Capítulo 1
INTRODUCCIÓN

En el presente Capítulo tendremos como objetivo principal una introducción al concepto de autómata celular. Para ello, y tras una primera definición, mencionaremos brevemente a los distintos enfoques que de este tópico, ya clásico en la Informática Teórica y en la Matemática Aplicada, pueden darse. Veremos la gran diversidad de comportamientos que soportan estos constructos, y se planteará la necesidad de clasificarlos. En particular, y esto será de gran interés para nosotros, se muestran capaces de generar números aleatorios. Igualmente introduciremos las nociones básicas de Criptografía que serán necesarias para abordar una primera lectura, ampliándolas a lo largo del texto cuando sea necesario. Puesto que el objetivo de esta Memoria de Tesis Doctoral es realizar la síntesis de dos paradigmas: Autómatas Celulares y Criptografía, indicaremos en qué estado se hallan actualmente las investigaciones en esta línea. Finalmente, presentaremos las líneas de trabajo que sigue la Memoria, en términos de los objetivos a lograr, de las hipótesis que nos hemos planteado para ello y de la metodología que hemos seguido para constrastar tales hipótesis.

1.1. Qué son los autómatas celulares

En su libro Cellular Automata Machines [Tof91], *Toffoli* y *Margolus* definen los autómatas celulares como "...universos sintéticos y estilizados... Tienen su propia clase de materia, que se arremolina en un espacio y un tiempo que le son propios". La descripción que efectúan del concepto que va a ser objeto de nuestro estudio, si bien imprecisa por cuanto que es abstracta, nos proporciona las claves básicas para, en el contexto adecuado, comprender qué son y para qué sirven estos modelos.

En la naturaleza resulta muy común encontrar sistemas cuyo comportamiento global es extremadamente complejo, a pesar de que sus elementos constituyentes son muy simples. El factor clave que contribuye a generar la complejidad es la cooperación entre dichos componentes. Sin embargo, habiéndose puesto al descubierto un enorme *corpus* de conocimiento sobre la naturaleza de los componentes constituyentes de los sistemas físicos y biológicos, es muy poco lo que sabemos sobre los mecanismos a través de los cuales estos componentes actúan unidos para dar lugar a la complejidad global que se observa[*].

Los autómatas celulares son ejemplos de sistemas matemáticos compuestos de

[*] Piénsese por ejemplo en la actividad más compleja que se conoce: la actividad racional de un cerebro humano. A pesar de los avances logrados en campos como la fisiología neuronal, seguimos sin saber cómo se organizan las neuronas para dar lugar a un pensamiento consciente.

múltiples componentes idénticos, los cuales, siendo muy simples individualmente, dan lugar cuando actúan juntos a comportamientos globales muy complejos. Su estudio nos permite, por una parte, derivar modelos específicos de sistemas particulares y por otra, tener la esperanza de formular principios generales aplicables a sistemas complejos ya conocidos en Física, Química o Biología, que nos expliquen cómo surge el comportamiento autoorganizado que observamos.

Un repaso histórico nos muestra que el estudio de los sistemas autoorganizados se ha basado siempre en las ecuaciones diferenciales de *Boltzmann* [Tip86], que describen la variación en el tiempo de cantidades macroscópicas. Las ecuaciones se obtienen sobre un grupo de estados microscópicos asumiendo que las colisiones sucesivas entre moléculas no guardan correlaciones estadísticas.

Investigaciones más generales sobre la autoorganización en sistemas dinámicos utilizan modelos matemáticos alternativos, como las ecuaciones de *Navier-Stokes* [Tip86] o sus variantes, donde la evolución en el tiempo del sistema a partir de unas condiciones iniciales viene descrita por una trayectoria en el espacio de estados de las variables. En los casos más simples, todas las trayectorias tienden a converger a puntos límite aislados, o a ciclos simples. En otros casos, las trayectorias pueden concentrarse sobre superficies complicadas aparentemente caóticas conocidas como atractores extraños.

Los modelos basados en autómatas celulares proporcionan un enfoque alternativo, ya que trabajan con variables y tiempo discretos. Exhiben comportamientos complejos análogos a los encontrados en los modelos diferenciales, pero debido a la simplicidad de su construcción nos permiten realizar sobre ellos un análisis más completo y detallado.

1.1.1. Concepto de Autómata Celular

Consideremos ahora el anillo residual de los enteros módulo k, dado por $\mathbb{Z}_k = \{0, 1, ..., k-1\}$. Veamos cómo definir un autómata celular cuyos componentes toman valores sobre él.

Definición 1 *Un autómata celular es una sucesión de elementos de* $(\mathbb{Z}_k)^{\mathbb{Z}}$. *Cada elemento se escribe como*

$$(..., a_{i-1}^{(t)}, a_i^{(t)}, a_{i+1}^{(t)}, ...)_{t \in N}$$

Cada coordenada $a_i \in \mathbb{Z}_k$ *es denominada célula. El t-ésimo elemento de la sucesión indica el estado del autómata en el instante t del tiempo. Este elemento está determinado por el elemento de la sucesión que le precede y una función de transición de modo que, para todo i*

$$a_i^{(t+1)} = \phi\left(a_{i-r}^{(t)}, a_{i-r+1}^{(t)}, ..., a_{i+r}^{(t)}\right) \tag{1.1}$$

Se dice que el autómata celular tiene rango de vecindad r.

En la práctica los autómatas celulares se componen de un número finito de células, fijándose por tanto una longitud $m \in \mathbb{N}$ de modo que cada elemento de la sucesión que define al autómata es de la forma

$$(..,0, ..., 0, a_1^{(t)}, a_2^{(t)}, ..., a_m^{(t)}, 0, ..., 0...)_{t \in N}$$

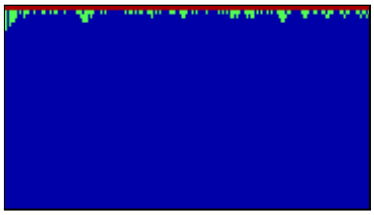

Figura 1 Autómata celular de clase 1

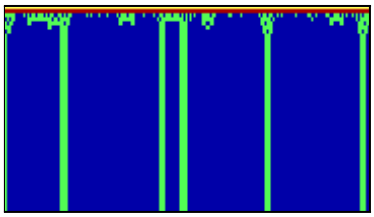

Figura 2 Autómata celular de clase 2

Incluso con valores paramétricos tan simples como $k = 2$ y $r = 1$ el comportamiento global ofrecido por un autómata celular puede llegar a ser extremadamente complejo.

Una vez establecidos los parámetros k y r, podemos en principio determinar cualquier función de transición ϕ, que establecerá el mecanismo evolutivo del autómata. Si entonces consideramos el patrón generado a partir de una configuración inicial establecida aleatoriamente podemos observar distintos comportamientos según sea la función ϕ elegida.

Un estudio empírico ya propuesto en algunas referencias [Wol94], y razonablemente exhaustivo, que expondremos a lo largo del Capítulo, parece sugerir que los patrones obtenidos siempre se sitúan en una de cuatro clases cualitativas.

Criterio 2 *Criterio cualitativo de clasificación de autómatas celulares*

1. El patrón desaparece con el tiempo. Es espacialmente homogéneo (Figura 1).
2. El patrón evoluciona de forma invariante. Aparecen estructuras periódicas o estables (Figura 2).
3. El patrón se comporta de forma caótica y no periódica (Figura 3).
4. El patrón crece y se contrae irregularmente, y pueden aparecer estructuras complejas muy localizadas que en ocasiones se propagan (Figura 4).

