

Técnicas de planificación para la generación automática de programas de control

Jaime Alberto Guzmán Luna¹

RESUMEN

En el presente artículo se estudia la aplicación de las técnicas de planificación de inteligencia artificial para la generación automática de programas de control secuencial. Para tal efecto, se presenta una aproximación al estado del arte sobre las técnicas de planificación de inteligencia artificial más conocidas, los diferentes enfoques en que se aplican en los sistemas de manufacturación y se ilustra su aplicación en el desarrollo de un prototipo de sistema inteligente, basado en el planificador STRIPS, cuya función es producir la secuencia de operaciones de una máquina de mecanizado.

Palabras Clave: Planificación de Inteligencia Artificial, Sistemas de Manufacturación, Programa de Control secuencial. Problema de Mecanizado.

I. INTRODUCCIÓN

Uno de los principales aspectos de las técnicas de planificación de inteligencia artificial [1, 15, 20] es la fuerte semejanza entre los procesos de planificación y los de programación y entre el concepto de un plan y el de un algoritmo. Aunque existió un fuerte interés y varios trabajos se llevaron a cabo al inicio bajo el aspecto de la síntesis automática de programas [10, 14] muy pocos avances se han producido desde entonces. En este sentido, el dominio de los sistemas de manufacturación se está convirtiendo en un área de creciente interés para los investigadores del campo de los sistemas de planificación de inteligencia artificial.

Este artículo estudia la aplicación de las técnicas de planificación de inteligencia artificial para la generación automática de programas de control secuencial en problemas específicos de manufactura. En la siguiente sección se presenta un resumen de los conceptos básicos relacionados con la planificación de inteligencia artificial. La sección III presenta la revisión de los diferentes enfoques en que se aplican las técnicas de planificación en los sistemas de manufacturación. La sección IV ilustra las características de un caso práctico en el dominio de un problema de mecanizado. La sección V presenta el diseño del sistema inteligente de planificación para el caso práctico. La sección VI detalla la implementación del sistema inteligente. La sección VII presenta las conclusiones y recomendaciones.

II. LA PLANIFICACIÓN DE INTELIGENCIA ARTIFICIAL

La Inteligencia Artificial se ha centrado durante mucho tiempo en el razonamiento sobre acciones y planes para enfrentar problemas reales a través de una de sus subdisciplinas más importantes: la planificación. Su objetivo es la construcción de programas de computador que permitan obtener de forma automática planes adecuados para alcanzar una meta. Estos programas se conocen como planificadores.

Las primeras aproximaciones a los problemas de planificación, que se podrían denominar aproximaciones clásicas [8, 10, 19], enuncian un problema de planificación con base en los siguientes componentes:

1. Una descripción del mundo circundante, los objetos que lo componen, sus características y las relaciones que existen entre ellos. Tal descripción es conocida como *Estado Inicial*.
2. Una descripción de la meta que se desea alcanzar (también conocida como *Objetivo*), es decir, una descripción de las relaciones y de las características que se desean alterar en los objetos que componen el estado inicial. Ambas descripciones se construyen mediante un lenguaje formal.
3. Una descripción de todas las posibles acciones que se pueden realizar, como afecta cada una de ellas a los objetos del mundo, a sus características y a sus relaciones, a partir de un lenguaje formal. Esta última descripción también se conoce como *Teoría del Dominio*.

Así el resultado esperado de un planificador es un comportamiento, esto es, una secuencia ordenada de acciones tal que, si se lleva a cabo sobre el mundo descrito en el Estado Inicial, permite alcanzar el estado objetivo. Esta secuencia de acciones es conocida como un *plan*.

A. Representación del mundo, acciones y cambio

Se ha utilizado una variedad de lenguajes para representar el estado del mundo, las acciones disponibles y el objetivo a alcanzar, y existe una serie de características comunes a todos ellos basadas en un conjunto de suposiciones preestablecidas para restringir las características del mundo que se pueden enfrentar. Algunas de estas suposiciones son:

¹ Miembro del grupo de investigación en Inteligencia Artificial, Universidad Nacional, Medellín.

Un algoritmo de planificación basado en un proceso de búsqueda puede calificarse como un planificador por regresión.

- El resultado de la ejecución de una acción en el mundo es perfectamente predecible.
- El conocimiento que el planificador posee sobre el mundo es completo.
- Las acciones que se conocen son las únicas capaces de alterar el mundo.

