

# Autómatas Celulares Aplicados en los Videojuegos

Univ. José Luis Zamora Guarachi  
 Universidad Mayor de San Andrés  
 Carrera de Informática  
 Teoría de la Información y Codificación  
 argosjlp@hotmail.com

## RESUMEN

Sin duda los videojuegos se consolidaron como el entretenimiento favorito para buena parte de la juventud, llegando a popularizarse de manera significativa.

Es por eso que en el afán de ofrecer videojuegos cada vez más novedosos, con graficas más reales, con mayor velocidad de ejecución, etc. que se busca la utilización de nuevas tecnologías en esta área. En general, simular fenómenos físicos como explosiones, fluidos y fuego, en forma realista es demasiado costoso numéricamente como para que se pueda hacer en tiempo real en un videojuego. Sin embargo, hay una técnica que permite obtener aproximaciones visualmente plausibles utilizando los llamados Autómatas Celulares.

## Palabras clave

Videojuego, autómatas, explosiones, grafos, impulso, personaje, desarrollo.

## 1. INTRODUCCION

Se puede utilizar autómatas celulares en aplicaciones practicas de diseño grafico para simular ciertos efectos (blur, emboss, detección de bordes y emulación de estilos de pintor, etc.), así como efectos gráficos tipo “plasma”, como nubes de vapor y efectos especiales utilizados en películas como **Final Fantasy**, **Matrix**, **El Señor de los Anillos**, también hay aportaciones en la generación de sonidos de instrumentos digitales y en la música sintética.

## 2. ¿QUE ES UN AUTOMATA CELULAR?

Los autómatas celulares son sistemas dinámicos discretos en el espacio y en el tiempo, cuyas variables pueden representar un número finito de estados. Son el arquetipo de sistemas discretos; sistemas complejos en los que emergen propiedades colectivas y fenómenos de autoorganización global a partir de interacciones locales muy simples.

Autómata Celular se puede pensar como una grilla cuyas celdas tienen asociado uno o varios valores. Cada célula tiene vecinos, generalmente 4. Al principio, se asignan valores iniciales, y luego se va haciendo evolucionar el AC siguiendo una regla matemática predefinida. Cada célula cambia de estado teniendo en cuenta su estado anterior y el de sus vecinos, de forma que todas las células del espacio celular cambian según la misma función de transición y que el cambio de estado es simultáneo en todo el espacio celular.

Con este esquema general se pueden simular muchos fenómenos biológicos y físicos interesantes.

Existen autómatas celulares de diversos tipos, pero todos deben tener:

- ⇒ Un espacio bidimensional o n-dimensional dividido en un número de subespacios homogéneos, llamados celdas.
- ⇒ Cada celda puede estar en uno de un conjunto finito de estados.
- ⇒ Una Configuración inicial que consiste en asignarle un estado inicial a cada celda del autómata.
- ⇒ Una Vecindad definida para cada celda, o definir las leyes que indican si una celda es contigua a otra, indicando sus posiciones relativas.
- ⇒ Una Regla de Evolución, o cómo debe cambiar de estado cada celda dependiendo de circunstancias verificables.
- ⇒ Un Reloj Virtual que generará pulsos que indican cuando debe establecerse los cambios de estado de las celdas según las Reglas de Evolución.

Es necesario hacer notar el gran interés que suscitan los autómatas celulares en la sociedad científica actual, desde que **Conway** en los años 70 describiera en un precioso programa (**figura 1**) el funcionamiento de un autómata celular, hasta las teorías existencialistas que se tejen en la actualidad en base a estos, al descubrirse su intrínseca presencia en temas tan diversos como la pigmentación, la astrofísica, la medicina y la biología, entre otras.

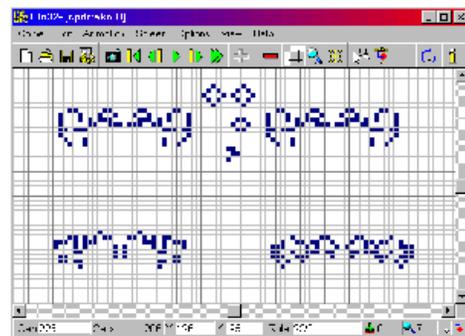


Figura 1.-pantalla del programa life 32 de Conway

## 3. GRAFOS DINAMICOS

Un grafo es un conjunto de **Vértices**, unidos mediante conexiones llamadas **Aristas**. Tanto los Vértices como las Aristas pueden tener propiedades asignadas.

Usando ese concepto general, se puede generalizar un Autómata Celular. El ejemplo básico de una grilla 2D, se traslada a un grafo en que cada Vértice tiene una posición (lo que originalmente era una coordenada en la grilla) y cuatro Aristas que conectan al Vértice con sus vecinos.