Los patrones de clase 3 son en ocasiones invariantes bajo escala, de modo que partes de los mismos, cuando son ampliadas, son indistinguibles del original. En estas ocasiones los patrones están caracterizados por tener una dimensión fractal [Man87] siendo el valor 1,59 la dimensión más común. De hecho, muchos patrones invariantes bajo escala presentes en sistemas naturales han sido generados mediante autómatas celulares.

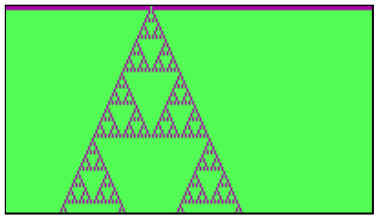

Figura 3 Autómata celular de clase 3

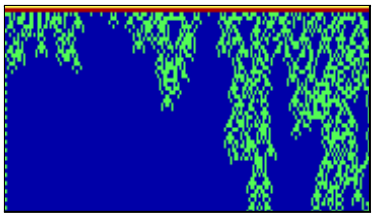

Figura 4 Autómata celular de clase 4

Las Figuras 5 a 7 muestran la evolución de varios autómatas celulares desde estados iniciales aleatorios, donde a cada célula se le asigna en el estado inicial un valor de $k = 5$ posibles con probabilidades iguales e independientes. El fenómeno de autoorganización aparece en algunos casos, generándose estructura a partir de estados iniciales desordenados, e incluso alcanzando gran complejidad en ocasiones.

Debe precisarse que la elección de la configuración inicial, siempre que se haga en forma aleatoria, no afecta en demasía al patrón global del autómata, ni a sus propiedades estadísticas, y sólo tiene una importancia relativa a nivel puramente local.

1.1.2. Relaciones con Otros Campos del Conocimiento

Los autómatas celulares han sido estudiados desde múltiples ópticas, y ha sido posible la consecución de resultados que lo relacionan con varios campos del conocimiento como son: la Matemática, donde se ha encontrado que sus conjuntos de configuraciones forman un conjunto de *Cantor* [Wol94], pasando a tener sentido sobre ellos medidas como los exponentes de *Lyapunov*; en la Teoría de Lenguajes Formales se demuestra que el conjunto de las configuraciones son un lenguaje formal que puede ser descrito por gramáticas encuadradas en la Jeraquía de *Chomsky*; se ha encontrado que conforme más complejo es el autómata celular, hemos de recurrir a un nivel superior de la jerarquía para su descripción. Como mecanismo de procesamiento

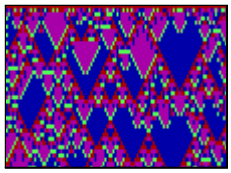

Figura 5 Autómata con 5 estados de clase 3

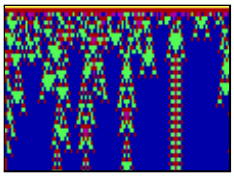

Figura 6 Autómata con 5 estados de clase 4

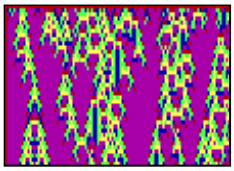

Figura 7 Autómata con 5 estados de clase 4

de información, se puede demostrar que los autómatas celulares definen un concepto de *función computable* que se enmarca en el ámbito que la *Tesis de Church* [Dav92] define. Incluso hay instancias concretas del modelo que realizan la más sofisticada función que cualquier máquina abstracta puede alcanzar: la computación universal.

1.1.3. Aplicaciones

Con objeto de encaminar la exposición de lo abstracto a lo concreto, realizaremos en este apartado un breve comentario sobre algunas posibles aplicaciones de los autómatas celulares en determinados campos. Ya hemos comentado cómo el autómata celular es una alternativa legítima para modelar sistemas naturales descritos hasta ahora por modelos basados en ecuaciones diferenciales. En sistemas lineales, el autómata celular puede ser el complemento perfecto a la descripción diferencial, mientras que en sistemas no lineales con variación brusca de determinadas variables, como son los químicos y los biológicos son incluso más apropiados para una descripción pormenorizada.

Es posible encontrar autómatas celulares que proporcionan un modelo del crecimiento de cristales dendríticos[***], como puede ser el caso de los copos de nieve. Comenzando desde una semilla simple, las células cuyos valores representan la fase sólida se van agrupando de acuerdo con una función de transición dos-dimensional.

Sistemas químicos no lineales, como algunas reacciones químicas, son también descritos por autómatas celulares que capturan las características principales de las ecuaciones en derivadas parciales que los definen, reproduciendo en una pantalla los mismos patrones reactivos que se observan bajo un microscopio.

Fenómenos como la turbulencia en el seno de un fluido, pueden a su vez ser modelados por autómatas celulares que describen las interacciones locales entre vórtices próximos dispuestos sobre las células de un autómata dos-dimensional.

Si a la función de transición se la somete a ruido[****] durante la evolución

[***] Los cristales crecen en asociación paralela y forma arborescente. (Diccionario Enciclopédico Planeta, Tomo 3. Planeta, 1984).

[****] Por supuesto, probabilísticamente hablando.

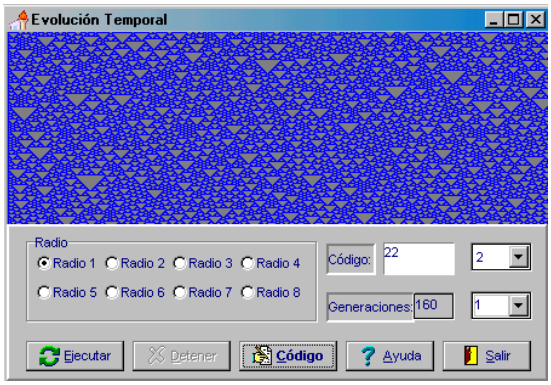

Figura 8 Evolución de la regla 22

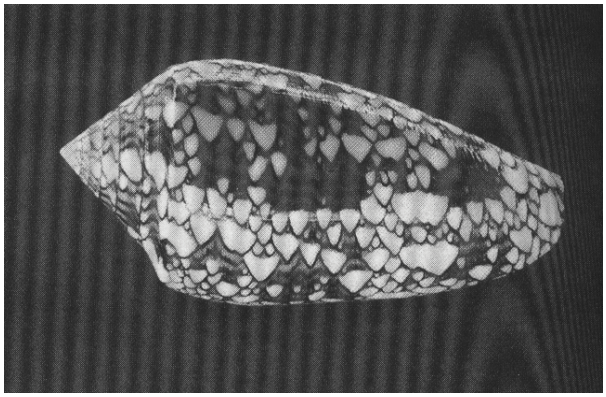

Figura 9 Pigmentación de un molusco univalvo

temporal, entonces podemos identificar el autómata celular con el modelo generalizado de *Ising*.

Interpretaciones próximas a las Ciencias de la Computación pueden ser también realizadas, viendo al autómata como un procesador de información, que toma un dato codificado en la configuración inicial, y lo modifica en función de un programa: su función de transición.

Finalmente, y aproximándonos a los sistemas biológicos, encontramos una amplia variedad de sistemas que han sido modelados mediante nuestro objeto de estudio. En concreto, mecanismos de formación de patrones como el de pigmentación de las conchas de determinados molucos han sido modelados con éxito mediante autómatas de las clases 2 y 3. Veáse al respecto la Figura 9 y compárese con el patrón evolutivo simulado por computadora de la Figura 2. ¿Está siguiendo el molusco para su patrón pigmentario una ley descrita por el autómata celular con código[**] 22? Respuestas a esta clase de preguntas aún no están disponibles hoy día, pero resulta tentador considerar la alternativa positiva.

[**] En el siguiente Capítulo se establecerá de forma rigurosa en qué consiste tal codificación.