A principios de los 70 surge el planificador STRIPS [8], el cual está basado en la arquitectura de GPS [19], un conocido resolutor de problemas de propósito general. En este se mantiene una representación basada en la lógica de predicados para los estados del mundo y considera representaciones ad-hoc para las acciones. Cada estado se representa mediante conjunciones de literales básicos. Su contribución técnica más importante se conoce como la representación de acciones de STRIPS. Esta representación y el algoritmo para modelar los efectos de una acción constituyen lo que podría denominarse la suposición de STRIPS, que consiste en que la ejecución de una acción afecta únicamente a aquellos aspectos recogidos explícitamente en la descripción del operador.

En la representación de acciones de STRIPS, cada acción se describe a partir de tres componentes:

- Lista de precondition: es una lista de predicados que deben ser ciertos en un estado para que pueda aplicarse una acción.
- Lista de adición: es una lista de predicados nuevos que se hacen ciertos tras la aplicación de la acción.
- Lista de supresión: es una lista de predicados antiguos que dejan de ser ciertos tras la aplicación de la acción.

Esta representación de acciones tuvo gran acogida por los investigadores en planificación y fue usada como la base de la mayoría de las técnicas de planificación automática desde principios de los años 70 como ABSTRIPS [26], NOAH [27], NONLIN [28] o TWEAK [6] hasta los años 90 como PRIAR [31], PRODIGY [31] o GRAPHPLAN [2], pasando por técnicas dotadas por una gran capacidad expresiva como UCPOP [23] y otras técnicas con mayor vocación práctica como OPLAN-2 [29] o SIPE2 [32].

B. La planificación como un problema de búsqueda en un espacio de estados

Dada la descripción de un problema de planificación a partir de estados iniciales, estados objetivos y un conjunto de operaciones o acciones disponibles, su solución podría ser vista como un problema de búsqueda, en el cual empezando por el estado inicial y, aplicando las acciones disponibles, una por una, e ir obteniendo nuevos estados se logre llegar a un estado que satisface la descripción del objetivo.

Para lograr esto, solo se necesitaría además de la descripción del problema, un algoritmo de búsqueda de los normalmente utilizados en cualquier curso de Inteligencia artificial [20, 21, 22, 24, 25]. Un algo-

ritmo como este bien podría llamarse un planificador. Es más, se podría denominar un planificador por espacio de estados, puesto que la búsqueda se realiza a través de un espacio de estados posibles como el mostrado en Fig. 1.

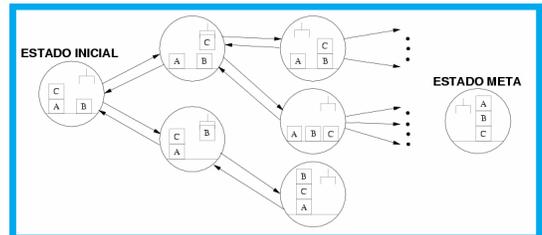


Fig. 1. Espacio de estados

En este espacio de estados, cada nodo representa una posible descripción del mundo circundante y cada arco que une dos nodos representa la aplicación de una acción. De esta forma, el camino que se busca hasta el estado meta constituye, realmente, el plan de acción deseado para resolver el problema. Estos planes obtenidos como resultado de una búsqueda en un espacio de estados son planes totalmente ordenados de acciones, en el sentido de que dos acciones se encuentran ordenadas entre sí formando una única secuencia como la mostrada en Fig. 2.



Fig. 2. Un plan como una secuencia ordenada de acciones

Este tipo de planificador, también se podría clasificar como un planificador por progresión puesto que la búsqueda se desarrolla hacia delante desde el estado inicial hasta un estado meta. El principal problema de este método de construcción hacia delante es que el factor de ramificación, es decir, el número promedio de elecciones que se presentan en cada paso, puede llegar a ser muy elevado, lo que hace que el proceso de búsqueda pueda ser ineficiente.

Una forma de ganar en eficiencia en un proceso de búsqueda en espacio de estados, consiste en construir este proceso de búsqueda en el sentido inverso, es decir, comenzando por el estado meta hasta el estado inicial. Basándose en la suposición de que el estado meta del problema puede contener un menor número de literales que el necesario para describir un estado completo, como el estado inicial, entonces este proceso de búsqueda hacia atrás tendrá un factor de ramificación menor y, por tanto, la búsqueda será más eficiente. Un algoritmo de planificación basado en un proceso de búsqueda como este se podría calificar como un planificador por regresión. El planificador STRIPS es un ejemplo de esta técnica.

