

Convocatoria de ayudas de Proyectos de Investigación Científica y Desarrollo Tecnológico (2002)

MEMORIA CIENTÍFICO-TÉCNICA DEL PROYECTO

1 RESUMEN DE LA PROPUESTA (Debe rellenarse también en inglés)

INVESTIGADOR PRINCIPAL: ENRIQUE ALBA TORRES (proyecto coordinado)
SUBPROYECTO 1 (TRACER::UMA) ENRIQUE ALBA TORRES
SUBPROYECTO 2 (TRACER::ULL) COROMOTO LEÓN HERNÁNDEZ
SUBPROYECTO 3 (TRACER::UC3M) PEDRO ISASI VIÑUELA
SUBPROYECTO 4 (TRACER::UPC) JOAQUÍN GABARRÓ VALLES
SUBPROYECTO 5 (TRACER::UNEX) JUAN MANUEL SÁNCHEZ PÉREZ

TÍTULO DEL PROYECTO: TRACER: Técnicas de Optimización Avanzadas para Problemas Complejos
RESUMEN (debe ser breve y preciso, exponiendo sólo los aspectos más relevantes y los objetivos propuestos):

El proyecto coordinado TRACER tiene como objetivo la promoción del conocimiento en áreas de investigación relativas a la algoritmia y técnicas de optimización avanzadas. El foco principal estriba en la resolución eficiente de problemas de complejidad controlable que puedan, en el futuro, servir de puente entre los problemas típicos de corte académico y los problemas de gran complejidad que son abordables mediante dichas técnicas de optimización.

Se propone el diseño, implementación y análisis de variantes sofisticadas (paralelas, híbridas y auto-guiadas) de algoritmos evolutivos, heurísticos en general y también exactos para resolver problemas complejos en los dominios de la optimización combinatoria tradicional, telecomunicaciones, bio-informática, y hardware reconfigurable (FPGA), entre otros.

Como valor añadido al proyecto se ofertan dos resultados concretos. El primero es un sistema cliente/servidor a través de Internet, útil para que la comunidad científica en general pueda contribuir sus propios problemas y elegir cualquiera de las técnicas de optimización desarrolladas. El segundo consiste en diseñar e implementar un repositorio sobre problemas incluyendo teoría, software, resultados prácticos y enlaces de interés.

PROJECT TITLE: TRACER: Advanced Optimisation Techniques for Complex Problems

SUMMARY:

The co-ordinated project TRACER aims at developing the baseline research foundations of algorithms and advanced optimisation techniques. The main goal of this project is to achieve an efficient resolution of complex problems by developing new optimisation procedures that can help in present and future applications, both in academia and the real world.

We propose the design, implementation and evaluation of sophisticated (parallel, hybrid, self-guided) variations of evolutionary algorithms, and, in general, heuristic and exact solving tools to solve complex problems arising in the domains of traditional combinatorial optimisation, telecommunications, bio-informatics, and FPGA's, among others.

As an added-value result, we offer two software systems. The first one is a client/server Internet system endowing researchers with the ability to submit their problems to be solved by our (newly developed) optimisation techniques. The second one is a problem repository including theoretical results, software, numerical results and related links.

ÍNDICE

- 2. INTRODUCCIÓN 3**
 - 2.1 ¿Qué Entendemos por “Técnicas Avanzadas”? 5
 - 2.2 ¿Qué Entendemos por “Problemas Complejos”? 5
 - 2.3 Sistemas de Valor Añadido: Cliente/Servidor para Optimización en Internet y Repositorio 6
 - 2.3.1 Sistema Cliente/Servidor para Optimización en Internet 6
 - 2.3.2 Sistemas Similares Existentes en Internet 8
 - 2.4 Problemas y Algoritmos 9
 - 2.4.1 Problemas 10
 - 2.4.2 Algoritmos 12

- 3. OBJETIVOS DEL PROYECTO 16**

2. INTRODUCCIÓN

Deben tratarse aquí: la finalidad del proyecto; los antecedentes y estado actual de los conocimientos científico-técnicos, incluyendo la bibliografía más relevante; los grupos nacionales o internacionales que trabajan en la misma materia específica del proyecto, o en materias afines; los logros anteriores del grupo en esta materia y en otras afines; etc.

- Si el proyecto es continuación de otro previamente financiado, deben indicarse con claridad los objetivos ya logrados y los resultados alcanzados.
- Si el proyecto aborda una nueva temática, deben indicarse los antecedentes y contribuciones previas del grupo, con el fin de justificar su capacidad para llevar a cabo el nuevo proyecto.

Se propone la realización de un proyecto coordinado realizado por grupos de investigación de cinco universidades españolas. El proyecto pretende cubrir una necesidad existente, tanto en el ámbito de la investigación como en el de la Sociedad, relativa a la resolución de problemas complejos. Además de usar tecnologías actuales para este fin, incorporamos a nuestra propuesta importantes valores añadidos en el terreno del uso de Internet y del avance en técnicas modernas de optimización.

Se presenta una propuesta tri-anual, donde el enfoque estará en la solución de problemas de elevada dimensión/complejidad, y en el avance teórico en cuanto a diseño de nuevas técnicas algorítmicas eficientes para este menester. De esta manera, se pretenden conseguir avances en el funcionamiento de las técnicas, así como mejorar su eficiencia. Los problemas se escogerán de acuerdo a su relación con los grupos de investigación de la propuesta y siguiendo las líneas prioritarias del Plan Nacional TIC. Esto contribuirá al interés de los resultados, ya que al mismo tiempo abordaremos las dificultades subyacentes de cada problema persiguiendo el mayor trasvase de conocimientos posible.

Consideraremos no únicamente problemas tradicionales de optimización combinatoria, sino que se apostará de manera decidida por aplicaciones de corte real que generen un interés en múltiples dominios de aplicación por sí mismas, permitiendo desafiar la capacidad de los distintos prototipos que se desarrollen para el sistema. Esto provocará necesariamente avances tanto en el conocimiento teórico sobre técnicas de resolución como en su calidad tecnológica desde el punto de vista de la aplicación final real.

Esta propuesta de proyecto coordinado de investigación tiene el valor añadido de ser ofrecida como un servicio a través de Internet a la terminación de sus tres años de vida. Este servicio permitirá a sus usuarios utilizar un programa cliente desde cualquier plataforma de trabajo para proponer a un servidor remoto un problema que éste debe resolver. Un segundo valor añadido lo constituye la creación de un repositorio web conteniendo instancias de alta dimensión de problemas complejos, así como identificando simplificaciones y extensiones de impacto social de dichos problemas, con miras a una repercusión y reconocimiento internacional. En este sentido, los componentes de los grupos crearán los servicios de valor añadido y, al mismo tiempo, serán también sus usuarios, lo que permitirá fomentar la colaboración de calidad.

Un sistema en Internet abierto y especializado como éste sería novedoso en investigación tanto como en el mundo empresarial, creando nuevas maneras de resolver problemas y generando expectativas comerciales que interesen a la Sociedad en su conjunto y fomenten la transferencia tecnológica con la Universidad. De hecho, se prevé también la creación de varias herramientas web en cada uno de los sitios participantes en los dominios de la optimización, las comunicaciones e Internet.

Debemos además mencionar que en este trabajo se contemplarán algunos de los avances más importantes actualmente en Internet: la reciente iniciativa .NET de Microsoft (ver [Platt01]) y el estándar de facto J2EE (*Java 2 Enterprise Edition*) [ZU01], ambas con notables implicaciones en la forma de diseñar y utilizar servicios. Adicionalmente, estos estándares serán evaluados en el campo del diseño de algoritmos y la optimización, lo cual no se tiene en cuenta normalmente en investigación debido al elevado hincapié que se realiza o bien en las técnicas en sí, o bien en las aplicaciones.

