitaria. Buenos Aires (1428) – ARGENTINA
E-mail:gabrielw@dc.uba.ar

[2] DIAM-IUSPIM. Université d'Aix-Marseille III
Av. Escadrille Normandie Niemen. 13397 Marseilles Cédex 20 – FRANCE
E-mail: Norbert.Giambiasi@iuspim.u-3mrs.fr

Abstract: Se muestra la aplicación de nuevos formalismos para el modelado y simulación de espacios de celdas. Los paradigmas están basados en los formalismos DEVS y Autómatas Celulares Asincrónicos, combinados con demoras inerciales o de transporte. Se facilita la definición automática de espacios de celdas, facilitando la verificación de los modelos. De esta forma se puede lograr el desarrollo eficiente y reducido en costo de soluciones de simulación para estos sistemas.

Keywords: Simulación a Eventos Discretos, Modelado, espacios de celdas.

1. INTRODUCCION

En la actualidad existen diversos paradigmas para especificar formalmente sistemas dinámicos formalmente; cada uno de ellos está mejor adaptado para un tipo distinto de sistema. El formalismo DEVS (Discrete EVents system Specification) fue propuesto con el fin de modelar sistemas a eventos discretos. Este formalismo (Zeigler 1976) permite descripción modular de los modelos, atacando la complejidad usando una aproximación jerárquica y orientada a objetos. Un sistema real modelado con este paradigma puede describirse como compuesto por distintos submodelos. Cada uno de ellos puede ser de comportamiento (atómico) o estructural (acoplado).

Como el paradigma es cerrado bajo clausura, los modelos acoplados pueden integrarse en una jerarquía. El uso de esta estrategia permite la reutilización de modelos ya chequeados, mejorando la seguridad de las simulaciones, reduciendo el tiempo de chequeo y mejorando la productividad.

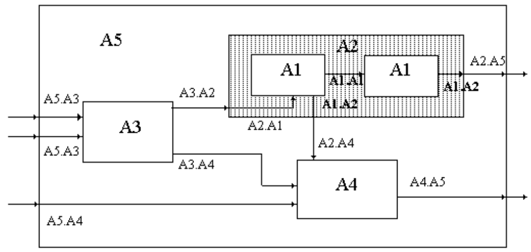


Figura 1. Ejemplo de acoplamiento de modelos DEVS (A1, A3, A4: modelos atómicos).

Los Autómatas Celulares (Wolfram 1986) usan variables discretas para el tiempo, espacio y estados del sistema. Un AC es una grilla regular n-dimensional e infinita, en donde cada celda puede tomar un valor de un conjunto finito. Los estados de las celdas cambian en pasos de tiempo discretos de acuerdo con una función de cálculo local. La función considera el estado actual de la celda y un conjunto finito de celdas cercanas (llamado la vecindad). El usar tiempo discreto impone restricciones en la precisión y eficiencia de los modelos. Para modelos complejos será necesario usar un alto grado de tiempo de procesamiento para obtener la precisión deseada. Sin embargo, en muchos casos la mayoría de las celdas del autómata no precisan actualización. El uso de un AC asincrónico (basado en eventos) puede evitar estos problemas.

En (Wainer y Giambiasi 1998) se definió el formalismo Cell-DEVS con tiempo. Este combina el formalismo de AC con el paradigma DEVS. Cada celda se define como un modelo atómico, y se incluye un procedimiento para acoplar celdas. Estos modelos pueden demorarse usando construcciones asociadas con componentes funcionales (Giambiasi y Miara 1976). Este trabajo ejemplifica el uso de este formalismo

2. Cell-DEVS: definición informal

La Figura 2 muestra informalmente los contenidos básicos de una celda binaria. Al recibir un evento externo, se ejecuta una función booleana, que consume el valor de entrada para el evento. Luego, la salida de la celda se demora durante **d** unidades de tiempo, y se planifica un evento interno. Para ello se usa una cola de eventos de entrada, ya que pueden llegar diversos eventos externos durante la demora. Cuando llega la hora del evento interno, se transmite la salida.