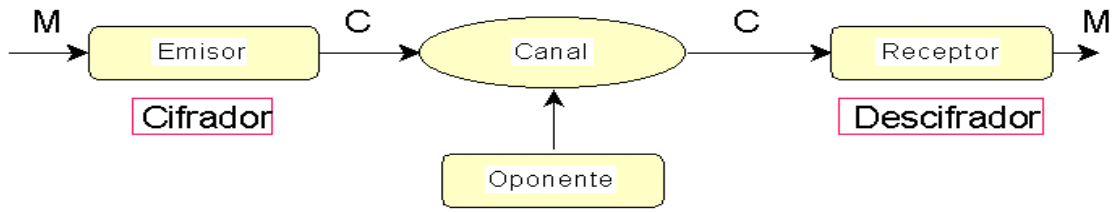

Figura 10 Esquema de una Cifra de Llave Privada

1.2. Principios Básicos de Criptografía

Dado que el propósito último de la presente Memoria de Tesis Doctoral es realizar la síntesis entre el modelo de los autómatas celulares y algunos protocolos criptográficos, avanzamos a continuación algunas definiciones criptográficas básicas.

Desde que el ser humano fue capaz a través de la escritura de plasmar su pensamiento sobre el papel, la preocupación por el problema del ocultamiento de la información ha estado presente. Por ello, no es exagerado decir que la ciencia de la Criptografía y su evolución, han corrido paralelas al resto de los avances que el hombre ha realizado en otros campos del conocimiento, si bien su relación con la Matemática ha sido, y aún continúa siendo, privilegiada.

1.2.1. Definición de Criptosistema

Realizamos a continuación las definiciones que serán necesarias en el transcurso del trabajo. La Figura 10 ilustra los elementos principales de un sistema de comunicación con cifrado. De acuerdo con ella, podemos establecer la definición de los siguientes agentes que intervienen en ella.

El *emisor* es el agente que envía información a través de un canal inseguro, y que se ve obligado a cifrarla para su protección. El *receptor* es el agente que recibe información cifrada a través de un canal inseguro, y debe descrifrarla para acceder a ella. El *oponente* es un agente con acceso al canal de información y cuyo propósito es conocer la información que circula por él. El *canal* es el medio a través del cual circula la información. El mensaje (M) es la información que se envía del emisor al receptor. El *texto llano* es la representación del mensaje en algún formato antes de ser cifrada. El *texto cifrado* (C) es la representación del mensaje en algún formato después de ser cifrada. También se le llama *criptograma*.

Un *criptosistema* es un método que permite cifrar y transmitir información de manera segura por un canal inseguro. Matemáticamente, un criptosistema no es más que una función que aplicada sobre el texto llano permite obtener el texto cifrado. Naturalmente, existe una transformación inversa que permite recuperar la información original al receptor.

En este punto es necesario distinguir entre cifrados de *clave privada* (simétricos) y de *clave pública* (asimétricos). Los primeros corresponden al modelo descrito en la Figura 10, y son aquellos en que la llave del cifrado debe ser secreta y conocida únicamente por emisor y receptor. Los segundos corresponden a un modelo diferente, donde se dispone de dos llaves distintas para cifrar (llave pública) y descrifrar (llave

Figura 11 Esquema de una Cifra de Llave Pública

privada). En la Figura 11 se aprecia el esquema de una cifra de llave pública. Como vemos, el emisor utiliza una clave para cifrar accesible a todo aquél usuario que desee enviar un mensaje codificado al usuario B. Éste, por su parte, utiliza para descifrar una clave (secreta) que sólo él conoce.

Todos los cifrados desarrollados en esta Tesis serán de clave privada (simétricos).

Finalmente, damos dos definiciones que establecen el grado de seguridad de los criptosistemas.

Definición 3 *Un criptosistema es incondicionalmente seguro si un atacante con tantos recursos como necesite en términos de tiempo y potencia de cálculo es incapaz de romper el cifrado.*

Definición 4 *Un criptosistema es computacionalmente seguro [Kel99] si un atacante con los mejores métodos y técnicas de descifrado por computadora* ***actualmente disponibles*** *es incapaz de romper la cifra.*

1.2.2. Requerimientos Habituales de un Criptosistema

- El método de cifrado y descifrado debe ser rápido y fiable.
- No debe existir retardo debido al cifrado o descifrado.
- La seguridad del sistema deberá residir solamente en el secreto de una clave y no de las funciones de cifra.
- La fortaleza del sistema se entenderá como la *imposibilidad computacional* de romper la cifra o encontrar la clave secreta.

1.3. Estado Actual del Tema

El campo de conocimiento que trata la Criptografía es prácticamente tan antiguo como la escritura, y ha evolucionado junto con el resto de los desarrollos teóricos que el hombre ha ido desarrollando. Por su parte, los autómatas celulares son introducidos por *John Von-Neumann* en 1945 y abandonados en su estudio hasta principios de los años 80, cuando *S. Wolfram* retoma la investigación sobre los mismos.

Desde entonces, este área de las Ciencias de la Computación no ha conocido descanso, y se ha investigado desde muchas y muy diferentes ópticas, desde el ámbito

estrictamente teórico de los modelos de Computabilidad, hasta el eminentemente práctico y aplicado de la Criptografía, tema que nos ocupa.

El estado actual del tema, en cuanto al desarrollo de protocolos de cifrado con autómatas celulares es de carácter primario, en el sentido de que la literatura recoge un autómata celular concreto, presentado por *Wolfram* [Wol91] en 1985, con el cual se implementa un sistema de cifrado en flujo muy simple con base en la función lógica XOR (OR exclusiva). Nosotros realizamos en su día [Tom96] el análisis de la misma y propusimos un sistema de cifrado con base en ella, si bien en la actualidad pretendemos el desarrollo de funciones alternativas que añadan complejidad al cifrado (Veáse el Capítulo 4 al respecto).

Sin embargo, el desarrollo de *Wolfram* quedó en un plano puramente teórico, sin implementarse algoritmos ni circuitos de cifrado construidos en la práctica. En el presente trabajo se desarrollarán ambas líneas.

Por su parte, el cifrado de bloque con autómatas celulares ha conocido una primera aproximación con *Howard Gutowitz*, que ha realizado un sistema de cifrado de estas características [Gut93] aunque con base en principios teóricos radicalmente diferentes al basado en las S-cajas que nosotros propugnamos. La elección de estas no es casual, pues han mostrado su eficacia en estándares de cifrado actualmente en explotación comercial como **DES** (**D**ata **E**ncypherment **S**tandard).

1.4. Líneas de Trabajo

1.4.1. Presentación de Objetivos

Tal y como se hizo constar en el Proyecto de Tesis Doctoral, admitido a trámite con fecha 10 de Septiembre de 2001, el objetivo fundamental de la Tesis Doctoral propuesta es el desarrollo, análisis, diseño e implementación (tanto *hardware* como *software*) de sistemas criptográficos con base en el modelo de cálculo teórico conocido como "autómatas celulares". El problema del hallazgo de sistemas de cifrado criptográficamente seguros es uno de los más interesantes y abiertos que tienen planteados conjuntamente las Ciencias de la Computación y la Matemática Aplicada.