Es necesario también especificar que muchos de los resultados que se obtengan supondrán la extensión y continuidad del proyecto MALLBA (TIC1999-0754-C03) centrado en paralelización LAN/WAN de algoritmos exactos, heurísticos e híbridos. Asimismo, se avanzará en los resultados que se están obteniendo en otros proyectos nacionales como APRIS (TIC2000-0475) en relación también a la ejecución paralela, principalmente sobre hardware reconfigurable. Nuestros objetivos básicos consisten en extender las técnicas desarrolladas para que resulten eficientes en problemas complejos y reales, en ofertar dichas técnicas y sus resultados en Internet para que puedan ser utilizados por la comunidad científica en general, y en incorporar a este campo las nuevas tecnologías surgidas en los últimos años.

Para alcanzar los objetivos de este proyecto coordinado participan cinco grupos de investigación de distintas universidades españolas, con un total de 33 investigadores implicados:

- **UMA**: grupo de investigadores de la Universidad de Málaga (el investigador principal del sub-grupo para esta propuesta, y del proyecto coordinado es Enrique Alba). < 6 investigadores>
- **ULL**: grupo de investigadores de la Universidad de La Laguna (investigador principal: Coromoto León). < 7 investigadores>
- **UC3M**: grupo de investigadores de la Universidad Carlos III de Madrid (investigador principal: Pedro Isasi). < 7 investigadores>
- **UPC**: grupo de investigadores de la Universidad Politécnica de Cataluña (investigador principal: Joaquín Gabarró). < 8 investigadores>
- **UNEX**: grupo de investigadores de la Universidad de Extremadura (investigador principal: Juan Manuel Sánchez). < 5 investigadores>

El acrónimo del proyecto es **TRACER** (*Techniques Required for Algorithms in Complex Environments of Research*). Cada sub-proyecto se identificará con una notación de ámbito dentro del proyecto coordinado. Así, los nombres oficiales para los cinco sub-proyectos son:

Su-proyecto 1: TRACER::UMA.

Su-proyecto 2: TRACER::ULL.

Su-proyecto 3: TRACER::UC3M.

Su-proyecto 4: TRACER::UPC.

Su-proyecto 5: TRACER::UNEX.

La justificación de esta composición es lógica atendiendo a la experiencia pasada y a la complementariedad de los trabajos y especialización de dichos sub-grupos. En primer lugar, los grupos UMA, ULL y UPC han venido colaborando durante los últimos tres años con especial dedicación en el proyecto MALLBA antes mencionado. Los intereses comunes en relación al paralelismo LAN y WAN han servido de base para realizar los trabajos que se aportan en los perfiles de cada grupo al final de la memoria.

Puesto que los principales objetivos de esta propuesta radican en el desarrollo de técnicas eficientes, los cinco grupos representan especializaciones con muchos puntos en común de este campo. Todos los grupos tienen especial interés en heurísticos y sus mezclas tanto con otros heurísticos como con técnicas exactas. En particular, existe una considerable bibliografía común en relación a los algoritmos evolutivos (algoritmos genéticos y programación genética en especial), campo de base para el trabajo desarrollado en el grupo de UC3M y también de UNEX, grupos que no participan en MALLBA.

Por otro lado, todos los grupos pretenden abordar problemas complejos e incluso reales que requieran mejorar las técnicas existentes, especialmente en eficiencia. De esta manera, la experiencia en el uso de hardware reconfigurable del grupo de la UNEX y los problemas de diseño y de series temporales aportados al conjunto son de mucho interés actual en investigación. Asimismo, el grupo de UC3M tiene líneas abiertas de investigación en predicción de series temporales y optimización en general.

Este último punto (la optimización en sus diferentes facetas) es el que aglutina a los cinco sub-grupos en un intento de avanzar en varios frentes a la vez. Todos los grupos, además, pretenden abordar los problemas del paralelismo LAN/WAN y su relación con Internet. De esta forma, los objetivos comunes respecto a optimización en general, Internet y problemas complejos confieren una uniformidad especialmente relevante tanto a la propuesta como a la selección de grupos para realizarla.

El ánimo de la propuesta es la interacción en esquemas de optimización modernos desarrollados por colaboración entre los distintos grupos, así como su evaluación. Esta evaluación se realizará sobre un conjunto de aplicaciones heterogéneas complejas que establezcan claramente la utilidad de los resultados. La novedad de las técnicas que se pretenden abordar y de las tecnologías sobre las que se establecerán son un valor añadido a los avances en eficiencia que esperamos obtener, tanto coordinadamente como en cada sub-grupo particular.

2.1 ¿Qué Entendemos por “Técnicas Avanzadas”?

Bajo la denominación de “técnicas avanzadas o modernas” de optimización se engloban varias familias de heurísticas que explotan ideas tomadas de diferentes dominios tales como la Biología, las Matemáticas o la Física entre otros. Así, pueden citarse los algoritmos evolutivos [BFM97] (basados en la metáfora de la Evolución Natural), el recocido simulado [KGV83] (basado en los procesos térmicos que dan lugar a la adopción de estados meta-estables de baja energía en sólidos sometidos a baños térmicos), los algoritmos meméticos [MC02] (basados en los procesos de evolución cultural y transmisión del conocimiento), los algoritmos de estimación de distribuciones [MP96] (basados en el aprendizaje de una distribución de probabilidad mediante muestreos guiados del espacio de búsqueda), o la búsqueda tabú [Glo89] (basada en el empleo de una memoria a corto plazo de sus acciones pasadas para la determinación de las acciones futuras).

Este tipo de técnicas se han revelado como la gran alternativa para afrontar la resolución de numerosos problemas de interés práctico, en campos que van desde la ingeniería [Yeh99] a la medicina [PS00], pasando por las telecomunicaciones [CGKK97] y la economía [Nov98]. A diferencia de enfoques de optimización más clásicos, las técnicas modernas de optimización no adolecen de problemas tan severos de escalabilidad, lo que permite su aplicación satisfactoria a problemas de tamaño real. Por ello, es también nuestro objetivo mejorar las técnicas en dominios exactos (por ejemplo programación dinámica) para llevarlas a mejores cotas de aplicación.

En cualquier caso, debe tenerse en cuenta que la incorporación de conocimiento dependiente del problema dentro de estas técnicas se ha revelado en la teoría [WM97] y en la práctica [Dav91] como un mecanismo fundamental para alcanzar soluciones de calidad a problemas complejos. En este sentido, la combinación de algoritmos de optimización clásicos y modernos resulta una opción común a tal efecto. Es por ello que la disponibilidad de ambos tipos de algoritmos de optimización, así como de herramientas para su integración es un requisito evidente para las bibliotecas algorítmicas para optimización (por ejemplo, [AAB+01]).

2.2 ¿Qué Entendemos por “Problemas Complejos”?