Como las celdas influenciadas sólo deben activarse cuando la celda actual cambia, los resultados de la función local sólo serán transmitidos si cambia el estado. Para ello se almacenan el estado presente y futuro de la celda. También se guarda una copia local de los valores de los vecinos, ya que el comportamiento de un modelo atómico debe ser independiente de los demás modelos. En otros casos, pueden introducirse demoras inerciales (Giambiasi y Miara 1976) para representar ciertos fenómenos en los que los cambios de estado dependen no sólo de valores externos sino de su duración. Esta construcción permite representar comportamientos con desalojo para una celda. La idea es que una entrada se descarta si su valor no se mantiene durante un período especificado (la demora inercial). En cambio, si el valor es constante durante ese tiempo, el estado de la celda debe cambiar.

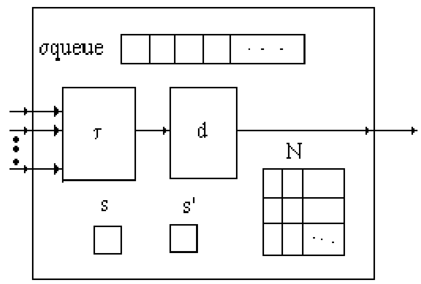


Figura 2. Celda con demora de transporte.

Este comportamiento se muestra a continuación, considerando una celda cuya función de cálculo es la identidad y usa una demora inercial.

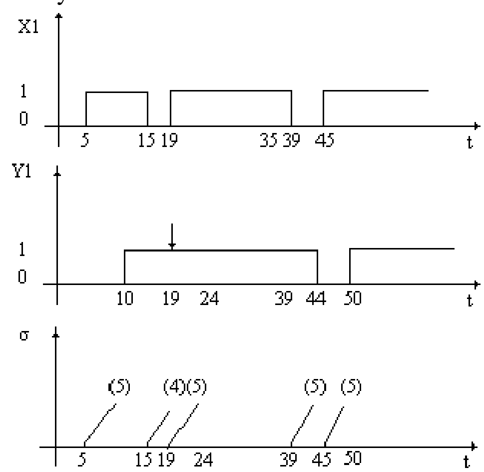


Figura 3. Trayectorias con demora inercial.

Aquí, el último evento futuro que ha llegado puede ser removido si hay una nueva entrada antes de consumir la demora inercial. Esto sólo ocurrirá cuando el nuevo valor externo sea diferente del almacenado previamente, lo que equivale a tener un único evento externo con una demora inercial mayor. Un ejemplo se ve en el instante 24: en la hora 19 llegó un nuevo evento externo con valor 1. Como el estado en ese momento era 1, el evento externo es removido y la función de salida no se ejecuta.

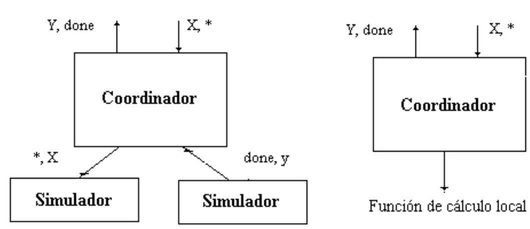
En (Wainer y Giambiasi 1998) puede encontrarse una especificación formal para estos modelos, que será utilizada en los ejemplos de la sección 4.

3. MODELOS DE IMPLEMENTACION

Con respecto al desarrollo de modelos de implementación, un modelador debería poder definirlos usando las especificaciones previas, verificando automáticamente la correctitud del modelo ejecutable. Deben poder representarse tres aspectos básicos de un espacio de celdas: dimensión, influencias y comportamiento. Con respecto a la dimensión, un usuario debe poder definir el tamaño del espacio de celdas. Luego, deben definirse las influencias de una celda incluyendo su vecindad. Los ports de entrada/salida serán creados siguiendo la especificación formal, así como el acoplamiento entre celdas. Para representar el comportamiento de la función de cálculo local de cada celda, se ha definido un lenguaje de especificación simple. Un compilador traduce la especificación en una representación interna Cell-DEVS.

