



Universidad Politécnica de Valencia
Departamento de Ingeniería de Sistemas y Automática

**Planificación de Robots
mediante
Descomposición, Abstracción,
Deducción y Predicción**

Tesis Doctoral

Eduardo Vendrell Vidal
Autor

Alfons Crespo i Lorente
Martin Mellado Arteché
Directores

Noviembre, 2000

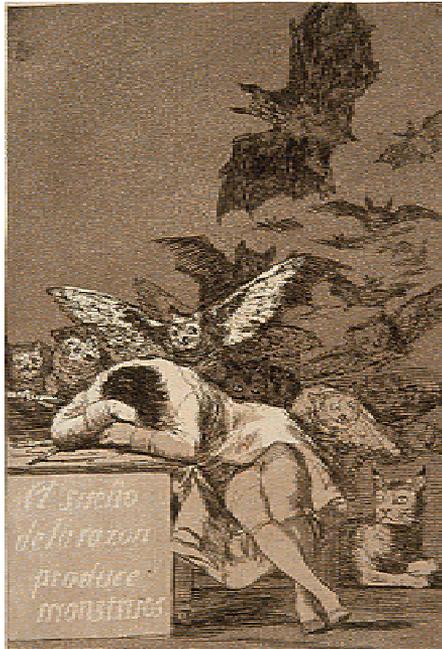
*a la memòria
de ma mare*

Agradecimientos

Sería poco honroso dejar pasar la ocasión de agradecer encarecidamente todo el apoyo prestado a lo largo y ancho del tiempo en que he estado trabajando en esta tesis doctoral a mis dos directores, Alfons Crespo y Martín Mellado. En especial a este último, mi punto de referencia en todo el tiempo que llevo trabajando en la universidad, que ya empieza a no ser poco.

También debo expresar mi agradecimiento a mis compañeros de departamento, y en especial a Josep Tornero, Pedro Albertos, Ángel Valera y Antonio Sánchez por la paciencia que han tenido conmigo, unos al acogirme en el inicio de mis labores investigadoras, otros simplemente por soportar mis neuras. A todos, muchas gracias.

Mi familia, y en especial María José, siempre me ha alentado para culminar este trabajo. Pues bueno, ya está. Muchas gracias a todos.



Índice

1	Introducción.....	1
	PREÁMBULO	3
	PLANTEAMIENTO DE LA TESIS DOCTORAL	4
	MOTIVACIÓN Y OBJETIVOS	6
	PRESENTACIÓN DE LA TESIS	8
2	Planificación, replanificación y lógica	
	temporal	11
	ESTADOS, EVENTOS Y ACCIONES	15
	MÉTODOS DE PLANIFICACIÓN	17
	CÁLCULO DE SITUACIONES	19
	REPRESENTACIÓN EN EL ESPACIO DE ESTADOS	20
	Búsquedas en el Espacio de Estados. Búsqueda en Grafos	20
	Redes de Petri	23
	Análisis <i>Means-End</i>	24
	La Representación STRIPS	25
	LÓGICA DE PREDICADOS	27
	SISTEMAS DE PRODUCCIÓN	29
	LÓGICA TEMPORAL	31
	PLANIFICACIÓN EN ROBÓTICA.....	40
	Planificación de Movimientos.....	41
	Planificación Dinámica.....	42
	Planificación de Ensamblajes.....	43
	Planificación mediante Sensorización	44
	Arquitecturas de Planificación	45
	Replanificación	47
	PLANTEAMIENTO DEL MÉTODO DE PLANIFICACIÓN	48
	Descomposición del Problema	51
	Abstracción Jerárquica	52
	Observación del Entorno (Deducción)	52
	Estimación Temporal (Predicción)	52
	CONCLUSIONES.....	53