Existe un conjunto de características que deben cumplir los problemas abordados para resultar de interés general a la comunidad investigadora. Muchos trabajos existentes mencionan algunos de ellos pero pocos profundizan en los puntos concretos cuantitativos que sirvan de guía para explorar complejidades crecientes. Sí que parece claro que podemos enunciar un conjunto de dificultades que hacen a un problema, si las posee, merecedor de estudio. En [WRDM96] podemos encontrar un buen ejemplo de este tipo de estudios, con enfoque en los algoritmos evolutivos pero en su mayoría aplicable a los problemas en general. Entre las dificultades que debe poseer un problema para ser de interés podemos listar las siguientes:

- A. **Multimodalidad.** El espacio de búsqueda posee un elevado ratio de soluciones localmente óptimas en relación al número de óptimos globales. Esto supone que las técnicas que se utilicen para resolver el problema tendrán una marcada tendencia a estancarse en los primeros.
- B. **Epistasis.** Existe un elevado grado de interacción y dependencia entre las variables de decisión del problema. Esto supone que los problemas suelen ser de difícil solución con técnicas de gradiente y que no deben tener una definición ni simétrica, ni lineal, ni separable.
- C. **Restricciones.** Un problema restringido divide el espacio de búsqueda entre soluciones factibles y no factibles. Esto supone una dificultad añadida importante para las técnicas de resolución debido a que deben tratar con estos dos tipos de soluciones, considerar un número considerable de restricciones (tiempo) y además optimizar la función objeto de estudio.
- D. **Multi-objetivo.** Un problema multi-objetivo (o multi-criterio) tiene como objetivo optimizar múltiples funciones al mismo tiempo, cada una dependiente de un elevado número de variables de decisión. Es muy común que no exista una única solución, sino que se existan múltiples soluciones que no admiten mejoras en una de las funciones objetivo más que en base a empeoramiento de otras de dichas funciones objetivo (soluciones llamadas “no dominadas”). Los problemas de este tipo definen la solución como un conjunto de valores no dominados según un criterio de óptimo de Pareto. La complejidad de este tipo de problemas (y su cercanía a la realidad) los hace de mucho interés y plantean graves dificultades a las técnicas existentes.

- E. **Dinámicos.** Tradicionalmente la optimización se ha enfocado en la resolución de problemas cuya definición, con mayor o menor dificultad, es estática. Sin embargo, existe todo un dominio de problemas que incorporan ruido aleatorio o que sólo pueden expresarse como optimización de funciones objetivo que cambian en el tiempo. Algunos ejemplos son los sistemas de control de los ascensores múltiples o de semáforos. Las técnicas tradicionales y las exactas suelen presentar graves deficiencias cuando un problema presenta este tipo de característica dinámica, haciendo necesario el estudio en este tipo de problemas.

En general, si deseamos que un resultado en investigación sea de impacto, necesitamos incorporar todas estas características, o al menos varias de ellas, en el conjunto de problemas que estudiar. De otra manera se pueden producir resultados falsamente alentadores o incluso se puede promocionar un indeseado desconocimiento de técnicas alternativas típicas de cada dominio.

Adicionalmente, un estudio incremental de técnicas y problemas debe utilizar los llamados *generadores de problemas*, quienes permiten evaluar a la clase-problema como clase y evitar evaluar instancias concretas de dicha clase, lo que podría introducir tendencias en los resultados. Actualmente, existen numerosos tipos de generadores de problemas que incluyen una o más de las características antes mencionadas, como es el caso de los paisajes NK [Kauffmann89] y de generadores multimodales como P-PEAKS [DPS97].

Necesariamente las técnicas de solución deberán enfrentarse a estas dificultades para que los resultados del proyecto coordinado tengan interés e impacto. Existen, además, dos consideraciones típicas más informales que pueden utilizarse para caracterizar la dificultad, como son el *tiempo de CPU estimado* para encontrar una solución y el *tamaño del espacio de búsqueda*. Está claro que no se trata de medidas determinantes en relación a la complejidad, ya que podemos encontrar problemas que requieran tiempos de CPU considerablemente altos y que presenten enormes espacios de búsqueda pero que al mismo tiempo no resulten de interés. Sin embargo, ambos parámetros pueden guiar cualitativamente el tipo de problemas al que nos referimos en cada momento del proyecto.

En cualquier caso, el enfoque prioritario debe ser la **eficiencia y competitividad** de los resultados.

2.3 Sistemas de Valor Añadido: Cliente/Servidor para Optimización en Internet y Repositorio

En este proyecto, además de los objetivos primarios de comprensión de las dificultades para abordar problemas de grandes dimensiones y complejidad con técnicas modernas, hemos querido añadir un segundo conjunto de valores añadidos. Estos valores añadidos no son objetivos primarios pero permitirán mejorar la interacción entre los grupos de este proyecto y ofrecer herramientas nuevas útiles a la comunidad de científicos españoles en general y europeos en particular.

Básicamente, los servicios de valor añadido que ofertaremos son dos:

- A. un sistema cliente/servidor para acceder a cualquiera de las técnicas que desarrollemos, de manera que el usuario envíe su problema codificado en Java y pueda elegir entre un abanico de resolutores, y
- B. un repositorio de problemas complejos con dimensiones crecientes.

2.3.1 Sistema Cliente/Servidor para Optimización en Internet

En la Figura 1 puede observarse un esquema del sistema que proponemos construir. Para utilizar el resultado de este proyecto se diseñará un cliente que permita el acceso al servicio de optimización. Este cliente podrá personalizarse para varios perfiles de usuario, contemplando al menos dos tipos: (a) usuarios técnicos que desconozcan los detalles de los servicios de optimización y (b) usuarios expertos en dichos servicios. Ambos tipos de usuarios tendrán a su disposición una serie de recursos remotos para resolver sus problemas. El usuario entrega su problema al cliente y lo instruye de acuerdo a sus conocimientos para llevar a cabo el proceso de resolución.

El cliente recoge la información del usuario (incluyendo la especificación del problema más los parámetros para el motor de optimización) y genera una petición de servicio para el servidor remoto. El intercambio de información entre el cliente y servidor se realizará a través de Internet en un formato flexible, como XML [Holzner01]. El servidor, en función de la información recibida en la petición, arrancará un motor de optimización apropiado para resolver el problema planteado.

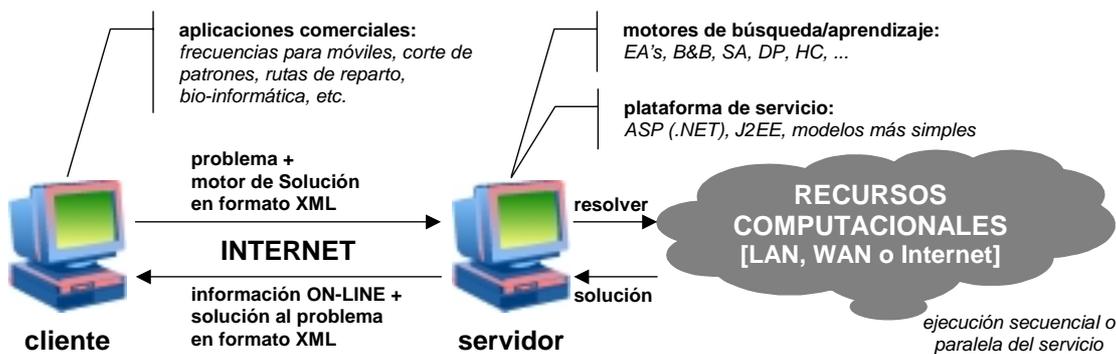


Figura 1. Arquitectura del sistema cliente/servidor a través de Internet.

El objetivo es que el usuario pueda estar informado en tiempo real sobre el progreso de su petición, así como realizar nuevas peticiones. El servidor deberá aceptar peticiones de uno o varios clientes y utilizar racionalmente los recursos computacionales a su disposición para resolver eficientemente cada petición. Naturalmente, en el camino a esta solución de valor añadido aparecerán resultados de interés en seguridad y comunicación de información en WAN e Internet en general.