La especificación de una celda se definirá como una lista de expresiones booleanas que devuelve un resultado (binario o ternario). Las expresiones pueden incluir operadores relacionales y llamadas a funciones predefinidas. Los resultados se calculan en base al estado actual de una celda y sus vecinos. Los parámetros se usan para crear un arreglo de objetos de celdas atómicas, y para definir las influencias de cada una. Para iniciar la simulación, deben definirse condiciones iniciales para cada celda, así como las condiciones de parada del ciclo de simulación.

Pueden detectarse diversos errores de la especificación. Si hay varios resultados para una celda con los mismos estados y vecinos, la especificación es ambigua. En cambio si no existe un resultado para un estado dado, la especificación está incompleta. Si varias expresiones producen el mismo resultado, la simulación puede seguir pero avisando al usuario. En cambio, si se encuentran resultados diferentes, la simulación debe detenerse (no puede determinarse el estado futuro de la celda). Este chequeo permite detectar inconsistencias en la definición del modelo, que deben ser reportadas al usuario que cambiará la especificación para asegurar su correctitud, verificando la validez del modelo.



(a) (b)

Figura 4. Reacción de los coordinadores ante mensajes (a) Jerárquico (b) Achatado.

El paradigma DEVS permite definición jerárquica de los módulos, y el mecanismo de simulación también tiene interacción jerárquica. Las entidades a cargo de manejar la simulación (coordinadores y simuladores) comparten información por medio del pasaje de mensajes. Esta interacción produce alto grado de overhead, que puede reducir el desempeño en tiempo de ejecución de las simulaciones Cell-DEVS, más aún en el caso en que los modelos estén compuestos por un gran número de celdas. Una forma de reducir el overhead es achatando la jerarquía de modelos. Un simulador achatado se implementa como un arreglo bidimensional de registros asociado con el espacio de celdas.

4. UN EJEMPLO DE APLICACION

La Figura 5 muestra un modelo cuya finalidad es simular tráfico en una sección de población urbana. El modelo está compuesto de distintos submodelos, cada uno de los cuales está construido usando distintas aproximaciones.

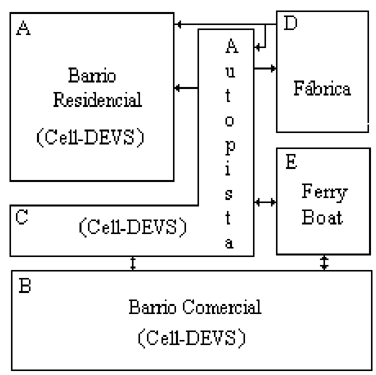


Figura 5. Un modelo Cell-DEVS (Wainer y Giambiasi 1998).

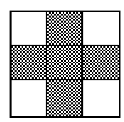
En este caso, A es un modelo Cell-DEVS que representa un barrio residencial. El modelo tiene la finalidad de estudiar valores de polución representando las partículas de humo en el aire. El tráfico de la autopista y el humo de los camiones de la fábrica se consideran generadores de smog. La función de cálculo local también debe modelar la influencia del viento sobre el smog. Si una partícula está en una celda durante un tiempo, se difunde a sus vecinos; sino, el viento la remueve. Se considerará que un automóvil o un camión generan una cantidad de smog equivalente a la cantidad de partículas necesarias para activar una celda del modelo A (si esta relación debiera ser distinta, podría utilizarse un modelo traductor intermedio). El modelo B representa el tráfico en un barrio comercial. Las calles son de una sola mano (con direcciones Norte o Este), y no hay semáforos. El último modelo Cell-DEVS, llamado C, representa una autopista de mano única que pasa a través de los barrios A y B. El modelo servirá para estudiar el flujo del tráfico en la autopista y su influencia en el resto de la ciudad.