III. LA PLANIFICACIÓN DE LOS SISTEMAS DE MANUFACTURACIÓN

Uno de los aspectos más atractivos de las técnicas de planificación de IA es la gran similitud entre el proceso de construcción de un plan de acción y el

proceso de construcción de un programa de computador, así como entre la representación del propio plan y la representación de un programa. La síntesis automática de programas en el dominio de los sistemas de manufacturación esta surgiendo como un área de interés cada vez mayor para los investigadores en planificación de IA, proporcionando un campo restringido para poder aplicar estas técnicas. Un caso particular, es la obtención automática del programa de control que coordina el funcionamiento de un sistema de manufacturación.

A. Los sistemas de manufacturación

Un sistema de manufacturación [3, 9, 12] es un conjunto de procesos, máquinas y fabricas donde se manejan materias primas para transformarlas en productos manufacturados de mayor valor (ver Fig. 3).



Fig. 3. Un sistema de manufacturación

Las transformaciones de distinto tipo que sufren estos productos iniciales son realizadas por las máquinas que componen el sistema de manufacturación, cuyo funcionamiento es controlado por un programa de control secuencial y la supervisión de un operador humano como muestra la Fig. 4. Las transformaciones se llevan a cabo mediante los actuadores, unas máquinas que realizan una cierta función. Estas transformaciones deben estar coordinadas con el estado de la planta, recogido a través de sensores y las ordenes de un operador humano, recogidas a través de un panel de control.

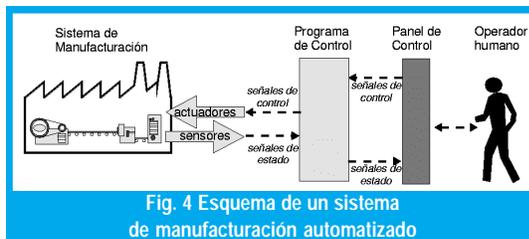


Fig. 4 Esquema de un sistema de manufacturación automatizado

El diseño del programa de control secuencial que gobierna un sistema de manufacturación es una tarea difícil que tradicionalmente se ha llevado a cabo de forma manual por ingenieros de control especializados. Esta elaboración manual del programa de control se ve afectada de los siguientes aspectos negativos:

- Es un proceso largo y complicado que consume mucho tiempo de ingeniería y que puede dar lugar a errores humanos en el programa de control.
- Este proceso se complica aun más por la necesidad de adaptarse a la evolución del ciclo de vida del propio sistema de manufacturación.

La aparición de nuevas técnicas de IA y la maduración de las técnicas ya existentes, esta permitiendo

el establecimiento de iniciativas conjuntas con el objetivo de desarrollar soluciones innovadoras para la automatización progresiva de las metodologías de construcción de los programas de control secuencial de forma que se puedan obtener de manera completamente autónoma por un computador, eliminando por completo cualquier intervención humana.

Aunque algunas de las aproximaciones se basan en técnicas generales basadas en el conocimiento [4, 5, 16], los aportes más importantes provienen de la utilización de técnicas de planificación de IA y, entre ellas, la disciplina mas extendida es la planificación de procesos.

B. La planificación de procesos

La automatización del proceso de manufacturación de partes discretas particularmente por procesos de mecanizado, como por ejemplo un tornillo o una tuerca, ha generado siempre un gran interés. Los sistemas de diseño asistido por computadora (CAD) permiten el modelado geométrico de componentes discretos mediante un programa de computador.

Los sistemas de manufacturación asistida (CAM) permiten utilizar la potencia de cómputo de un computador para secuenciar el proceso de manufacturación y su adecuación a las necesidades de producción. Los sistemas de planificación asistida de procesos (CAPP) se centran en la selección y definición de una secuencia de tareas de mecanizado necesarias para obtener una pieza final a partir de un determinado estado inicial [14].

Un problema de mecanizado consiste en la construcción de un programa de control secuencial, es decir, una secuencia de operaciones de mecanizado que permitan obtener una pieza final procesada a partir de una pieza inicial en bruto. Algunos problemas se pueden resolver analíticamente mediante algoritmos basados en satisfacción de restricciones. Sin embargo, la mayor parte de estos problemas requieren la modelación e inclusión del conocimiento especializado disponible en este dominio particular y un proceso de razonamiento adecuado a este conocimiento. Es en este punto donde entran en juego las técnicas de planificación de inteligencia artificial.

Una de las primeras aproximaciones a la aplicación de técnicas de planificación en problemas de mecanizado es GARI [9]. Es un sistema que utiliza conocimiento experto sobre las operaciones de transformación conocidas y genera, de forma automática, un plan de mecanizado.