El lado servidor implementará motores de búsqueda variados en Java. Entre ellos se incluirán técnicas exactas como ramificación y acotación (B&B) o programación dinámica (DP), técnicas de gradiente descendiente (HC) y procesos de búsqueda local tales como recocido simulado (SA) o búsqueda tabú (TS), así como técnicas heurísticas inspiradas en procesos naturales, principalmente variantes de algoritmos evolutivos (EA). Puesto que se generará una interfaz para poder interactuar con el servidor de manera genérica, el hecho de que los motores nuevos estén escritos en Java puede ser sólo una forma más de implementación del servicio, es decir, pueden añadirse al sistema motores ya existentes (por ejemplo escritos en C, C++ o incluso C#) simplemente haciendo que implementen la interfaz de interacción genérica con el servidor.

La utilización de estos motores de optimización supone investigar su aplicabilidad a las clases de problemas enumerados arriba, así como desarrollar resolutores modernos competitivos con técnicas existentes para dichos problemas. En muchos casos, esto supondrá generar resolutores que trabajen en paralelo y que incorporen información sobre los problemas resueltos (hibridación). En este sentido, el desarrollo en sí de técnicas resolutoras representa una **extensión y actualización del trabajo desarrollado en el proyecto financiado CICYT TIC1999-0754-C03** denominado "*Entornos Geográficamente Distribuidos: Una Librería de Optimización Combinatoria*" donde se han investigado técnicas con las características mencionadas (para más detalle consulte <http://lsi.upc.es/~mallba>).

Para proporcionar el servicio se analizarán y utilizarán técnicas para Internet. En particular, es de interés estudiar la viabilidad del proyecto utilizando *Java 2 Enterprise Edition* (J2EE) frente a la nueva plataforma de servicio .NET de Microsoft, que utiliza extensivamente *Active Server Pages* (ASP) y promueve el uso de lenguajes de programación como C# [Gunnerson 01] y protocolos de comunicación como SOAP (*Simple Object Access Protocol*) (<http://www.w3.org/TR/SOAP/>).

La arquitectura software del sistema debe cumplir varios requisitos. Además de ofrecer configuraciones por defecto y una interfaz de uso para el cliente, deben incluirse en el diseño mecanismos de extensión futura y de conexión a motores de búsqueda existentes. Esto último es necesario tras comprobar que existen numerosos sistemas de optimización en activo proporcionando buenas prestaciones en algunas de las aplicaciones que se pretenden cubrir en el proyecto.

La Figura 2 presenta un esquema de dicha arquitectura interna. Puede comprobarse que es necesario incluir un modelo de interacción abierto entre los motores de búsqueda y el lado servidor, con la intención de ofertar a los clientes la misma interfaz de trabajo independientemente de si se trata de un algoritmo desarrollado en este proyecto o existente. Naturalmente, esto no es posible a coste cero en el caso de los algoritmos existentes. En ese supuesto, debe desarrollarse un conjunto de adaptadores en Java (*wrappers*) que permitan componer los servicios requeridos usando los ya existentes.

Se estudiarán las ventajas de utilizar Java, C# y otros lenguajes para cada uno de los elementos del sistema: cliente, servidor, interfaz y algoritmos. La plataforma .NET y el nuevo *Common Language Runtime* (CLR) [Gough01] abren nuevas perspectivas para la compartición de código a través del IL (*International Language*).

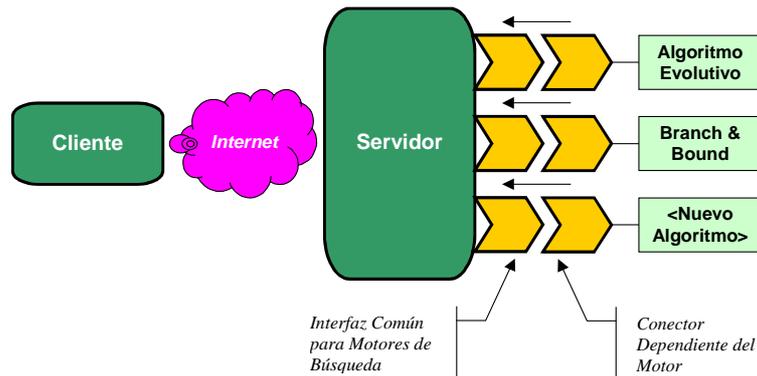


Figura 2. Arquitectura software para el servicio de optimización a través de Internet.

En la arquitectura software presentada, la interfaz con el servidor permite establecer y consultar el estado de los motores de optimización para, por ejemplo, ponerlos en funcionamiento y para hacer seguimiento en línea de la búsqueda. La interfaz en sí no tiene ningún componente algorítmico que determine la solución de los problemas, sino que es una herramienta para comunicación con el servidor o incluso entre varios algoritmos entre sí. Esto último permite la generación de nuevos algoritmos por composición (hibridación débil [CT98a]).

Los algoritmos se ejecutarán idealmente en máquinas distintas de la del servidor, y pueden requerir más de un procesador para proporcionar el servicio. Todo esto es independiente del servicio en sí, del cliente y de otros posibles algoritmos, aunque nada impide que tanto el servidor como los algoritmos utilicen políticas sofisticadas para asignación de tareas a procesadores basándose en su carga, uso, potencia relativa, etc. Asimismo, el servicio puede satisfacerse en ordenadores situados geográficamente lejos del servidor, dando lugar a un uso más eficiente de Internet.

Como reflexión final, debemos notar que no se pretende que el foco ni la mayor parte del proyecto se dirija a la implementación del sistema que provee el servicio, sino a los motores de búsqueda y a la resolución con ellos de problemas de entidad real que tengan impacto en la Sociedad. Proporcionar el servicio a través de Internet es adecuado para su difusión y resulta en un valor añadido importante, pero el peso está en las aplicaciones y en la ratio eficacia/eficiencia de los algoritmos que se desarrollen.

2.3.2 Sistemas Similares Existentes en Internet

Para realizar una aportación de interés actual resulta imprescindible estudiar sistemas que definan servicios y/o medios de proporcionarlos similares a los que pretendemos crear en este proyecto.

El trabajo de comparación podemos dividirlo en dos partes: sistemas que permitan optimización usando redes de ordenadores y sistemas que oferten servicios de optimización a través de Internet. En el primer caso nos referimos a sistemas, normalmente paralelos, para resolver un problema complejo. En este sentido, existen bibliotecas genéricas para paso de mensajes como MPI o sistemas de meta-computación o GRID como Globus, Condor o incluso CORBA, que han permitido a los investigadores diseñar algoritmos en red. Nuestro acercamiento en este aspecto diferirá en cuanto al modelo básico de ejecución, ya que proponemos una implementación tanto en Java como re-utilizando código existente en C++; consideraremos también la interacción con algoritmos descritos en MPI o sobre Condor allí donde resulte de interés para formar un cuerpo de resolutores de alta calidad y eficiencia.