El modelo atómico D representa una fábrica con camiones que llegan desde la autopista y camiones que salen de la fábrica hacia la autopista. Los resultados de esta simulación podrían utilizarse para planificar la entrada y salida de camiones a la fábrica. Finalmente, el modelo atómico E representa la entrada a un Ferry-Boat que conecta la ciudad y una isla. Este modelo podría se usa para estudiar el flujo de tráfico a los ferrys (determinando el número óptimo de botes que deben ser usados).

4.1. Comportamiento de los espacios de celdas

A continuación se analizará el comportamiento de los espacios de celdas de la Figura 5, especificando las funciones de cálculo locales con el lenguaje definido. En este caso la demora inercial utiliza para modelar la difusión de la polución: si el viento remueve una partícula, el smog no se expande. Para ello se considera una demora inercial de 3 unidades de tiempo. Ante eventos simultáneos, la función select considera viento con dirección SO.

Vecindad: { (a_{12}, a_{21}, a_{22}, a_{23}, a_{32}) };
Select: { (a_{12}, a_{23}, a_{22}, $a_{21,}$ a_{32}) };
viento = 3;
m = 9; n = 10; in_delay(viento).

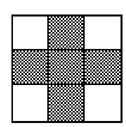


	Vecindad
1	(a_{22} = 1 AND NOT (ALL = 0)) OR (a_{22}=0) AND (a_{12}=1 OR a_{21}=1 OR a_{23}=1 OR a_{32}=1)
0	(ALL = 0) AND (a_{22} = 1)

Figura 6. Especificación del modelo A

La especificación del modelo B representa el movimiento de los vehículos en calles de mano única. La primera regla representa la entrada de un nuevo vehículo en una celda vacía, o un vehículo frenado que no se puede mover. La segunda regla representa la partida del vehículo de la celda de origen, o un espacio rodeado por celdas vacías.

Vecindad: { (a_{12}, a_{21}, a_{22} ,a_{23}, a_{32}) };
Select: { ($a_{32,}$ $a_{21,}$ a_{22}, a_{12}, a_{23}) };
velocidad = 5;
m = 50; n = 10;
tr_delay(velocidad).

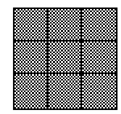


s	Vecindad
1	(a_{22} = 0 AND a_{32} = 1) /* Norte */ OR (a_{22} = 0 AND a_{21} = 1)) /* Oeste */ OR (a_{22} = 1 AND a_{23} = 1) OR (a_{22} = 1 AND a_{32} = 1)
0	(a_{22} = 1 AND a_{12} = 0) /* Norte */ OR (a_{22} = 1 AND a_{23} = 0) /* Oeste */ OR (a_{22} = 0 AND a_{32} = 0 AND a_{21} = 0)

Figura 7. Especificación del modelo B.

El desplazamiento de un vehículo no se registra explícitamente, sino que el cambio de estado se divide en dos movimientos diferentes. El mecanismo de simulación lo expandirá a todas las celdas influenciadas (vecinos), pero sólo se registra en aquellas donde el movimiento es válido. Si hay un cruce, el vehículo de la derecha pasa primero: el flujo hacia el norte es el primero en evaluarse en la función Select. Las demoras de transporte modelan la aceleración de los vehículos.

Vecindad: { (a_{11}, a_{12}, ..., a_{33}) };
Select = { a_{23}, a_{13}, a_{33}, a_{22}, a_{12}, a_{32}, a_{11}, a_{21}, a_{31} };
velocidadH = 2;
m = 15; n = 4;
tr_delay(velocidad).



