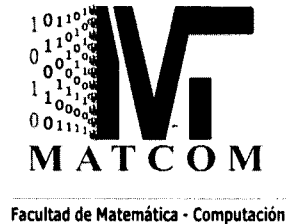


570.113
Oso
P

TD-0029-04-01

Universidad de La Habana.
Facultad de Matemática-Computación.



Plataforma Computacional para el Desarrollo de la Biología de Sistemas



Tesis de grado en opción al título de Licenciado en Ciencia de la Computación.

Autor:

Karel Osorio Ramírez.

Tutores:

Lic. Noel Moreno Lemus.

Lic. Rafael Arturo Trujillo Rasúa.

Ciudad de la Habana, Junio de 2004.
Año del 45 aniversario del triunfo de la Revolución.

Dedicatoria

A mi mamá,

A mi papá,

A Eiquel y a Yari.

A mis abuelos,

A mi familia toda,

A mi grupo,

Y a mis amigos.

Agradecimientos

Agradezco profundamente a mis padres por el cariño y el apoyo que siempre me han dado y sin el cual no hubiese sido posible llegar hasta aquí. A toda mi familia en general por estar siempre presentes.

Agradezco a mi grupo por tantos buenos momentos que pasamos juntos, fiestas, campismos, trabajo, estudio, también por su ayuda siempre presente cuando la necesité, especialmente a los becados que han estado siempre conmigo.

A mis profesores que con su profesionalidad han contribuido enormemente en mi formación.

A mis tutores, Noel y Trujillo, por su ayuda incondicional.

Al grupo de BioInformática de la UCI, por el apoyo brindado en la realización de este trabajo.

A la Revolución Cubana por darme la posibilidad de estudiar, de realizar mis sueños y contribuir al desarrollo de mi país.

Muchas gracias por todo...

Karel Osorio Ramírez.

¿Por qué la tierra es mi casa?

¿Por qué la noche es oscura?

¿Por qué la luna es blanca

que engorda como adelgaza?

¿Por qué una estrella se enlaza

con otra, como un dibujo?

Y ¿por qué el escaramujo

es de la rosa y del mar?

Yo vivo de preguntar:

Saber no puede ser lujo.

Silvio Rodríguez.

Índice

RESUMEN	1
----------------------	----------

INTRODUCCIÓN	2
---------------------------	----------

OBJETIVO GENERAL	5
-------------------------------	----------

TAREAS	5
---------------------	----------

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA.	6
---	----------

1.1 MÉTODO CLÁSICO DE OBTENCIÓN DE MEDICAMENTOS.....	6
---	----------

1.2 MODELACIÓN MATEMÁTICA Y SIMULACIÓN DE SISTEMAS BIOLÓGICOS.....	7
---	----------

1.2.1 ¿QUÉ ES UN MODELO MATEMÁTICO?.....	7
--	---

1.2.2 SIMULACIÓN COMPUTACIONAL.....	8
-------------------------------------	---

1.2.2.1 Ventajas y desventajas.....	9
-------------------------------------	---

1.3 MODELACIÓN DE SISTEMAS BIOLÓGICOS.....	10
---	-----------

1.3.1 AUTÓMATAS CELULARES.....	10
--------------------------------	----

1.3.1.1 Descripción de Autómatas Celulares.....	10
---	----

1.3.1.2 Definición Formal de un Autómata Celular.....	11
---	----

1.3.1.3 Autómatas Celulares de 2 Dimensiones.....	12
---	----

1.3.1.4 Propiedades y clases de autómatas celulares.....	12
--	----

1.3.1.5 Autómatas celulares probabilísticos.....	14
--	----

1.3.2 DINÁMICA DE POBLACIONES.....	16
------------------------------------	----

1.3.2.1 Interacciones presentes en el modelo.....	17
---	----

1.3.2.2 Ecuaciones que describen el modelo.....	22
---	----

1.4 SOLUCIÓN NUMÉRICA DEL SISTEMA DE ECUACIONES DIFERENCIALES.....	24
---	-----------

1.4.1 MÉTODOS DE RUNGE-KUTTA.....	24
-----------------------------------	----

1.4.2 MÉTODOS DE ADAMS.....	25
-----------------------------	----

1.5 HERRAMIENTAS PARA EL DESARROLLO DEL SISTEMA.....	25
---	-----------

1.5.1 TECNOLOGÍA .NET.....	26
----------------------------	----

1.5.2 LENGUAJE DE PROGRAMACIÓN C#.....	26
--	----

1.5.3 MATLAB.....	27
-------------------	----

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA.....	28
---	-----------

2.1 ANÁLISIS DEL SISTEMA.....	28
--------------------------------------	-----------

2.1.1 REQUISITOS DEL SISTEMA.....	28
-----------------------------------	----

2.2 DISEÑO E IMPLEMENTACIÓN DEL SISTEMA.....	29
---	-----------

2.2.1 MÓDULO DE MODELACIÓN.....	30
---------------------------------	----

2.2.1.1 Modelación de sistemas biológicos.	30
2.2.1.2 Fichero de ecuaciones diferenciales.	35
2.2.1.3 Interacción con Matlab.	36
2.2.1.4 Interfaz visual de modelación.	37
2.2.2 MÓDULO DE GRAFICACIÓN.	39
2.3 DESCRIPCIÓN DEL SISTEMA PCDBS.	41
2.3.1 PALETA DE COMPONENTES.	42
2.3.2 ÁREA DE MODELACIÓN.	43
2.3.3 INSPECTOR DE PROPIEDADES.	43
2.3.4 EXPLORADOR DE OBJETOS.	44

CAPÍTULO 3. MODELACIÓN DE UN SISTEMA BIOLÓGICO CONCRETO.....45

3.1 MODELO MATEMÁTICO DE LOTKA-VOLTERRA.....	45
3.1.1 MODELACIÓN CON PCDBS.....	48

CONCLUSIONES53

RECOMENDACIONES54

BIBLIOGRAFÍA55

Resumen

Nuestro país ha obtenido resultados importantes en el campo de la biotecnología y la obtención de medicamentos; el estudio, análisis y comprensión adecuada del funcionamiento de los sistemas biológicos es una tarea primordial en esta esfera.

En el presente trabajo se desarrolla un prototipo de una Plataforma Computacional para el Desarrollo de la Biología de Sistemas; la cual es de gran utilidad en el desarrollo de nuevas vacunas y medicamentos debido a que en nuestro país no se cuenta con herramientas de este tipo.

Introducción

Las enfermedades y los virus han sido unos de los males que a través de la historia han azotado a nuestra civilización. Muchas epidemias masivas han cobrado la vida de millones de personas a lo largo de los siglos y el hombre no ha cesado en su empeño de poder vencerlas.

En años tan lejanos como el (464-404 A.C.) Tucídides escribía: “en una epidemia acaecida durante la guerra del Peloponeso, los enfermos eran atendidos sólo por aquellos que habían sobrevivido previamente a la enfermedad, con la seguridad de que éstos no volverían a ser contagiados” [17].

Estas observaciones fueron pasando de generación en generación y así ya en mayo del año 1776, el médico inglés Edward Jenner(1749-1823), tras su constatación de que las vaqueras que habían adquirido la viruela vacunal(una forma benigna de enfermedad que sólo producía pústulas en las manos) no eran atacadas por la grave y deformante viruela humana, inoculó a un niño fluido procedente de las pústulas vacunales de Sarah Nelmes, semanas después el niño fue inyectado con pus de una pústula de un enfermo de viruela, comprobando que no quedaba afectado por la enfermedad [17].

El primer abordaje plenamente científico del problema se debió a Louis Pasteur. Estudiando la bacteria responsable del cólera: Aviar(más tarde conocida como *Pasteurella aviseptica*), observó que la inoculación en gallinas de cultivos viejos, poco virulentos, las protegía de contraer la enfermedad cuando posteriormente eran inyectadas con cultivos normales virulentos. De esta forma se obtuvo la primera vacuna a base de microorganismos atenuados [17].

Fue este el inicio de un camino largo que ha salvado la vida a millones de personas. Las enfermedades que ayer eran incurables se han convertido en algo cotidiano y perfectamente curable en la actualidad.

Pero la naturaleza es muy dinámica. Han surgido nuevos medicamentos, y casi al unísono, nuevas enfermedades. Los adelantos científico-técnicos han permitido acelerar los procesos de producción de medicamentos, pero no lo suficiente. Así, enfermedades como el SIDA, por sólo citar un ejemplo, con más de 20 años de existencia, sigue cobrando vidas sin que se vislumbre una solución palpable. Además, las grandes compañías farmacéuticas en poder de transnacionales sólo son capaces de resolver los problemas de la minoría que tiene el dinero suficiente para pagar los altos precios de sus productos, mientras que enfermedades perfectamente curables siguen privando de la vida a millones de personas, tanto en el mundo subdesarrollado como en el desarrollado.

Aun estas grandes compañías, equipadas de la mejor tecnología presentan un gran problema, la obtención y puesta en el mercado de un nuevo producto demora entre 12-15 años y el costo total de dicha producción asciende aproximadamente a \$500 millones de dólares, \$150 millones de los cuales se invierten en la búsqueda y experimentación de nuevas drogas [6].

En esta búsqueda incesante de nuevos métodos de investigación, más rápidos y menos costosos, ha surgido la modelación *in silico*.

La modelación *in silico* tiene el potencial de disminuir los tiempos de obtención de medicamentos de 12 años a tan solo 12-24 meses obteniéndose nueva y mejores drogas sin la necesidad de ser blanco de cuestionamientos en materia de ética profesional y humana [10].

Así, en los últimos años, el desarrollo impetuoso de la Biología y el abandono cada vez más creciente de los paradigmas reduccionistas de la Biología Molecular, se abre paso hoy a una nueva biología que hace creciente uso de las modelaciones matemáticas de sus problemas. Esta biología es la que se ha dado a llamar "biología de sistemas", en alusión a que tendrá por objeto de estudio la comprensión integrada de un sistema biológico y no de sus partes independientes.

La percepción de esta necesidad de la biología ya va llegando a los sectores comerciales de la biotecnología. Recientemente en los Estados Unidos se han creado las primeras empresas de biología de sistemas cuyos productos están intrínsecamente relacionados con la simulación de problemas biológicos concretos.

Ejemplo de tales empresas son:

- Gene Network Sciences (www.gnsbiotech.com, fundada 2000).
- Entelos (www.entelos.com, fundada 1996).
- Physiome (www.physiome.com, fundada 2001)
- Genómica (www.genomatica.com, fundada 2001)

Todas estas empresas parten de la propiedad intelectual sobre una plataforma de modelación (software) para integrar información, representar matemáticamente y simular sistemas biológicos. No obstante siguen como estrategia de negocios la formación de alianzas comerciales con empresas biotecnológicas y no la comercialización directa del software. El producto ofrecido por estas empresas se vincula con la simulación y análisis de un problema biológico concreto, donde se le ha añadido valor adicional al software incorporando y analizando información biológica específica. Por ejemplo Entelos, la más antigua y exitosa de estas compañías, comercializa un paquete para el estudio de efectos de drogas en la diabetes y Physiome tiene entre otros un paquete para simular la cardio-toxicidad de drogas moleculares [6, 10].

Nuestro país ha alcanzado un gran nivel en las ramas biotecnológicas y en el desarrollo de aplicaciones médicas en las que se han obtenido muy buenos resultados. Es por ello que una plataforma de modelación de sistemas biológicos que permita acelerar el proceso de obtención de medicamentos sería una herramienta, más que necesaria, imprescindible para poner a la biotecnología cubana en igualdad de condiciones con el resto de las compañías del mundo en este campo.

El Centro de Inmunología Molecular, cuya misión es obtener y producir nuevos biofármacos destinados al tratamiento del cáncer y otras enfermedades crónicas no transmisibles, hace un uso creciente de la modelación matemática de los problemas biológicos que estudia. En este centro existe un gran interés en contar con una herramienta computacional que les facilite este tipo de investigaciones.

Objetivo General

Realizar un prototipo de una plataforma de modelación computacional para el desarrollo de la biología de sistemas, que permita realizar análisis simples sobre el comportamiento de los sistemas biológicos.

Tareas

Realizar un estudio sobre los métodos más utilizados en la modelación de sistemas biológicos; realizar una implementación de uno de estos métodos como base para un primer prototipo; diseñar una interfaz amigable que permita realizar modelaciones de manera sencilla y modelar algún sistema biológico conocido y estudiado con anterioridad con el objetivo de comparar los resultados del software con los que aparecen en la literatura.

Capítulo 1. Fundamentación teórica.

En el presente capítulo se exponen los procedimientos tradicionales para la obtención de nuevos medicamentos y como la modelación computacional puede ser una herramienta útil en este proceso. Además se realiza un estudio de dos de los métodos más utilizados en la modelación de sistemas biológicos: Sistemas de Ecuaciones Diferenciales y Autómatas Celulares, así como un análisis breve sobre las posibles herramientas de desarrollo a utilizar en la implementación.

1.1 Método clásico de obtención de medicamentos.

Los métodos clásicos de obtención de medicamentos, a pesar de los desarrollos tecnológicos actuales siguen siendo demasiado tediosos y caros [4].

Se puede decir que desde el momento en que se obtiene un producto por síntesis en el laboratorio, hasta que llega al mercado pueden transcurrir 10 años, pues tuvo que haber pasado por los ensayos *in vitro*, los ensayos *in vivo*, los exámenes de toxicidad, verificación de efectos secundarios, ensayos clínicos, la patentización y la autorización de mercado [5].

En este tipo de investigación nunca existe la seguridad de que el producto saldrá al mercado. Aproximadamente cinco de cada cinco mil compuestos ingresados para pruebas preclínicas llegan a someterse a pruebas en humanos y solamente uno de estos cinco obtienen la aprobación para su venta [4]. No obstante, la experiencia obtenida puede servir como referencia a otros grupos de investigación que deseen poner a prueba otras sustancias de similar naturaleza [5].

Durante este largo proceso (aproximadamente 15 años) la enfermedad en estudio puede haber cobrado la vida a millones de personas, y si las investigaciones constituyen un fracaso desde el punto de vista de la obtención de un producto efectivo, otro numeroso grupo de personas seguirá a la espera de una solución viable y responsablemente humana.

1.1 Método clásico de obtención de medicamentos.

La fortaleza de este método de investigación radica en que los medicamentos que se obtienen son muy eficaces y de buena calidad [4]. Como hemos podido constatar las deficiencias de este método radican en el largo tiempo de ejecución [4, 5, 6], los elevados gastos en reactivos, utensilios y animales de laboratorio que ascienden a \$150 millones [5, 6], y problemas éticos relacionados con la utilización de humanos como animales de laboratorio.

