

Resolución de Problemas en Ingeniería Utilizando Técnicas de Inteligencia Artificial

C. Ansótegui, R. Béjar, A. Cabiscol y F. Manyà
Departamento de Informática e Ingeniería Industrial
Universitat de Lleida, Jaume II, 69, E-25001 Lleida
e-mail: {carlos,ramon,alba,felip}@eup.udl.es

Resumen

En este artículo presentamos el temario para una asignatura sobre resolución de problemas en ingeniería utilizando técnicas de inteligencia artificial. Por un lado, el temario cubre diferentes métodos automáticos (algoritmos) para resolver problemas de decisión y de optimización, tanto exactos como inexactos. Por otro lado, el temario contempla el estudio de problemas de interés en la ingeniería, centrándose básicamente en problemas de scheduling y de planificación, de forma que se enseña a especificar los problemas de forma adecuada para su resolución mediante los métodos vistos y a comprender las dificultades que estos problemas plantean en la práctica. El temario contempla la introducción y uso, mediante prácticas, de sistemas reales para la resolución de problemas de scheduling y planificación.

Palabras clave: inteligencia artificial, docencia, scheduling, planificación

1. Introducción

Muchos problemas de decisión y optimización que deben resolverse en ingeniería son difíciles de resolver utilizando métodos que basan su estrategia de resolución en métodos tradicionales tales como la programación lineal o entera. En estos casos, la alternativa consiste en utilizar métodos que incorporan técnicas inteligentes de resolución de problemas. De hecho, estos métodos han surgido como necesidad de resolver problemas reales en el ámbito industrial o de servicios (organización de las telecomunicaciones, logística, transportes o localización de servicios, etc...).

En este artículo describimos y justificamos el programa de una asignatura de Inteligencia Artificial que estamos impartiendo desde el curso 2000/2001 en los estudios de Ingeniería Técnica Industrial, especialidad Mecánica, de la Universidad de Lleida. Se trata de una asignatura optativa que pueden cursar los alumnos de tercer curso, y que tiene una carga lectiva de 6 créditos.

El programa se divide en tres bloques básicos. El primer bloque es un bloque introductorio que pretende contextualizar el tipo y características de los problemas que se van a tratar. El segundo bloque aborda problemas de decisión y algoritmos para su resolución. Los problemas tratados son básicamente problemas de scheduling. El tercer bloque aborda el tema de los problemas de optimización y sus algoritmos. Los problemas tratados son problemas de scheduling y planificación.

Un punto diferencial de este curso, respecto a un curso orientado a estudiantes de Informática, es que aquí se hace más hincapié en los aspectos asociados a la comprensión y correcta representación, mediante lenguajes de especificación, de los problemas a resolver, y a las técnicas de evaluación experimental del rendimiento de los algoritmos. Se deja de lado aquellos aspectos asociados a la programación meticulosa de algoritmos eficientes, ya que no se considera un curso orientado a alumnos con fuertes conocimientos en programación. La idea de este curso es que el alumno aprenda a comprender las restricciones de diferentes problemas de decisión y optimización, sea capaz de razonar sobre las dificultades para su resolución y sea capaz de seleccionar posibles algoritmos y sistemas reales disponibles que puedan resolver los problemas, una vez especifique correctamente el problema.

2. Temario

Como hemos dicho, el temario de la asignatura se encuentra dividido en tres bloques. A continuación mostramos el temario de la asignatura, para pasar seguidamente a su descripción y discusión.

- I. Introducción a los problemas de decisión y optimización
 - 1.1. Problemas combinatorios. Definición y ejemplos
 - 1.2. Espacio de búsqueda y explosión combinatoria
 - 1.3. Problemas de decisión
 - 1.3. Problemas de optimización
 - 1.4. Algoritmos no inteligentes

- II. Algoritmos y problemas de decisión
 - 2.1. Un problema simple: SAT
 - 2.2. Un problema más general: CSP
 - 2.3. Búsqueda por backtracking para SAT
 - 2.3. Búsqueda por backtracking para CSP
 - 2.4. Algoritmos incompletos: búsqueda local
 - 2.5. Problemas de scheduling de decisión
 - 2.6. Backtracking para scheduling
 - 2.7. Especificación y resolución de problemas de decisión con el sistema *Mozart*

- III. Algoritmos y problemas de optimización
 - 3.1. Un problema simple: MAX-SAT con pesos
 - 3.2. Un problema más general: MAX-CSP con pesos
 - 3.3. Búsqueda por ramificación y acotación
 - 3.4. Optimización por búsqueda local
 - 3.5. Problemas de scheduling de optimización
 - 3.6. Problemas de planificación
 - 3.7. Búsqueda por ramificación y cota para scheduling
 - 3.8. Especificación y resolución de problemas de scheduling de optimización con el sistema *Mozart*
 - 3.9. Especificación y resolución de problemas de planificación con el sistema *Blackbox*

El primer bloque, Introducción a los problemas de decisión y optimización, pretende contextualizar los contenidos que se verán en la asignatura, así como introducir los conceptos fundamentales que se emplearán. Estos conceptos se introducen de forma bastante intuitiva, y de manera que no se supone que el alumno ha tenido anteriormente contacto con tales conceptos. De entre los conceptos introducidos consideramos fundamentales los de espacio de búsqueda y explosión combinatoria. La comprensión de los mismos es fundamental para iniciarse en el problema de la tratabilidad de los problemas de decisión o de optimización que se verán en la asignatura.