Sin embargo, el enfoque orientado al campo de los autómatas celulares es relativamente novedoso, al tiempo que proporciona un espacio de desarrollo amplio y de múltiples posibilidades. Así, en primera instancia, nos planteamos el análisis del espacio de los autómatas celulares binarios con el objetivo inicial de encontrar *generadores seudoaleatorios* capaces de superar las baterías de tests de aleatoriedad estándar (Capítulo 3). El siguiente paso será el desarrollo de funciones de cifrado orientadas al bit (también conocidas como *cifrados de flujo)*, y la construcción de sistemas de cifrado simétricos con base en ellas y en los autómatas celulares previamente identificados, probando su resistencia frente al criptoanálisis y desarrollando un *software* de propósito específico para cifrar información (Capítulo 4). Se procederá a desarrollar también protocolos adicionales de cifrado de mayor rango, no orientados ya al bit, sino al bloque. Para ello se utilizárán técnicas como las **S**-cajas, habituales en esta clase de cifrados, y se cifrará la información empleando múltiples autómatas celulares que combinen linealidad y ausencia de ella (Capítulo 5).

Concluida la Tesis Doctoral se habrán cubierto los siguientes objetivos:

Identificación de autómatas celulares con secuencias seudoaleatorias
Diseño de funciones de cifrado orientadas al bit
Diseño de un protocolo de cifrado orientado al bloque
Criptoanálisis de los sistemas de cifrado anteriores
Implementación *software* de los sistemas de cifrado indicados
Implementación *hardware* de algunos de los sistemas de cifrado indicados

1.4.2. Hipótesis Planteadas

Las hipótesis por tanto planteadas, y que deberán ser objeto de demostración o refutación durante la realización de la Tesis Doctoral son las siguientes:

1. Existencia de autómatas celulares que generen secuencias seudoaletorias de bits. Con base en trabajos previos publicados por nosotros, desarrollaremos una búsqueda, a través del espacio de reglas de los autómatas celulares, para identificar aquellas que se comporten como buenos generadores de números aleatorios. Las validaremos con base en los tests de aleatoriedad estándares en el Capítulo 3. El resultado, una serie de tablas que demuestran, para unos determinados valores parámetricos de k y r, la existencia de autómatas celulares con comportamientos aleatorios. Se incluyen medidas de entropía y distancia de *Hamming*.

2. Existencia de funciones de cifrado orientadas al bit criptográficamente seguras. La literatura da noticia de una de estas funciones muy simples (regla 30), ya que solo trabaja con un nivel de puertas lógicas. Nuestro propósito es proceder a la propuesta de funciones más elaboradas y por lo tanto, criptográficamente más seguras. En el Capítulo 4 expondremos la metolodogía general a seguir para ello, con base en la regla 30, para mostrar cómo debería aplicarse a los autómatas celulares hallados en el Capítulo 3.

3. Existencia de sistemas de cifrado de bloques criptográficamente seguros. Nuevamente existe una propuesta muy base netamente mejorable, que utilizaremos de base para desarrollar un protocolo de cifrado de bloque más elaborado.

4. Viabilidad técnica para desarrollar un *software* de cifrado en base a lo anterior.

5. Viabilidad técnica para desarrollar un *hardware* de cifrado.

1.4.3. Metodología de Contraste de Hipótesis

Para contrastar la hipótesis número uno utilizaremos la simulación *software* de los autómatas celulares en combinación con una batería de test de aleatoriedad estándares tales como Chi-cuadrado, de huecos, de distancias, de rachas y otros. De trabajos previos que han sido publicados hemos derivado el *software* de simulación, mientras que los test de aleatoriedad son programables a partir de las definiciones teóricas de [Knu67]. Dada la infinitud del espacio de búsqueda, la consecución de autómatas que actuén aleatoriamente es segura.

La hipótesis número dos necesita de una labor previa de síntesis, aplicando los principios clásicos del Álgebra de Boole y del diseño combinacional estándar. Construir funciones es tan simple como formularlas, y eso haremos. Para contrastar posteriormente su seguridad como criptosistema, realizaremos el criptoanálisis estadístico habitual. En casos concretos, en que la linealidad de los autómatas celulares utilizados (funciones) no permita cifras seguras, realizaremos un análisis de linealidad y, aplicando la metodología estándar existente, corregiremos la situación con alguna de las variantes no lineales que la misma propone.

La tercera hipótesis seguirá un esquema de trabajo similar al descrito para la segunda: construcción del sistema de cifrado, comprobación de su fortaleza frente al criptoanálisis y, en caso de fallo frente a este, aplicación de principios de reingeniería hasta lograr el sistema de cifrado deseado. Para ello, comenzaremos por integrar los autómatas celulares en un cifrado *Feistel* sin *S-cajas*, para en posteriores versiones desarrollar éstas e introducirlas. Adicionalmente, y como aproximación final, propondremos (únicamente a nivel teórico), una cifra *Feistel* No Equilibrada integrando autómatas celulares.

Las hipótesis cuarta y quinta serán contrastadas aplicando los principios de la Programación Orientada a Objetos y de la Ingeniería del Software, generando como producto final los siguiente programas:

1. Simulador de Autómatas Celulares.
2. Filtro de Análisis de Aleatoriedad, implementando la batería de test escogidos.
3. Software de Cifrado en Flujo y en Bloque.

Capítulo 2
DEFINIÓN Y ANÁLISIS FORMAL DE LOS AUTÓMATAS CELULARES

En este Capítulo, y con objeto de desarrollar herramientas de análisis y medida que serán necesarios en fases posteriores del estudio, comenzamos con las definiciones y notaciones necesarias relativas al concepto de autómata celular. Veremos cómo representar funciones de transición mediante números naturales y demostraremos que en general, cualquier función de transición es computable. Presentaremos a continuación una pieza experimental ya clásica en la literatura, y reproducida por nosotros, cuyo propósito es estimar las clases de comportamiento que los autómatas celulares presentan, y las frecuencias de las mismas. Posteriormente se introducirán algunas herramientas como la distancia de *Hamming*, la Entropía de *Shannon* y otras que propiciarán un conocimiento profundo de este espacio de reglas, y nos darán las pautas de generalización a autómatas con un mayor número de estados, de vecinos o de ambos.

2.1. Definiciones y Notación

En este apartado precisaremos más formalmente el concepto de autómata celular como modelo matemático, al tiempo que introduciremos la notación que seguiremos a lo largo del trabajo.

Como ya se ha indicado, un autómata celular es básicamente una sucesión de arrays[*] de células a_i tomando sus valores en $\mathbb{Z}_k$. En lo sucesivo, $a_i^{(t)}$ denotará el valor de la célula situada en la posición i del array en el instante de tiempo t.

Definición 5 *Una configuración o estado de autómata celular en tiempo t será un vector* $C^{(t)}$ *de* $(\mathbb{Z}_k)^{\mathbb{Z}}$ *. En el caso de tratar con un autómata celular finito dispondremos de N células* $a_0, a_1, ..., a_{N-1}$.

Notación 6 *Dada una configuración de autómata celular* $C^{(t)}$*, el estado de la i-ésima célula en la configuración se notará por* $C^{(t)}(i)$.

Definición 7 *Extensión de la función de transición* ϕ *a una configuración: función* ϕ_g*. La función de transición global* $\phi_g : (\mathbb{Z}_k)^{\mathbb{Z}} \to (\mathbb{Z}_k)^{\mathbb{Z}}$*siendo* $(\mathbb{Z}_k)^{\mathbb{Z}}$ *el conjunto de todas las configuraciones posibles está dada por*

$$C^{(t+1)} = \phi_g\left(C^{(t)}\right) = \phi\left(a_i^{(t)}\right)_{i\in\mathbb{Z}} \tag{2.1}$$

[*]Utilizamos la expresión *array* por su significado, computacionalmente obvio, sin perjuicio de la expresión vector.