1.2 Modelación matemática y simulación de sistemas biológicos.

En las primeras décadas del siglo XX el italiano Vito Volterra (1860 - 1940), uno de los creadores del análisis funcional, realizó un estudio sobre las pesquerías en el Mar Adriático obteniendo la primera teoría sobre las relaciones presa-depredador; simultáneamente Alfred Lotka desarrolló su teoría basándose en ciertas reacciones químicas y obtuvo los mismos resultados que Volterra.

En la búsqueda de nuevas vías para realizar investigaciones biológicas y producir nuevos medicamentos se ha abierto el camino hacia la modelación matemática de los sistemas biológicos. Esta herramienta unida a la simulación computacional de los modelos obtenidos proporciona una serie de ventajas en comparación con los métodos clásicos, que ha posibilitado que muchos científicos de todo el mundo vean, en esta dirección, una alternativa para llevar a cabo estudios biomédicos [2, 6, 7, 9, 10].

1.2.1 ¿Qué es un modelo matemático?

Un modelo matemático es una representación para un objeto o proceso, en el cual quedan plasmadas sus principales características, a partir de conjeturas o suposiciones iniciales estableciendo un compromiso entre complejidad y exactitud; en esta idealización es posible utilizar entes, estructuras y leyes de la matemática; permitiendo interpretar los resultados del modelo en términos del objeto o proceso estudiado.

Necesariamente la definición anterior lleva a pensar que existen infinidad de modelos para infinidad de problemas biológicos, hecho que es muy real. Por otra

1.2 Modelación matemática y simulación de sistemas biológicos.

parte, existen diferentes modelos para el mismo problema, pues, como norma, todo modelo matemático que arroje resultados acordes con los obtenidos experimentalmente debe ser aceptado [1]; además un mismo modelo es aplicable a diferentes procesos o fenómenos.

1.2.2 Simulación computacional.

La simulación es la acción de diseñar y dar solución a un modelo para un objeto o proceso con el objetivo de lograr una mejor comprensión del objeto o proceso en estudio. La simulación computacional hace uso de las computadoras para la solución del modelo propuesto.

Los métodos de simulación pueden ser clasificados atendiendo varios criterios; si analizamos el comportamiento del sistema en el tiempo podemos decir que existen las simulaciones estáticas, donde solo nos interesa representar el estado del sistema en un instante de tiempo determinado; una simulación que represente la evolución del sistema en el tiempo se clasifica como simulación dinámica. Dentro de las simulaciones dinámicas existen simulaciones continuas que representan cambios continuos en el tiempo y suelen representarse mediante ecuaciones diferenciales; y los discretos que cambian el estado del sistema en instantes concretos de tiempo o eventos. Otro criterio muy utilizado es la presencia o no de variables aleatorias, una simulación se dice determinista si no contiene variables aleatorias; y se dice estocástico si contiene al menos una variable aleatoria.

En la simulación de sistemas biológicos un objetivo importante es la obtención gráficas que describan el comportamiento fármaco-cinético de las poblaciones involucradas en el proceso, entre otros.

Es importante destacar que estas gráficas de comportamiento fármaco-cinético aportan la información necesaria para validar si un producto puede ser utilizado ante una enfermedad determinada.

1.2 Modelación matemática y simulación de sistemas biológicos.

1.2.2.1 Ventajas y desventajas.

Son muchas las ventajas de estos nuevos métodos de investigación biológica. Casi todas ellas se derivan del hecho de que los tiempos requeridos para realizar un experimento son sumamente pequeños.

En un laboratorio de investigación para realizar los estudios *in vivo* se requiere de meses de trabajo. Cada vez que un animal de laboratorio es inyectado con un producto se hace necesario esperar horas y en determinadas ocasiones hasta días, para tomar las muestras de sangre y hacer los estudios fármaco cinéticos. Cada vez que se quiere variar la concentración del medicamento es necesario repetir todo el ensayo otra vez [4,5].

Estas debilidades de los métodos clásicos, constituyen la fortaleza de los métodos de simulación. Los ensayos que durarían meses y consumirían miles de dólares en reactivos químicos y animales de laboratorio, en un simulador sólo necesitan del cambio de un parámetro en las ecuaciones del modelo (una "concentración" o una "constante de velocidad" por citar algunos ejemplos) [2].

Resumiendo, la simulación de sistemas biológicos:

- Aceleran el descubrimiento de posibles drogas [7].
- Minimizan los tiempos de obtención de un nuevo producto (2-3 años) [10].
- Reducen los costos de investigación y producción [6, 9].
- Eliminan los problemas éticos vinculados al uso de humanos como animales de laboratorio.
- Posibilitan la predicción de nuevos modelos a partir de los ya existentes [2, 7].
- Brindan facilidades para variar parámetros y condiciones experimentales [2].

La principal desventaja de estos métodos de simulación es que se requiere un personal altamente preparado en el tema [6, 7, 9, 10].

1.3 Modelación de sistemas biológicos.

Para la modelación de sistemas biológicos es válido usar cualquier modelo que obtenga resultados acordes con los experimentales. Dos de los modelos más utilizados con este fin son los Autómatas Celulares y la Dinámica de Poblaciones; los cuales serán descritos a continuación.

1.3.1 Autómatas celulares.

Alrededor de 1940, John von Neumann estuvo trabajando en una teoría general de autómatas para el procesamiento de información que fuese aplicable tanto a sistemas biológicos como a aparatos tecnológicos.

Fue Stanislaw Ulam quién sugirió a von Neumann que utilizara un arreglo cuadrulado con un autómata de estados finitos en cada celda. Cada una de estas celdas estaría en un estado y utilizaría información sobre los estados de las celdas a su alrededor para pasar al siguiente estado. Von Neumann no llegó a terminar su prueba escrita sobre este trabajo, perdiéndose así el interés por los autómatas celulares hasta que en los años 70 se hicieron populares por la publicación de llamado “Juego de la vida” de John H. Conway [18].

1.3.1.1 Descripción de Autómatas Celulares.

El mundo de los autómatas celulares es un arreglo uniforme de celdas, donde cada sitio contiene cierta información y donde el tiempo transcurre en pasos discretos. Las leyes de este universo consisten en las reglas que determinan cuál será el próximo estado de la celda.

Este arreglo puede tener n dimensiones, aunque con más de tres dimensiones son muy difíciles de representar. Como ya habíamos dicho cada celda actualiza su estado tomando información de ella misma y de un grupo de celdas a su alrededor que constituyen un vecindario; los más comunes son el de Neumann, donde son tomados los 4 vecinos ortogonales (norte, sur, este, oeste) además de el mismo (Figura 1.1 a), y el de Moore, que además de los vecinos ortogonales tiene en cuenta los diagonales y el mismo (Figura 1.1 b) [18].

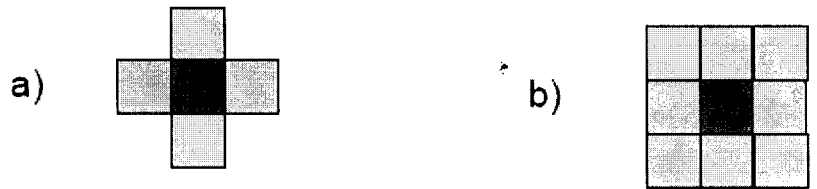


Fig. 1.1

La cantidad de estados en que puede estar una celda tiene que ser finito. Las reglas que determinan el siguiente estado de la celda se basan en los estados de las celdas que componen el vecindario, de esta forma si k es el número de estados posibles y n el número de celdas en el vecindario, existen k^n posibles reglas de transición.

1.3.1.2 Definición Formal de un Autómata Celular.

Podemos definir un Autómata Celular M como el cuádruplo siguiente:

$$M = \{A, Q, \Delta, N\}$$

Donde A representa un arreglo de D dimensiones, Q es el conjunto de estados de la celda, Δ la función de transición de estados y N el vecindario.

Un Autómata Celular (AC) es un arreglo D -dimensional con un Autómata de Estados Finitos (AEF) en cada posición o celda del arreglo [18]. Cada AEF toma como entradas los estados de las celdas de una región local finita N , que llamaremos vecindario; y además contiene un número de estados finitos Q , un alfabeto finito Σ , y una función de transición Δ que va desde el conjunto de las posibles configuraciones de los estados del vecindario al conjunto de estados. Sea $n = |N|$, la cantidad de celdas del vecindario, Δ queda definida de esta manera:

$$\Delta: Q^n \rightarrow Q$$

El alfabeto Σ esta formado por las posibles combinaciones de los estados Q del vecindario, por lo que $\Sigma = Q^n$. Sea $k = |Q|$, entonces

$$|\Sigma| = k^n$$

1.3.1.3 Autómatas Celulares de 2 Dimensiones

Los AC de dos dimensiones son los más utilizados en el estudio de sistemas físicos y biológicos [19]. Veamos un AC conocido como la regla de "Parity", el mismo se define utilizando el vecindario de von Neumann, con $Q = \{0,1\}$ y la función de transición es la siguiente:

$$a_{i,j}^{t+1} = (a_{i-1,j}^t + a_{i,j-1}^t + a_{i,j}^t + a_{i+1,j}^t + a_{i,j+1}^t) \bmod 2$$

Al iniciar este AC con una semilla en el centro, es decir una celda en el centro con 1 y las demás en 0, se obtienen los resultados que muestra la Fig. 1.2:

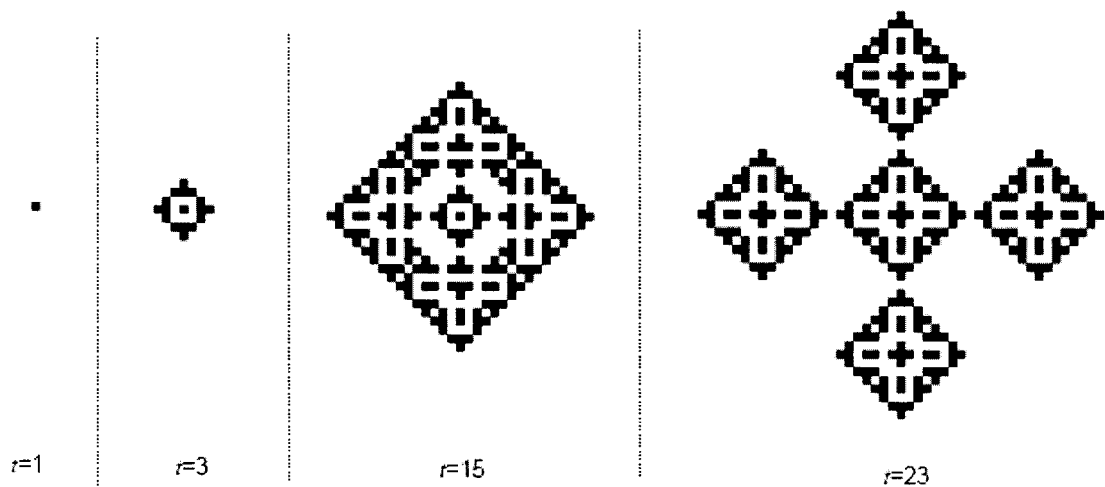


Fig. 1.2

1.3.1.4 Propiedades y clases de autómatas celulares.

Los AC son capaces de producir comportamientos complejos, que no es más que la formación de estructuras organizadas, como vimos en la reglas de Parity [20]. Esta complejidad es irreducible, por lo que teniendo un AC y una configuración inicial no podemos determinar a priori cual será su configuración trascurridas t generaciones.

S. Wolfram realizó uno de los más importantes intentos de clasificación de los AC en cuanto a su comportamiento; él propone las siguientes clases [21, 22]:

1.3 Modelación de sistemas biológicos

- Clase I: Evolución del AC lleva a un estado homogéneo.
- Clase II: Evolución del AC lleva a estructuras estables simples.
- Clase III: Evolución del AC lleva a patrones caóticos.
- Clase IV: Evolución del AC lleva a patrones complejos, de largo período.

Estas clases de AC propuestas por Wolfram son análogas al comportamiento encontrado en las soluciones de Sistemas de Ecuaciones Diferenciales (SED). Cuando la solución de un SED llega al mismo punto para cualquier condición inicial en un tiempo considerable, estamos en presencia de la Clase I; si dicha solución depende de parámetros y su comportamiento es estable entonces corresponde a la Clase II; a los SED que generan soluciones caóticas en dependencia de sus condiciones iniciales se les ubica en la Clase III; la Clase IV debido a su complejidad se sugiere que se incluyan los AC que no correspondan a ninguna de las clases anteriores [21, 22].

Auto Replicación.

Los AC auto replicables son aquellos que tienen la propiedad de crear una copia de si mismo, y fueron los que motivaron a von Neumann a desarrollar la teoría de AC [18]. El ejemplo más sencillo de un autómata auto replicable lo representa el autómata de Byl [18]. Este autómata funciona con 6 estados, 5 vecinos y 43 reglas de transición; la Fig. 1.10 muestra como este autómata partiendo de una configuración determinada es capaz de auto replicarse en 26 generaciones:

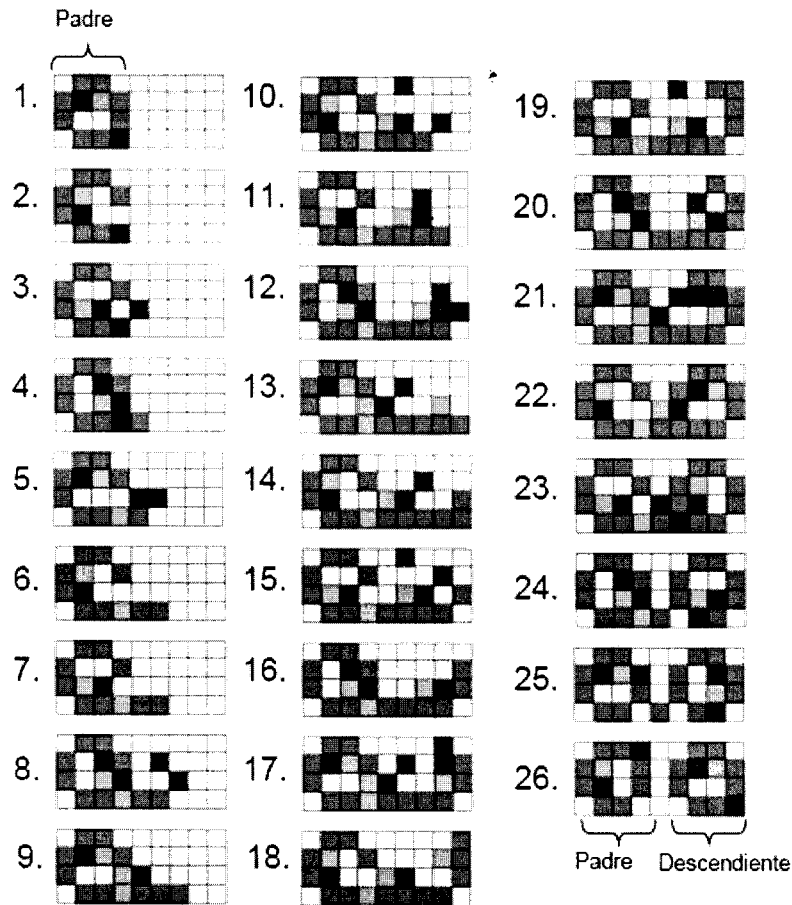


Fig. 1.3

1.3.1.5 Autómatas celulares probabilísticos.

Los AC probabilísticos pueden definirse de dos formas; una donde las reglas del autómata son aplicadas uniformemente a todas las celdas del espacio celular, existiendo para cada cambio de estado una probabilidad de ocurrencia, a este caso se le denomina AC probabilístico uniforme (ACPU); y otra donde las reglas no son uniformes y cada celda puede aplicar una regla u otra con distintas probabilidades, denominándose este caso AC probabilístico no uniforme (ACPN) [18].