3	Planificación mediante descomposición y abstracción	55
	DESCOMPOSICIÓN DEL PROBLEMA.....	59
	DESCOMPOSICIÓN EN PARTES	61
	BÚSQUEDA Y SELECCIÓN DE LA SOLUCIÓN EN CADA PARTE	62
	Resolución de Conflictos entre Soluciones	63
	Combinación de Soluciones.....	65
	DESCOMPOSICIÓN EN ROBÓTICA	71
	Criterios para la Agrupación de Estados y Acciones	71
	Factores para la Descomposición en Partes	73
	ABSTRACCIÓN JERÁRQUICA.....	80
	ABSTRACCIÓN EN ROBÓTICA	83
	Misiones.....	86
	Tareas.....	88
	Acciones	93
	Movimientos.....	93
	Trayectorias	95
	Replanificación en la Abstracción Jerárquica	96
	COMBINACIÓN DE LA DESCOMPOSICIÓN EN PARTES Y DE LA ABSTRACCIÓN EN NIVELES.....	99
	EJEMPLO DE APLICACIÓN A UN ROBOT ARTICULADO....	100
	EJEMPLO DE APLICACIÓN A UN ROBOT MÓVIL.....	107
	EL PROYECTO RETRARO.....	111
	CONCLUSIONES	123
4	Replanificación: El proceso deductivo	125
	EL PROCESO DE REPLANIFICACIÓN	129
	Clasificación de los Agentes.....	131
	Las Trayectorias de los Móviles.....	133
	EL PROCESO DEDUCTIVO.....	139
	Observación del Entorno	141
	Deducción de las Trayectorias.....	142
	Reconocimiento de Nuevas Trayectorias.....	146
	Mantenimiento del Histórico de Trayectorias.....	152
	CONCLUSIONES	153
5	Replanificación: El proceso predictivo.....	155
	EL PROCESO PREDICTIVO	159
	Predicción de Interacciones.....	162
	Replanificación por Ajuste de Velocidades.....	165
	ESTUDIO DE LOS CASOS	166
	El Caso del Punto Compartido	166

El Caso del Segmento Compartido en el mismo Sentido.....	175
El Caso del Segmento Compartido en Sentido Contrario.....	190
REPLANIFICACIÓN DE MÚLTIPLES MÓVILES	198
El Planteamiento Básico.....	198
Un Sistema de Control para cada Móvil	199
Un Sistema de Control Centralizado para todos los Móviles	
Controlados	200
Alternativas y Consideraciones	211
CONCLUSIONES.....	216
6 Arquitectura software	219
ARQUITECTURAS SOFTWARE DE CONTROL.....	223
ARQUITECTURA PROPUESTA.....	225
Planificador de Misiones	227
Planificador de Tareas	228
Planificador de Movimientos	229
Planificador de Trayectorias.....	229
Monitor del Sistema.....	231
Deductivo+Predictivo	231
LA ARQUITECTURA RETRARO	233
CONCLUSIONES.....	239
7 Conclusiones	241
CONTRIBUCIONES	243
Planificación	244
Replanificación	245
TRABAJOS FUTUROS	246
Bibliografía y referencias.....	249

Índice

1 *Introducción*

Introducción

PREÁMBULO

En la resolución de todo problema bien planteado tiene una especial importancia la *planificación*. Cuando aparece la necesidad de llevar a cabo una misión o tarea, ésta se resuelve teniendo en cuenta una serie de objetivos parciales planteados en el inicio de la misma. Adicionalmente, el propio desarrollo del problema conlleva la aparición de nuevos objetivos que deben ir resolviéndose conforme avanzan los pasos de resolución de dicha tarea.

Pero, ¿qué significa la planificación?. En general, la planificación tiene que ver con el hecho de *deducir las acciones necesarias para cumplir unos objetivos*. Evidentemente, dependiendo del objetivo a cumplirse, la planificación tiene un significado u otro. Es más, dependiendo del nivel en el que se aplica la planificación, ésta tiene un significado u otro.