S	Vecindad
1	(a_{22} = 0 AND a_{21} = 1) OR /* Flujo normal */ (a_{22}=0 AND a_{21}=0 AND a_{31}=1 AND a_{32}=1) OR /* Sobrepaso por izquierda */ (a_{22}=0 AND a_{21}= 0 AND a_{11}=1 AND a_{12}=1) /* Sobrepaso por derecha */ OR (a_{22} = 1 AND $Column_3$ = 1) /* Embotellamiento */
0	(a_{22} = 1 AND a_{23} = 0) OR /* Flujo normal */ (a_{22}=1 AND a_{23}=1 AND a_{13}=0 AND a_{12}=0) OR /* Sobrepaso por izquierda */ (a_{22}=1 AND a_{23}=1 AND a_{33}=0 AND a_{13}=1 AND a_{32}=0) /* Sobrepaso por derecha */ OR (a_{22} = 0) /* Celda vacía no analizada en la regla 1 */

Figura 8. Especificación del modelo C.

La especificación de este modelo representa una autopista de una sola mano con cuatro carriles. En este caso el movimiento de los vehículos también se modela dividiéndolo en dos fases distintas. La primer regla incluida representa la llegada de un nuevo vehículo a la celda actual, o un vehículo demorado en una celda.

4.2. Especificación formal de los submodelos

En esta sección se muestra un detalle de la especificación formal de los modelos utilizados. Los modelos D y E son modelos DEVS tradicionales, y su especificación detallada puede encontrarse en (Wainer y Giambiasi 1998). En este caso enfocaremos la especificación de los modelos celulares.

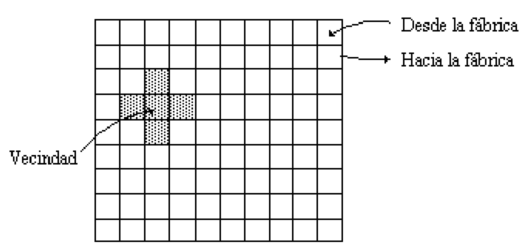


Figura 9. ***Espacio de celdas A.***

El espacio de celdas A puede especificarse como:

$$M_A = < I_A, X_A, Y_A, Xlist_A, Ylist_A, h_A, N_A, \{m_A, n_A\}, C_A, B_A, Z_A, select_A >$$

$\mathbf{Xlist}_A = \{ (1,10); (2,10) \}$; $\mathbf{Ylist}_A = \{ \varnothing \}$. $\mathbf{h}_A = 5$;
$\mathbf{I}_A = < P^X, P^Y >$, donde $P^X = \{ <X(1,10), \text{binario}>, <X(2,10), \text{binario}> \}$; $P^Y = \{ \varnothing \}$;
$\mathbf{N}_A = \{ (0,0), (-1,0), (1, 0), (0,1), (0,-1) \}$;
$\boldsymbol{X}_A = \boldsymbol{Y}_A = \{0, 1\}$; $\mathbf{m}_A = 9$; $\mathbf{n}_A = 10$; $\mathbf{B}_A = \{\varnothing\}$;
$\mathbf{C}_A = \{C_{Aij} \,/\, i \in [1,9], j \in [1,10]\}$; $\mathbf{select}_A = \{ (-1,0), (0,1), (0,0), (0,-1), (1, 0) \}$;
$\mathbf{Z}_A$ se construye usando el mecanismo de acoplamiento del formalismo (Wainer y Giambiasi 1998), y en este caso está definida por:

$$
\begin{array}{ll}
P_{ij}^{Y_1} \rightarrow P_{i,j-1}^{X_1} & \qquad P_{i,j+1}^{Y_1} \rightarrow P_{ij}^{X_1} \\
P_{ij}^{Y_2} \rightarrow P_{i+1,j}^{X_2} & \qquad P_{i-1,j}^{Y_2} \rightarrow P_{ij}^{X_2} \\
P_{ij}^{Y_3} \rightarrow P_{i,j+1}^{X_3} & \qquad P_{i,j-1}^{Y_3} \rightarrow P_{ij}^{X_3} \\
P_{ij}^{Y_4} \rightarrow P_{i-1,j}^{X_4} & \qquad P_{i+1,j}^{Y_4} \rightarrow P_{ij}^{X_4} \\
P_{ij}^{Y_5} \rightarrow P_{ij}^{X_5} & \qquad P_{ij}^{Y_5} \rightarrow P_{ij}^{X_5}
\end{array}
$$

Figura 10. Interconexión de ports para Z_A.

