

# Problemas Inversos, Tecnicas Evolutivas y Agentes inteligentes: Estrechando las Fronteras

DAGMAR MONETT y HANS-DIETER BURKHARD

HUMBOLDT UNIVERSITY OF BERLIN. *Artificial Intelligence, Department of Computer Science.*

diaz@informatik.hu-berlin.de ; ladb@informatik.hu-berlin.de

Recibido para revisión Feb de 2004, aceptado Mar de 2004, versión final recibida May de 2004

---

**Resumen:** En este artículo se exponen las características esenciales de la modelación matemática y la simulación de procesos reales. Con la intención de delegar capacidades humanas en las computadoras al aplicar técnicas evolutivas a problemas inversos, se extienden los principios de la Inteligencia Artificial en la automatización de dichas técnicas. También se describen la interacción entre categorías de agentes inteligentes para tales propósitos, así como el procesamiento del conocimiento de los mismos a un meta-nivel.

**Palabras Clave:** Problemas Inversos, Algoritmos Evolutivos, Agentes inteligentes.

**Abstract:** This paper resumes the essential characteristics of the mathematical modeling and simulation of real processes. With the intention of delegating most human capabilities to computers when applying evolutionary techniques to inverse problems, the principles of Artificial Intelligence regarding the automation of those techniques, are here extended. The interaction among intelligent agents categories for those purposes, as well as its knowledge processing on a meta-level are also described.

**Keywords:** Inverse Problems, Evolutive Algorithms y Intelligents Agents.

---

## 1 INTRODUCCIÓN

Los sistemas reales que se presentan en variados escenarios son conocidos a través de su actividad o manifestación externa. La necesidad de estudiar los objetos y relaciones matemáticas que consideran determinadas propiedades de los sistemas reales, entre ellas, las causas que originan su comportamiento y su desarrollo bajo ciertas condiciones iniciales, ha constituido un vasto campo de aplicación para la modelación matemática y la matemática Numérica. Numerosos sistemas reales se han representado a través de ecuaciones diferenciales ordinarias y en derivadas parciales, sistemas de ecuaciones lineales y no lineales, o combinaciones de las mismas, por sólo mencionar algunas de las clases de modelos matemáticos desarrollados por la matemática Numérica. Sin embargo, los modelos, al no poder tener en cuenta toda la riqueza del sistema real, constituyen tan solo una aproximación.

Los métodos de optimización, por su parte, como procesos de aproximación a las soluciones de numerosos modelos propuestos, han sido muy considerados en la búsqueda de la mejor correspondencia entre los mismos y los sistemas reales que representan. El desarrollo exitoso de algoritmos evolutivos ha demostrado su aplica-

bilidad a numerosos problemas de optimización [véase Goldberg (1989), Lucasius y Kateman (1989), Monett (1997), Patton, Punch y Goodman (1997)]. Sin embargo, las técnicas evolutivas en general son complejas de aplicar aún por expertos, particularmente cuando un problema real, difícil de resolver, está involucrado. Así, la idea de concebir a las computadoras como herramientas apropiadas para paliar estas problemáticas, sin la necesidad de la intervención humana para ello, nos ha servido como motivación para el desarrollo de sistemas inteligentes. Es por ello que en este artículo, como continuación de (Monett, 2001), intentamos vincular a tan disímiles ramas de la ciencia, al considerar agentes inteligentes como medios para automatizar, cual expertos humanos, diferentes estadios de la aplicación de algoritmos evolutivos a la modelación matemática de sistemas reales.

La estructura de este artículo es la siguiente: la sección 2 contiene una breve descripción de conceptos relacionados con la modelación y la simulación de sistemas reales. La sección 3 introduce las características fundamentales de los agentes inteligentes. En la sección 4 se presenta la composición básica de un sistema multi-agente para la aplicación de técnicas evolutivas. La sección 5 contiene las conclusiones del artículo.

## 2 MODELACION Y SIMULACION DE SISTEMAS REALES

En la actualidad, importantes investigaciones están dirigidas a la obtención de medicamentos atendiendo a factores complejos relacionados con el efecto que producen dichos compuestos en el hombre. Dada la complejidad de los sistemas reales, diversas reacciones químicas se estudian en la medicina contemporánea con vistas a obtener sustancias compatibles con el organismo humano. En este sentido trabaja la *modelación*, que no es más que una teoría que se desarrolla teniendo en cuenta las características del sistema real, su comportamiento, así como su relación con otros sistemas. Un modelo, como representación simplificada e intencional de un sistema real, es el camino a través del cual se representan las propiedades o componentes del sistema real. La modelación más común es la modelación matemática, donde se relacionan expresiones matemáticas entre medidas y cantidades medibles, o datos observables y variables del sistema, que definen el comportamiento del sistema real (por ejemplo: ecuaciones diferenciales que representen la variación de las concentraciones de una sustancia en el tiempo).