Posteriormente a este trabajo inicial, se han sucedido muchas aproximaciones a la planificación de procesos desde distintos puntos de vista tales como SIPP [18] e IMACS [17], basados en técnicas de planificación jerárquica de tareas HTN. TOLTEC [30] es un planificador basado en casos que elabora planes de

mecanizado a partir de fragmentos de planes a distintos niveles de abstracción previamente almacenados en memoria.

Esta última parte ha mostrado como la aplicación de algunas de las técnicas de planificación de IA supone un impacto considerable en la resolución de problemas de mecanizado, en concreto en la obtención de planes que puedan utilizarse como los programas de control que gobiernen las máquinas que intervienen en él.

IV. CASO PRÁCTICO

Con el fin de evaluar de manera práctica la aplicación de las técnicas de planificación de IA para la generación automática de programas de control secuencial se desarrolló un prototipo de un sistema inteligente orientado a resolver un problema de mecanizado específico.

A. Características del sistema de mecanizado

El sistema de mecanizado consiste en una máquina compuesta de varios elementos interrelacionados cuyo funcionamiento coordinado permite la transformación de la materia prima en un producto manufacturado. Sus características básicas son:

- 1) **La máquina de mecanizado.** Consiste en un robot cartesiano el cual realiza su movimiento en las tres dimensiones y está compuesto por los siguientes elementos: *El Brazo*, cuerpo de la máquina que permite el desplazamiento en el espacio de trabajo, *la Mano*, elemento terminal del robot que permite el agarre de la herramienta, *el Rotor*, elemento que permite la rotación para realizar la tarea de hacer huecos.
- 2) **El conjunto de herramientas.** Para realizar los huecos en la materia prima se utilizan brocas de diferentes tamaños. Estas brocas están colocadas en una caja de herramienta con una localización determinada.
- 3) **El proceso de mecanizado.** Se lleva a cabo en una sola máquina regulada por un sistema de control secuencial, de forma que solo se puede hacer una acción a la vez. La máquina se dedica a la realización de huecos en una pieza proporcionada como materia prima convirtiéndola al final del proceso en un producto manufacturado de mayor valor.

B. La representación de un problema de mecanizado como un problema de planificación

Nuestro sistema permite producir el programa de control para una máquina de mecanizado en el dominio de hacer huecos sobre una materia prima. Su estructura básica está conformada por un sistema de planificación que dada una geometría de los huecos y unas condiciones iniciales y finales de los componentes de la máquina y las herramientas que ma-

neja, genera la secuencia de operaciones de los componentes de esta máquina.

Producir la secuencia de operaciones de mecanizado mediante las técnicas de planificación de inteligencia artificial implica la necesidad de utilizar un planificador que represente las características del sistema de mecanizado (componentes y proceso) para obtener un plan de acciones que sea el esqueleto del programa de control. Para esto seleccionamos el planificador STRIPS.

1) Representación del problema mediante estados y acciones

Dado que el planificador STRIPS utiliza una descripción del mundo basada en estados y acciones y enfoca la planificación como un problema de búsqueda en este espacio de estados, por lo cual es necesario representar nuestro problema de mecanizado bajo esta perspectiva.

El funcionamiento de cada uno de los componentes de la máquina de mecanizado, llamados también *actuadores*, se pueden describir mediante el uso del autómata de estados finitos [3, 9, 11] donde los estados del autómata describen todos los estados en que se puede encontrar el actuador y cada transición entre estados representa una acción del actuador (ver Fig. 5). Estos actuadores son las únicas entidades capaces de realizar transformaciones mediante sus acciones y las relaciones existentes entre ellos.

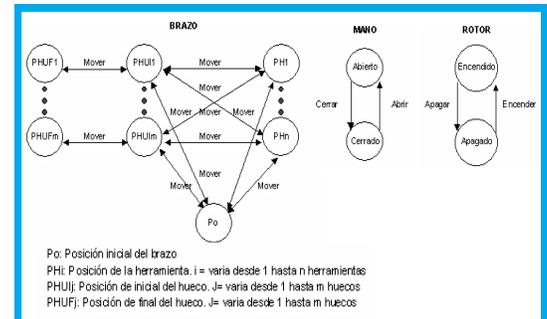


Fig 5. Estados y acciones de los componentes de la máquina

Cada herramienta puede ser vista como un elemento que cambia de estado como resultado de una acción del actuador *Mano* como se muestra en Fig. 6.