Cada celda en el espacio se describe como:

$$C_{Aij} = < I_A, X_A, S_A, Y_A, N_A, d_{intA}, \delta_{extA}, d_A, \tau_A, \lambda_A, D_A >$$

$\mathbf{I}_A = < 5, P^X, P^Y >$, donde
$P^X = \{ (X_1, \text{binario}), (X_2, \text{binario}), (X_3, \text{binario}), (X_4, \text{binario}), (X_5, \text{binario}) \}$;
$P^Y = \{ (Y_1, \text{binario}), (Y_2, \text{binario}), (Y_3, \text{binario}), (Y_4, \text{binario}), (Y_5, \text{binario}) \}$.
$\boldsymbol{X}_A = \boldsymbol{Y}_A = \{0, 1\}$;
$\mathbf{S}_A$:
<u>Variables descriptivas:</u>

$$s = \begin{cases} 1 \text{ si hay smog en la celda;} \\ 0 \text{ sino.} \end{cases}$$

<u>Parámetros:</u>
viento: demora relacionada con la influencia del viento.
$\mathbf{h}_A = 5$; $\mathbf{d}_A = $ viento;
$\mathbf{N}_A = \{ (0,0), (-1,0), (1, 0), (0,1), (0,-1) \}$;
$\boldsymbol{\tau}_A$: $S_A \times N_A \rightarrow S_A$ se ha definido en la sección previa.

El espacio de celdas C, que se muestra en la siguiente figura, se define como:

$$C = < I_C, X_C, Y_C, Xlist_C, Ylist_C, h_C, N_C, \{m_C, n_C\}, C_C, B_C, Z_C, select_C >$$

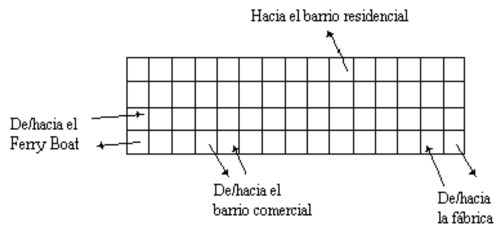


Figura 11. Espacio de celdas C.

$\mathbf{Xlist}_c$ = { (4,14), (4,5), (3, 4) }; $\mathbf{Ylist}_c$ = { (1,10), (4,4), (4,15), (4,1) }.
$\mathbf{I}_C$ = < P^X, P^Y>, donde
P^X = {<X(1,3), binario>,<X(4,14), binario>,<X(4,5), binario>,<X(4,3), binario> };
P^Y = {<Y(1,10), binario>,<Y(4,4), binario>,<Y(4,15), binario>,<Y(4,1), binario> }.

$\mathbf{h}_C$= 9; $\boldsymbol{X}_C$ = $\boldsymbol{Y}_C$ ={0, 1}; $\mathbf{m}_C$ = 4; $\mathbf{n}_C$ = 15; $\mathbf{B}_C$= {∅};
$\mathbf{N}_C$ = { (-1,-1), (-1,0), (-1,1), (0,-1), (0,0), (0,1), (1,-1), (1, 0), (1, 1) };
$\mathbf{C}_C$ se define como en el ejemplo anterior.
$\mathbf{S}_C$:
Variables descriptivas:

$$s = \begin{cases} 1 \text{ si hay un vehículo en la celda;} \\ 0 \text{ sino.} \end{cases}$$

Parámetros:
VelocidadH: demora relacionada con la velocidad de los vehículos.
$\mathbf{Z}_C$ también se define como en el ejemplo anterior.