Respecto al segundo punto, el otro gran objetivo es ofertar el servicio resultante en Internet. En la actualidad, el número de sistemas que hacen esto es reducido. Un ejemplo de especial interés es la iniciativa NEOS (<http://www-neos.mcs.anl.gov/neos>). Este sistema presenta a los usuarios una colección de solucionadores ordenados por el tipo de algoritmos (deterministas, estocásticos, programación lineal, optimización con restricciones y sin ellas, etc.) de manera que un cliente puede especificar su problema y resolverlo. Aunque el objetivo de nuestro sistema de valor añadido es similar, **nuestra propuesta es distinta en cuatro aspectos muy importantes:**

1. NEOS impone un lenguaje concreto de tipo algebraico para especificar los problemas (usualmente el denominado AMPL). En nuestro caso, los problemas vendrán especificados en clases de objetos de algún lenguaje moderno (Java, C++ o C#), con una mucho mayor generalidad y flexibilidad para los usuarios. Esto además, no requerirá que el usuario aprenda ningún otro nuevo lenguaje para usar el sistema.
2. NEOS no es realmente un sistema coherente. Esto se debe a que lo que hace es dar de alta a resolutores que han sido implementados por terceras partes. Aunque el sistema de gestión es así más eficiente, nosotros proponemos un sistema coherente y estructuralmente similar para todos los solucionadores, sin perjuicio de que puedan extenderse en el futuro el tipo y número de motores de búsqueda que inicialmente dispongamos.
3. NEOS implementa la solución de problemas utilizando muy diversos sistemas de computación paralela internos. En nuestro caso proponemos una implementación multi-plataforma en Java y otros lenguajes de los solucionadores, de manera que se puedan reutilizar módulos entre ellos y que la conexión a la web sea más simple que con sistemas de meta-computación o paso de mensajes existentes en la actualidad.
4. NEOS oferta los servicios indirectamente y a través de una interfaz web muy básica. Nosotros proponemos una oferta de servicios más actualizada, flexible y completa utilizando el sistema .NET o J2EE, incluyendo posibles migraciones a lenguajes de última generación como C# para los resolutores o la provisión del servicio.

Además de NEOS, existen otros proyectos basados en ofrecer servicios de super-computación a través de la WEB, entre los que se pueden citar WebSubmit [MKD99] y WebFlow [HAFF99].

WebSubmit se compone de una utilidad que permite acceder a aplicaciones que se ejecutan sobre una colección de sistemas de computación heterogéneos, y tiene como objetivo principal el facilitar el uso de dichos recursos mediante la Web, sin requerir que los usuarios tenga conocimientos detallados de los sistemas que están usando. Los sistemas usados son máquinas UNIX, e incluyen utilidades para ejecutar y monitorizar los trabajos sobre dichos sistemas. WebFlow es una infraestructura que tiene la finalidad de acceder a sistemas de meta-computación basados en Globus, usando un navegador, Java y CORBA.

La propuesta de valor añadido de TRACER se diferencia claramente de estos sistemas en su clara orientación a ofrecer un servicio especializado en la resolución de un tipo específico de problemas. Por el contrario, WebSubmit y WebFlow se centran principalmente en la infraestructura, más que en ofrecer un determinado tipo de servicio.

En lo que se refiere a repositorios, son notablemente conocidos repositorios de problemas tales como la OR-library (<http://mscmga.ms.ic.ac.uk/info.html>) en el campo de la optimización combinatoria o el repositorio UCI (<http://www-netra.ics.uci.edu/~mlearn/>) sobre aprendizaje automático. Uno de nuestros objetivos es crear un repositorio de problemas en los dominios mencionados que permita interactuar y colaborar a los miembros de los grupos participantes así como ser ofertado a la comunidad internacional como resultado del presente proyecto.

En lo referente a algoritmos, pretendemos continuar la línea que permite ejecutar esqueletos genéricos del tipo desarrollados en el proyecto coordinado **MALLBA** (TIC1999-0754-C03) pero con un énfasis en la capacidad multi-plaforma de los resultados (usando principalmente Java) y en la resolución eficiente de problemas reales. En este sentido, MALLBA (<http://www.lsi.upc.es/~mallba/>) **no se oferta como un servicio en Internet, sino como una biblioteca en C++** para ejecución paralela LAN/WAN, y además está enfocado principalmente a la evaluación de algoritmos, no a la resolución de problemas de dimensiones reales. Estos son los puntos que se pretenden extender con el presente proyecto.

2.4 Problemas y Algoritmos

En esta sección describiremos tanto los problemas que se pretenden resolver como los algoritmos cuyos motores se proveerán en la versión final del sistema. Cada una de las secciones siguientes está dedicada a estos puntos.

La Figura 3 muestra un resumen con técnicas y problemas que se abordarán en este proyecto. Las siguientes secciones explican en detalle ambos apartados.

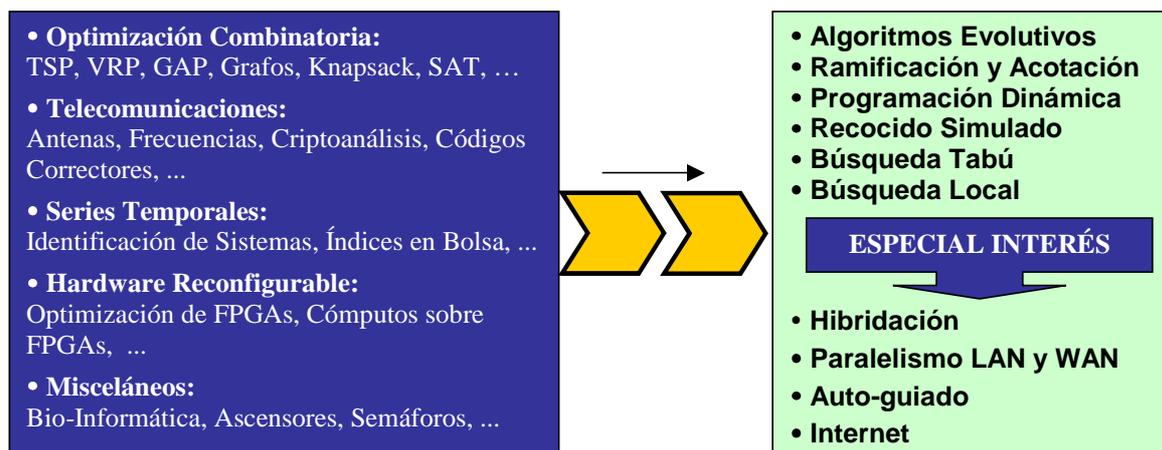


Figura 3. Conjunto de problemas y técnicas de resolución consideradas.

2.4.1 Problemas

Los problemas a los que se aplicará el esquema de servicio propuesto deben cumplir entre otros requisitos el ser de interés actual y de dimensiones reales. No sólo incorporaremos problemas con características interesantes como multi-modalidad, elevada interacción entre sus parámetros (epistasis) o sujetos a restricciones. Los problemas deben presentar estos atributos distintivos y además ser directamente útiles a la comunidad científica y al mercado actual.

La propia selección de dominios de los que extraer problemas debe estar orientada a cumplir estos requisitos. De ahí que hayamos pensado directamente en aplicaciones combinatorias industriales, en el mercado de las telecomunicaciones, y en sistemas de aprendizaje también industriales y científicas.

En relación al dominio de la **optimización combinatoria aplicada**, pretendemos abordar la provisión de servicios en red para tareas aplicadas de interés. Las más importantes se resumen a continuación:

1. **Optimización de Rutas para Flotillas de Vehículos (VRP).** Este tipo de problemas permite elaborar planes de visita para una flotilla de vehículos de distintas capacidades de manera que el tiempo de servicio y el camino recorrido sea el mínimo posible [LGPS99]. Para disponer de una aplicación real debemos contemplar también la disponibilidad de los clientes (horarios), restricciones de los conductores (por ejemplo paradas obligatorias de los conductores), la capacidad de cada vehículo y la posibilidad de mejorar (incluso en tiempo real) el modelo de tráfico que permita diseñar rutas óptimas. Para mejorar el interés de los resultados, deben proporcionarse soluciones a los clientes en pocos minutos, lo que requiere una extensiva investigación en el campo de los heurísticos y del procesamiento paralelo. Numerosas empresas en el sector de la automoción estarían interesadas en los resultados: autobuses escolares, empresas distribuidoras con más de 5 vehículos, servicios de paquetería urgente, etc.
2. **Corte Óptimo de Patrones de Ropa Industrial (PCP).** Este problema consiste en maximizar el número de prendas de vestir que puede extraerse de un determinado área de tela para reducir las pérdidas de empresas de corte y confección. El problema existe en numerosas otras áreas como el corte óptimo de barras de acero o de bloques de madera, a veces en versiones más simples de corte en una única dimensión.
3. **Aplicaciones de Problemas en Grafos (GPA).** Hay un considerable número de aplicaciones de optimización combinatoria, sobre todo en relación a los grafos, que admiten una extensión real inmediata de interés. Éste es el caso del problema del viajante de comercio (TSP) y el diseño de rutas o circuitería, la partición de grafos (MAXCUT) y problemas de decisión, o la satisfacción de cláusulas lógicas (MAXSAT) y problemas en ingeniería. En el proyecto se proponen abordar estos problemas y algunos relacionados que dispongan de una orientación práctica.