Si nos centramos en el campo de la robótica, los objetivos que deben cumplirse tienen que ver con la deducción de las acciones necesarias, desde el punto de vista del sistema robotizado en cuestión, para que éste lleve a cabo una misión encomendada. Esta planificación se puede considerar a distintos niveles. Así, a alto nivel, hay ciertas acciones que se pueden deducir independientemente de las características que tenga el robot que debe cumplir la tarea, mientras que las acciones que directamente significan órdenes para el sistema robotizado, suponen un nivel más bajo de planificación. Pero tanto en un caso como en otro, la metodología empleada para realizar la planificación puede ser la misma, aún considerando aspectos distintos en cada nivel.

En la vida cotidiana, al realizar las tareas más comunes, no hay un planteamiento inicial de una planificación por niveles. Esto es porque las acciones cotidianas quedan almacenadas en nuestra mente de una forma clara, aunque tenemos la información necesaria para planificarlas y esta planificación o ya se hizo en su momento o se hace de forma inconsciente. Es más, la conciencia de que se realiza una planificación sólo aflora en los casos en los que para la realización de una tarea común se presenta un imprevisto y entonces es cuando se recapacita y se busca una solución alternativa.

La resolución de tareas robotizadas se puede plantear de igual manera. Cuando se aborda por primera vez la resolución de una tarea, se debe planificar y resolver atendiendo a una serie de aspectos, acudiendo a la información que se tiene (del propio sistema

robotizado, del entorno, de la descripción de la tarea a realizar...). Posteriormente, esta tarea queda completamente descrita, tanto sus objetivos como su resolución, con lo que la planificación no vuelve a ser necesaria hasta que cambia la información asociada (por ejemplo, que la sensorización del sistema robotizado aporte datos que difieran de los disponibles y conocidos). Es en ese momento cuando se aborda una replanificación en los niveles en los que la información requerida para ello haya cambiado.

Así, la información sensorial del sistema robotizado no sólo nos debe permitir contrastar los diferentes estados que se alcanzan durante la resolución/ejecución del problema, validando los pasos de ejecución, sino que permite asimismo incorporar nuevos estados. Esta incorporación de nuevos estados hace que la información asociada cambie y por tanto que se plantee replanificar las acciones que debe realizar el sistema robotizado.

La sensorización nos va a permitir realizar una tarea muy importante en situaciones en las que aparecen distintos robots en un mismo entorno realizando tareas que pueden ser comunes o independientes unas de otras. Estas situaciones son especialmente conflictivas en lo que respecta al riesgo de interacción no deseada entre los distintos robots. La sensorización nos permitirá entonces extraer información del entorno que ayude a plantear la planificación de las tareas para cada robot. Es lo que llamamos una observación del entorno.

De igual forma que el ser humano realiza una observación del entorno y deduce acciones en un futuro inmediato, este planteamiento es perfectamente válido para el contexto de la robótica. Así, ayudándonos de la observación, se acude a la extracción de datos del entorno que permita plantear las acciones inmediatas de acuerdo a una planificación temporal.

PLANTEAMIENTO DE LA TESIS DOCTORAL

En la tesis doctoral que aquí se presenta se aborda la planificación y replanificación de las actividades que lleva a cabo un robot en un entorno dinámico. De un lado, se plantea una metodología de planificación que considera una descomposición del problema en partes y niveles de abstracción distintos. Las situaciones nuevas se contemplan a partir de la observación del entorno, susceptibles de ser incorporadas a la información necesaria para llevar a cabo la planificación. Por otro lado, la replanificación se plantea a partir de un razonamiento temporal sobre las trayectorias que siguen los

robots a lo largo de la ejecución de sus actividades. Este razonamiento temporal permite, a partir de la confirmación de acciones en el tiempo, exponer una proyección de futuro en lo que respecta a los movimientos a seguir por los robots del entorno. Las interacciones previstas se evitan a partir de un ajuste de las velocidades de los robots que se controlan.