Autómata Celular Probabilístico No Uniforme.

Un ACPN se define de la siguiente manera:

$$M = \{A, Q, \Delta, N, P\}$$

Donde A, Q, Δ , N tiene el mismo significado que en los AC, P se define como una función $P: \Delta \rightarrow Q$. La probabilidad de que una transición δ origine el nuevo estado q la denotaremos por $p(\delta, q)$. Se cumple que:

$$\forall \delta \in \Delta, \sum_{q \in Q} p(\delta, q) = 1$$

Autómata Celular Probabilístico Uniforme.

Un ACPU se define análogamente a los ACPN redefiniendo la función probabilística $P: \Delta \times Q \rightarrow Q$. La probabilidad de que la transición δ produzca cambio de estado de q a q' se denota por $p(\delta, q' | q)$ y la probabilidad de que no se produzca el cambio es $1 - p(\delta, q' | q)$.

Un Autómata Probabilístico.

Veamos un ejemplo de un ACPU que simula el crecimiento del liquen en un AC con 2 estados, 9 celdas en el vecindario y la siguiente regla de transición:

- Una celda pasa a 1 o se mantiene en 1 si tiene solo un vecino en el estado 1. Esto ocurre con una probabilidad $p = 0.1$.

A continuación mostramos el estado del AC después de 100 generaciones partiendo de una semilla central. La Figura 1.4 a) muestra el AC que acabamos de describir y la b) muestra lo que ocurre si $p = 1$.

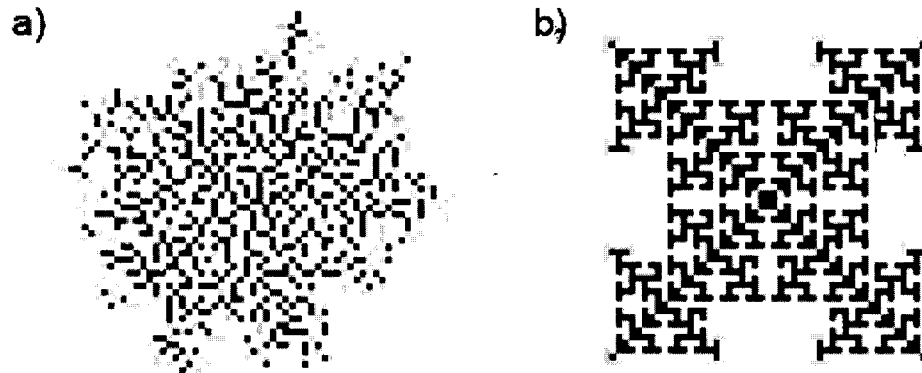


Fig. 1.4

Los AC probabilísticos incrementan el desorden pero simulan con mayor exactitud los procesos biológicos y físicos.

1.3.2 Dinámica de poblaciones.

Los sistemas biológicos que se pueden analizar desde la óptica de la dinámica de poblaciones son descritos de forma general por la Ley de Malthus [1, 25, 16]; que no es más que una ecuación diferencial con condiciones iniciales o Problema de Cauchy.

$$dP / dt = k \cdot P, P(0) = P_0$$

donde:

P: población (cantidad de miembros de la población).

t: tiempo (tiempo de estudio de la población).

k: Es una función $f(P_1, \dots, P_n, t)$, donde P_1, \dots, P_n representan las poblaciones presentes en el modelo. Es muy común que estas funciones sean definidas como constantes, $f(P_1, \dots, P_n, t) = a$, $a \in \mathfrak{R}$.

La solución para este problema de Cauchy es:

$$P = P_0 \cdot e^{k \cdot t}$$

1.3 Modelación de sistemas biológicos

Finalmente podría pensarse que estudiar aquellos sistemas biológicos que puedan ser descritos como un problema de dinámica de poblaciones es cuestión sencilla, pero la realidad es muy diferente si analizamos que nuestro modelo quedará totalmente descrito por un número de ecuaciones diferenciales que está en concordancia con el número de poblaciones interactuantes en el sistema objeto.

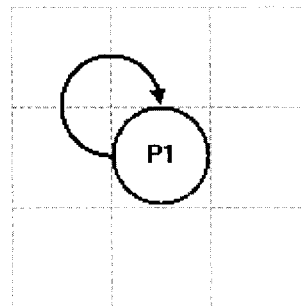
1.3.2.1 Interacciones presentes en el modelo.

Para el modelo realizado en este trabajo, basado íntegramente en la dinámica de poblaciones, se tienen en cuenta las formas más comunes de interacción entre dos poblaciones para las que se tienen las ecuaciones matemáticas que lo describen. Es necesario destacar que después de analizar intensamente la dinámica de las poblaciones se decidió tener en cuenta sólo los casos que se describen a continuación, por considerar que los mismos abarcan un gran número de modelos de sistemas biológicos.

Proliferación o nacimiento de una población.

Este diagrama de interacción (Fig. 1.5) representa la proliferación o nacimiento de una población. Aquí la velocidad con que prolifera la misma depende de la cantidad existente [3, 11]. La ecuación diferencial asociada a esta interacción es:

$$dP/dt = k*P$$

**Fig. 1.5*****Muerte de una población.***

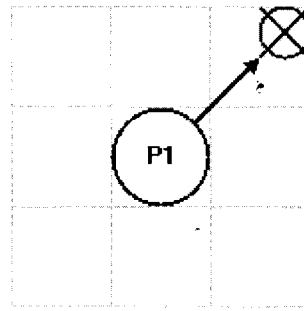


Fig. 1.6

De igual forma que la proliferación, la muerte de una población depende de la cantidad existente de la misma [3]. Así la ecuación que representa la muerte de una población está dada por:

$$dP/dt = -k*P$$

Diferenciación.

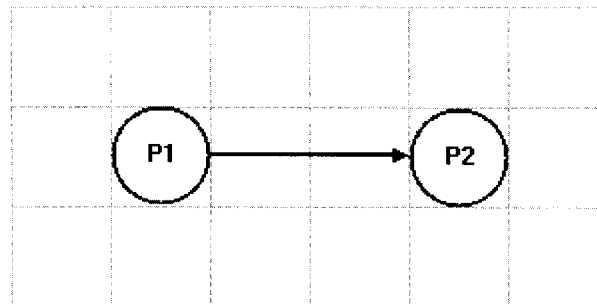


Fig. 1.7

Diferenciación o tránsito directo de una población a otra. Estas pueden ser diferentes o pueden ser dos estados de una misma población. En muchas ocasiones la cantidad de individuos de una población que son necesarios para obtener otra es más de uno, y por esto es necesario incluir en este tipo de relación los coeficientes estequiométricos de las poblaciones interactuantes, que no son más que las cantidades de individuos necesarios para obtener otra cierta cantidad de individuos de la población resultante; por ejemplo, 2 representantes de la población P_1 originan 1 representante de la población P_2 . En este tipo de relación se pierde cierta cantidad de individuos de la población P_1 y se ganan individuos de

la población P_2 , estas cantidades están determinadas por el número de sujetos de la población que se diferencia y por coeficientes o expresiones que determinan en que medida se ganan y se pierden individuos de las poblaciones involucradas. Las ecuaciones diferenciales que describen este proceso son las siguientes [13, 14]:

$$dP_1/dt = -c_1 * k_1 * P_1^{C_1}$$

$$dP_2/dt = c_2 * k_2 * P_1^{C_1}$$

donde:

c_1 y c_2 son los coeficientes estequiométricos para esta relación de las poblaciones P_1 y P_2 respectivamente.

k_1 y k_2 son coeficientes o expresiones que determinan la velocidad de pérdida y ganancia respectivamente.

Este caso se puede generalizar y hacer que más de una población se combinen para su diferenciación en otra población (Fig. 1.8).

En este caso las poblaciones P_1 y P_2 se diferencian en la población P_3 . Para este ejemplo las ecuaciones serian:

$$dP_1/dt = -c_1 * k_1 * P_1^{C_1} * P_2^{C_2}$$

$$dP_2/dt = -c_2 * k_1 * P_1^{C_1} * P_2^{C_2}$$

$$dP_3/dt = c_3 * k_2 * P_1^{C_1} * P_2^{C_2}$$

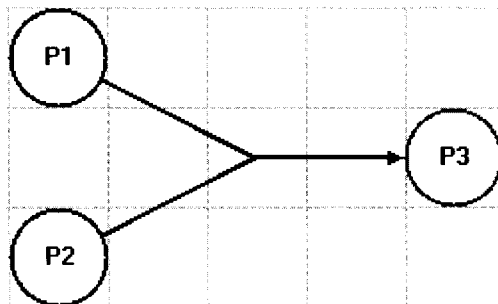


Fig. 1.8

Generalizando, si se tienen las poblaciones P_1, P_2, \dots, P_n que se diferencian en la población P_{n+1} , sea c_j el coeficiente estequiométrico de la población P_j en esta relación con $j = 1, \dots, n+1$, y k_1 y k_2 las expresiones que determinan la velocidad de pérdida y ganancia respectivamente, las ecuaciones que caracterizan este proceso son las siguientes:

$$dP_j/dt = -C_j * k_1 * \prod_{i=1}^n P_i^{C_i}$$

$$dP_{n+1}/dt = C_{n+1} * k_2 * \prod_{i=1}^n P_i^{C_i}$$

con $j = 1, 2, \dots, n$.

Producción de compuestos químicos.

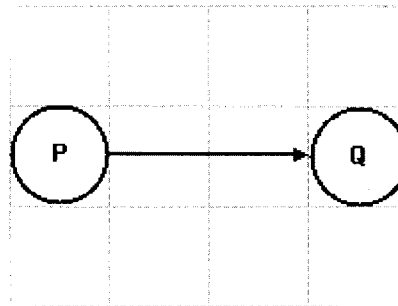


Fig. 1.9

Como resultado del metabolismo de una población se pueden formar determinados compuestos químicos. Aquí incorporamos una nueva característica a nuestro modelo; además de poblaciones podemos incluir compuestos químicos que también pueden ser modelados mediante una ecuación diferencial que representa el comportamiento de la cantidad de sustancia de dicho compuesto en el tiempo. La particularidad de este proceso es que no afecta la cantidad de individuos de la población o las poblaciones que originan la sustancia. La ecuación correspondiente es:

$$dQ / dt = c_q * k * P_p^C$$

donde:

c_q y c_p son los coeficientes estequiométricos² del compuesto químico Q y la población P respectivamente.

Catalizadores e inhibidores.

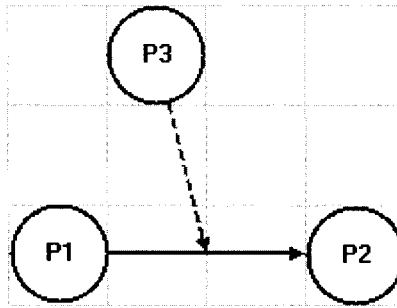


Fig. 1.10

Las poblaciones y sustancias químicas pueden actuar como reguladores de la velocidad de un proceso, bien sea como catalizador o inhibidor de la misma. Los catalizadores o inhibidores tienen como característica que ellos aceleran o retardan la velocidad del proceso sin variar su cantidad de sustancia o de individuos del mismo, según sea el caso. Por esta razón este tipo de relación materializa su acción en la función de velocidad de la relación que él regula de aquí que la presencia de este tipo de relación solamente nos indica que en la ecuaciones que caracterizan el proceso que el regula de alguna manera debe aparecer el regulador realizando la función correspondiente.

A continuación se muestran algunas de las funciones más usadas:

- $k \frac{P^C}{1 + P^C}$
- $k \frac{P^C}{1 - P^C}$
- $C \cdot \ln(P)$
- $k \cdot a^{-P}$

1.3 Modelación de sistemas biológicos

Si utilizamos la primera de las funciones que aquí exponemos las ecuaciones del modelo serían:

$$dP_1/dt = -C_1 * k_1 * \frac{P_3^{C_3}}{1 + P_3^{C_3}} * P_1^{C_1} \quad \}$$

$$dP_2/dt = C_2 * k_2 * \frac{P_3^{C_3}}{1 + P_3^{C_3}} * P_1^{C_1}$$

1.3.2.2 Ecuaciones que describen el modelo.

Como se ha podido apreciar en el modelo se pueden representar poblaciones y compuestos químicos que interactúan mutuamente mediante los procesos que ya describimos. La variación en el tiempo de la cantidad de población y de sustancia respectivamente son descritas por ecuaciones diferenciales que son originadas por cada uno de los procesos en que toma parte la población o el compuesto químico en cuestión.

Hasta ahora solo se han visto los distintos tipos de procesos y como influyen ellos en el desarrollo de las poblaciones interactuantes de forma aislada. Evidentemente la ecuación que describa el comportamiento de una población tiene que de alguna manera tener en cuenta todos los procesos en que participa dicha población. Esto se logra expresando la ecuación como una sumatoria donde cada término representa la acción de un proceso en la población en cuestión.