**Definición 1:** Una interpretación  $\mu$  de un sistema  $S$  es un modelo particular que lo representa, tal que  $\mu : E \rightarrow R$ , donde  $E$  es el conjunto de entradas o datos de entrada de  $S$ , y  $R$ , el conjunto de salidas o resultados del modelo una vez aplicada  $\mu$  a  $S$ .

Generalmente se define una métrica con el objetivo de evaluar el grado de ajuste de los valores de  $R$ , al aplicar  $\mu$ , con respecto a los valores observados experimentalmente o comportamiento del sistema real como tal. En dependencia del valor de la métrica seleccionada se determina si el modelo escogido es válido o no.

Usualmente un sistema real no tiene una única representación a través de un único modelo. La relación entre sus componentes y cómo interactúan entre sí, puede describirse utilizando diferentes expresiones. Además, todas las interpretaciones válidas para un sistema real no reproducen la conducta del mismo con igual relevancia. Cada una de ellas no es más que una aproximación teórica de lo que se supone ocurre verdaderamente.

**Definición 2:**  $M_s$  es el conjunto de interpretaciones  $\mu$  que modelan a un sistema  $S$ . Esto es:

$$M_s = \{ \mu / \mu \text{ modela a } S \}$$

La comparación que se establece entre las interpretaciones de  $M_s$  y el comportamiento del sistema real se realiza a través de la *simulación*. El objetivo de la

misma es establecer relaciones entre los modelos y las herramientas que se seleccionen para corroborar su representación, exitosa o no, del sistema real. La simulación más extendida en la actualidad se basa en la utilización de modelos computacionales, los cuales, a través de representaciones simbólicas y numéricas en las computadoras, reproducen la conducta de modelos matemáticos o permiten determinar parámetros desconocidos en los mismos, por ejemplo.

En algunos sistemas reales los parámetros del sistema son conocidos *a priori*, pudiéndose predecir el conjunto  $R$  a partir de determinados valores en  $E$  (véase Definición 1). Esto significa que, para ciertos datos de entrada al sistema, se puede predecir el comportamiento del mismo durante el tiempo. Este tipo de problemas se conoce con el nombre de *problemas directos*.

En una modelación de un sistema real puede ocurrir, sin embargo, que los parámetros que caracterizan al sistema se desconozcan, no siendo posible la medición o cálculo de los valores respectivos. Si fuera importante entonces conocer el valor de dichos parámetros, pueden inferirse sus valores observando las salidas del sistema o los resultados del modelo, relacionadas con las correspondientes entradas o datos al mismo. De esta forma el sistema puede ser interpretado lo más completamente posible para luego predecir su posterior comportamiento.

Este tipo de aplicaciones se enmarca dentro del área de los *problemas inversos*. La naturaleza general de un problema inverso consiste en determinar las causas basadas en la observación de los efectos. Por el contrario, la solución del correspondiente problema directo determina los efectos del modelo basados en una completa descripción de sus causas.

El objetivo de nuestra investigación se basa en la determinación de parámetros en sistemas reales cuyos valores son desconocidos total o parcialmente. Por lo tanto, nuestro trabajo está relacionado con problemas inversos y no con su representación directa. Es por ello que el tipo de problema inverso que nos interesa tratar es el *problema inverso con coeficientes* o *problema clásico de estimación de parámetros*, donde deben ser determinadas las constantes o parámetros desconocidos del modelo propuesto.

Por todos debe estar claro que, si el objetivo es ahorrar tiempo, recursos humanos y de laboratorio, la determinación de los coeficientes de un problema inverso no debe considerar inferir los mismos experimentalmente, sino a través de la simulación computacional. Además, el conjunto  $E$  de datos de entrada puede contener valores imprecisos como consecuencias lógicas de mediciones experimentales realizadas por el hombre, como también el conocimiento acerca del sistema real puede ser vago, o su modelación incompleta. Por lo tanto, la decisión primero de que técnicas emplear para dar soluciones lo más aproximadas posibles al comportamiento

real del sistema, así como la estimación apropiada de los parámetros desconocidos, son aspectos importantes a tener en cuenta durante la modelación y simulación de los sistemas reales que queremos considerar.