Se trata de un tema abordado con cierta frecuencia en robótica. La mayoría de planificadores aplican métodos e ideas aportadas del campo de la inteligencia artificial, considerándose en la formulación típica de un problema relacionado con robots la existencia de un robot equipado con sensores y un conjunto de acciones básicas que pueden ser llevadas a cabo en un mundo sencillo de entender. Con lo que una acción de robot significa modificar un estado, o la configuración del mundo [FU *et al.*, 88].

Muchos de las propuestas de planificación existentes adolecen de una formalización y se plantean, en muchos casos, dependientes de los objetivos a cumplirse.

Por su parte, el planteamiento de esta tesis doctoral implica la definición de una serie de conceptos relacionados directamente con el desarrollo de la misma. Así, el establecimiento de una jerarquía de niveles está directamente relacionada con los conceptos de misión, tarea o acción, descritos a partir de la dependencia de la actividad del contexto y del agente que la desempeña. Esta división en niveles permite abordar una planificación teniendo en cuenta para cada nivel aspectos distintos, que en muchas ocasiones no variarán aunque varíe la información que se tiene en cuenta a niveles distintos.

Otro de los puntos importantes al establecer la metodología de planificación es la consideración de la información necesaria para cada nivel de descomposición. Datos relativos al entorno, características del sistema robotizado y características propias de la tarea a realizarse conforman de forma general la información a considerarse para poder abordar la planificación a partir de una descomposición en partes.

En lo que respecta a la replanificación de las actividades del robot que se controla, dos son los procesos que se plantean: deducción y predicción. Las trayectorias que siguen otros móviles en el entorno se descubren a partir de un proceso de deducción, basado en la observación del entorno a partir de un sistema de sensorización adecuado. La obtención de información del paso de los móviles por puntos del entorno, combinado con la existencia de una base de datos de trayectorias conocidas es la base del proceso deductivo.

Por último, un proceso predictivo, basado en la estimación de las acciones que los robots deben realizar para llevar a cabo las tareas encomendadas permite replanificar las trayectorias a seguir por el robot controlado. El ajuste de velocidades a través de unas reglas de comportamiento basadas en la estimación temporal es el método elegido para replanificar las trayectorias.

MOTIVACIÓN Y OBJETIVOS

La motivación para el desarrollo y consecución final de esta tesis doctoral parte fundamentalmente de dos precedentes: la actualidad de la temática, y la pertenencia del autor a un grupo de investigación cuyas actividades están estrechamente relacionadas con esta temática.

La robótica, y en especial la robótica móvil, es una de las fuentes de investigación más actuales, con avances que tienen un impacto a corto plazo los sistemas de producción y, a más largo plazo, sobre la vida cotidiana. En especial, la planificación de robots atiende a uno de los problemas fundamentales, esto es, la consecución de los objetivos marcados de una manera eficiente. Para ello, existen distintas técnicas que en la actualidad son motivo de investigación relacionadas con esta temática. Muchos de los avances se realizan en campos de investigación colaterales, como por ejemplo la Inteligencia Artificial, pero con resultados aplicables en la robótica. Es de esta manera como el razonamiento temporal deviene en un método de especial utilidad para la replanificación de actividades de robot.

El autor de la presente tesis doctoral pertenece a la línea de investigación en **Tele-Robótica**, inmersa dentro del grupo de investigación en **Sistemas Robotizados** del Departamento de Ingeniería de Sistemas y Automática de la Universidad Politécnica de Valencia. En el seno de este grupo de investigación y en colaboración con otros departamentos, el autor ha venido trabajando en distintos proyectos de investigación nacionales e internacionales relacionados con la planificación y programación de robots.