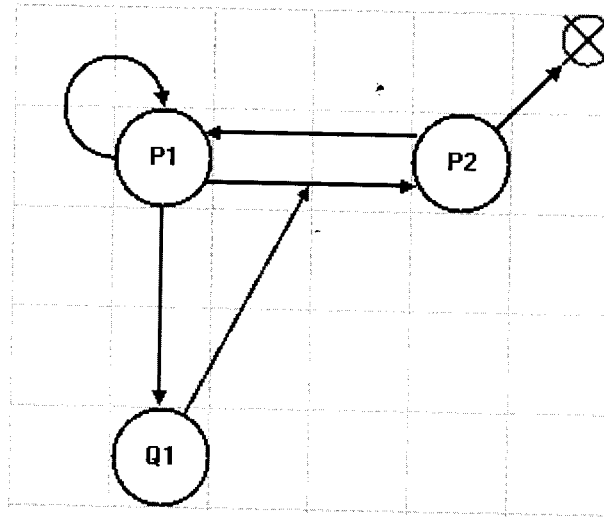


Fig. 1.11

Mediante este pequeño ejemplo se mostrará la forma de obtener el sistema de ecuaciones diferenciales que describe un modelo (Fig. 1.11). Consideremos que los coeficientes estequiométricos son todos iguales a 1, y además que en las dos relaciones de diferenciación existentes entre P_1 y P_2 los coeficientes de pérdida y ganancia son iguales a k_2 para la diferenciación de P_1 en P_2 y k_3 para la de P_2 en P_1 , k_1 es el coeficiente de la relación de proliferación de P_1 y k_4 el de muerte de P_2 , por último k_5 es el de la relación de producción de Q_1 y la expresión utilizada para manifestar la catálisis de Q_1 sobre el proceso de diferenciación de P_1 en P_2 es $\frac{Q_1}{1+Q_1}$. El sistema de ecuaciones diferenciales que describe este modelo es el siguiente:

$$dP_1/dt = k_1P_1 - k_2 \frac{Q_1}{1+Q_1} P_1 + k_3P_2$$

$$dP_2/dt = -k_4P_2 + k_2 \frac{Q_1}{1+Q_1} P_1 - k_3P_2$$

$$dQ_1/dt = k_5P_1$$

1.3 Modelación de sistemas biológicos

Este sistema se convierte en un problema con valores iniciales al considerar la cantidad de población y la cantidad de sustancia en el instante inicial.

1.4 Solución numérica del sistema de ecuaciones diferenciales.

El sistema de ecuaciones diferenciales a que nos enfrentamos tiene la forma general $dP_j/dt = f_j(P_1, \dots, P_n, t)$ con j variando de $1 \dots n$, siendo n el número de poblaciones y compuestos químicos del modelo. La complejidad que pueden llegar a tener las funciones f_j y el desconocimiento total de su naturaleza hacen que sea impracticable la obtención de la solución por métodos analíticos; quedando los métodos numéricos como solución más adecuada en este caso.

Existen diferentes métodos que podemos utilizar para la solución de nuestro sistema. A continuación veremos las principales características de los más utilizados.

1.4.1 Métodos de Runge-Kutta.

Los métodos de Runge-Kutta son basados en la Serie de Taylor, donde la cantidad de términos que se decidan tomar de la serie determinará la precisión del método. En estos métodos se le denomina paso(h) a $X_{n+1} - X_n$, este paso puede ser fijo obteniéndose puntos equidistantes; y también se puede variar haciendo aumentar el paso al doble en caso de detectar errores pequeños o reducirlo a la mitad en presencia de errores grandes, esto complica el programa de cálculo pero aumenta la eficiencia y en la mayoría de los casos son más rápidos [23].

El método de segundo orden requiere evaluar cada función 2 veces en cada paso de integración y el error local es del orden h^3 , de la misma forma el método de cuarto orden necesita 4 evaluaciones de cada función todos los pasos y posee un error local del orden h^5 ; esto posibilita que se pueda integrar con un paso mayor para lograr un error similar al que obtendríamos si utilizásemos el de orden dos, mejorando así el tiempo de ejecución del algoritmo [23].

1.4 Solución numérica del sistema de ecuaciones diferenciales.

Los métodos de Runge-Kutta son métodos de paso simple, que sólo requieren de los resultados del paso anterior, esto les brinda la posibilidad de autoiniciarse ya que parten de las condiciones iniciales. Al igual que todos los métodos de paso simple no presentan inestabilidad numérica para paso h suficientemente pequeño, esto implica que pequeños cambios en las condiciones iniciales del sistema solo origina cambios acotados en la solución [23].

1.4.2 Métodos de Adams.

Los métodos de Adams utilizan procedimientos de integración numérica utilizando un polinomio de interpolación para aproximar la función $f(x, y(x))$ de $y' = f(x, y)$, las diversas variantes de tomar este polinomio originan múltiples métodos de Adams [23].

Estos métodos son de paso múltiple ya que necesitan de los valores iniciales en varios puntos equidistantes para su iniciación, esta información se puede obtener aplicando primero algún método de paso simple como los de Runge-Kutta, constituyendo una desventaja de los métodos de paso múltiple ante los de paso simple. Otra característica desventajosa de los métodos de Adams comparándolos con los de Runge-Kutta es que poseen un error local mayor en procedimientos del mismo orden, teniendo así menos precisión. Una desventaja de los métodos Runge-Kutta es que requieren múltiples evaluaciones de la funciones en cada paso, y estos solo requieren una evaluación, siendo más rápidos. Los métodos de Adams poseen estabilidad condicional [23].

1.5 Herramientas para el desarrollo del sistema.

Para la elaboración del sistema se realizó un estudio sobre las posibles herramientas a utilizar en su desarrollo. Tomando en cuenta las tendencias actuales y las posibilidades que brindan los diferentes sistemas se hace la siguiente propuesta.

1.5.1 Tecnología .Net

Microsoft.NET es el conjunto de nuevas tecnologías en las que Microsoft ha estado trabajando durante los últimos años con el objetivo de obtener una plataforma sencilla y potente para distribuir el software en forma de servicios que puedan ser suministrados remotamente y que puedan comunicarse y combinarse unos con otros de manera totalmente independiente de la plataforma, lenguaje de programación y modelo de componentes con los que hayan sido desarrollados. Esta es la llamada plataforma .NET, y a los servicios antes comentados se les denomina servicios Web.

Para crear aplicaciones para la plataforma .NET, tanto servicios Web como aplicaciones tradicionales (aplicaciones de consola, aplicaciones de ventanas, servicios de Windows NT, etc.), Microsoft ha publicado el denominado kit de desarrollo de software conocido como .NET Framework SDK, que incluye las herramientas necesarias tanto para su desarrollo como para su distribución y ejecución y Visual Studio.NET, que permite hacer todo lo anterior desde una interfaz visual basada en ventanas.

El concepto de Microsoft.NET también incluye al conjunto de nuevas aplicaciones que Microsoft y terceros han (o están) desarrollando para ser utilizadas en la plataforma .NET. Entre ellas podemos destacar aplicaciones desarrolladas por Microsoft tales como Windows.NET, Hailstorm, Visual Studio.NET, MSN.NET, Office.NET, y los nuevos servidores para empresas de Microsoft (SQL Server.NET, Exchange.NET).

1.5.2 Lenguaje de programación C#.

C# es el nuevo lenguaje de propósito general diseñado por Microsoft para su plataforma .NET. Aunque es posible escribir código para la plataforma .NET en lenguajes como Visual Basic, C++ y otros; C# es el único que ha sido diseñado específicamente para ser utilizado en ella, por lo que la programación usando C# es mucho más sencilla e intuitiva que en cualquiera de los otros lenguajes ya que C# carece de elementos heredados innecesarios en .NET.

1.5 Herramientas para el desarrollo del sistema

En resumen, C# es un lenguaje de programación que toma las mejores características de lenguajes preexistentes como Visual Basic, Java o C++ y las combina en uno solo. El hecho de ser relativamente reciente no implica que sea inmaduro, pues Microsoft ha escrito la mayor parte de la BCL(Base Class Library) usándolo, por lo que su compilador es el más depurado y optimizado de los incluidos en el .NET Framework SDK

1.5.3 Matlab.

Matlab es un programa interactivo con un gran desempeño para el cálculo numérico computacional, con un enfoque directo hacia la computación científica. Posee un lenguaje de programación sencillo, que cuenta con un gran número de instrucciones dirigidas a la resolución de problemas científicos; además de una vasta colección de algoritmos computacionales para el trabajo con funciones matemáticas sencillas y complejas. Es muy utilizado en el desarrollo de algoritmos, simulación y modelación de prototipos, y en general en cualquier problema que necesite cálculos numéricos.

Matlab incluye librerías que le permiten interactuar con programas escritos en otros lenguajes como C, Visual Basic o Fortran. Matlab Funciona en Windows y Unix.

Capítulo 2. Características del sistema.

2.1 Análisis del sistema.

Como se ha podido ver en el Capítulo 1 pueden existir diferentes modelos que arrojen resultados similares a los experimentales, pudiendo ser aceptados cualquiera de ellos. En la modelación de sistemas biológicos el más utilizado y tradicional es mediante sistemas de ecuaciones diferenciales, y los autómatas celulares vienen siendo una alternativa muy interesante.

En este trabajo se pretende desarrollar un sistema que permita realizar modelos de sistemas biológicos por medio de sistemas de ecuaciones diferenciales, permitiendo hacer análisis simples sobre el comportamiento de los sistemas modelados.

El nombre escogido para el sistema es PCDBS (Plataforma Computacional para el Desarrollo de la Biología de Sistemas).

2.1.1 Requisitos del sistema.

Basado en lo descrito en el Capítulo 1, y teniendo en cuenta que el objetivo fundamental esta primera etapa es realizar un prototipo con algunas funcionalidades elementales, esta primera versión debe cumplir con los siguientes requisitos:

- Creación de modelos de sistemas biológicos, descritos mediante dinámica de poblaciones, donde se puedan expresar todas las posibles interacciones que existen entre poblaciones y compuestos químicos.
- Formación y solución numérica del sistema de ecuaciones diferenciales que describe el sistema biológico en estudio.

2.1 Análisis del sistema.

- Interfaz amigable que permita la realización del modelo de forma sencilla e intuitiva, que le permita a la persona que realiza el experimento abstraerse del modelo matemático que determina al mismo.
- Mostrar las gráficas de dinámica de las poblaciones y el mapa de fases.

2.2 Diseño e implementación del sistema.

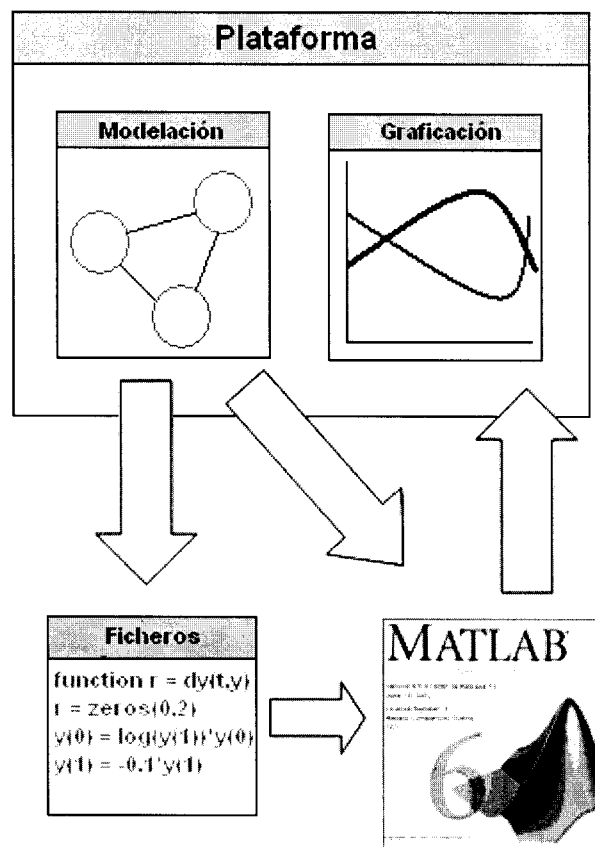


Fig. 2.1

La aplicación está compuesta por dos módulos fundamentales, el de modelación y el de graficación, además interactúa con Matlab para la solución numérica del sistema de ecuaciones diferenciales (Fig. 2.1).

 2.2 Diseño e implementación del sistema.

2.2.1 Módulo de Modelación.

Este módulo es que permite realizar las modelaciones de los sistemas biológicos. En él se pueden representar poblaciones y compuestos químicos como los componentes fundamentales de los sistemas biológicos; además de las posibles relaciones que puedan existir, descritas en el capítulo anterior. Otra función de este módulo es generación del sistema de ecuaciones diferenciales, realizando una salida a ficheros de la misma en formato de una función Matlab, además interactúa con este software matemático para la solución numérica del problema.

2.2.1.1 Modelación de sistemas biológicos.

Para la modelación de los sistemas biológicos se utilizó una estructura similar a un grafo orientado, en el cual los vértices o nodos representan las poblaciones o compuestos químicos y los arcos las relaciones que existen entre ellos. Las particularidades que diferencian la estructura utilizada de los grafos convencionales es que pueden existir arcos de nodos a arcos y arcos que se originen en más de un nodo.

Veamos una formalización de la estructura descrita. Un grafo G es definido por una terna $G = (N, A, \Delta)$ donde N es el conjunto de los nodos que conforman el grafo, A el conjunto de arcos, y Δ una aplicación definida como sigue:

$$\Delta: A \rightarrow ((\wp(N) - \phi) \times (N \cup A))$$

Mediante esta aplicación es asignado a cada arco de A un conjunto de nodos orígenes no nulo y un nodo o arco de llegada.

En la implementación de dicha estructura se utilizaron listas, donde el grafo esta compuesto por una lista de nodos que representa a N y una lista de procesos o relaciones que representa a A (Fig. 2.2).