En el esquema básico de un grafo que reemplaza a una grilla (Figura 2). Cada vértice de las paredes está conectado con un vértice de la pared opuesta, para crear un "mundo infinito". Todo lo que "sale" por la derecha "entra" por la izquierda. Al igual que un mapa de Karnaght.

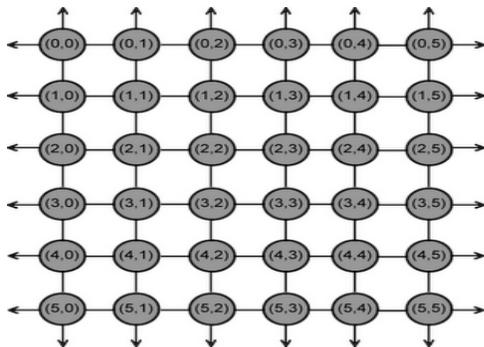


Figura 2.-grafo en forma de grilla

Este esquema es más general en varios sentidos. Primero, permite desacoplar fácilmente el espacio "real" de nuestro Autómata Celular, de esa forma ahora cada nodo del grafo puede estar en **posiciones arbitrarias** del espacio y, en particular, ubicarlos en forma no uniforme.

Se puede entonces simular con más detalle la física en algunas partes que en otras. Solamente es cuestión de ser cuidadoso a la hora de crear nuestro algoritmo. Otra virtud que tiene este esquema es que se pueden generar **obstrucciones** en el grafo para simular obstrucciones reales.

Simular obstrucciones es solo cuestión de romper Aristas o Vértices. Se puede relacionar un grafo con un mapa de nuestro juego (figura 3). En esquemas más complicados se pueden mantener todos los Vértices y Aristas y asignar pesos a las Aristas. De esta forma se pueden simular distintos tipos de "rigidez" o incluso destrucción de paredes.

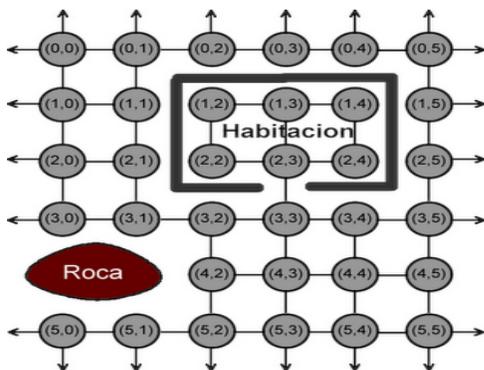


Figura 3.-grafo con obstrucciones

#### 4. SIMULANDO EXPLOSIONES

Para simular una explosión se necesita un grafo que tenga definida una presión en cada Vértice. En cada paso de tiempo, se mira Vértice por Vértice y se compara la presión del Vértice con la presión de sus vecinos. Luego se calcula un flujo de "aire" entre el Vértice y el vecino y se lo utiliza para modificar sus presiones.

El esquema es básico e intuitivo: El nodo que tiene más presión le entrega un poco a cada uno de los vecinos que tienen menos. Y así, paso por paso, se genera una fuente de presión.

Gráficamente se vería la siguiente reacción (figura 4, 5, 6 y 7):

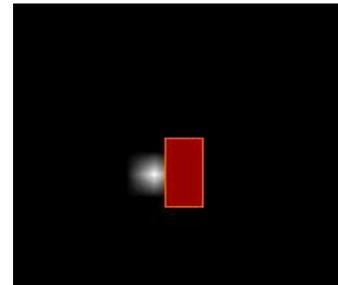


Figura 4.- Explosión al lado de un obstáculo pequeño. t=1.

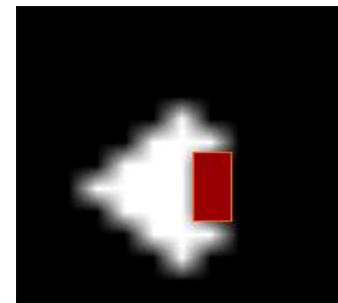


Figura 5.-Explosión al lado de un obstáculo pequeño t=10.

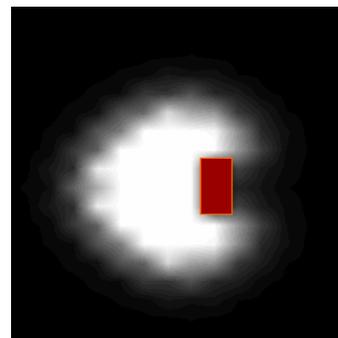


Figura 6.-Explosión al lado de un obstáculo pequeño t=40.

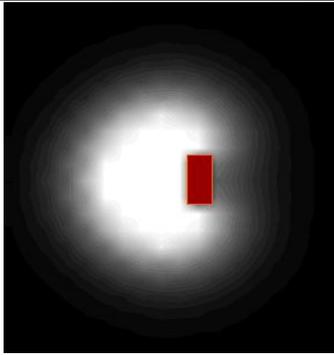


Figura 7.-Explosión al lado de un obstáculo pequeño  $t=100$ .