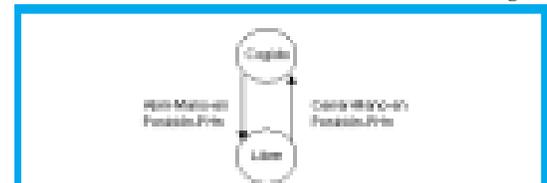


Fig. 6. Comportamiento de una herramienta

Las descripciones de un problema de mecanizado, para nuestro sistema, consisten en una serie de transformaciones que deben realizarse sobre la materia prima para poder obtener el producto manufacturado. Estos procesos de transformación se representan mediante la secuencia de acciones a seguir por

cada uno de los componentes de la máquina que permita la realización de los diferentes huecos que deberá tener el producto manufacturado. Esto es representado mediante *el plan final* que resulta al aplicar el planificador STRIPS para el caso específico del problema.

2) Representación del problema en el STRIPS

La representación de un problema de mecanizado como un problema de planificación en el STRIPS se manejan como la unión de tres componentes: el estado inicial, el estado final y el modelo de acción.

El estado inicial es una conjunción de literales que representa un conjunto de hechos que son ciertos, inicialmente, en el sistema de mecanizado. Estos hechos hacen referencia al estado en que se encuentran los actuadores y al estado, características y situación en que se encuentra la materia prima y las herramientas al inicio del problema. En Fig. 7 se describe las posibles situaciones en que puede estar el estado inicial para nuestro problema

Mano	Herramienta X	Materia prima ²	Brazo
Estado=abierto ó cerrado	Estado=Libre	Vacia	Posición= (0,0,0)

Fig. 7. Descripción del estado inicial del problema

El estado final de nuestro problema es una conjunción de literales que representa el conjunto de hechos que son ciertos luego de elaborar el producto manufacturado a partir de la materia prima presentada en el estado inicial mediante las acciones de los actuadores. En la Fig. 8 se describe las condiciones de nuestro estado final para el proceso de una materia prima.

Brazo	Mano	Herramienta	Materia prima
Posición= (0,0,0)	Estado=Abierto ó cerrado	Estado=Libre	Hueco1... Hueco m

Fig. 8 Descripción del estado inicial del problema

El modelo de acción representa la descripción del conjunto operaciones de los actuadores en los diferentes estados del sistema de mecanizado que se utilizan para resolver el problema de manufacturación. Para el STRIPS cada acción se describe a partir de los tres componentes: lista de precondición, lista de adición y lista de supresión. A continuación, se describe el modelo de acción.

Modelo de Acción para el problema de Mecanizado

Actuador: Mano
Regla: Cerrar-mano
Lista Precondición: Mano-Estado =Abierto, Brazo- Posición = Herramienta(x)- Posición en caja de Herramienta, Herramienta(x)-Estado = Libre
Lista Adición: Mano-Estado = Cerrado, Herramienta(x)-Estado = Cogido
Lista supresión: Mano-Estado = Abierto, Herramienta(x)-Estado = libre.
Actuador: Mano
Regla: Abrir-mano
Lista Precondición: Mano-Estado =Cerrado, Brazo- Posición = Herramienta(x)-

Posición en caja de Herramienta, Herramienta(x)-Estado = Libre
Lista Adición: Mano-Estado = Abierto, Herramienta(x)-Estado = Libre
Lista supresión: Mano-Estado = Cerrado, Herramienta(x)-Estado = Cogido
Actuador: Brazo
Regla: Mover Brazo (P1, P2)
Lista Precondición: P1<>P2, Brazo-Posición = P1
Lista Adición: Brazo-Posición = P2
Lista supresión: Brazo-Posición = P1
Actuador: Brazo
Regla: Hacer Hueco
Lista Precondición: Brazo-Posición = Hueco(Y)- Posición Inicial, Herramienta (x) -Estado = Cogido
Lista Adición: Hueco (y)

En la descripción del problema con el STRIPS, un hueco se representará como la secuencia de movimientos de los componentes de la máquina (ver Fig. 9). Por lo tanto en nuestro sistema, un hueco es la acción de incluir en la lista del plan de la máquina la secuencia indicada anteriormente. Dado que las acciones del rotor están incluidas en la definición del hueco no se incluirá este actuador en la descripción del problema.