2.2 Diseño e implementación del sistema.

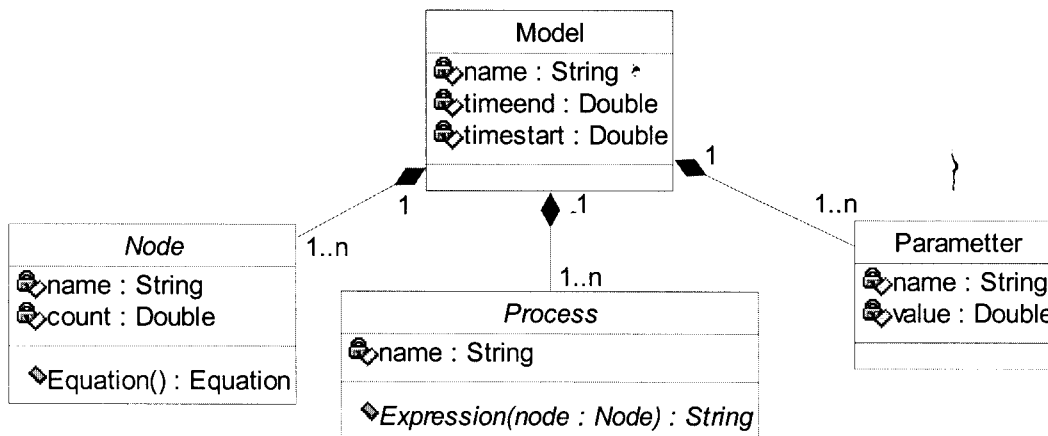


Fig. 2.2

Además el modelo posee los atributos timestart y timeend que determinan el intervalo de tiempo en que será estudiado el modelo. Es posible también definir parámetros que pueden estar involucrados en las expresiones de velocidad de uno o más procesos, siendo interesante analizar el comportamiento del sistema al hacer variar estos parámetros en un conjunto de valores.

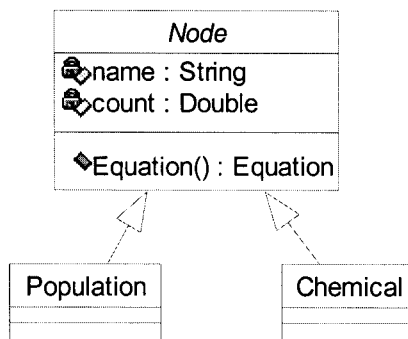


Fig. 2.3

La clase Node es una clase abstracta que es utilizada para representar los posibles tipos de nodos o vértices de nuestro grafo; las poblaciones son representadas por la clase Population y los compuestos químicos por la clase Chemical (Fig. 2.3).

2.2 Diseño e implementación del sistema.

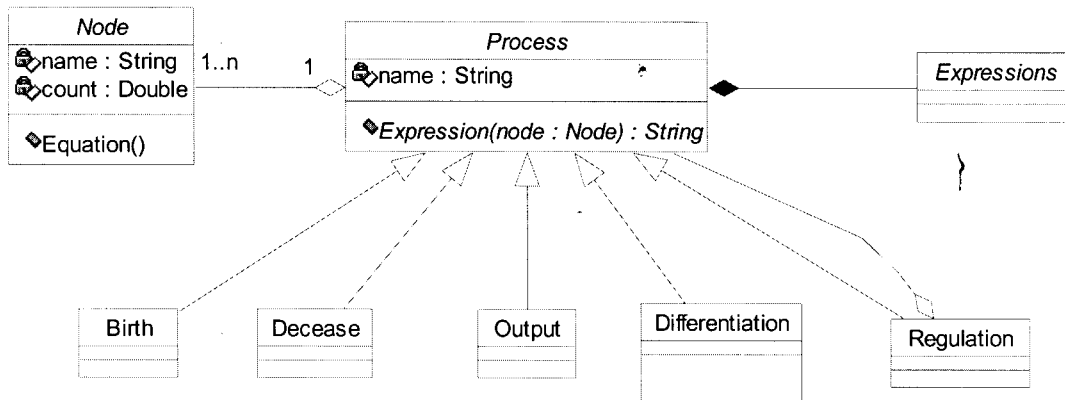


Fig. 2.4

La clase Process también es abstracta, útil para definir los procesos que pueden existir en el modelo, la misma esta compuesta por un conjunto de nodos, donde cada clase que deriva de ella conoce los nodos que involucra (Fig. 2.4).

Los procesos están compuestos también por expresiones que determinan la velocidad del proceso. En el modelo sólo son incluidos procesos que poseen una o dos expresiones, de aquí surgen las clases OneExpression y TwoExpression que son utilizadas en los procesos que necesitan una y dos expresiones respectivamente (Fig. 2.5).

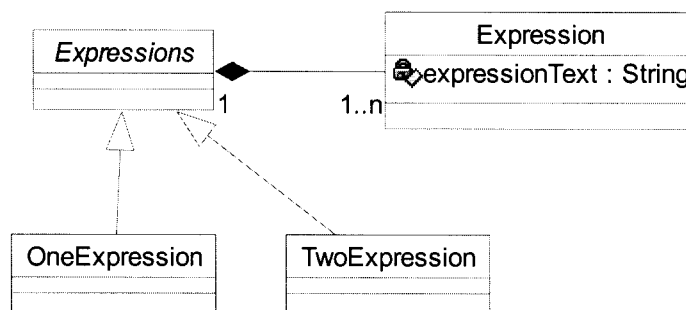


Fig. 2.5

La clase Expression representa una expresión matemática que determina la velocidad de un proceso. Esta expresión matemática es tratada como una cadena

2.2 Diseño e implementación del sistema.

de caracteres y para la misma se definió un lenguaje muy simple que es descrito a continuación:

- “+” → operador de suma.
- “-” → operador de resta.
- “*” → operador de multiplicación.
- “/” → operador de división.
- “^” → operador de potencia.
- “sin” → función seno.
- “cos” → función coseno.
- “tan” → función tangente.
- “log” → función logaritmo natural.
- “exp” → función exponencial de base e (euler).
- “sqrt” → función raíz cuadrada.
- “pi” → constante π .
- “t” → variable tiempo.

La gramática utilizada para este lenguaje fue la siguiente:

$$S \rightarrow A+S \mid A-S \mid A$$

$$A \rightarrow B*A \mid B/A \mid B$$

$$B \rightarrow C^B \mid C$$

$$C \rightarrow \langle id \rangle \mid \langle num \rangle \mid t \mid pi \mid +C \mid -C \mid FD \mid D$$

$$D \rightarrow (S)$$

$$F \rightarrow sin \mid cos \mid tan \mid sqrt \mid log \mid exp$$

Donde $\langle id \rangle$ y $\langle num \rangle$ se definen de la siguiente manera:

2.2 Diseño e implementación del sistema.

<id>: [a.. z, A..Z, _] [a...z, A...Z, 0...9, _]*

<num>: [0...9]+.[0..9]+ | [0...9]+

En el sistema de ecuaciones diferenciales cada nodo genera una ecuación que se obtiene a partir de los procesos que involucran al nodo. Para la representación de una ecuación del sistema de ecuaciones que representa al modelo se utilizó la clase Equation (Fig. 2.6). Esta clase esta compuesta por el nodo que la origina y por una expresión que es obtenida a partir de las expresiones de los procesos en los cuales participa dicho nodo.

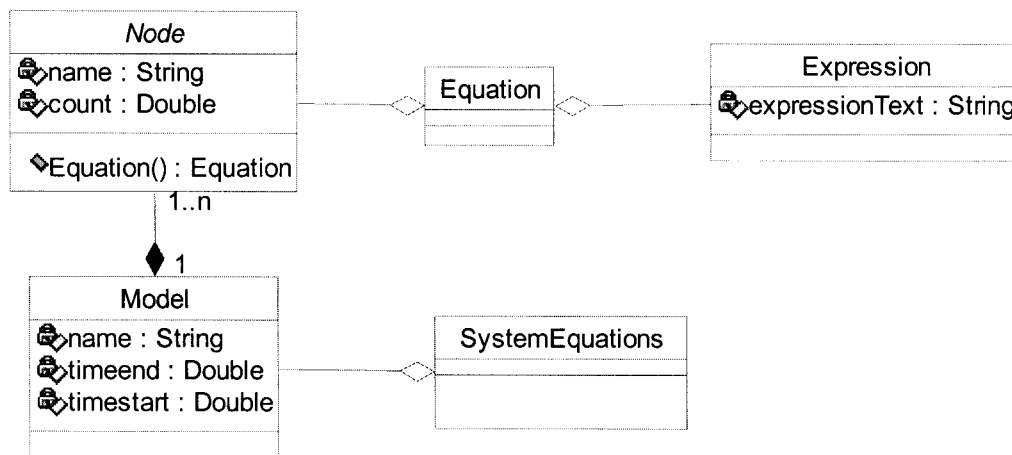


Fig. 2.6

Con este conjunto de clases que se han descrito brevemente es posible representar el grafo explicado y con él podemos modelar un conjunto amplio de sistemas biológicos.

2.2 Diseño e implementación del sistema.

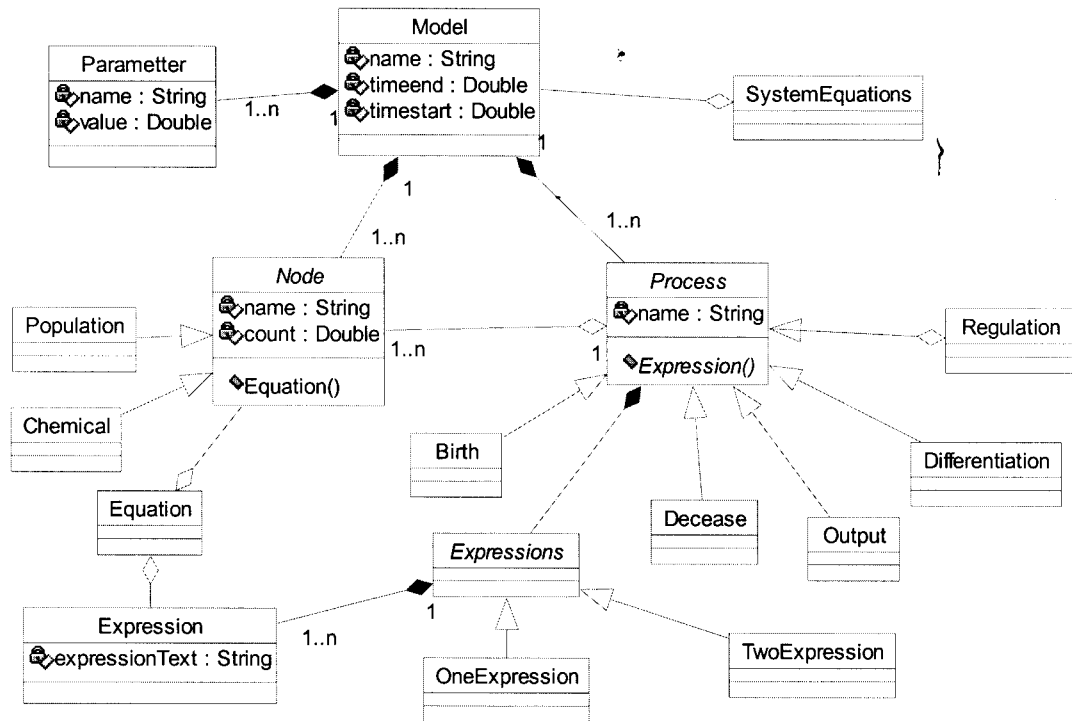


Fig. 2.7

2.2.1.2 Fichero de ecuaciones diferenciales.

Una de las funciones del módulo de modelación es generar un fichero que represente el sistema de ecuaciones diferenciales y que será utilizado por Matlab para la solución numérica del mismo.

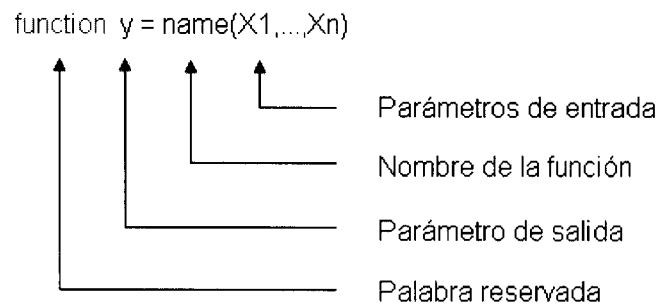


Fig. 2.8

2.2 Diseño e implementación del sistema.

Este fichero tiene el formato de las funciones en Matlab, el cual tiene un encabezado donde se define el nombre de la función y los parámetros de entrada y de salida (Fig. 2.8), en nuestro caso llamaremos “dy” a la función, y consta de dos parámetros de entrada “t” y “y”, el parámetro “t” representa el tiempo y el parámetro “y” es un vector columna con los valores de las variables involucradas en el sistema; además la función retorna un parámetro “r” que no es más que el vector columna resultante de evaluar los valores de “t” y “y” en el sistema de ecuaciones. La Fig. 2.9 muestra un ejemplo de un fichero utilizado en el cálculo de la solución numérica de un sistema de ecuaciones diferenciales.

```
function r = dy(t,y)
r = zeros(3,1)
r(1) = -(0.1)*y(1)+1*(0.3)*y(3)^1-1*(0.4)*y(1)^1
r(2) = (0.2)*y(2)+1*(0.4)*y(1)^1-1*(0.5)*y(2)^1
r(3) = -1*(0.3)*y(3)^1+1*(0.5)*y(2)^1
end
```

Fig. 2.9

2.2.1.3 Interacción con Matlab.

Una vez generado el sistema de ecuaciones diferenciales que modela al sistema biológico el próximo paso sería la obtención de su solución numérica con la utilización del Matlab. Para ello se crea una instancia de la aplicación Matlab, a través de la cual es posible ejecutar comandos, funciones del matlab y funciones definidas en ficheros externos, además es posible obtener el valor de las variables.

Para la solución numérica de sistemas de ecuaciones diferenciales Matlab brinda un conjunto de funciones donde las más utilizadas son ode45, ode23 y ode113. La función ode45 implementa el método de Runge-Kutta, combinando los de orden 4 y 5, de la misma forma el ode23 combina los métodos de orden 2 y 3; la función ode113 implementa un método de Adams.

La función utilizada es la ode45 con la siguiente sintaxis:

$$[T, Y] = \text{ode45}(\text{odefun}, \text{tspan}, y0)$$

2.2 Diseño e implementación del sistema.

Donde:

odefun: es la función que representa el sistema de ecuaciones diferenciales, en nuestro caso es la función "dy" que fue comentada en la sección anterior.

tspan: intervalo de tiempo para el cual se desea conocer la solución del sistema de ecuaciones diferenciales, se define mediante un vector de dos componentes donde la primera de ellas es el tiempo inicial y la segunda el tiempo final.

y0: valores de las variables que intervienen en el sistema en el instante inicial.

T: vector columna donde es devuelto el valor del tiempo en cada paso de integración.

Y: matriz donde es devuelto el valor obtenido de cada variable para cada paso de integración.

2.2.1.4 Interfaz visual de modelación.

Con el objetivo de crear una interfaz amigable y en la cual se puedan crear las modelaciones de sistemas biológicos se desarrollo un componente que brindara las facilidades necesarias para el diseño de un modelo.