En el terreno de las **telecomunicaciones** los problemas seleccionados presentan una equilibrio entre los requisitos expresados:

1. **Error Correcting Codes (ECC).** Este tipo de problemas consiste en diseñar alfabetos de palabras-código que presenten una distancia de Hamming mínima lo más elevada posible (maximización de distancia mínima entre palabras). Este problema aparece con frecuencia en el dominio de las transmisiones digitales por red, y ha sido abordado con algunos heurísticos en el pasado [CFW98].
2. **Radio Network Design Problems (RND).** El objetivo de este problema es situar un conjunto de antenas repetidoras en un área geográfica dada, de manera que se cubra la mayor superficie posible con el mínimo número de antenas [CGKK97]. La relación con otro tipo de problemas y el impacto económico de esta aplicación para las empresas proveedoras de servicio la hacen merecedora de interés para este proyecto.
3. **Radio Link Frequency Assignment Problem (RLFAP).** Se trata de una tarea de optimización muy dura debido a que se debe asignar un número mínimo de frecuencias distintas a un conjunto muy elevado de enlaces radio usados para la comunicación de terminales móviles [KRS95]. El elevado conjunto de restricciones proviene de la separación en dominios de valores de las frecuencias que se deben asignar (un enlace sólo puede admitir ciertas frecuencias) y un elevado número (varios miles) de restricciones para evitar la interferencia entre las frecuencias asignadas en una solución final.
4. **Criptoanálisis.** En general, la evaluación de los sistemas de criptoanálisis es una tarea compleja y no ha sido resuelta de una manera eficiente. Es posible abordar este problema con la utilización de un sistema basado en algoritmos genéticos que nos dé una nueva medida eficaz capaz de valorar si un sistema de criptoanálisis es seguro o no. La base de estos métodos consiste en detectar si el sistema de cifrado tiene vulnerabilidades en la entrada a partir de regularidades encontradas en la salida cuando dicha entrada toma unos valores específicos [HRIS01].
5. **Otros Problemas.** En el dominio de las telecomunicaciones existen numerosos otros problemas de interés que pueden plantearse como servicio de optimización. Algunos de ellos son la disposición óptima de terminales para conectarlos a un concentrador, el diseño óptimo de redes de fibra óptica SONET para resultar en alta resistencia a fallos, implementación de algoritmos óptimos para encaminamiento de la información en redes, sistemas de criptoanálisis, etc.

El tercer bloque de problemas podemos encontrarlo en las aplicaciones sobre arquitecturas de **hardware reconfigurable**, tanto como objetivo de optimización como plataformas para la ejecución de algoritmos. En particular, las líneas que trabajar son:

1. **Optimización usando Programación Genética.** El paradigma de programación genética (PG) será evaluado en tareas de optimización del conexionado de circuitos sobre hardware reconfigurable. Diferentes modelos de PG, especialmente modelos paralelos, serán evaluados en esta aplicación, así como en otras tareas de optimización típicas de este dominio para evaluar las técnicas PG desarrolladas.
2. **Modelado de Series Temporales.** Dado un fenómeno que siga una serie temporal, es decir, que se vayan generando datos a lo largo del tiempo, se trataría de poder predecir qué datos van a generarse en el futuro. Existen un gran número de problemas que pueden catalogarse de esta manera. Algunos ejemplos de este tipo de problemas pueden ser la evolución de las cotizaciones bursátiles, la evolución de las divisas, la evolución del nivel de las mareas, la evolución de las temperaturas, etc [OFI99]. Este problema se abordará usando algoritmos evolutivos; adicionalmente, a partir de los resultados actuales en el proyecto TIC2000-0475 para identificación de sistemas, se pretende sentar las bases teóricas y tecnológicas para adecuar una arquitectura paralela reconfigurable al modelado matemático de series temporales en tiempo real que representen un sistema dinámico genérico.
3. **Inspección Automática de Calidad.** A partir del proyecto regional IPR98A030 de la UNEX, se pretende extender los resultados para desarrollar una plataforma reconfigurable con capacidad de acción directa en entornos industriales para la inspección de materiales (por ej. corcho).

Por último, englobamos con el nombre genérico de "**aplicaciones modernas**" a un conjunto de problemas que tienen interés práctico además de una complejidad importante. De hecho, es posible que durante la vida del proyecto se identifiquen nuevas aplicaciones en este tipo de dominios:

2. **Bio-Informática (BIO)**. Se trata de problemas de optimización cuya solución permita ayudar en la detección de posibles enfermedades o bien en el modelado de tareas en este dominio.
3. **Ingeniería (ING)**. Estos problemas son de interés en varios campos tales como el diseño de coches o computadoras y también en aplicaciones civiles.
4. **Problemas de Asignación (GAP)**. En general, la asignación mínima de recursos a tareas es de gran interés. En particular, proponemos el trabajo con problemas de asignación de personas a trabajos, procesos a procesadores en un sistema paralelo, y aplicaciones similares en este campo que puedan surgir.

El conjunto de problemas final podría ampliarse o reducirse un poco en el transcurso de los tres años de proyecto en relación a contactos con otros centros de investigación o explotación que puedan expresar su interés en algún dominio concreto. Naturalmente, la concesión de personal asociado de investigación (becarios o contratados) a este proyecto mejoraría notablemente el espectro de aplicaciones abordadas.

2.4.2 Algoritmos

En el proyecto se aboga por el diseño y futuras extensiones de motores de optimización que incluyan algoritmos de última generación modernos. Esto supone que se incluirán algoritmos de cualquier procedencia con el objetivo de conseguir un sistema de respuesta eficiente y efectiva según los criterios del usuario. Así, se incluirán tanto algoritmos heurísticos como exactos, y cualesquiera extensiones (paralela, híbrida, sujeta a restricciones, multi-objetivo, etc.) que puedan mejorar la implementación del servicio.

En particular, se implementarán los siguientes motores de optimización:

1. Algoritmos Evolutivos (EA) en sus distintas familias conocidas, tales como algoritmos genéticos (GA), estrategias evolutivas (ES), programación genética (GP) y posibles conexiones con otras técnicas.
2. Algoritmos de ramificación y acotación y programación dinámica, haciendo hincapié en el estudio de nuevas funciones de acotación más eficientes para problemas complejos.
3. Algoritmos de inspiración natural tales como el recocido simulado (SA) y sus múltiples variantes.
4. Algoritmos de seguimiento del gradiente (HC), tanto genéricos como dependientes del problema (por ejemplo para la resolución del problema del viajante de comercio).
5. Algoritmos híbridos (HA), tanto genéricos como siguiendo un patrón concreto de búsqueda, como es el caso de los algoritmos meméticos basados en búsqueda local (LSMA).
6. Algoritmos multi-objetivo en el campo de los algoritmos evolutivos (MOEA), para problemas que requieran optimización atendiendo a múltiples criterios a la vez.
7. Otros algoritmos, tales como estimación de distribuciones (EDA), sistemas de colonias de hormigas (ACO) [DMC96], búsqueda tabú (TS), búsqueda dispersa (SS), mapas auto-organizados de características (SOFM) [Isasi93][DW87], o técnicas específicas para la satisfacción de restricciones (RES).