$\mathbf{select}_C$ = { (0,1), (-1,1), (1,1), (0,0), (-1,0), (1,0), (-1,-1), (0,-1), (1,-1) };

La definición para el modelo B es similar que en los casos anteriores. Finalmente, consideremos la especificación formal para el modelo completo. En este caso,

$$M = < X, Y, D, \{M_i\}, \{I_i\}, \{Z_{ij}\}, select >$$

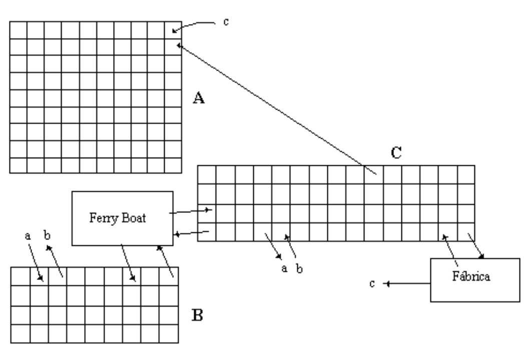


Figura 12. Estructura del modelo M.

X = ***Y*** = $\{\emptyset\}$;
D = { A, B, C, D, E }, y $\forall i \in$ D, M_i es uno de los modelos básicos definidos previamente.
$\mathbf{I_i}$ es el conjunto de influencias del modelo i:
$I_A = \{\emptyset\}$; I_B = { C, E }; I_C = { A, B, D, E }; I_D = { A, C }; e I_E = { B, C }.

$\mathbf{Z_{ij}}$ se define como:
Z_{BC}: $Y(1,3)_B \rightarrow X(4,5)_C$
Z_{BE}: $Y(1,9)_B \rightarrow IN_f$
Z_{CA}: $Y(1,10)_C \rightarrow X(2,10)_A$
Z_{CB}: $Y(4,4)C \rightarrow X(1,2)_B$
Z_{CD}: $Y(4,15)_C \rightarrow IN_f$
Z_{CE}: $Y(4,1)_C \rightarrow IN_b$
Z_{DA}: $OUT_f \rightarrow X(1,10)_A$
Z_{DC}: $OUT_f \rightarrow X(4,14)_C$
Z_{EB}: $OUT_b \rightarrow X(1,7)_B$
Z_{EC}: $OUT_b \rightarrow X(3,4)_C$

Finalmente, **select** = { C, A, B, D, E}.

5. CONCLUSION

En este trabajo se muestra la aplicación de un nuevo paradigma de especificación para modelización y simulalación de espacios de celdas. El paradigma Cell-DEVS permite especificar comportamientos complejos de forma simple, independientemente de la complejidad cuantitativa de los modelos.

Los métodos permiten la definición automática de espacios de celdas en base al formalismo DEVS. Se pueden integrar múltilples puntos de vista para cada submodelo, permitiendo integrar diversos paradigmas de modelización de forma eficiente.

En la actualidad se han implementado dos herramientas de simulación basadas en el formalismo, que ejecutan secuencialmente. En estos casos, la obtención de una muestra significativa que permita obtener resultados confiables, es un proceso que consume tiempo, además que modelos inherentemente paralelos deben ejecutarse serialmente. Para solucionar estos problemas, se están integrando las herramientas desarrolladas junto con un entorno de simulación paralela, permitiendo, de esta forma, logran mayor grado de aceleración en las simulaciones. Todas las herramientas son de dominio público y pueden obtenerse en: http://www.dc.uba.ar/people/profesores/wainer

REFERENCIAS

Giambiasi, N., Miara, A. SILOG: A practical tool for digital logic circuit simulation. Proceedings of the 16th. D.A.C., San Diego, U.S.A. (1976).

Wainer, G.; Giambiasi, N. Specification, modelling and simulation of timed Cell-DEVS models. Technical Report TR-98-006, Departamento de Computación, FCEN/UBA. Enviado para publicación. (1998).

Wolfram, S. Theory and applications of cellular automata. Vol. 1, Advances Series on Complex Systems. World Scientific, Singapore (1986).

Zeigler, B. Theory of modeling and simulation. Wiley (1976).