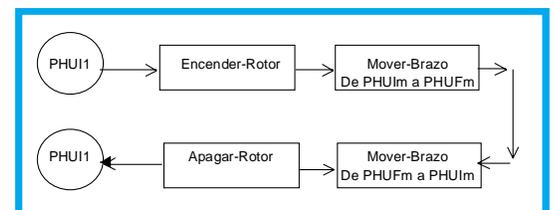


Fig. 9. Un hueco como secuencia de acciones de los actores de la máquina

V. DISEÑO DEL SISTEMA INTELIGENTE

El diseño del sistema inteligente se basa en las técnicas de orientación a objetos tomando como punto de entrada los planteamientos del problema expuestos en la sección anterior.

A. Arquitectura

El sistema está compuesto por un único módulo que se compone de los elementos interfaz de usuario, el sistema de planificación y el mecanismo de almacenamiento de la secuencia de control y la especificación del problema (ver Fig. 10).

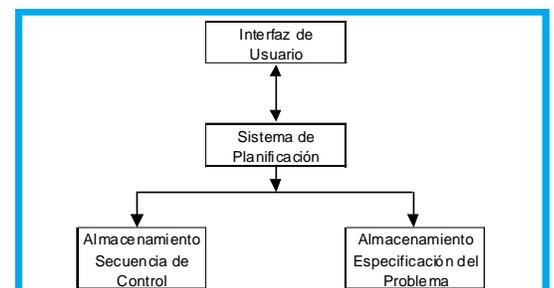


Fig. 10 Arquitectura del sistema

La *interfaz de usuario* permite la interacción del sistema con el usuario para:

- Organizar la entrada de los datos del problema de acuerdo a la forma de representar un problema el STRIPS.
- Mostrar la salida de los datos del problema mostrando la secuencia de control que deberá seguir el sistema de mecanizado

El *sistema de planificación* contiene el modelo lógico del sistema de manufacturación basado en el STRIPS y permite crear la secuencia de control para el proceso de mecanizado.

El sistema posee un *mecanismo de almacenamiento* en forma de archivos para guardar la especificación del problema y la secuencia de control.

B. Definición de las clases del sistema

El sistema inteligente está compuesto básicamente por las siguientes clases:

- 1) La clase problema: Provee la descripción básica de un problema de planificación que puede plantearse al STRIPS. En esta se describe: el estado inicial del problema (atributo estadoProblema), el estado final (atributo objetivos) e implementa el algoritmo STRIPS mediante los diferentes métodos que posee.
- 2) La clase regla: Las reglas que permiten al planificador construir un plan son instancias de esta clase. Describe las siguientes características de una regla: Nombre de la acción (atributo acción), la lista de precondiciones (atributo precondiciones), la lista de adición (atributo listaAdicionar) y la lista de supresión (atributo listaBorrar).
- 3) La clase DatosAdicProgAutoControl: Permite adicionar los datos necesarios para conformar la secuencia de control. En nuestro caso se emplea para generar la secuencia de acciones que representan un hueco.

Implementa tres métodos: *getPuntoFinal()* para devolver el punto final donde termina la perforación del brazo mecánico, *getPuntoInicial()* para devolver el punto inicial donde inicia la perforación del brazo mecánico y *toString()* que devuelve una cadena específica con base en sus atributos describiendo las acciones que describe un brazo mecánico para realizar un hueco.

- 4) La clase SubProbProgramasControl: Clase que sobrecarga el mensaje de la regla utilizada por la clase problema para que haga uso de los datos adicionales asociados a los objetivos. Permite personalizar los mensajes que saldrán cada vez que se ejecute una regla. Por defecto se imprime el nombre de la regla y en casos especiales los datos adicionales descritos en la clase DatosAdicProgAutoControl.

- 5) La clase GeneracionProgramasIU: Permite especificar un problema en el dominio de hacer huecos para una materia prima: Permite al usuario: especificar el estado inicial del problema, los objetivos que se pretenden lograr y genera las reglas específicas del problema. La interfaz hace uso del planificador STRIPS para encontrar un plan de acciones secuenciales para lograr los objetivos propuestos.

VI. IMPLEMENTACIÓN

La implementación de nuestro prototipo de sistema inteligente fue hecha inicialmente en CommonLisp y al final se implementó en java utilizando en ambos los fundamentos de programación orientada a objetos.

El desarrollo del programa en CommonLisp se realizó utilizando el entorno CLOS (Common Lisp Object System) que viene incluido en la versión 6.2 de este lenguaje. Aunque este ambiente de programación está basado en listas presenta una adecuada forma de trabajar la orientación a objetos y esta versión presenta un módulo para crear una adecuada interfaz visual.