En concreto, la participación en el proyecto de investigación ESPRIT “Use of the Expert System Platform REAKT for Transport Robot Guidance (RETRARO)”¹ supuso la espoleta que desencadenó el desarrollo de este trabajo de investigación. El objetivo principal del

¹ ESPRIT IV Project N° 20.788 “Use of the Expert System Platform REAKT for Transport Robot Guidance (RETRARO)”. 1995 a 1997.

proyecto era el de controlar las actividades de un robot móvil autónomo en un entorno industrial real. Para ello, se aplicó el sistema experto en tiempo real REAKT [Mensch *et al.*, 1993] y se diseñó una arquitectura software modular para la planificación y control de las actividades del móvil. La arquitectura propuesta mostraba una primera estructura en distintos niveles (misiones, tareas, trayectorias) y consideraba la descomposición de las actividades del móvil autónomo según la zona de la factoría que recorría en cada momento. Estos principios supusieron el germen de la metodología de planificación que se aquí se presenta.

Con todo, los objetivos marcados en el desarrollo del trabajo de investigación se esbozan a continuación:

- *Revisión de las metodologías de planificación existentes aplicadas a la robótica.*
- *Aplicación de las intuiciones humanas como base para establecer la presente metodología de planificación de robots.*
- *Desarrollo de un método de planificación de robots basado en los conceptos de descomposición y abstracción jerárquica.*
- *Establecimiento de los criterios y factores que permiten decidir una correcta descomposición de un problema en robótica.*
- *Generación de un índice de bondad para la estimación de una correcta descomposición en partes del problema.*
- *Establecimiento de unos niveles de abstracción en robótica a partir de la relación del problema con el robot y el entorno.*
- *Definición de un método de replanificación a bajo nivel (trayectorias) basado en los conceptos de deducción y predicción.*
- *Desarrollo de una estrategia de reconocimiento de trayectorias de móviles mediante su deducción.*
- *Aplicación del razonamiento temporal en el futuro como base del método predictivo.*
- *Definición de los algoritmos y reglas de la replanificación de trayectorias.*
- *Definición de una arquitectura software que plasme la metodología propuesta.*
- *Utilización de la metodología de planificación en un proyecto de investigación con aplicación industrial.*

PRESENTACIÓN DE LA TESIS

La memoria de la tesis doctoral que aquí se presenta se estructura en siete capítulos, de los que el primero supone esta introducción.

El segundo capítulo presenta un estado del arte relativo a los métodos de planificación para la resolución de problemas en general y en el campo de la robótica en particular. Así, se analizan métodos de planificación existentes, basados en gran parte en ideas provenientes del área de la inteligencia artificial, considerando la estructura de datos que plantean (información necesaria para abordar la planificación). Se aborda también el problema de la replanificación de las acciones, analizándose las posibilidades de replanificación de las actividades de un robot. Este capítulo termina con el planteamiento de la metodología tema de esta tesis, contrastándola con los métodos analizados con anterioridad.

El proceso de planificación se trata en el tercer capítulo. En él se describen las dos intuiciones humanas en que se basan la descomposición en partes del problema a resolver y la abstracción en niveles de las partes obtenidas. Se abordan ambos puntos en general, particularizándose su tratamiento en el ámbito de la robótica. Para ello, se definen una serie de factores y parámetros que permiten decidir y seleccionar una correcta descomposición en partes y abstracción jerárquica del problema. El capítulo se completa con dos ejemplos de aplicación sobre robots articulados y móviles, y con la aplicación de la génesis del proceso de planificación en un proyecto de investigación, como principal aportación.

El cuarto capítulo aborda el proceso deductivo a partir de la observación del entorno, desarrollo que permite el reconocimiento de las trayectorias que siguen otros móviles del entorno. Así, se presenta la estructura básica del proceso a partir de un algoritmo básico de deducción de las trayectorias de los móviles como secuencias de puntos y tiempos de paso. El capítulo incluye una clasificación de los móviles y la formalización de las trayectorias de los mismos.