Valga en este momento citar como ejemplo práctico, la determinación de propiedades químicas de nuevos materiales para la liberación controlada de medicamentos en el organismo humano. Es por esto que las propiedades del sistema que se desconozcan deben ser estudiadas con la mayor precisión posible.

En la edición especial de 1995 por los veinte años de la revista *Byte*, se destacan los agentes inteligentes como parte de las veinte tecnologías de mayor impacto, y entre las veinte contribuciones más importantes de las ciencias de la computación a la sociedad, se relacionan las aplicaciones en la medicina, la biología y la salud humana.

En nuestra investigación pretendemos apoyar en agentes inteligentes tareas propias de la modelación y la simulación expuestas con anterioridad. En la sección siguiente se profundiza en algunas de las características esenciales de dichos agentes.

### 3 AGENTES INTELIGENTES

La Inteligencia Artificial (IA) constituye, sin dudas, una herramienta novedosa para facilitar al hombre el uso de las computadoras. Las interfaces de usuarios se han convertido en un terreno fértil para las aplicaciones basadas en agentes inteligentes. Como parte integrante de dichas interfaces, los agentes hacen más fácil al hombre el operar las computadoras.

Es por esto que las aplicaciones con agentes inteligentes constituyen, hoy en día, el área de aplicación más amplia en Internet: agentes para filtrar el correo electrónico, agentes para la navegación en la red, agentes asistentes para manipular información, por sólo citar algunas.

En este trabajo proponemos agentes inteligentes que asistan al usuario y actúen en su beneficio, evitando que el hombre diseñe nuevas sustancias a partir de experimentos costosos en recursos humanos y de laboratorio. Los agentes en este contexto pueden definirse como *agentes asistentes de software*: ellos guían al usuario al manipular o ejecutar determinado software.

Por ejemplo, si consideramos técnicas evolutivas para su aplicación a problemas inversos con coeficientes, los agentes pueden asistir al usuario para introducir datos experimentales o propios del sistema, ejecutar las técnicas evolutivas seleccionadas, manipular los resultados y procesar la información obtenida, escoger los parámetros adecuados para cálculos futuros, entre otras tareas. De hecho, los agentes asistentes de software le permiten al usuario delegar funciones que de otra forma

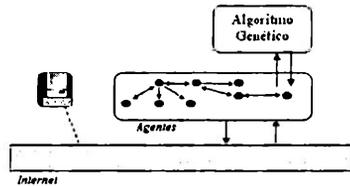


Figura 1: Agentes inteligentes en Internet.

tuvieran que realizar por sí mismos, así como automatizar, simplificar, aprender, y recomendar vías para encontrar respuestas adecuadas, sin introducir complejidad adicional cuando se trata de aplicar técnicas evolutivas particularmente.

La aplicación de métodos de IA emerge como una valiosa alternativa al permitir a los usuarios delegar en las computadoras determinadas funciones.

No todos los procesos pueden ser automatizados debido a la precisión del análisis y la especificación de los procedimientos que las computadoras necesitan. Aún así, una cantidad considerable de trabajo intelectual puede ser simulada o llevada a cabo autónomamente por *agentes inteligentes*. Esta autonomía puede entenderse como la determinación del comportamiento del agente basado en su propia experiencia.

Para ser considerados con capacidad inteligible, los agentes inteligentes asistentes de software deben ser entidades autónomas que colaboren con el usuario en la obtención de información para ejecutar las técnicas evolutivas, realizar las acciones sin supervisión por parte del usuario, aprender de experiencias pasadas para mejorar sus acciones en el futuro, así como extraer por sí mismos las conclusiones adecuadas de la información procesada, entre otras acciones.

En la Figura 1 se muestra el uso, por parte de agentes inteligentes asistentes de software, de una técnica evolutiva en particular (un algoritmo genético) con independencia de la acción del usuario. Los agentes, en este caso, actúan en el mundo virtual del Internet.

Nuestra investigación se basa en el desarrollo de agentes de este tipo, flexibles y fáciles de usar, como parte de un sistema multi-agente para la ejecución automática de técnicas evolutivas. La idea principal radica en proveer a los agentes inteligentes de la habilidad necesaria para manipular dichas técnicas, tal y como un experto en las mismas puede hacerlo. Esto significa que el usuario no tiene por que conocer a fondo las mismas ni las particularidades de dichos métodos para usarlos.