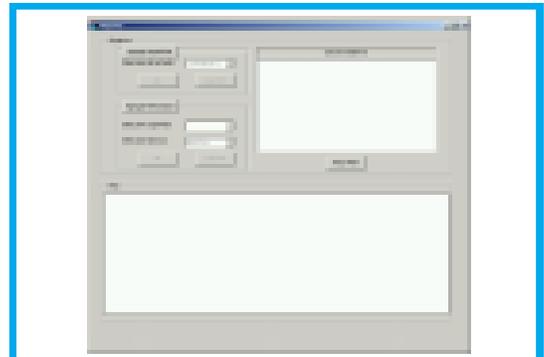


Fig. 11. Interfaz del sistema en el ambiente CommonLisp

El desarrollo del programa en java se realizó utilizando la versión 1.4.1. Dada su fortaleza para el desarrollo de programas orientados a objetos su implementación fue mucho más sencilla que la hecha en el ambiente de programación anterior (Fig. 12).



Fig.12. Interfaz del sistema en el ambiente JAVA

VII. CONCLUSIONES Y RECOMENDACIONES

Este artículo ha presentado una aproximación del estado del arte en la aplicación de las técnicas de planificación de inteligencia artificial al diseño automático de los programas de control para sistemas de manufactura, específicamente para problemas de mecanizado. La necesidad de buscar nuevas técnicas que permitan que el diseño de programas de control sea libre de errores, rápido y de bajo costo, ha motivado la aplicación de las técnicas de planificación de inteligencia artificial en este campo. Se ha mostrado que el dominio de los sistemas de manufacturación tiene algunas propiedades básicas las cuales deben ser tenidas en cuenta por los sistemas de planificación para asegurar el correcto razonamiento sobre las acciones que se realizaran en este dominio, y lograr un adecuado nivel de detalle del plan generado. En esta dirección los esfuerzos se han orientado en adecuar la expresividad de los planificadores ya existentes y aplicar los diferentes esquemas de planificación que permitan tratar con estas características.

La construcción del sistema inteligente mostrado en este trabajo, nos ha permitido verificar que las técnicas de planificación ofrecen un medio natural para la generación automática de programas de control para sistemas de mecanizado. Aunque el sistema aún es un prototipo, se encontró que utilizar el planificador STRIPS ayudó a la estructuración del problema mediante su mecanismo de representación y a la generación automática de la secuencia de operaciones mediante su mecanismo de búsqueda para obtener el plan. Como trabajos futuros, dado que el sistema inteligente presentado en este trabajo es un prototipo, como tal presenta algunos problemas que se deben mejorar para su correcta aplicación:

- Dada la naturaleza del STRIPS que implementa la planificación lineal, el sistema no asegura la solución mas adecuada (mínimos pasos de la máquina) por lo cual se deberá incluir la aplicación de heurísticas que mejoren su funcionamiento en el dominio específico.
- La expresividad del STRIPS es muy limitada para representar problemas complejos de manufacturación. Su modelo de acción por si mismo tiene dificultades para recoger un representación por intervalos de la ejecución de una acción y por tanto, para garantizar la ejecución correcta de dicho intervalo, condición importante en los sistemas de manufacturación. Esto conlleva a estudiar otras técnicas de planificación que permitan mayor expresividad al sistema si se quiere extender su dominio de aplicación.

REFERENCIAS BIBLIOGRÁFICAS

- [1] Allen J. F. Hendler, J. y Tate, A. Readings in Planning, Morgan Kaufmann, 1990.
- [2] Cai, Y., Martins-Shih, H. y Sekiguchi, T., "Acquisition of knowledge in sequential control systems", en IEEE Intern. symp. on industrial

electronic, págs. 300-303, 1992.