El proceso predictivo a través de la estimación temporal centra el quinto capítulo de esta tesis doctoral. Una vez definidos todos los conceptos y estructurados los datos necesarios, de acuerdo con el procedimiento deductivo, las trayectorias de los móviles se comparan y reajustan de cara a evitar posibles interacciones. Este capítulo aborda el estudio de diferentes casos tipo, extrayéndose los algoritmos básicos y las reglas que configuran el método.

El sexto capítulo plantea la definición de una arquitectura software que contempla todas las características del método propuesto. La definición de esta arquitectura software, con la descripción de todos los módulos que la componen y los enlaces y relaciones entre los mismos, suponen el objeto de este apartado.

El séptimo capítulo expone las conclusiones que se extraen de la metodología propuesta. En este capítulo se resumen las principales características y ventajas del mismo, así como las líneas de trabajo a seguir para completar y mejorar el método.

La tesis doctoral se cierra con un apéndice bibliográfico que incluye todas las referencias contenidas en el documento.



Introducción

2 *Planificación, replanificación y lógica temporal*

Planificación, replanificación y lógica temporal

El espíritu de este trabajo de investigación se centra en las posibilidades de planificación y replanificación de las actividades que realiza un robot industrial en un entorno de producción. Mientras que la planificación hace referencia a un proceso previo de organización y proyección en el tiempo de las distintas acciones que debe realizar el robot para conseguir un objetivo prefijado, la replanificación alude a la posibilidad de reorganizar en tiempo real un plan preestablecido, con el objetivo de optimizar su ejecución y evitar anomalías imprevistas.

Este capítulo hace repaso a definiciones y procedimientos de planificación y replanificación conocidos, haciendo especial énfasis en aquellos métodos relacionados con la robótica. Así mismo, se presenta la metodología de planificación y replanificación objeto de esta tesis doctoral, que será desarrollada en los capítulos posteriores.

Planificación, replanificación y lógica temporal

ESTADOS, EVENTOS Y ACCIONES

Planificar es generar un método para la resolución eficiente de un problema dado. Este método, el plan, es en general una secuencia de pasos que deben seguirse para la resolución del problema. El acto de planificar se puede aplicar a multitud de disciplinas y contextos. La planificación de problemas es abordada de manera frecuente en el ámbito de la Ingeniería Artificial [Allen *et al.*, 1990] [Hendler *et al.*, 1990] [Weld, 1998], aunque existen multitud de ejemplos en los que se puede aplicar un plan. Así, por ejemplo, en el contexto de un partido de fútbol, un plan será una estrategia diseñada, una jugada, con el objetivo de que un equipo dado consiga un gol. Los pases, regates y movimientos en general de los jugadores serán los pasos que se deben seguir para la ejecución del plan. Este ejemplo nos muestra otra de las características de los planes: la consecución de un objetivo. Todos los planes tienen como meta un objetivo, resolución de un problema dado.

En robótica, planificar significa obtener la mejor secuencia de pasos para completar una tarea de robot [Latombe, 1991] [Hwang y Ahuja, 1992]. Un paso dentro de esta secuencia sería cualquier acción del robot, tanto si esta acción supone un movimiento como si se trata de cualquier acción que no suponga movimiento, por ejemplo, la apertura de una pinza. Así, un plan (resultado del planificador) podría ser, por ejemplo, la secuencia adecuada de acciones que permita realizar la tarea de coger una pieza en el periodo más corto de tiempo posible, con un menor número de movimientos de robot y verificando las restricciones propias de la tarea (detección y evitación de colisiones en los movimientos del robot, establecer el agarre más adecuado según la pieza, etc.).

En el contexto de la robótica móvil, las actividades del robot tienen, si cabe, una mayor importancia. El hecho de que el robot pueda moverse, desplazarse en un entorno determinado, hace que aumente el número de factores que pueden afectar a sus actividades. Así, dentro del entorno de un robot móvil podremos encontrar otros agentes que desarrollan la misma tarea que el robot móvil (participando en fases distintas de la misma tarea y colaborando, por tanto, con el robot móvil), o incluso móviles no controlados (otros robots móviles desarrollando tareas distintas, operadores en la misma planta, etc.). Todos estos factores hacen más complicado el proceso de planificación.