En general, el objetivo es proporcionar resultados óptimos de manera eficiente, por lo que resultará imprescindible incluir conocimiento sobre los problemas en el interior de los algoritmos y también estudiar posibles combinaciones novedosas y extensiones paralelas que aumenten la eficiencia final.

Los algoritmos desarrollados serán en buena parte nuevos, y estarán sujetos a unos mínimos requisitos de interfaz para su correcto funcionamiento uniforme y conjunto. Se pretende un desarrollo lo más desacoplado posible que pueda reforzar la reutilización futura del software. **De hecho, el procedimiento de uso de los motores de búsqueda debería permitir utilizar algoritmos ya desarrollados en Java o C++ tras extenderlos con los *wrappers* adecuados para interaccionar como sistemas.**

Bibliografía

- [AAB+01] Alba E., Almeida F., Blesa M., Cotta C., Díaz M., Dorta I., Gabarró J., González J., León C., Moreno L., Petit J., Roda J., Rojas A., Xhafa F. (2001). "MALLBA: Toward a Combinatorial Optimization Library for Geographically Distributed Systems", XII Jornadas de Paralelismo, pp. 105-110, Editorial U.P.V.
- [AK01] Alba E., Khuri S. (2001). "Applying Evolutionary Algorithms to Combinatorial Optimization Problems". In Alexandrov V.N., Dongara J.J., Juliano B.A., Renner R.S., Tan C.J.K. (eds.), *Proceedings of the International Conference on Computational Science (ICCS'01)*, LNCS vol. 2074, Part II, Springer-Verlag, Berlin, Heidelberg, pp. 689-700
- [ANT02] Alba E., Nebro A.J., Troya J.M. (2002 to appear). "Heterogeneous Computing and Parallel Genetic Algorithms". *Journal of Parallel and Distributed Computing*
- [ADGGGLRRS 00] Almeida F., Dorta M.I., García F., González J.A., González D., León C., Roda J.L., Rodríguez C., Sande F. (2000). "Algoritmos Exactos de Optimización Combinatoria: Una aproximación Geográficamente Distribuida". En *XI Jornadas de Paralelismo*. Granada
- [BFM97] Bäck T., Fogel D.B., Michalewicz, Z. (1997). *Handbook of Evolutionary Computation*. Oxford University Press, New York NY
- [CDG99] Corne D., Dorigo M., Glover F. (1999). *New Ideas in Optimization*. McGraw-Hill, Cambridge
- [CGKK97] Calégari P., Guidec F., Kuonen P., Kobler D. (1997). "Parallel Island-Based Genetic Algorithm for Radio Network Design". *Journal of Parallel and Distributed Computing*, 47:86-90
- [CFW98] Chen H., Flann N.S., Watson D.W. (1998). "Parallel Genetic Simulated Annealing: A Massively Parallel SIMD Algorithm". *IEEE Transactions on Parallel and Distributed Systems*, 9(2), pp. 126-136
- [CASL01] Cotta C., Alba E., Sagarna R., Larrañaga P. (2001). "Adjusting Weights in Artificial Neural Networks using Evolutionary Algorithms". In P. Larrañaga, J.A. Lozano (eds.) *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*, chapter 18, pp. 357-373, Kluwer Academic Publishers
- [DMC96] Dorigo M., Maniezzo V., Colorni A.(1996). "The Ants Systems: Optimisation by a Colony of Co-operating Agents". *IEEE Transactions on Systems, Man, and Cybernetics*, Part B, 26:9-41
- [DPS97] De Jong K.A., Potter M.A., Spears W.M. (1997). "Using Problem Generators to Explore the Effects of Epistasis". In Bäck T. (ed.) *Proceedings of the 7th International Conference of Genetic Algorithms*, pp. 338-345, Morgan Kaufmann
- [DW87] Durbin R., Willshaw D. (1987). "An analog approach to the Travelling Salesman Problem using an elastic net method". *Nature* 326. pp. 688-691.
- [FFO01] Fogel D.B., Fogel G.B., Ohkura K. (2001). "Multiple-vector Self-adaptation in Evolutionary Algorithms". *BioSystems*, 61:155-162
- [FI01] Fernández F., Isasi P. (2001). "Designing Nearest Neighbour Classifiers by the Evolution of a Population of Prototypes". *European Symposium on Artificial Neural Networks*
- [GJ79] Garey M.R., Johnson D.S. (1979). "Computers and Intractability: A Guide to the Theory of NP-Completeness". Freeman W.H. and Co., San Francisco, CA
- [Glo89] Glover F. (1989). "Tabu Search", *ORSA Journal of Computing*, 1(3):190-206
- [Gough01] Gough J. (2001). *Compiling for the .NET: Common Language Runtime (CLR)*. Prentice Hall
- [Gunnerson01] Gunnerson E. (2001). *A Programmer's Introduction to C#*.
- [Holzner01] Holzner S. (2001). *XML Complete*. McGraw-Hill

- [IMS97] Isasi P., Molina J.M., Sanchís A. (1997). "Unsupervised Neural Network for Forecasting Alarms in Hydroelectric Power Plant". *International Work-Conference on Artificial and Neural Networks (IWANN'97)*
- [HRIS01] Hernández J.C., Ribagorda A., Isasi P., Sierra J.M. (2001). "Using Genetic Algorithms to Design Linear Congruential Pseudo-random Number Generators". *Genetic and Evolutionary Computation Conference*. pp. 1292-1299..
- [Isasi93] Isasi, P. (1993). "Aprendizaje no supervisado: Redes de Kohonen". *Redes neuronales artificiales: fundamentos y aplicaciones*. Editorial Universidad Alcalá de Henares. pp. 83-102
- [IOFF96] Isasi P., Olmeda I., Fernández E., Fernández C. (1996). "Neural Networks Forecast Intra-Day Futures and Cash Returns". *First International Conference on Machine Learning, Forecast Optimization (MALFO'96)*
- [Kauffmann89] Kauffmann, S.A. (1989). "Adaptation on Rugged Fitness Landscapes". In Stein D.L. (ed.), *Lectures in the Sciences of Complexity*, Volume 1, 527-618. Addison Wesley
- [KGV83] Kirkpatrick S., Gelatt Jr. C.D., Vecchi, M.P. (1983). "Optimization by Simulated Annealing", *Science* 220(4598):671-680
- [KRS95] Kapsalis A., Rayward-Smith V.J., Smith G.D. (1995) "Using Genetic Algorithms to Solve the Radio Link Frequency Assignment Problem". In Pearson D. W., Steele N. C., Albrecht R. F. (eds.), *Proceedings of the International Conference on Artificial Neural Nets and Genetic Algorithms*. Springer-Verlag., pp. 37-40
- [LGPS99] Laporte G., Gendreau M., Potvin J.Y., Semet F. (1999). *Classical and Modern Heuristics for the Vehicle Routing Problem*. Tech. Rep. G-99-21, Univ. Montreal
- [MC02] Moscato P., Cotta C. (2002). "A Gentle Introduction to Memetic Algorithms". *Handbook of Metaheuristics*, Glover F., Kochenberger G. (eds.), Kluwer Academic Publishers, Boston
- [MF98] Michalewicz Z., Fogel D.B. (1998). *How to Solve It: Modern Heuristics*. Springer Verlag, Berlin Heidelberg
- [MKD99] McCormack R., Koontz J., Devaney J. (1999). "Seamless Computing with WebSubmit". *Concurrency: Practice and Experience* 11(15), December 25. <http://math.nist.gov/mcsd/savg/websubmit/>
- [HAFF99] Haupt T., Akarsu E., Fox G., W. Furmanski. (1999). "Web Based Meta-computing". <http://www.npac.syr.edu/users/haupt/WebFlow/>
- [MMS00] Merkle D., Middendorf M., Schneck H. (2000). "Ant Colony Optimisation for Resource Constrained Project Scheduling". *Genetic and Evolutionary Computation Conference*. pp. 893-902
- [MP96] Mühlenbein H., Paass G. (1996). "From Recombination of Genes to the Estimation of Distributions I. Binary Parameters". In *Parallel Problem Solving From Nature IV* (pp. 179-197), LNCS 1411. Springer Verlag, Berlin Heidelberg
- [MS02] Michaelis M., Spokas P. (2002). *C# Developer's Headstart*. McGraw-Hill
- [NDV97] Naudts B., Dominique S., Verschoren A. (1997). "Epistasis as a Basic Concept in Formal Landscape Analysis". In Bäck T. (ed.) *Procs. of the 7th ICGA*, 65-72. Morgan Kaufmann
- [Nov98] Novkovic S. (1998). "A Genetic Algorithm Simulation of a Transition Economy: An Application to Insider-Privatization in Croatia". *Computational Economics*, 11(3):221-243
- [OFI99] Olmeda I., Fernández E., Isasi P. (1999). "Exploring nonlinearity in ultra-high frequency stockmarket with artificial neural networks". *Neural Network World*. 7(3):179-192
- [Platt 01] Platt D.S. (2001). *Introducing Microsoft® .NET*. Microsoft Press
- [PS00] Pena-Reyes C.A., Sipper M. (2000). "Evolutionary Computation in Medicine: An Overview", *Artificial Intelligence in Medicine* 19(1):1-23