- [3] Cai, Y. y Sekiguchi, T., "Petri-net-based representation and acquisition of knowledge in sequential control systems", en IEEE conference on industrial electronics, págs. 1293-1297, 1990.
- [4] Chapman, D., "Planning for conjunctive goals", Artificial Intelligence, tomo 32, págs. 333-377, 1987.
- [5] Comission, E., "Intelligent Manufacturing Systems. Guide for Proposers", Http://www.cordis.lu/esprit/src/imsguide.htm.
- [6] Crouette, D., Methodologie de conception des systèmes complexes a evenements discrets: application a la conception et la validation hierarchisée de la commande de cellules flexibles de production dans l'industrie manufacturière, Tesis Doctoral, Ecole Centrale de Lille, 1991.
- [7] Descotte, Y. y Latombe, J., "GARI: A problem solver that plans how to machine mechanical parts", en IJCAI 1981, págs. 766-772, 1981.
- [8] Fikes, R. E. y Nilsson, N. J., "STRIPS: A new approach to the application of theorem proving to problem solving", Artificial Intelligence, tomo 2, págs. 189-208, 1971.
- [9] Fisher, T., Batch control systems: Design, application and implementation, Instrument Society of America (ISA), 1990.
- [10] Green, C., "Application of theorem proving to problem solving", en IJCAI, págs. 741-747, 1969.
- [11] Gruver, W. A. y Boudreaux, J. C., Intelligent manufacturing: programming environments for CIM, Springer-Verlag, London, 1993.
- [12] Instrument Society of America (ISA), Batch control Part 1: models and terminology (SP88), 1995.
- [13] Kambhampati, S., "A validation-structure-based theory of plan modification and reuse", Artificial Intelligence, tomo 55, págs. 193-258, 1992.
- [14] Manna, Z. y Waldinger, R. J., "Knowledge and reasoning in program synthesis" Artificial Intelligence, tomo 6, págs. 175-208, 1975.
- [15] McDermott, D. y Hendler, J., "Planning: what is, what it could be, an introduction to the special issue on planning and scheduling", Artificial Intelligence, tomo 76, págs. 1-16, 1995.
- [16] Mizutani, H., Nakayama, Y., Ito, S., Namioka, Y. y Matsudaira, T., "Automatic programming for sequence control", en Innovative applications of artificial intelligence, A. C. Scoot y P. Kahr, eds., págs. 315-331, AAAI Press, 1991.
- [17] Nau, D., Gupta, S. K. y Regli, W. C., "AI planning versus manufacturing-operation planning: A case study", en IJCAI95, págs. 1670-1676, 1995.
- [18] Nau, D. S., "Hierarchical representation of problem solving knowledge in a frame-based process planning system", International Journal of Intelligent Systems, tomo 1, págs. 29-44, 1986.
- [19] Newell, A. y Simon, H. A., "GPS: a program that simulates human thought", en Computers and thought, E. A. Feigenbaum y J. Feldman, eds., R. Oldenburg KG, 1963.
- [20] Nilsson, N. J., Principles of Artificial Intelligence, Tioga Publishing, 1980.
- [21] Norvig, P., Paradigms of Artificial Intelligence programming: Case studies in Common LISP, Morgan Kaufmann, 1992.
- [22] Pearl, J., HEURISTICS: Intelligent search strategies for computer problem solving, AddisonWesley, 1984.
- [23] Penberthy, J. S. y Weld, D. S., "UCPOP: A sound, complete, Partial Order Planner for ADL", en 3rd. Int. Conf. on Principles of Knowledge Representation and Reasoning, págs. 103-114, 1992.
- [24] Rich, E. y Knight, K., Inteligencia Artificial, McGraw-Hill, 1994.
- [25] Ruseell, S. y Norvig, P., Inteligencia Artificial: Un enfoque moderno, Prentice Hall Iberoamericana, 1996.
- [26] Sacerdoti, D. E., "Planning in a hierarchy of abstraction spaces", Artificial Intelligence, tomo 5, págs. 115-135, 1974.
- [27] Sacerdoti, E. D., "The nonlinear nature of plans", en IJCAI 1975, págs. 206-214, 1975.
- [28] Tate, A., "Generating project networks", en IJCAI 1977, págs. 888-893, 1977.
- [29] Tate, A., Drabble, B. y Kirby, R., "O-PLAN2: An open architecture for command, planning and control", en Intelligent scheduling, M. Zweben y M. Fox, eds., Morgan Kaufmann, 1994.
- [30] Tsatsoulis, C. y Kashyap, R. L., "Casebased reasoning and learning in manufacturing with the TOLTEC planner", IEEE Transactions on Systems, Man a Cybernetics, tomo 23, págs. 1010-1023, 1993.
- [31] Veloso, M., Carbonell, J., P#rez, A., Borrajo, D., Fink, E. y Blythe, J., "Integrating planning and learning: The PRODIGY architecture", Journal of Experimental and Theoretical Artificial intelligence, tomo 1, n o 7, 1995.
- [32] Wilkins, D. E., "Can AI planners solve practical problems?", Computational intelligence, tomo 6, págs. 232-246, 1990.

Jaime Alberto Guzmán Luna

Ingeniero Civil, Magister en Ingeniería Civil, Universidad Nacional. Fue docente de la Facultad de Ingeniería de la Universidad Distrital Francisco José de Caldas, actualmente es profesor de la Universidad Nacional de Colombia, Sede Medellín.