Definición 8 *Si $C^{(i)}$ y $C^{(j)}$ son dos configuraciones de un autómata celular, se dice que $C^{(j)}$ es sucesora de $C^{(i)}$ si y solo si $C^{(j)} = \phi_g\left(C^{(i)}\right)$.*

Notación 9 *Si $C^{(j)}$ es sucesora de $C^{(i)}$, lo notamos por $C^{(i)} \Vdash C^{(j)}$.*

Definición 10 *Si $C^{(i)}$ y $C^{(j)}$ son dos configuraciones de un autómata celular, se dice que $C^{(j)}$ deriva de $C^{(i)}$ si y solo si, existen configuraciones distintas $C^{(1)}, ..., C^{(n)}$ tales que $C^{(1)} \Vdash C^{(2)} \Vdash ... \; C^{(n-1)} \Vdash C^{(n)}$ y además $C^{(i)} = C^{(1)}$ y $C^{(j)} = C^{(n)}$.*

Notación 11 *Si $C^{(j)}$ deriva de $C^{(i)}$, lo notamos por $C^{(i)} \overset{*}{\Vdash} C^{(j)}$. Si $i = 1$ (configuración inicial), a la derivación $C^{(1)} \Vdash C^{(2)} \Vdash ... \; C^{(j-1)} \Vdash C^{(j)}$ la llamamos computación de longitud j.*

Una propiedad muy interesante respecto al conjunto de reglas de autómata celular, una vez fijados los parámetros k y r, y muchas veces probada en la literatura [Wol94] es la siguiente.

Proposición 12 *Una vez especificados k y r, el conjunto de reglas de autómata celular disponibles es cerrado bajo composición.*

En el contexto en que nos moveremos, la configuración $a_i^{(t)} = 0$ para todo i puede considerarse como distinguida y la notaremos simplemente por 0 cuando no haya lugar a ambigüedad.

Por otra parte, la función de transición no es más que una aplicación entre dos conjuntos, por tanto es legítimo efectuar la siguiente

Definición 13 *Un autómata celular es inyectivo si su función de transición global ϕ_g es uno a uno.*

Definición 14 *Un autómata celular es reversible si existe otra autómata celular que realiza la computación inversa.*

El siguiente resultado aparece de forma inmediata.

Proposición 15 *Un autómata celular es reversible si y solo si es inyectivo.*

Demostración. Si la función global de un autómata dado ϕ_g es inyectiva, entonces dadas dos configuraciones C y C' se tendrá que si $\phi(C) = \phi(C')$ entonces $C = C'$ o lo que es lo mismo cada configuración tiene una sola imagen por ϕ_g. Es sencillo entonces construir otro autómata a partir del inicial, con iguales k y r, donde se tenga una función global que revierta a ϕ_g.

Por otra parte, si un autómata celular es reversible, necesariamente ϕ_g debe ser uno a uno. ■

Esta clase particular de autómatas es sumamente útil ya que, cuando el array de células es finito, preserva la información almacenada en la configuración inicial a lo largo de toda su evolución espacio-temporal, y además disponemos de otro autómata [Kar91] (su "inverso") para recuperar tal información.

Definición 16 *Un autómata celular tiene una función de transición ϕ lineal si para cualesquiera $u, v \in (\mathbb{Z}_k)^{2r+1}$ y cualesquiera $\lambda, \mu \in \mathbb{Z}$ se satisface*

$$\phi(\lambda u + \mu v) = \lambda\phi(u) + \mu\phi(v) \tag{2.2}$$

Se dice que no es lineal en caso contrario.

Nota 17 *Si un autómata celular tiene función de transición lineal, ésta se puede escribir de la forma*

$$\phi\left(a_{i-r}^{(t)}, a_{i-r+1}^{(t)}, ..., a_{i+r}^{(t)}\right) = \alpha_{i-r}a_{i-r}^{(t)} + \alpha_{i-r-1}a_{i-r-1}^{(t)} + ... + \alpha_{i+r}a_{i+r}^{(t)} \tag{2.3}$$

Definición 18 *Una función de transición no lineal ϕ será considerada legal cuando se verifiquen las dos condiciones siguientes sobre la misma:*

1. *El estado nulo es un punto fijo:* $\mathbb{F}(0) = 0$.
2. *Principio de simetría, que establece que* $\mathbb{F}(a_{i-r}, ..., a_{i+r}) = \mathbb{F}(a_{i+r}, ..., a_{i-r})$.

Un cálculo sencillo prueba que el número de autómatas celulares sobre $\mathbb{Z}_k$ con vecindad de tamaño r es $k^{k^{2r+1}}$, y que el número de autómatas celulares con función de transición legal y vecindad de tamaño r es $k^{k^{r+1}\frac{(k^r+1)}{2}-1}$.

Si consideramos autómatas celulares finitos, se compondrán de un array de tamaño N formado por $a_0, ..., a_{N-1}$ células. En este caso, el cálculo de la configuración sucesora de una dada requiere la aplicación de alguna condición de frontera, en orden a definir el valor de la función de transición ϕ en los extremos del array.

Definición 19 *Condición de Frontera Cilíndrica. Establece que $a_N = a_0$ de modo que la configuración adopta la forma de un círculo y la evolución espacio-temporal la de un cilindro.*

Definición 20 *Condición de Frontera Nula. Establece que $a_i = 0$ siempre que $i < 0$ o $i > N$.*

En los modelos finitos de autómata celular, el espacio de configuraciones es naturalmente finito y tiene cardinal k^N. La evolución puede representarse por grafos de estado finito, cuyos nodos representarán configuraciones, y donde un arco enlazará dos configuraciones si y solo si una sucede a la otra mediante la aplicación de la función de transición ϕ_g. Como es natural, después de un número de pasos de tiempo lo bastante largo, aunque por supuesto menor o igual que k^N, el autómata entrará en un ciclo, formado por un circuito de configuraciones del grafo.

2.1.1. Codificación Numérica de las Funciones de Transición

Proponemos a continuación un método, delineado por *S. Wolfram* [Wol91], que nos permite una representación de las funciones de transición mediante dígitos naturales[**]. Podemos así especificar directamente un autómata celular sin más que dar un número. En principio podría pensarse que dados k y r fijos, existe un universo infinito de posibles funciones de transición. Sin embargo, muchas de ellas son equivalentes, produciendo las mismas salidas a partir de las mismas entradas. Por ejemplo, si $k = 2$ y $r = 1$, hay sólo $2^8 = 256$ reglas distintas, puesto que sólo hay 8 combinaciones posibles de entrada (3 dígitos binarios) y dos posibles valores de salida (0 ó 1). En consecuencia, una función de transición se podría describir mediante un número binario de 8 dígitos, representando cada uno la salida de la función para cada posible valor de entrada de la misma.

(a_{i-1}, a_i, a_{i+1})	111	110	101	100	011	010	001	000
Valor salida	0	1	0	1	1	0	1	0

(2.4)

En la Tabla 2.4 puede verse un ejemplo del procedimiento. En general, la literatura se refiere a ese número mediante su expresión decimal, siendo la regla recogida en la tabla la "regla número 90".

Este procedimiento podría aplicarse para cualesquiera valores de k y r fijos, pero no resultaría práctico, ya que al existir $k^{k^{2r+1}}$ reglas diferentes, valores bajos de k y r generarían códigos de regla enormes[***], intratables por una computadora. Para no renunciar a las ventajas derivadas de tratar a las reglas como números enteros, se limita el espacio de reglas a codificar numéricamente, utilizando lo que se conoce como códigos o reglas de carácter aditivo, los cuales reducen el número de reglas posibles a codificar hasta $k^{(2r+1)(k-1)+1}$ con valores tratables en una computadores para magnitudes razonables de k y r. El siguiente resultado muestra cómo hacerlo.