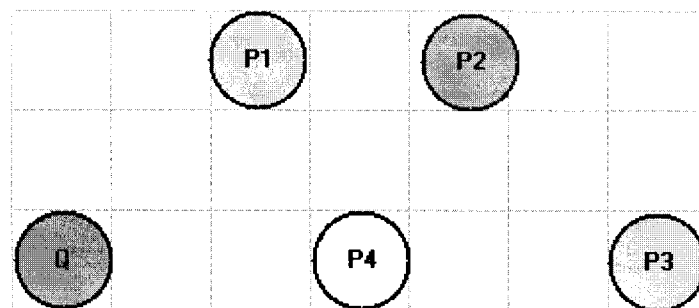


Fig. 2.10

En dicho componente las poblaciones y los compuestos químicos serán representadas con círculos de diferentes colores, donde cada color determina una especie o sustancia química según sea el caso (Fig. 2.10).

Para la representación de los procesos que tienen lugar en los sistemas biológicos se utilizaron arcos que determinan las poblaciones o compuestos químicos que

intervienen en cada proceso. La forma de representación es la misma utilizada en la sección 1.4.1 para la descripción de cada uno de estos procesos.

El componente desarrollado también brinda la posibilidad de encapsular un subgrafo del grafo original. De esta forma el usuario puede simplificar la visualización de la estructura del modelo (Fig. 2.11 y Fig. 2.12).

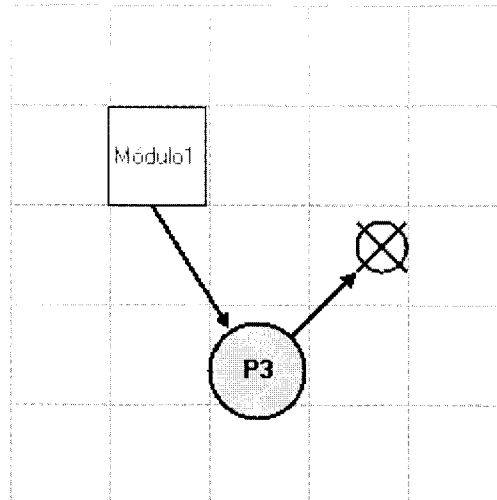


Fig. 2.11

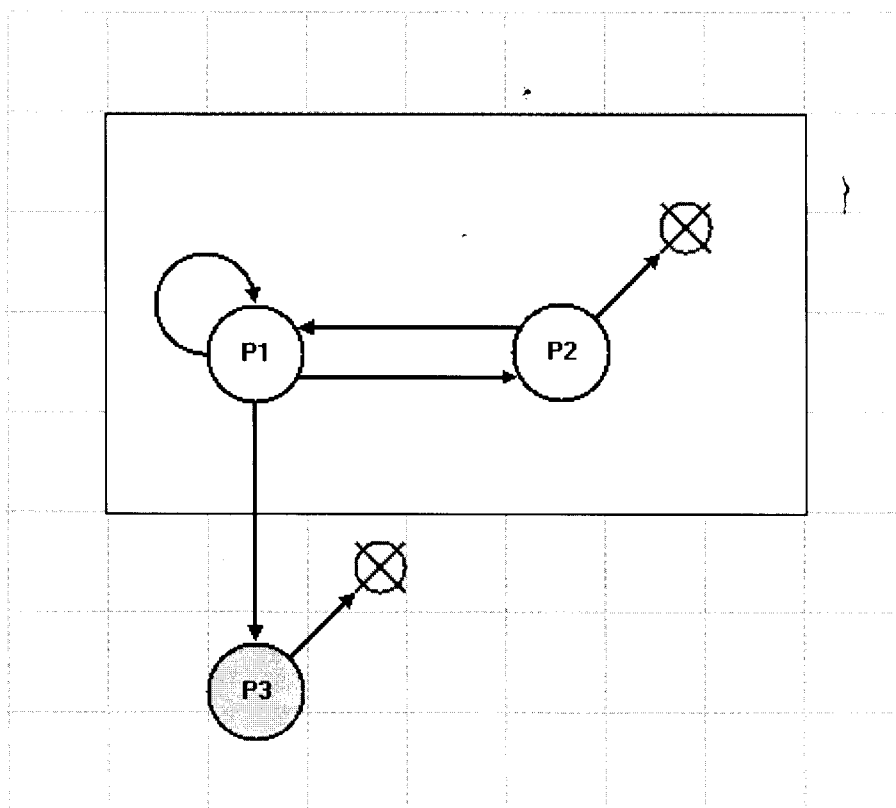


Fig. 2.12

2.2.2 Módulo de graficación.

Este módulo es el responsable de visualizar las gráficas que muestran el desarrollo de las poblaciones y compuestos químicos que intervienen en el modelo a través del tiempo (Fig. 2.13); además muestra los mapas de fases que origina el modelo (Fig. 2.14). Estas gráficas son conformadas con las soluciones numéricas del sistema de ecuaciones diferenciales que se obtienen con la ayuda de Matlab.

2.2 Diseño e implementación del sistema.

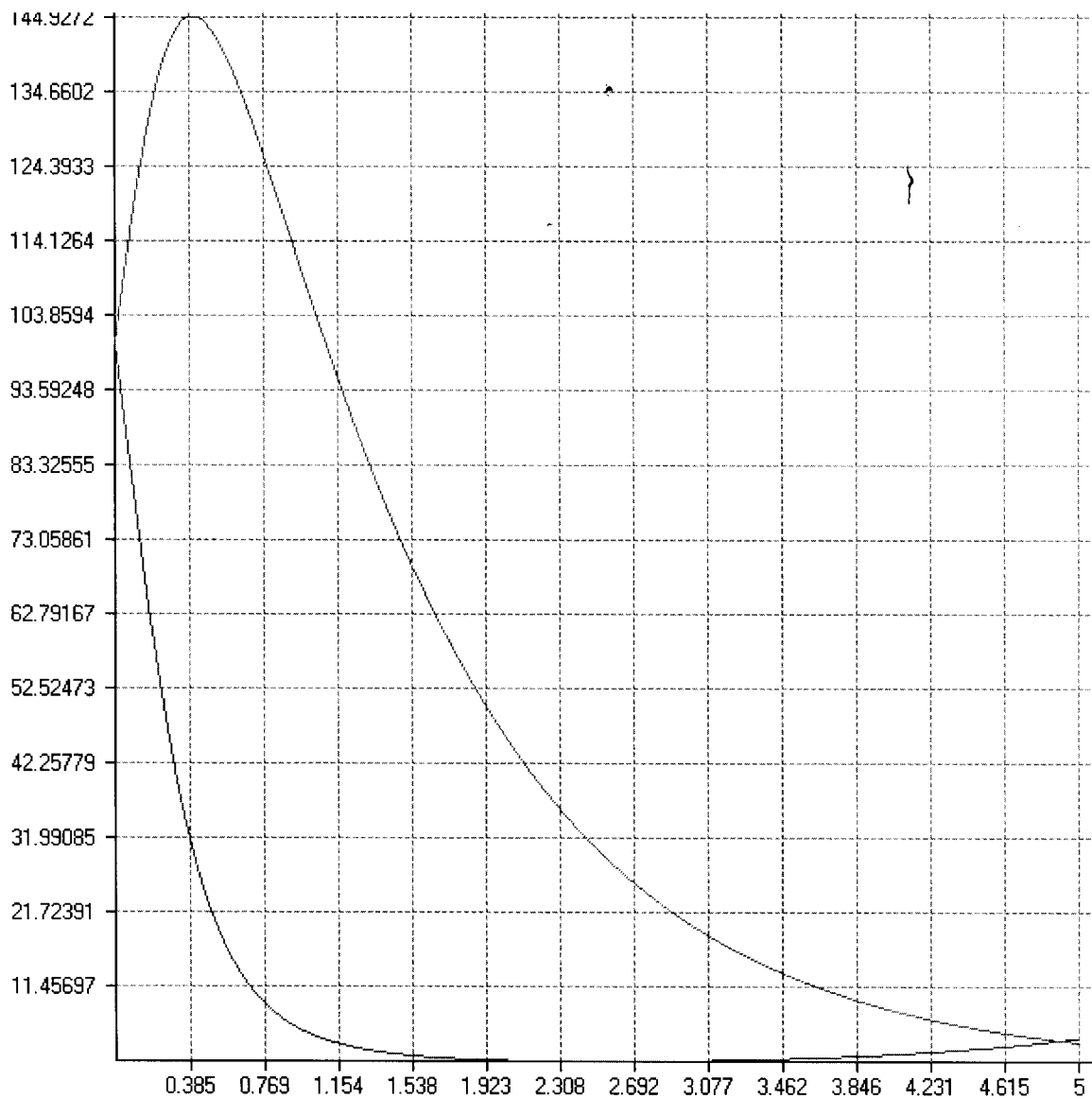


Fig. 2.13

Un mapa de fases se construye tomando un par de variables del sistema y representando el valor que ocupan dichas variables en el mismo instante de tiempo (Fig. 2.14). Estos mapas son útiles para la representación geométrica de la estabilidad del sistema.

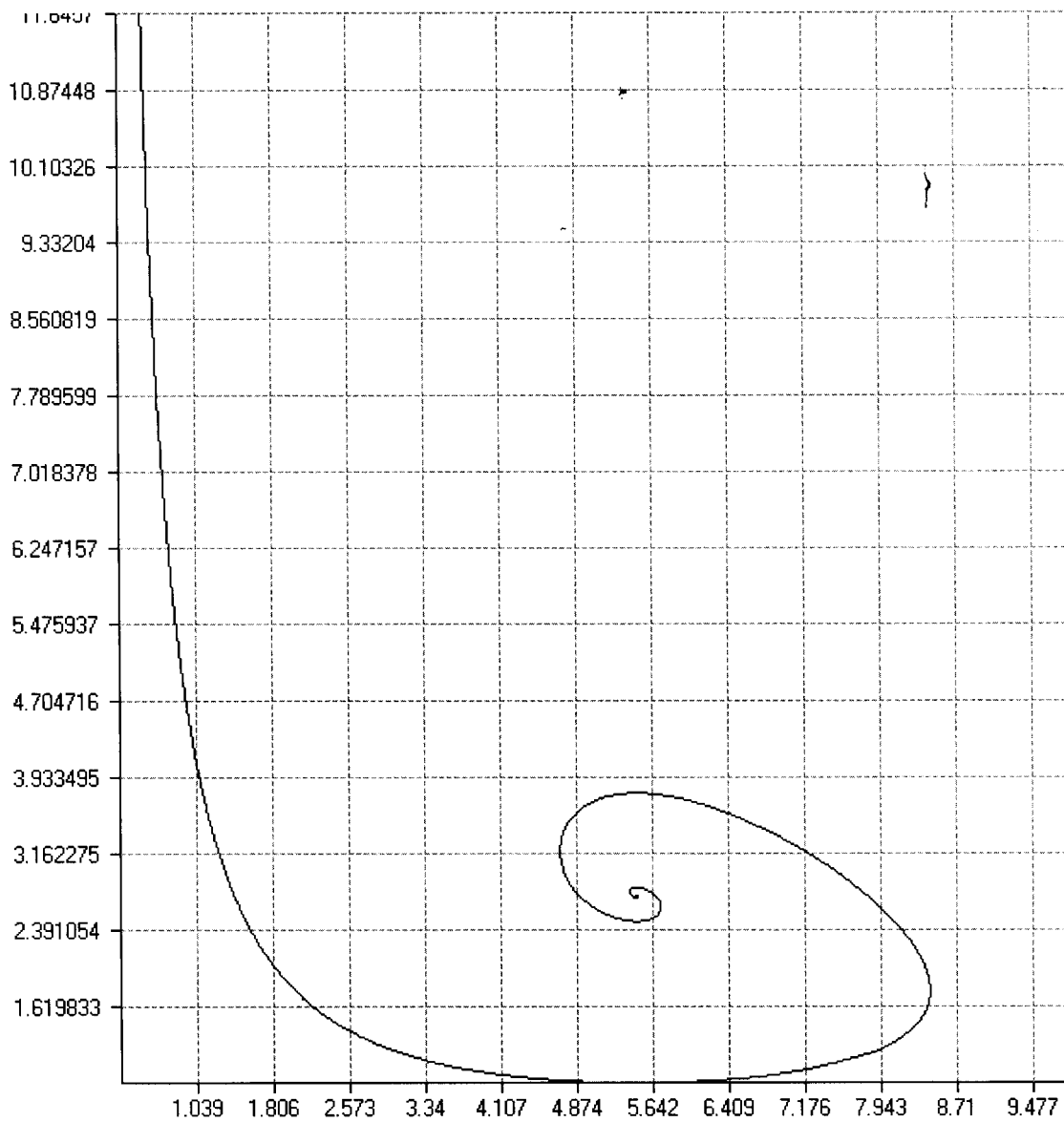


Fig. 2.14

2.3 Descripción del sistema PCDBS.

Como ha sido mencionado, el sistema se denomina PCDBS; el mismo posee una ventana principal que es donde se desarrollan la mayoría de las utilidades que brinda el sistema (Fig 2.15).