La generación de estrategias efectivas en la forma en que un experto concibe, implementa, prueba y desarrolla algoritmos evolutivos, pueden ser incluidas en los agentes como parte de su comportamiento. En la

sección que a continuación presentamos profundizaremos en algunos de estos aspectos.

#### 4 SISTEMA MULTI-AGENTE Y PROCESAMIENTO DEL CONOCIMIENTO

Una de las divisiones de la IA Distribuida la constituyen los Sistemas Multi-Agentes. En la IA clásica surgió la idea de subdividir el problema original en subproblemas. Si cada uno de ellos podía encargarse de una solución parcial, entonces de alguna manera se continuaba después el proceso una vez aportada la solución del subproblema en cuestión.

Es por ello que se desarrollaron los modelos basados en el razonamiento concurrente, permitiendo la ejecución simultánea de múltiples agentes bajo requerimientos esenciales de comunicación y sincronización entre los mismos.

Por ejemplo, en Burkhard (1995), García, González y Morales (1997) y Lindemann, Münch, Dittmann, Gnoth, Torres y Hannebauer (1999) se definen e implementan sociedades multi-agentes para aplicaciones diversas.

La necesidad de completar tareas en un tiempo requerido en el que los resultados tengan validez, necesita de un balance entre el gasto del tiempo de los procesos de deliberación y de acción de los agentes. Las arquitecturas de agentes han diferenciado bien ambos procesos.

Las conductas *deliberativas*, por lo general, descansan en algoritmos lentos y no tienen en cuenta eventos imprevistos. Sin embargo, dichas conductas están motivadas por una representación simbólica del mundo, explícita y amplia, como conocimiento de los agentes. Por su parte, las conductas *reactivas* solamente reaccionan ante estímulos del entorno resultando difíciles de aplicar a problemas complejos. Es por ello que surgen las arquitecturas *híbridas*, donde los agentes reaccionan a eventos imprevistos pero a la vez manipulan conocimiento explícito acerca del mundo en que actúan (Bussmann y Demazeau, 1994).

Basándonos en las ventajas de la combinación de los procesos de deliberación y de acción, a continuación profundizaremos en la representación del conocimiento de los agentes inteligentes asistentes de software en la manipulación de un algoritmo genético como caso particular de técnica evolutiva.

##### 4.1 Interacción entre los Agentes

Los agentes, en la arquitectura híbrida que proponemos, son modelos simplificados del pensamiento y comportamiento humanos. Son considerados autónomos e "inteligentes" para ejecutar software matemáticos sin intervención directa del hombre: son procesos de com-

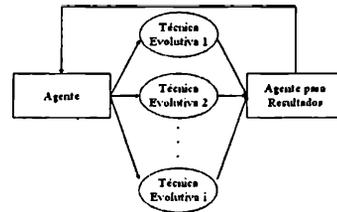


Figura 2: Organización básica de los agentes en el sistema.

putación independientes que deciden por sí mismos cómo desarrollar ciertas tareas. Sus propiedades esenciales son las siguientes:

- **Autonomía:** operan sin intervención directa del hombre y llevan un control de sus acciones y estados internos,
- **Habilidad social:** interactúan con otros agentes y humanos,
- **Reactividad:** perciben el medio y responden a los cambios que ocurren en él (carácter reactivo),
- **Pro-actividad:** capaces de exhibir una conducta dirigida hacia objetivos al tomar la iniciativa (carácter deliberativo)

En este artículo no haremos referencias a la comunicación entre los agentes, el traspaso de mensajes, los procesos específicos de deliberación y acción, la representación y control de los estados internos de los agentes, entre otras características imprescindibles en todo sistema de este tipo. Solamente se tiene en cuenta, la estructura general del sistema multi-agente, así como el procesamiento del conocimiento de los mismos en un meta-nivel.

En la Figura 2 se describe la organización básica de agentes para manipular técnicas evolutivas. El primer agente a la izquierda determina de alguna forma la técnica evolutiva a aplicar en dependencia del problema a resolver. La técnica evolutiva se ejecuta y se devuelven ciertos resultados que otro agente procesa. Finalmente dichos resultados fluyen al agente inicial, quien tomará decisiones y perfeccionará su conducta.