Construcción de Código Aditivos

Sea entonces un autómata celular con k estados por célula y un rango de vecinos de tamaño r. Consideremos la siguiente expresión, dada la t-ésima configuración:

$$s = \sum_{j=-r}^{r} a_{i+j}^{(t)} \tag{2.5}$$

Se tiene que

$$0 \leq s \leq (2r+1)(k-1) \tag{2.6}$$

[**] Esta técnica de codificación está implementada en el núcleo del *software* que hemos derivado como resultado de este estudio, ya que acelera considerablemente los cálculos, evita tener que diseñar un traductor de expresiones funcionales, y para el usuario no experto permite una más fácil selección y clasificación de las funciones de transición.

[***] Para $k = 4$ y $r = 1$ tendríamos 2^{128} reglas posibles, cuando la masa del Sol en gramos de 2^{110}. Para $k = 10$ y $r = 1$ tendríamos 10^{1000} reglas, cuando el volumen estimado del Universo en m^3 es de 10^{82}.

Sea ahora el vector formado por todos los posibles valores de la suma s, ordenados de mayor a menor y dado por

$$((2r+1)(k-1), ..., 2, 1, 0)$$

Asociando a la componente i-ésima del vector un dígito $d_i \in \mathbb{Z}_k$, obtendremos un vector numérico de la forma

$$\left(d_{(2r+1)(k-1)}, ..., d_2, d_1, d_0\right)$$

El vector numérico así formado constituye un número en base k, sin más que considerar la expresión $d_{(2r+1)(k-1)}...d_2d_1d_0$. Obtendremos la expresión decimal de dicho dígito mediante la siguiente sumatoria:

$$regla = \sum_{i=0}^{(2r+1)(k-1)} (d_i k^i) \tag{2.7}$$

Consideraremos la función de transición ϕ completamente especificada mediante un código aditivo cuando fijemos el valor de los parámetros k, r (que describen el dominio de la función ϕ), y el valor de $regla$ que especifica completamente a la función, sujeto a la evidente restricción: $0 \leq regla \leq k^{[(2r+1)(k-1)]+1} - 1$.

Ahora bien, hemos establecido que el dígito $regla$, obtenido mediante la ecuación 2.7 determina por completo la función de transición ϕ del autómata celular pero, ¿en qué forma lo hace? Veamos que el proceso es realmente muy sencillo y puede ser descrito mediante la siguiente tabla:

s	código de la regla
0	d_0
1	d_1
$\vdots$	$\vdots$
$(2r+1)(k-1)$	$d_{(2r+1)(k-1)}$

(2.8)

La lectura de la función de transición especificada numéricamente es ya inmediata. Para conocer el nuevo estado al que evoluciona la célula *i-ésima*, se suman los estados de las células vecinas. Esa suma será un dígito s que aparecerá en la columna izquierda de la tabla. Se considera como nuevo estado de la célula *i-ésima* a aquel dígito en base k indizado en la columna derecha de la tabla por el valor de la suma s.

Ejemplo 21 *Como ejemplo, supongamos $k = 2$ y $r = 1$. Sabemos que el espacio de reglas aditivas está formado por $\left\{regla : 0 \leq regla \leq 2^{[3]+1} - 1\right\}$. Luego especificar una regla es dar un número en ese rango como es el 9. Del número derivamos la tabla sin más que pasar 9 a su representación en base 2 (en general el número se representa en base k si el autómata tiene k estados).*

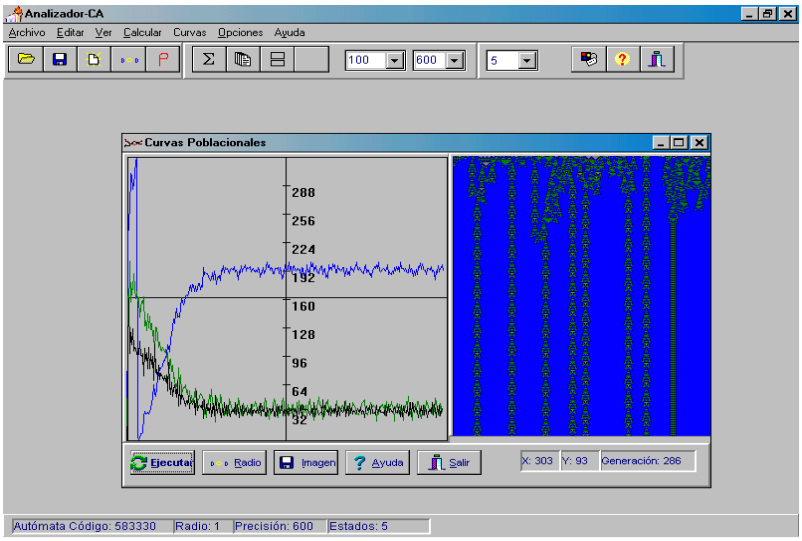

Figura 1 El analizador-CA en ejecución

s	código de la regla
0	1
1	0
2	0
3	1

(2.9)

Por lo tanto, una vez escogida la regla número *9* para nuestro autómata celular, actualizaremos el valor de la célula $a_i^{(t)}$ sumando el estado de la propia célula y el de sus vecinas a derecha e izquierda. Esa suma estará entre *0* y *3*, valor que usaremos para indizar en la tabla y obtener el nuevo estado de la célula en cuestión.

2.2. Análisis Experimental y Clasificación

En este apartado, procederemos a desarrollar la primera pieza empírica del estudio. El objetivo: proceder a realizar una clasificación cerrada de los autómatas celulares. La metodología, la seguida ya por otros autores a partir del trabajo pionero de [Wol94]. El medio, la simulación *software*. El propósito, habilitar técnicas de programación que posteriormente nos serán de gran utilidad. De lo hasta ahora comentado se deben desprender dos conclusiones claras:

1. La diversidad de comportamientos que puede seguir el patrón de evolución de un autómata celular es amplísima, al menos desde el punto de vista del observador.

2. Se plantea la necesidad de proceder a la sistematización de ese conjunto de comportamientos encuadrándolos en clases con características comunes.

Como ya hemos indicado, el medio y herramienta que usaremos para lograr nuestro propósito será un simulador *software*[****] desarrollado por nosotros como parte de este estudio el cual, entre otras características de mayor relevancia, presenta una utilidad que permite visualizar el patrón de evolución de un autómata celular dado. Nos referimos al *Analizador-CA*, de cuya ejecución vemos un ejemplo en la Figura 1, y cuyas características principales se exponen en el Capítulo 6.

2.2.1. Computabilidad de los Autómatas Celulares

El resultado teórico que nos llevó a considerar en primer lugar y a implementar posteriormente el simulador en cuestión fue el siguiente.

Proposición 22 *Un autómata celular finito puede ser simulado con complejidad* $O(n^2)$ *siendo* n *el número de células del autómata y el número de generaciones a computar*

Demostración. Consideremos cualquier autómata celular y sea el algoritmo de simulación que se muestra más abajo.

Algoritmo de Simulación de Autómatas Celulares

INPUT: código aditivo del autómata celular y número de generaciones n.
OUTPUT: evolución sobre n generaciones.
1. *For* $i \leftarrow 1$ *to* n *do* $a_i^{(0)} \leftarrow$random.
2. For $j \leftarrow 1$ to n do
 2.1 For $i \leftarrow 1$ to n do
 2.2 $a_i^{(j)} = \phi\left(a_{i-r}^{(j-1)}, a_{i-r+1}^{(j-1)}, ..., a_{i+r}^{(j-1)}\right)$

Como vemos, el bucle de 2.2 requiere n pasos y sus operaciones internas, incluida la llamada a ϕ son $O(1)$. Si combinamos lo anterior con el bucle externo encontramos que ambos requieren $O(n^2)$ operaciones. ■

Por tanto, y dada la sencillez y comodidad de la simulación con procesadores relativamente potentes, procedimos a la construcción del *Analizador-CA*.