2.3 Descripción del sistema PCDBS.

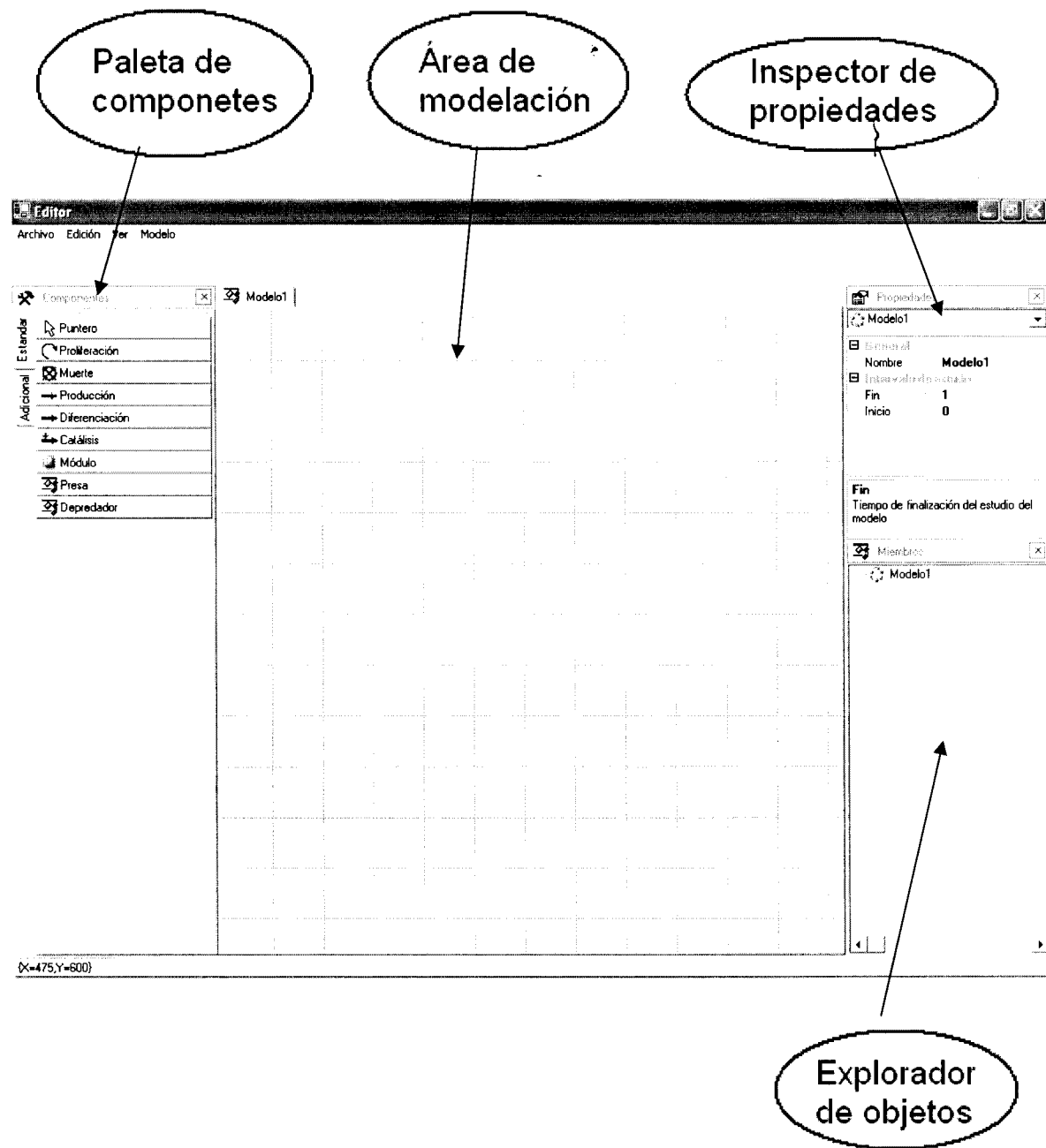


Fig. 2.15

2.3.1 Paleta de componentes.

La función de la paleta de componentes es contener un conjunto de modelos básicos para la construcción de modelos biológicos más complejos, las relaciones

2.3 Descripción del sistema PCDBS.

que se pueden crear en los modelos así como las poblaciones, compuestos químicos que el usuario quiera definir. En este primer prototipo solo se encuentran las relaciones y el usuario solo puede definir poblaciones y compuestos químicos, quedando para futuras versiones la opción de añadir componentes validados a la paleta.

2.3.2 Área de modelación.

En esta área es donde ponen las poblaciones, compuestos químicos y relaciones que se toman de la paleta de componentes, su función fundamental es mostrar el modelo de una forma sencilla y fácil de manipular; donde el usuario puede dar la estructuración visual del modelo que considere adecuada. En esta área también se brindan las salidas gráficas del sistema.

2.3.3 Inspector de propiedades.

El inspector de propiedades juega un papel muy importante en el sistema; ya que es aquí donde el usuario define las expresiones y parámetros que definen el sistema, así como realizar modificaciones a los nombres de las variables y demás propiedades de los diferentes miembros del modelo que sean modificables (Fig. 2.16).

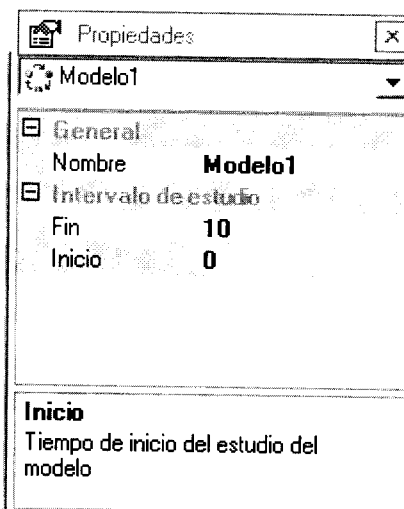


Fig. 2.16

2.3.4 Explorador de objetos.

En el explorador de objetos se muestran todos los componentes que pertenecen al modelo, brindando de esta forma una opción complementaria al área de modelación para acceder a dichos componentes.

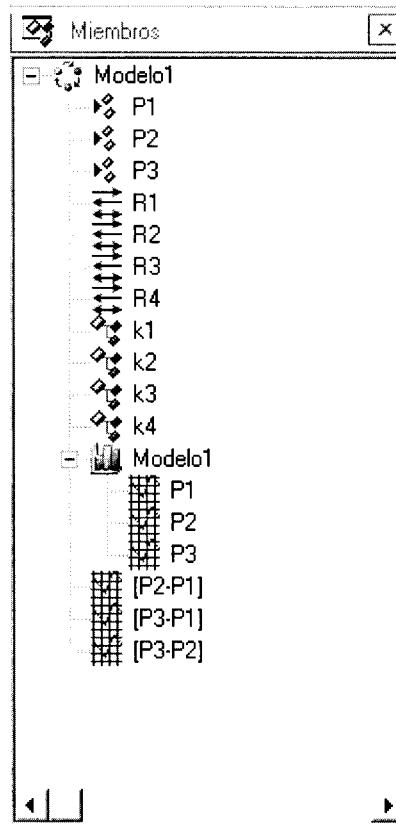


Fig. 2.17

Capítulo 3.

Modelación de un sistema biológico concreto.

En este capítulo se tomará un modelo conocido y estudiado en la dinámica de poblaciones, para realizar una modelación mediante el sistema y comparar los resultados obtenidos con los ya conocidos.

El modelo escogido es el que muestra la relación presa-depredador descrito por las ecuaciones de Lotka-Volterra; es un modelo muy simple cuyo comportamiento ha sido muy bien estudiado.

3.1 Modelo matemático de Lotka-Volterra.

El matemático italiano Vito Volterra (1860-1940), después de haberse interesado por la ecología matemática, estudió los registros de las pesquerías del Mar Adriático Superior y observó que, durante y después de la Primera Guerra Mundial, cuando la pesca había disminuido drásticamente, la proporción de los depredadores había aumentado.

Este hecho lo llevo a estudiar ese problema de una manera más general, logrando encontrar un modelo que representa este fenómeno.

Como se ha explicado para realizar un modelo es imprescindible precisar que parámetros y restricciones tendrá el mismo, escogiendo siempre las características más importantes y de mayor interés de estudio en el sistema. Volterra para realizar su estudio estableció las siguientes normativas:

- Que la especie depredadora se alimenta exclusivamente de la especie presa, mientras que ésta se alimenta de un recurso que se encuentra en el hábitat en grandes cantidades y se considera inagotable.
- Que ambas poblaciones eran homogéneas, es decir, no intervienen factores como la edad o el sexo.
- Medio ambiente homogéneo, es decir, no existen cambios en las características físicas, biológicas y demás del hábitat.

3.1 Modelo matemático de Lotka-Volterra.

- Que los encuentros de la especie depredadora con la especie presa son igualmente probables.

Con estas suposiciones encontró que solo necesitaba dos variables para representar el problema: el tamaño poblacional de la especie depredadora y el de la especie presa; además supuso que las mismas solo dependen del tiempo y de las relaciones internas del sistema.

Los primeros resultados que obtuvo fueron que en ausencia de depredadores, el desarrollo de la población de presas se podía expresar mediante la Ley de Malthus, si hacemos X la cantidad de población de la presa tenemos que:

$$dX / dt = aX(t)$$

de la misma manera, si hacemos Y la cantidad de población depredadora, en ausencia de presas se tiene que:

$$dY / dt = -cY(t)$$

La acción de depredación supone un incremento en la población de depredadores y una disminución en la población de presas. Basándose en el hecho de que cuantos más encuentros existan por unidad de tiempo entre las dos poblaciones mayor será la actividad depredadora, concluyó que el número de encuentros es igual al producto algebraico del tamaño de las poblaciones. Por lo que nuestro modelo queda de la siguiente manera:

$$dX / dt = aX(t) - bX(t)Y(t)$$

$$dY / dt = -cY(t) + dX(t)Y(t)$$

donde :

a : tasa instantánea de aumento de la población de presas en ausencia de depredadores.

c : tasa instantánea de disminución de la población de depredadores en ausencia de presas.

3.1 Modelo matemático de Lotka-Volterra.

b: susceptibilidad de las presas a la depredación.

d: habilidad de depredación de esta especie.

Este modelo determina un comportamiento periódico en forma de oscilaciones de las dos poblaciones del modelo presa-depredador (Fig. 3.1). El mapa de fases de este modelo muestra una curva cerrada en forma de circunferencia (Fig 3.2).

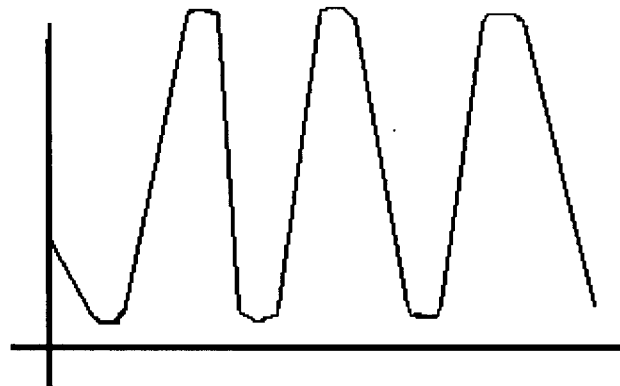


Fig. 3.1

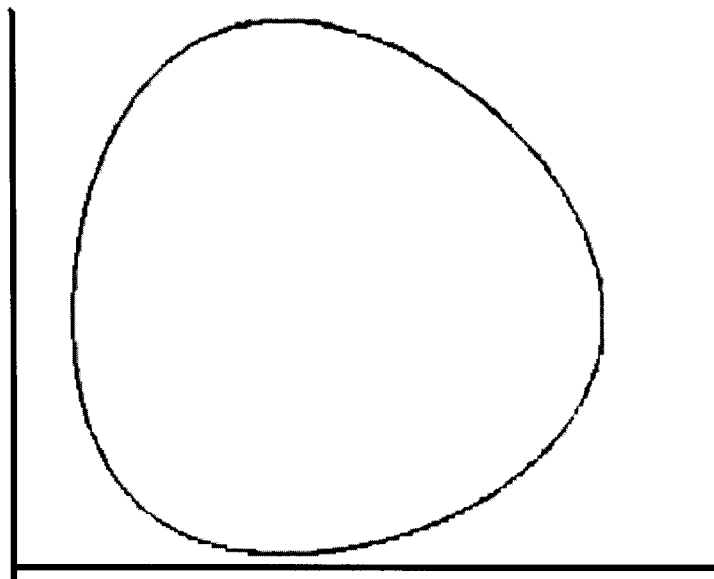


Fig. 3.2

3.1 Modelo matemático de Lotka-Volterra.

3.1.1 Modelación con PCDBS.

A continuación se describe como es posible realizar el modelo depredador-presa, del cual han sido descritas sus principales características, con la utilización de PCDBS.

Se definen dos poblaciones, la población X actúa como presa y la población Y de depredadora; se define una relación de proliferación o nacimiento para la población X, una proceso de muerte para la población Y, y una diferenciación de la población X en la población Y (Fig. 3.3).

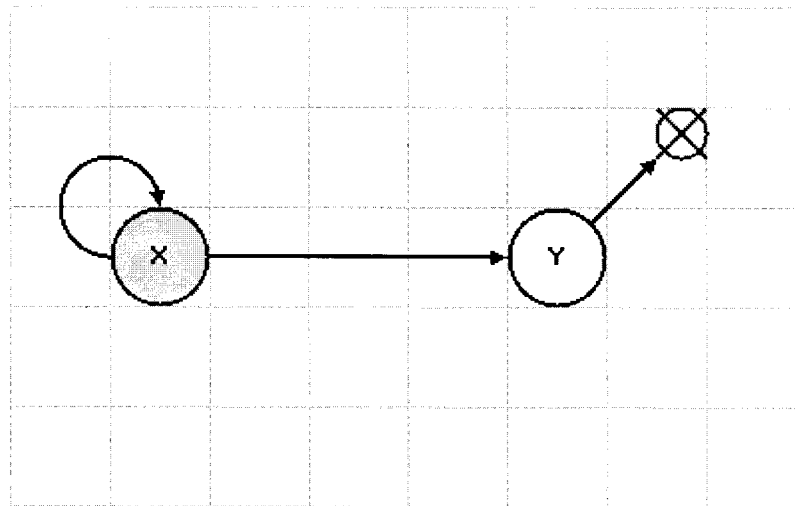


Fig. 3.3

Haciendo clic en las relaciones y a través del inspector de objetos es posible definir las expresiones de cada una de ellas (Tabla 3.1, Fig. 3.4).

Relaciones	Expresiones
Proliferación de X	a
Muerte de Y	c
Diferenciación de X en Y(Ganancia)	d*Y

3.1 Modelo matemático de Lotka-Volterra.

Diferenciación de X en Y(Pérdida)	$b \cdot Y$
-----------------------------------	-------------

Tabla 3.1

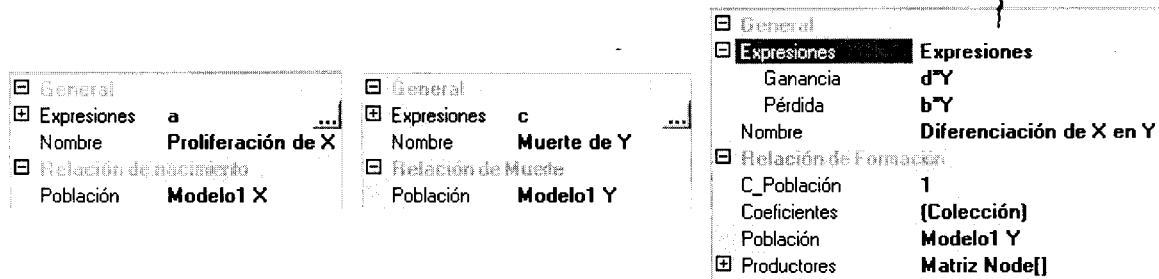


Fig. 3.4

De la misma manera y con la utilización del explorador de objetos se definen los valores de las constantes a, b, c, y d (Tabla 3.2).