- [Ree93] Reeves C.R. (1993). *Modern Heuristic Techniques for Combinatorial Problems*. Blackwell Scientific Publications, Oxford
- [RORS96] Rayward-Smith V.J., Osman I.H., Reeves C.R., Smith G.D. (1996). *Modern Heuristic Search Methods*. John Wiley & Sons, Chichester
- [TFVF01] Teich T., Fisher M., Vogel A., Fisher J. (2001). "A New Ant Colony Algorithm for the Job-Shop Scheduling Problem". *Genetic and Evolutionary Computation Conference*. pp. 803--806
- [VR01] Vawter C., Roman E. (2001). "J2EE vs. Microsoft.NET: A Comparison of Building XML-based Web Services". <http://www.theserverside.com/resources/article.jsp>
- [WM97] Wolpert D.H., Macready W.G. (1997). "No Free Lunch Theorems for Optimization". *IEEE Transactions on Evolutionary Computation* 1(1):67-82.
- [WRDM96] Whitley D., Rana S., Dzubera J., Mathias K.E. (1996). "Evaluating Evolutionary Algorithms". *Artificial Intelligence*, 85:245-276
- [Yeh99] Yeh I.C. (1999). "Hybrid Genetic Algorithms for Optimization of Truss Structures". *Computer Aided Civil and Infrastructure Engineering* 14(3):199-206
- [ZU01] Zaman Ahmed K., Umrysh C.E. (2001). *Developing Enterprise Java Applications with J2EE™ and UML*. Addison-Wesley

3. OBJETIVOS DEL PROYECTO

(máximo dos páginas)

- ◆ Deben **enumerarse brevemente** y describirse con claridad, precisión y de manera realista (es decir, acorde con la duración prevista del proyecto) los objetivos concretos que se persiguen, los cuales (salvo en el caso de proyectos presentados al Programa Nacional de Promoción General del Conocimiento) deben adecuarse a las líneas temáticas prioritarias del Programa Nacional o Acción Estratégica al que se adscribe el proyecto (*ver Anexo de la convocatoria*).
- ◆ La novedad y relevancia de los objetivos (así como la precisión en la definición de los mismos) se mencionan explícitamente en los criterios de evaluación de las solicitudes (*ver apartado 11º.1 de la Convocatoria*)

El objetivo primario del presente proyecto es avanzar en el diseño de técnicas de optimización modernas hasta llegar a la resolución eficiente de problemas complejos de gran dimensión.

Como objetivos secundarios o de valor añadido se proponen: (a) ofertar las técnicas desarrolladas como servicio en Internet usando tecnología de última generación accesible para la comunidad científica y (b) crear un repositorio de problemas de dificultad considerable con información que fomente la interacción y colaboración con grupos internacionales.

En la actualidad, las herramientas de optimización y aprendizaje permiten abordar problemas cada vez más desafiantes que en el pasado se resolvían de manera poco eficiente o incluso para los que no existía una solución admisible. El elevado tiempo de cómputo y el desconocimiento de paradigmas de optimización modernos (especialmente, heurísticas) para muchos investigadores han definido un conjunto nutrido de problemas para los que se necesita avanzar en eficiencia.

Nuestro objetivo primario consiste en abordar problemas complejos para hacer trasvase de conocimientos entre ellos, y poder extender las técnicas modernas de optimización de forma que resulten eficientes y competitivas. Esto supone un estudio teórico de los problemas y los algoritmos, y la realización de numerosos estudios comparativos considerando además técnicas existentes en los dominios de donde provienen dichos problemas.

Como vemos, el objetivo es múltiple, pues además de diseñar nuevos algoritmos y resolver problemas reales pretendemos realizar labores de valor añadido importantes como son realizar repositorios de problemas y algoritmos, y ofertar un servicio *on-line* de ejecución para abordar nuevos problemas futuros no contemplados actualmente en la definición del proyecto.

En general, los problemas que pretendemos abordar en el presente proyecto pertenecen a dominios muy diferentes, como son:

1. **Optimización Combinatoria:** diseño de rutas óptimas de reparto en flotillas de vehículos, corte óptimo de prendas en patronaje industrial, problemas reducibles a instancias del viajante de comercio (TSP), satisfacción de cláusulas lógicas (SAT), particionado de grafos (MAXCUT), planificación de tareas, etc.
2. **Telecomunicaciones:** problemas en el ámbito de diseño de redes, criptoanálisis, encaminamiento de la información, diseño de códigos correctores de errores óptimos, posicionamiento de antenas repetidoras, asignación de frecuencias a enlaces de radio, etc.
3. **Hardware Reconfigurable:** diseño de circuitos FPGA, problemas de clasificación en visión artificial, etc.
4. **Aplicaciones Modernas:** selección de problemas de interés en dominios actuales como la bio-informática o la ingeniería, o la predicción del comportamiento de series temporales, con especial hincapié en que tengan componentes multi-objetivo, satisfacción de restricciones o que sean problemas dinámicos.

La selección de problemas e instancias para cada fase del proyecto se realizará en atención al tipo de dificultad intrínseca que presenten. Adicionalmente, es muy acentuado en la actualidad la necesidad de usar paralelismo si se desea eficiencia (tanto en LAN como en WAN), así como incorporar de manera estructurada información del problema para mejorar la eficiencia numérica [WM97]. Estas dos características, junto con cierta capacidad de auto-guiado de la búsqueda, serán incorporadas también en el plano de los algoritmos como objetivos de calidad.