Sin duda esto optimiza el flujo y optimiza los efectos, creando además un efecto más vistoso que el que se podría obtener usando las antiguas técnicas de diseño (figura 8). Que si bien lograban obtener efectos gráficos aceptables, necesitaban más recursos.



Figura 8.- Los videojuegos “antiguos” necesitaban más recursos para conseguir efectos de calidad aceptable.

### 5. TRASLADANDO LOS EFECTOS DE LA EXPLOSION AL JUEGO

Supongamos que el objeto que nos interesa se encuentra cerca de la explosión. Podemos calcular el vector de flujo sumando los vectores de flujo entre un Vértice y sus vecinos. En el ejemplo de la imagen siguiente, el vector de flujo sobre el nodo en que se encuentra el personaje tiene dirección sur oeste. Ese flujo, multiplicado por algún valor que represente la sección de área del personaje, nos indica una **fuerza** que actúa sobre el personaje.

Es interesante notar que en una explosión, todo sucede muy rápido en los primeros instantes de tiempo. En un juego uno podría simular decenas de pasos de un autómata celular por cada frame (capa de acción determinado por un lapso de tiempo) del juego y utilizar un promedio en el tiempo del vector de flujo. Este promedio es el **impulso** a transferir a los objetos

del juego en cada frame. De esta forma se puede simular el efecto medio de la onda expansiva en los jugadores, objetos ubicados en el mapa y el mapa mismo.

También se puede sumar las magnitudes de flujo, para simular efectos como sordera momentánea, ruptura de vidrios, o daño general.

### 5.1 Cómo trasladar el efecto de la onda expansiva al personaje?

En el siguiente gráfico (figura 8) las flechas azules son los vectores de flujo entre el nodo del personaje y sus vecinos. La flecha roja es la suma de los vectores azules, que multiplicadas por el cross-section del personaje dan la **fuerza** que actúa sobre el personaje debido a la explosión. Promediando esta fuerza durante varios pasos del autómata celular se obtiene el **impulso** sobre el personaje.

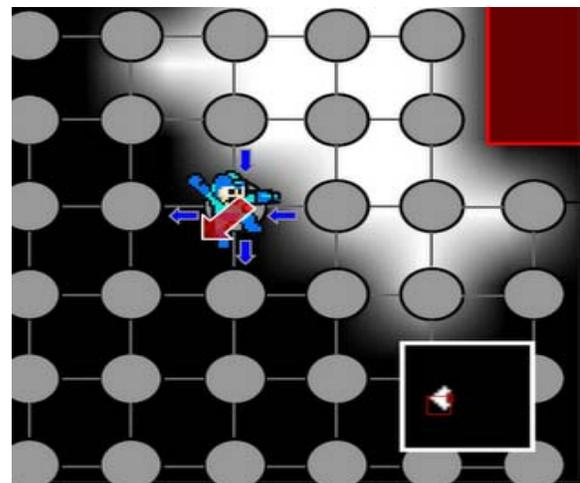


Figura 9.- Sumando las magnitudes de flujo se puede simular las explosiones en conjunción con el personaje.

### 6. CONCLUSION

Dada la gran versatilidad que tienen los autómatas celulares se los puede poner en uso en un campo tan vanguardista como son los videojuegos.

Aplicando conceptos matemáticos en un autómata celular se puede simular fenómenos físicos interesantes como fuego y explosiones entre otras.

El uso de estas tecnologías llevan a disminuir el uso de los recursos computacionales, consiguiendo mejores efectos y gráficos, disminuyendo el tiempo de ejecución, sin duda optimizando un videojuego (figura 9, 10), Lo cual es sin duda uno de los objetivos primordiales de la informática. La búsqueda de la optimización en cualquier área.



Figura 9.-



Figura 10.-Los videojuegos “modernos” ya utilizan esta tecnología para simular efectos como fuego, humo y explosiones.

## 7. REFERENCIAS

[1] Cordova A. 2008. “*Herramientas para sistemas complejos: autómatas celulares y algoritmos autoorganizativos*” Departamento de Física de la Materia Condensada. Universidad de Sevilla

[2] Pozzo E. 2009, “*Autómatas Celulares para la simulación de explosiones en los videojuegos*” Artículo.

- [3] Pozzo E. , 2009, “*MVC en la creación de videojuegos*” Artículo.  
<http://ezequelpozzo.blogspot.com/2009/11/mvc-en-la-creacion-de-videojuegos.html>

[4] Hooft G., (premio Nobel de fisica),“*El universo es un autómata celular y la realidad es la interpretación de una compleja computadora*”, Artículo:

<http://axxon.com.ar/noticias/automata-celular.html>