El segundo bloque, Algoritmos y problemas de decisión, pretende introducir al alumno en los problemas de decisión y en los algoritmos existentes para su resolución. La idea general es introducir los problemas a resolver y seguidamente ver los algoritmos disponibles para su resolución. Se empieza introduciendo uno de los problemas de decisión conceptualmente más simples en la informática, el problema SAT, que sin embargo contiene toda la complejidad y riqueza de cualquier otro problema de decisión, por ser un problema NP-completo. SAT es importante porque es uno de los problemas para los que existen algoritmos altamente eficientes para su resolución, y frecuentemente es empleado como un lenguaje para especificar otros problemas de decisión para los cuales no existen algoritmos eficientes. A continuación se introduce un problema más general, el CSP, y seguidamente se empiezan a ver los algoritmos disponibles para SAT y CSP: algoritmos sistemáticos, basados en búsqueda por backtracking, y algoritmos incompletos, basados en búsqueda local. A continuación se introducen los problemas de scheduling de decisión y algoritmos para su resolución. Finalmente, se inician las clases de prácticas donde el alumno empleará el sistema Mozart para la especificación y resolución de problemas de scheduling. En la siguiente sección se habla con más detalle sobre las clases de prácticas.

El tercer bloque, Algoritmos y problemas de optimización, persigue objetivos análogos a los del segundo bloque, pero ahora centrados alrededor de los problemas de optimización. Los problemas de optimización se introducen de forma natural como una generalización de los problemas de decisión, en el sentido de que lo que se nos pide resolver en estos últimos es más complejo que en los problemas de decisión. Una respuesta a un problema de optimización nos permite responder a muchos problemas de decisión. Los primeros problemas vistos en este bloque son MAX-SAT y MAX-CSP, generalizaciones naturales de sus correspondientes problemas de decisión SAT y CSP. A continuación se introducen diversos algoritmos para resolver estos problemas, también basados en búsqueda sistemática y búsqueda local. Como problemas reales se introducen los problemas de scheduling de optimización y los problemas de planificación. Se comenta en detalle un posible algoritmo de optimización para

scheduling, debido principalmente a la inmensa cantidad de investigación que se ha realizado sobre algoritmos de optimización para scheduling. Finalmente se entra en la parte de clases de prácticas, donde el alumno empleará los sistemas *Mozart* y *Blackbox* para especificar y resolver problemas de optimización de scheduling y de planificación.

3. Clases de prácticas

Las clases de prácticas en el laboratorio forman una parte esencial en este temario. Muchos de los conceptos e ideas que se pretenden mostrar al alumno sólo pueden verse a la hora de emplear las herramientas que se verán en el laboratorio. Básicamente, el alumno utilizará dos sistemas: el entorno de programación con restricciones *Mozart*¹ y el sistema de resolución de problemas de planificación *BlackBox*².

El sistema *Mozart* es un entorno de programación con restricciones que permite emplear de forma mixta los diferentes paradigmas de programación existentes: declarativo, funcional e imperativo. A la hora de especificar el problema a resolver *Mozart* nos permite emplear una sintaxis meramente declarativa, que es la forma más natural de especificar un problema y sus restricciones. A la hora de resolver el problema, *Mozart* nos permite emplear programas tanto funcionales como imperativos o mezclas de ambos. Introducir el problema de forma declarativa es muy importante para el ingeniero, ya que le permite abstraerse de detalles tediosos sobre estructuras de datos.

El sistema *Blackbox* es considerado como uno de los sistemas de libre distribución más potente y versátil para la resolución de problemas de planificación. Se basa en la utilización de un lenguaje declarativo para la especificación de los problemas de planificación y en su posterior transformación al problema SAT para resolver el problema mediante potentes algoritmos para SAT. El empleo indistinto de algoritmos tanto sistemáticos como de búsqueda local para SAT le permite resolver instancias de naturaleza muy diferente.

¹ El sitio web del sistema Mozart es <http://www.mozart-oz.org>

² El sitio web del sistema BlackBox es <http://www.cs.washington.edu/homes/kautz/blackbox>

4. Bibliografía

1. M. Ginsberg. Essentials of Artificial Intelligence, Morgan Kaufmann, 1993.
2. N.J. Nilsson. Inteligencia Artificial, McGraw-Hill, 2001.
3. E. Rich and K. Knight. Artificial Intelligence, 2nd Edition, McGraw-Hill, 1991.
4. S. Russell and P. Norvig. Artificial Intelligence: A Modern Approach, Prentice Hall, 1995.
5. P.H. Winston. Artificial Intelligence, 3rd Edition, Addison-Wesley, 1992.