Constantes	Valor
a	0,9
b	0,01
c	0,9
d	0,01

Tabla 3.2

Solo queda definir las cantidades de individuos iniciales de cada población y el intervalo de tiempo que queremos observar del desarrollo del sistema (Tabla 3.3).

	Valores
Población presa (X)	100
Población depredadora (Y)	100

3.1 Modelo matemático de Lotka-Volterra.

Inicio del estudio del modelo	0
Fin del estudio del modelo	20

Tabla 3.3

Mediante la opción de menú **Modelo**→**Obtener Resultados** se realiza el proceso de obtención de la solución numérica de modelo y se visualizan las gráficas del comportamiento de las poblaciones en el tiempo, así como los mapas de fases.

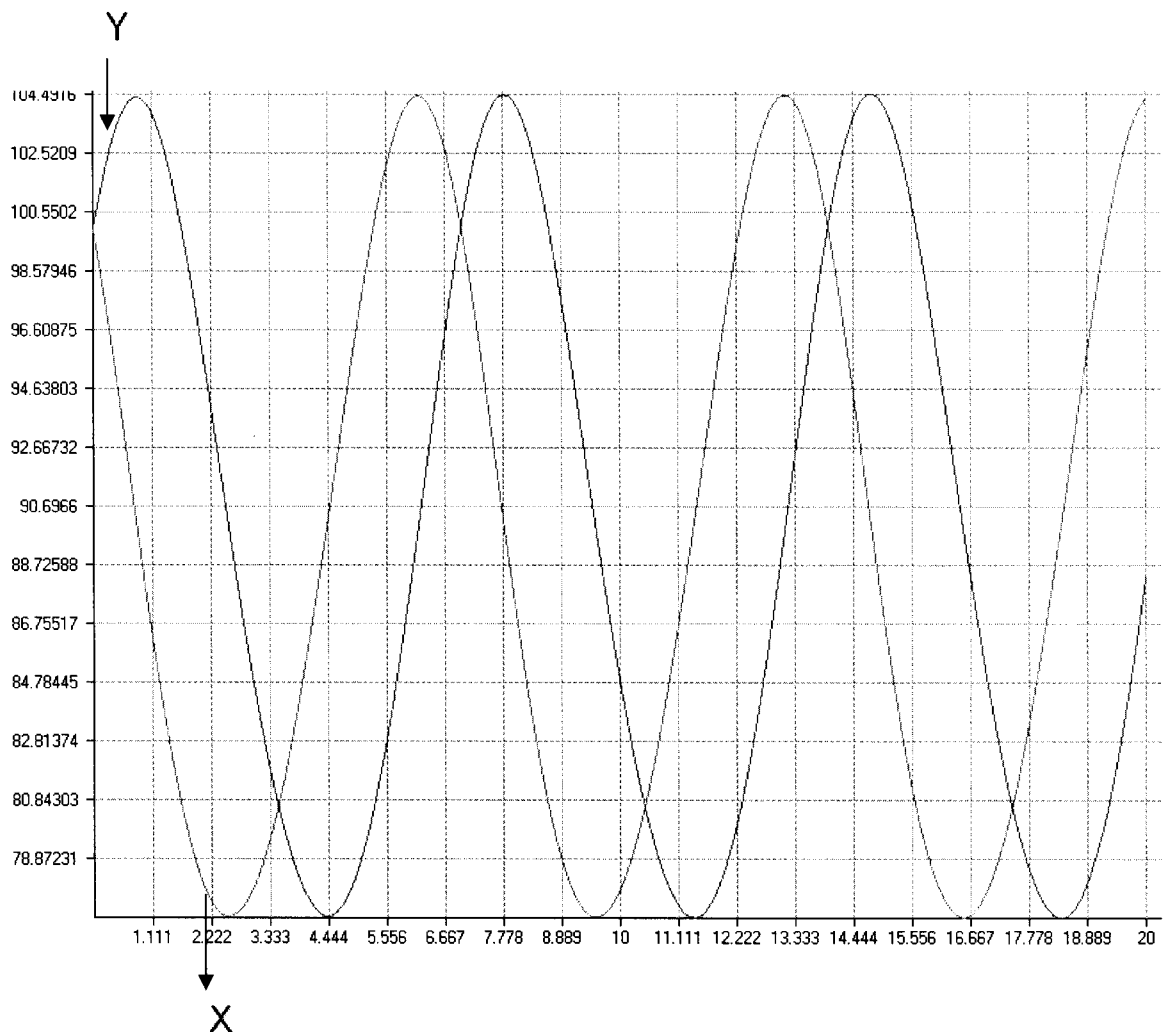


Fig. 3.5

3.1 Modelo matemático de Lotka-Volterra.

Como muestra la Fig. 3.5 los resultados obtenidos son similares a los planteados con anterioridad; en el instante inicial de este modelo que hemos creado, las poblaciones de presa y depredadores son iguales, lo que motiva un aumento en la población de depredadores con la consiguiente disminución en la población de las presas, esta situación cambia en un intervalo de tiempo pequeño, debido a que la población de presas será insuficiente para alimentar la creciente población de depredadores y por lo que los depredadores comienzan a morir hasta que llega el momento que las presas abandonan su comportamiento descendente y comienzan un incremento motivado por el gran número de muertes de depredadores; esta recuperación de la población de presas también motiva al cabo de un tiempo que la población de depredadores comience a crecer y seguirán creciendo las dos hasta que los depredadores comiencen otra vez a disminuir la población de presas. Esto explica el comportamiento periódico y oscilatorio de las poblaciones.

En la Fig. 3.6 muestra el mapa de fases conformado con los valores de las variable X y Y durante todo el tiempo de estudio, su estructura de curva simple cerrada corrobora el comportamiento periódico y oscilatorio característico de este modelo.

Como se ha podido observar los resultados obtenidos con el sistema PCDBS en el modelo presa-depredador son muy similares a los descritos en la literatura.

3.1 Modelo matemático de Lotka-Volterra.

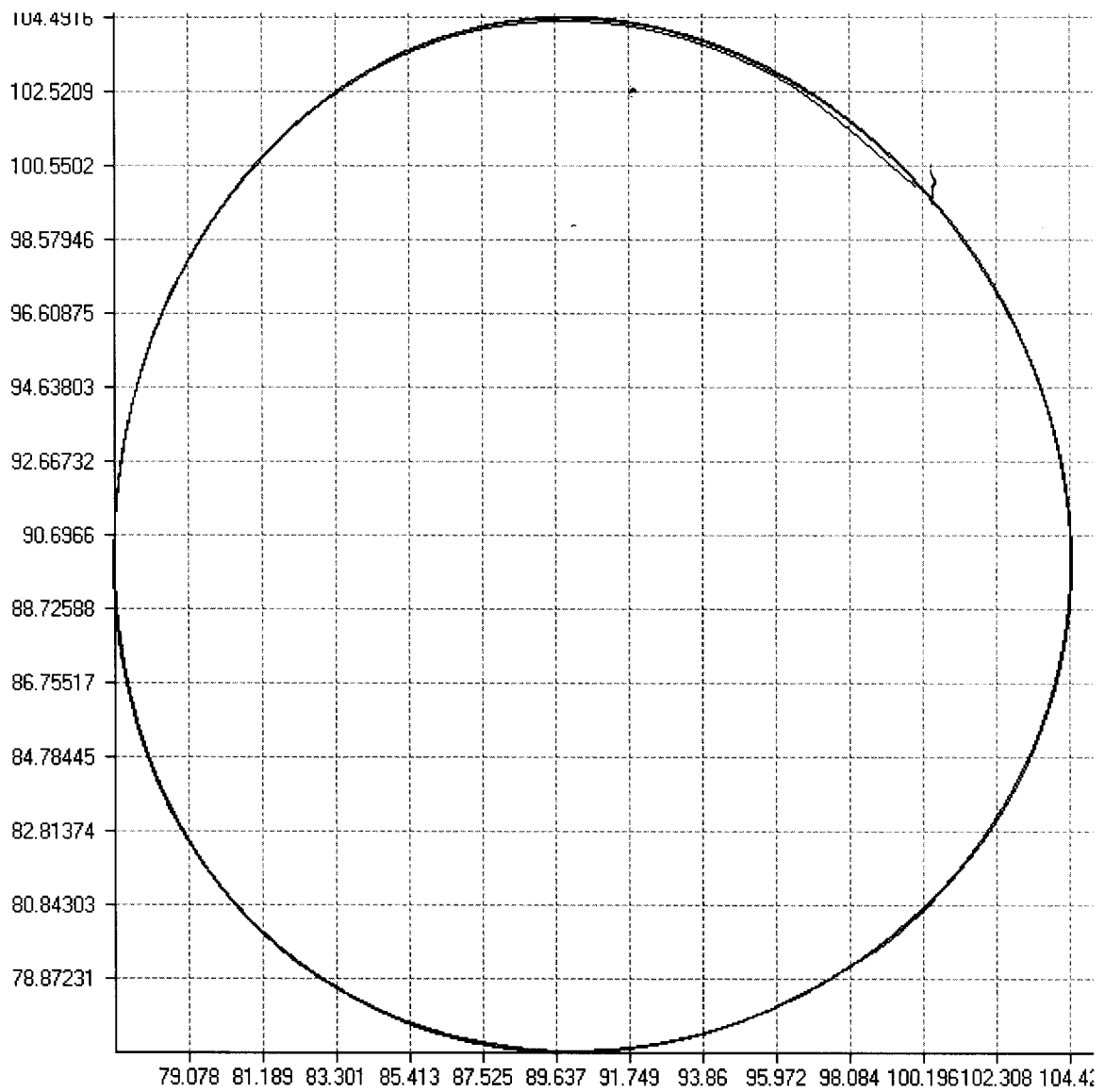


Fig. 3.6

Conclusiones

En el presente trabajo se desarrolló un prototipo de una Plataforma Computacional para el Desarrollo de la Biología de Sistemas que permite realizar análisis simples sobre el comportamiento de los sistemas biológicos.

Se realizó un estudio sobre los métodos más utilizados en la modelación de sistemas biológicos y se implementó dicha modelación mediante la utilización de sistemas de ecuaciones diferenciales. El sistema PCDBS permite la creación de modelos de sistemas biológicos de forma sencilla y fácil de comprender.

Con la utilización de PCDBS fue modelado el sistema presa-depredador descritos por la ecuaciones de Lotka-Volterra y se obtuvieron resultados muy similares a los descritos en la literatura.

Recomendaciones

Teniendo en cuenta que en este trabajo sólo se desarrolló un prototipo del sistema PCDBS que brinda las funcionalidades primarias del sistema, se realizan las siguientes recomendaciones:

- Incorporar al software análisis más complejos que se pueden obtener del sistema de ecuaciones diferenciales, como graficas de bifurcaciones y obtención de los estados de equilibrio.
- Incorporar y mejorar las opciones de diseño y edición de la plataforma con el objetivo de lograr una mayor eficiencia en su utilización.
- Incorporar a la plataforma un módulo que permita realizar modelaciones de sistemas biológicos mediante Autómatas Celulares.
- Validar los resultados de la plataforma mediante la modelación de sistemas biológicos estudiados en las instituciones científicas de nuestro país.
- Dotar el sistema de un conjunto de modelos validados que puedan ser utilizados como base para la modelación de sistemas biológicos más complejos.

Bibliografía

1. Van den Berg, H. *Lecture notes for an Introduction to Mathematical Biology*. Mathematics Institute: University of Warwick.: January, 2000.
2. Head, M., Meryhew, N., Runquist, O. *Mechanism and Computer Simulation of immune complex formation, opsonization, and clearance*. Mosby-Year Book, Inc. : Hamline University, 1996.
3. Stillianakis, N., Boucher, C., De Jong, M., Van Leeuwen, R., Schuurman, R., De Boer, R. *Clinical Data Sets of Human Immunodeficiency Virus Type 1. Reverse Transcriptase-Resistant Mutants Expained by a Mathematical Model*. Journal of Virology, Jan. 1997, p. 161-168.
4. <http://www.biobolsa.net/modules.php?name=Sections&sop=viewarticle&artid=2>. Noviembre, 2003.
5. Aragón Dávalos, R., Weniger, B. *Unidos en la Lucha Contra la Leishmaniosis. Proyecto de Investigación Colombia-Francia*. Departamento de química, Universidad del Valle: Agencia AUPEC. 1998.
6. www.entelos.com. Noviembre, 2003.
7. www.genomatica.com. Noviembre, 2003.
8. www.systemsbiology.org Noviembre, 2003.
9. Korsan, B., Kingsbury, L. *The Collaborative and Strategic Benefits of Exploring Drug Models in Clinical Development*. DIA 39th Annual Meeting June 19, 2003
(http://www.pharsight.com/news/sci_presentations.php)
10. www.physiome.com Noviembre, 2003.

11. Proyecto de Educación de Dinámica de Sistemas del MIT. Supervisado por Dr. Jay W. Forrester. Massachusetts Institute of Technology. Marzo, 1996.
12. Perelson, A., Weisbuch, G. *Immunology for physicists*. *Reviews of Modern Physics*, Vol. 69, No. 4, October 1997.
13. Hirayama, H., Yoshii, K., Ojima, H., Hawaii, N., Gotoh, S., Fukuyama, Y. *Linear Systems Analysis of Activating Processes of Complement System as a Defense Mechanism*. *BioSystems* 39, 1996, p. 173-185.
14. Muller, J. *Optimal Vaccination Strategies –for Whom?* *Mathematical Biosciences* 139, 1997, p. 133-154.
15. Glasstone, S., Aguilar. *Tratado de Química-Física*. Madrid, 1979, 7ª edición.
16. Atkins, P. *Química-Física*. Omega, Barcelona, 1999, 6ª edición
17. Jañez Pareja, E. *Curso de Inmunología General*. Departamento de Inmunología, Universidad de Granada, España, 2000.
18. De Camino Beck, Tomás. Un lenguaje para la especificación de autómatas celulares con aplicaciones en biología. Instituto tecnológico de Costa Rica, 2000.
19. Wolfram, S; Packard, N. Two-dimensional cellular automata. *Journal of Statistical Physics*, 38, 901-946, March 1985.
20. Wolfram, S. Cellular Automata as Mode of Complexity. *Nature*, 311, 419-424, October 1984.
21. Wolfram, S. The Cellular Automata. *Los Alamos Science*, 9, 2-21, 1983.
22. Wolfram, S. Twenty problems in the theory of cellular automata. *Physica Scripta*, T9, 170-183, 1985.

23. Álvarez Blanco, Manuel; Gómez Montenegro, Arnaldo; Guerra Hernández, Alfredo; Lau Fernández, Rogelio. Matemática numérica, 368 – 426, 2003.