Profundizando un poco en las características de los agentes, a continuación proponemos tres tipos de ellos de acuerdo con los objetivos anteriormente descritos:

- **Agente Especialista:** Como su nombre lo indica, simboliza el razonamiento de los expertos humanos. Este agente solicita al usuario los datos relativos al modelo matemático del sistema real en estudio. Dada su experiencia en técnicas evolutivas, determina cuál estrategia considerar en dependencia de

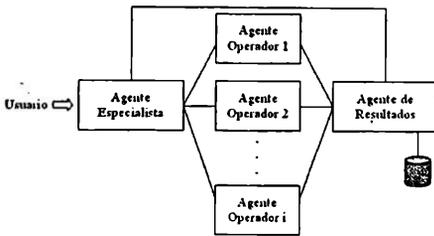


Figura 3: Agentes especialista, operadores y de resultados.

las características del modelo matemático y del sistema real en cuestión. Es, por excelencia, el agente que asiste a los usuarios en la manipulación del software como tal.

- **Agentes Operadores:** Ejecutan, independientemente, las técnicas evolutivas propuestas. Conocen los pormenores necesarios para la ejecución de dichos algoritmos y manipulan los datos autónomamente de acuerdo con las especificaciones del software que operan. Deducen explícitamente la información que no es suministrada por los usuarios pero que es considerada esencial para su correcto funcionamiento.
- **Agente de Resultados:** Recibe los resultados parciales y globales de todas las ejecuciones que realizaron los agentes operadores. Organiza dichos resultados y los almacena en dependencia de la relevancia de los mismos, pues en futuras ejecuciones pudieran servir como solución a problemas similares.

En la Figura 3 se asocia un agente operador por cada técnica evolutiva, responsabilizándose cada agente con los detalles del funcionamiento de una técnica específica. Analicemos un ejemplo de aplicación para profundizar en el procesamiento del conocimiento de los agentes y las interacciones entre los mismos. Consideremos para ello un algoritmo genético como software aplicado a problemas inversos con coeficientes [véase Monett (1997) y Monett (2000) para profundizar en los aspectos relativos al algoritmo genético].

La variación de un parámetro de control del algoritmo genético provoca, en cada ejecución, la obtención de resultados diferentes.

Concluir que una interpretación  $\mu$  es la que ajusta el sistema real con mayor precisión, determinando si el parámetro seleccionado tiene el valor óptimo, no es una tarea expedita y sencilla. Es por ello que, usualmente, se ejecutan varias veces los algoritmos genéticos hasta ajustar los valores de sus parámetros de control en dependencia del problema real de que se trate. Entonces, el agente

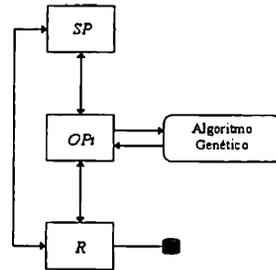


Figura 4: Interacción entre los agentes.

especialista  $SP$ , tal y como se representa en la Figura 4, puede ser quien evalúe las diferentes posibilidades de aplicación del algoritmo genético en dependencia de los valores de sus parámetros de control.

Recordemos que el agente especialista simula el comportamiento del experto que decide cuál método seleccionar y cómo, para resolver el problema real inicial. De esta forma, está también capacitado para definir criterios de parada para el algoritmo genético, por ejemplo, con relación a su ejecución, así como obtener del usuario toda información que necesite.

El agente especialista  $SP$ , además, debe interactuar con el agente de resultados  $R$  (véase Figura 4) con el objetivo de conocer si se tienen resultados previos, calculados con anterioridad, para iguales parámetros de control del algoritmo genético. En estos casos, no solamente los parámetros de control del algoritmo genético deben coincidir con los calculados en el pasado, sino también el modelo  $\mu$  a simular en ambos problemas.

El agente especialista  $SP$  en este momento está en condiciones de seleccionar al agente operador adecuado para la simulación.

Supongamos, sin perder generalidad, que el agente operador escogido fue el  $OP_i$  (véase Figura 4), especializado en la ejecución del algoritmo genético. Una vez que el agente operador  $OP_i$ , esté disponible, el agente especialista  $SP$  suministra al mismo los datos relativos a la técnica evolutiva particular (el algoritmo genético en este caso). El agente operador  $OP_i$  verifica los datos para la ejecución del algoritmo genético, selecciona aquellos que no han sido suministrados pero que son importantes para la ejecución (pues conoce las particularidades del algoritmo y actúa como un experto en el mismo), ejecuta el algoritmo, acumula los resultados una vez concluida la ejecución, y envía los mismos al agente de resultados  $R$  para que los procese.