El enfoque que vamos a seguir con esta herramienta ya disponible es muy simple: considerar espacios de reglas de autómatas celulares sencillos, para poder realizar un análisis individualizado de cada uno de sus miembros a fin de encuadrarlo en una de las cuatro clases de comportamiento descritas en la literatura. Una vez realizado dicho análisis, procederemos a obtener las frecuencias representativas de la importancia de cada clase.

El criterio que seguiremos será el único que en este momento tenemos disponible. Es cualitativo, y fue introducido en el Capítulo 1 (ver Criterio 2).

Un amplio conjunto de ejemplos ha sido objeto de análisis, como evidencia para soportar la conjetura que clasifica el comportamiento de cualquier autómata celular en una de las cuatro clases definidas (Es necesario aclarar que el criterio de

[****]Para el desarrollo del *software* se han utilizado los compiladores ©Delphi 2.0 y ©Delphi 4.0 bajo ©Window 9x.

clasificación no es formal en absoluto, sino puramente observacional. No obstante, damos por buenos los resultados obtenidos al ajustarse de forma muy aproximada a los obtenidos previamente por algunos autores).

Como ejemplo de la metodología seguida, sea entonces el conjunto de los autómatas celulares con dos estados por células y amplitud de vecindad uno ($k = 2$ y $r = 1$). En estas condiciones, sabemos que hay un total de $2^{2^3} = 256$ funciones o reglas de transición posibles (no estamos ahora utilizando la codificación aditiva sino el modelo general), de las cuales son legales 32. De hecho, y adicionalmente, se han analizado los comportamientos para las combinaciones paramétricas siguientes: ($k = 2$ y $r = 2$), ($k = 3$ y $r = 3$) y ($k = 3$ y $r = 1$). Mostramos más adelante los resultados obtenidos.

En todos los casos, la configuración inicial $C^{(0)}$ del autómata celular se ha establecido aleatoriamente, de modo que cada célula tuviese un valor inicial escogido en $\mathbb{Z}_k$ independientemente con probabilidad $\frac{1}{k}$. A pesar de la ausencia de estructura del estado inicial, muchas de las reglas parecen generar patrones con un nivel de estructuración evidente. A pesar de que los patrones obtenidos con reglas diferentes difieren en los detalles, todos parecen encuadrarse en una de las cuatro clases de comportamiento postuladas. Concretamente se observa que para ($k = 2$ y $r = 1$):

1. La evolución lleva a un estado homogéneo en los siguientes casos: reglas con código 0, 4, 16, 32, 36, 48,54, 60 y 62.

2. La evolución lleva a un conjunto de estructuras periódicas, o simples y estables separadas entre sí: reglas con código 8, 24, 40, 56 y 58.

3. La evolución lleva a patrones caóticos: reglas 2, 6, 10 ,12, 14, 18, 22, 26, 28, 30, 34, 38, 42, 46 y 50.

4. La evolución lleva a estructuras localizadas complejas, en ocasiones de larga duración: reglas con código 20 y 52.

Algunos patrones, como por ejemplo el dado por la regla con código 12 y que hemos asignado a la clase 3, contienen triángulos muy definidos y parecen más regulares que los obtenidos con otras reglas, como por ejemplo la número 10. La razón de esta aparente discrepancia no es otra que el distinto grado de irreversibilidad que prensentan ambas reglas, y que se manifiesta a través del grado de regularidad del patrón espacio temporal.

También podemos reseñar la prácticamente nula influencia de la configuración inicial en el aspecto global del patrón. Tal y como puede verse en la Figuras 2 y 3, los patrones obtenidos con la misma regla (en este caso la número 22) y configuraciones iniciales distintas, pero establecidas aleatoriamente, se diferencian únicamente en sus detalles concretos, mostrando las mismas características cualitativas desde un punto de vista macroscópico.

2.2.2. Conclusiones

La Tabla 2.10 lista las fracciones de autómatas celulares que hay en cada una de las cuatro clases. Ha sido obtenida utilizando el *Analizador-CA* para las combinaciones paramétricas especificadas, simulando las reglas mediante representación numérica,

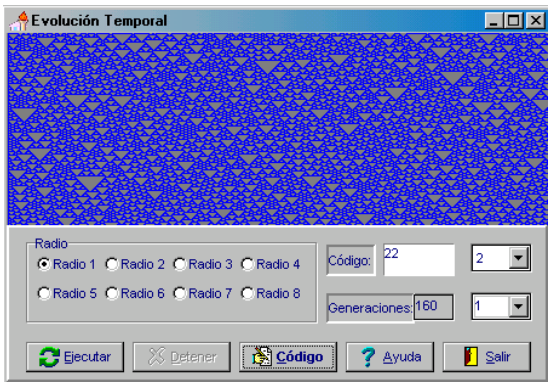

Figura 2 Evolución de la regla 22

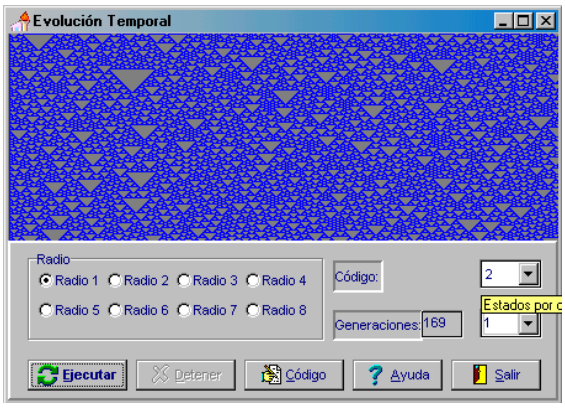

Figura 3 Evolución alternativa de la regla 22

y contabilizando la pertenencia de estas en a una de las cuatro clases ya definidas según el Criterio 2 . Se observa que el incremento de los parámetros k y r llevan a la clase 3 a ser la más común en detrimento de las clases 1 y 2. La clase 4, por su parte es comparativamente más bien rara.

	$k=2$	$k=2$	$k=3$	$k=3$
Clase	$r=1$	$r=2$	$r=3$	$r=1$
1	0,50	0,25	0,09	0,12
2	0,25	0,16	0,11	0,19
3	0,25	0,53	0,73	0,60
4	0	0,06	0,06	0,07

(2.10)

En [Wol94] se introducen algunos métodos cuantitativos de carácter formal para el análisis de los autómatas celulares que apoyan la caracterización básica obtenida aquí de modo experimental, y que no trataremos por no ser aplicados en este estudio.

2.3. Distancia de Hamming

Como hemos visto, la regla o función de transición de un autómata celular define una transformación de una secuencia de bits[*****] o configuración en otra. Dado el espacio de configuraciones del autómata celular, podemos definir varias distancias posibles en el mismo. Sin embargo, nosotros nos quedaremos con una de las más simples y ampliamente utilizada en las Ciencias de la Computación: la distancia de *Hamming*.

Definición 23 *La distancia de Hamming $H(C_1, C_2)$ entre dos configuraciones de un autómata celular se define como el número de bits (o dígitos) en que difieren las configuraciones C_1 y C_2.*

Es trivial demostrar la siguiente proposición.

Proposición 24 *$H(C_1, C_2)$ es una distancia.*

La distancia de *Hamming* será un importante parámetro de caracterización de la evolución de un autómata celular, y lo utilizaremos con profusión. Es por tanto necesario probar que es computable en el sentido de [Dav92] con razonable facilidad.