El agente de resultados  $R$ , por su parte, recibe los resultados de todas las ejecuciones y almacena los que considera necesarios, positivos o negativos, con el objetivo de evitar similares ejecuciones en el futuro.

Extiéndase por “positivos” y “negativos” los resultados con los cuales se concluye que un modelo en  $M_s$  es adecuado o no para un sistema real determinado (véase Definición 2).

Además, éste agente debe interactuar nuevamente con el agente especialista  $SP$ , pues éste último es quien expone al usuario del sistema los resultados de todo el proceso, y brinda una valoración -como experto al fincorde con el tipo de problema y la técnica que se aplicó.

Hasta aquí hemos hecho hincapié en las particularidades relativas a los agentes en el sistema que proponemos. La investigación actual se basa en los procesos de comunicación entre los agentes utilizando como soporte una red local de computadoras. Al suponer que los agentes pueden estar distribuidos geográficamente, en específico los relacionados con los software matemáticos, nuestra intención es extender su uso en Internet para poder aplicarlos desde lugares distantes entre sí.

## 5 CONCLUSIONES

Durante el desarrollo de lo anteriormente expuesto se presentaron las características más generales de la modelación matemática y la simulación de sistemas reales. Como fue explicado en las secciones 3 y 4, con esta investigación se pretende delegar la mayor cantidad de actividades en agentes inteligentes asistentes de software, con el objetivo de automatizar el uso de técnicas evolutivas. Estos objetivos facilitan sobremanera el manejo de las mismas, así como estrechan, una vez más, disciplinas científicas aparentemente independientes entre sí.

## AGRADECIMIENTOS

Los autores agradecen el apoyo financiero que brinda el Servicio Alemán de Intercambio Académico (DAAD) a la M.Sc. Dagmar Monett Díaz, para el disfrute de una beca de estudios de postgrado en la Universidad de Humboldt de Berlín, bajo el código A/99/03570.

## REFERENCIAS

- Burkhard, H. D. (1995), Agent oriented programming for open systems, in 'Proceedings of the ECAI-94 Workshop on Agent Theories, Architectures and Languages, The Netherland'. Lecture Notes in AI.
- Bussmann, S. y Demazeau, Y. (1994), An agent model combining reactive and cognitive capabilities, in 'Proceeding of the IEEE International Conference on Intelligent Robots and Systems', Munich, Alemania.
- García, A., González, N. y Morales, F. (1997), Diseño e implementación de sistemas multiagentes, in 'Proceedings del Primer Simposio de Inteligencia Artificial, Conferencia Internacional de Ciencia y Tecnología para el Desarrollo CIMAF-97', La Habana, Cuba.
- Goldberg, D. E. (1989), *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Reading, Mass.
- Lindemann, G., Münch, I., Dittmann, I., Gnoth, M., Torres, E. y Hannebauer, M. (1999), ChariTime: An agent-oriented approach for appointment management in distributed working communities, in 'Proceeding of the KI-99 Workshop Agententechnologie- Multiagentensysteme in der Informationslogistik und Wirtschaftswissenschaftliche Perspektiven der Agenten-Konzeptionalisierung', number 16, TZI-Bericht, Bonn, Alemania.
- Lucasius, C. y Kateman, G. (1989), Application of genetic algorithms in chemometrics, in J.D.Shaffer, ed., 'Proceeding of the Third International Conference on Genetic Algorithms', Morgan Kaufmann.
- Monett, D. (1997), Genetic algorithms for the reactivity ratios determination in copolymerization, in 'Proceedings del Primer Simposio de Inteligencia Artificial, Conferencia Internacional de Ciencia y Tecnología para el Desarrollo CIMAF-97', La Habana, Cuba.
- Monett, D. (2000), Towards and intelligent agent desing for handling genetic algorithms., in B. H. D., L. Czaja, A. Skowron y P. Starke, eds, 'Proceedings of the Workshop on Concurrency, Specification and Programming CS&P-2000', Vol. 1, Berlin, Alemania, pp. 149-155.
- Monett, D. (2001), On the automation of evolutionary techniques and their application to inverse problems form chemical kinetics, in 'Porceedings of the GECCO 2001, Genetic and Evolutionary Computation Conference', San Francisco, California.
- Patton, A., Punch, W. y Goodman, E. (1997), 'A standard GA approach to native protein conformation prediction'. Genetic Algorithm Research and Application Group, Computer Science Department, Michigan State University.