Proposición 25 *Dadas dos configuraciones C_1 y C_2 de un autómata celular finito con n células, la distancia de Hamming entre ellas es computable en $O(n)$ pasos.*

Demostración. Sea el Algoritmo de Cáculo de $\mathbf{H}(C_1, C_2)$ que se muestra , que calcula efectivamente tal distancia.

Como vemos, 2.1 requiere $O(1)$ pasos de cálculo, y considerando el bucle externo tenemos que $\sum_{i=1}^{n} 1 = n$ tal y como queríamos. ■

El parámetro complementario inmediato es la distancia media de *Hamming*, que definimos como sigue:

[*****] O más generalmente, define una aplicación $\phi_g : \mathbb{Z}_k^N \to \mathbb{Z}_k^N$ donde N representa el número de células que componen el autómata celular, supuesto que este es finito.

Algoritmo de Cálculo de $\mathbf{H}\,(C_1, C_2)$.

INPUT: Configuraciones C_1 y C_2 de un autómata celular finito con n células.
OUTPUT: $H\,(C_1, C_2)$.
1. *Inicialización: Hacer* $d \leftarrow 0$.
2. *For* $i \leftarrow 1$ *to* n *do*
 2.1 *If* $C_1(i) \neq C_2(i)$ *then* $d \leftarrow d + 1$.
3. $H\,(C_1, C_2) \leftarrow d$.

Definición 26 *Dada una sucesión de* n *configuraciones* $C_0, C_1, ..., C_{n-1}$ *tales que* $C_i \vdash C_{i+1}$ *para* $i = 0, ..., n-2$ *se define su distancia de Hamming media* $\overline{H}$ *como*

$$\overline{H} = \frac{\sum_{i=0}^{n-2} H(C_i, C_{i+1})}{n} \tag{2.11}$$

De igual forma, es computable de forma inmediata y ambas han sido incluidas en el núcleo del *Analizador-CA*.

Corolario 27 *La distancia de Hamming media para una sucesión de* n *configuraciones de un autómata celular finito de* n *células se calcula en* $O(n^2)$ *pasos.*

Demostración. El Algoritmo de Cálculo de $\overline{H}$ que presentamos lo prueba.

Algoritmo de Cálculo de $\overline{\mathbf{H}}$.

INPUT: *Sucesión de* n *configuraciones de un autómata celular finito de* n *células.*
OUTPUT: $\overline{H} = \sum_{i=0}^{n-2} H(C_i, C_{i+1})/n$.
1. *Inicialización:* $p \leftarrow 0$.
2. *For* $i \leftarrow 0$ *to* $n-2$ *do*
 2.1 *Hacer* $p \leftarrow p + H(C_i, C_{i+1})$ *vía Alg. Cálculo de* $\mathbf{H}\,(C_1, C_2)$.
3. *Hacer* $\overline{H} = p/n$.

Es evidente que $\sum_{i=0}^{n-2} n = n(n-2) = n^2 - 2n$ y por tanto el parámetro $\overline{\mathbf{H}}$ es computable en $O(n^2)$ pasos. ■

Con carácter general podemos establecer el siguiente criterio en relación con la evolución espacio-temporal de los autómatas celulares cuando se les aplica la distancia de *Hamming* $H\,(C_1, C_2)$:

Criterio 28 *Los autómatas celulares de las clases* 1 *y* 2 *presentan, en general, distancias de Hamming constantes en el tiempo. Los autómatas celulares de las clases* 3 *y* 4 *presentan distancias de Hamming crecientes o fluctuantes en su valor.*

Si consideramos las configuraciones C_1 y C_2 como iniciales, verificando que $H\,(C_1, C_2) = 1$, y realizamos τ pasos de cálculo con el mismo autómata partiendo de

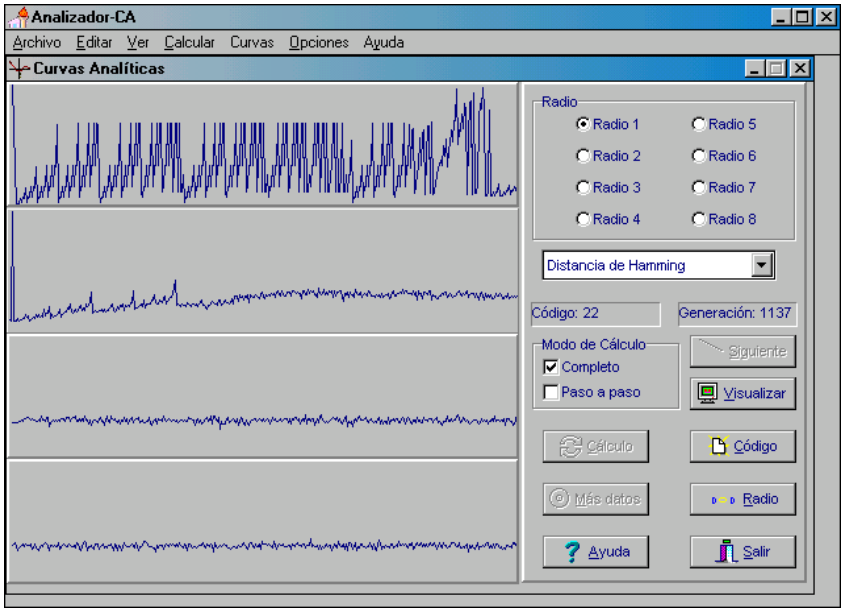

Figura 4 Distancia de Hamming para la regla 22

ambas configuraciones, sabemos que el patrón global final será el mismo[******] desde un punto de vista macroscópico. Sin embargo, ¿qué ocurrirá con la distancia? Es claro que puesto que la diferencia inicial es de un bit, la diferencia final, no podrá superar el valor 2τ. Estudios usando el *Analizador-CA* prueban que para autómatas celulares simples la distancia de *Hamming* tiende rápidamente a un valor constante relativamente pequeño.

En general, el estudio realizado con el *Analizador-CA* sugiere que el comportamiento para autómatas celulares complejos en cuanto a la distancia se refiere, presenta grandes diferencias según consideremos modelos de autómatas aditivos o no. En concreto, para reglas aditivas encontramos que la diferencia obtenida tras τ pasos de tiempo viene dada por la evolución a partir de la diferencia inicial. Si consideramos una configuración inicial con una única célula no nula y con regla número 90, encontramos que la distancia de cada configuración respecto de la inicial está dada por la función $H_\tau = 2^{\#_1(\tau)}$, tal y como puede verse en la Figura 5. De hecho, si consideramos la distancia media de *Hamming*, a lo largo de muchos pasos de tiempo discreto, encontramos que se comporta como $\overline{H_\tau} = \tau^{\log_2 3-1} \simeq \tau^{0,59}$. De igual forma, y para la regla 22, vemos (Figura 4) bruscas oscilaciones de la distancia en las primera etapas de la evolución espacio-temporal, que tienden a estabilizarse posteriormente en torno a un valor fijo, con pequeñas fluctuaciones en torno al mismo en ambos sentidos.

Para autómatas celulares no aditivos, con idénticas condiciones iniciales, la distancia a la configuración inicial suele incrementarse de forma bastante lineal, tendiendo para grandes valores de τ a una expresión de la forma $H_\tau \simeq \tau$. Esto nos

[******] Recuérdese al respecto que en el Capítulo 1 establecimos que cambios en pocas células no afectan al comportamiento global de manera apreciable.

Por supuesto, esta forma de hablar es sólo una aproximación, para indicar que el incremento de la variable temporal τ lleva aparejado un crecimiento del valor de la distancia de Hamming, H_τ.