De cara a representar los problemas a solucionar y, en definitiva, el propio mundo o entorno en el que los problemas tienen lugar y los distintos mecanismos o agentes los resuelven, resulta conveniente hablar de estados y eventos.

Se trata de una aproximación común en la mayoría de textos relacionados con la temática. En [Allen y Ferguson, 1994] se tiene una buena aproximación a estas definiciones, aunque otros textos como [Fu *et al.*, 1988] incluyen referencias a estas definiciones centradas en el campo de la robótica.

En primer lugar, es necesario definir el concepto de estado. Un estado es una configuración determinada del mundo en un instante o intervalo de tiempo dado. Cuando se tiene un problema determinado, este consistirá en pasar de un estado inicial a un estado final u objetivo.

Por otro lado, un evento tiene que ver con el concepto de cambio. Un evento describe el cambio que se produce en el mundo o entorno en el que participan uno o varios agentes. Mientras que el concepto de estado hace referencia a una situación que se mantiene durante un cierto intervalo de tiempo, el concepto de evento hace relación a un cambio que se produce en el entorno en que estamos.

El concepto de estado cumple la propiedad de *homogeneidad*, esto es, si un estado se mantiene a lo largo de un intervalo de tiempo t dado, también lo hará en un subintervalo de este intervalo temporal. Por su parte, los eventos no cumplen esta propiedad, sino que cumplen una propiedad inversa que podríamos denominar de *anti-homogeneidad*. Esto es, si un evento se produce durante un intervalo de tiempo t , en un subintervalo de este espacio temporal no se produce el evento en su totalidad. Esto es así precisamente porque el evento está relacionado con el concepto de cambio. Por ejemplo, consideremos el evento

poner el libro encima de la mesa

que se desarrolla durante el intervalo de tiempo que va desde t_1 a t_2 . En un subintervalo dado de este espacio de tiempo no se está desarrollando el evento en su totalidad. Se estará cogiendo el libro de la estantería, o cargando el libro, o dejándolo encima de la mesa, pero no se puede considerar el evento en su totalidad.

Esta propiedad nos lleva a la conclusión de que los eventos están constituidos por un conjunto de nuevos eventos que se desarrollan en subintervalos del espacio de tiempo en el que se da el evento global.

Un concepto relacionado con el de estado y evento es el de acción. Una acción es lo que potencialmente puede realizar un agente. Así, hablamos de las acciones que realiza o puede realizar un robot. De hecho, al ejecutar una acción, el agente produce un evento determinado, que llevan a nuevos eventos que se van produciendo.

Cuando se dice que un estado se mantiene a lo largo de un intervalo de tiempo o que un evento se produce en un intervalo de tiempo determinado, se están ligando estos términos al concepto de tiempo. Más adelante se trata el tema de la lógica temporal y se vuelve a incidir en los conceptos de estado y evento dentro de este contexto.

MÉTODOS DE PLANIFICACIÓN

La utilización de técnicas de Inteligencia Artificial para desarrollar planificadores es muy común. Existen diversos métodos basados en la existencia de un conjunto de estados que el móvil debe alcanzar a lo largo de la resolución de un problema determinado. Normalmente, este conjunto de estados se puede representar como un grafo cuyos nodos son los diferentes estados y los arcos son la forma de pasar de un estado a otro. De esta manera, existen distintas técnicas que permiten explorar un grafo y obtener la solución óptima en forma del camino más corto para alcanzar un estado objetivo a partir de un estado inicial. Estas técnicas, basadas en su mayoría en algoritmos de búsqueda en grafos o en la aplicación de reglas, consideran distintos costes para los arcos de los grafos, significando con ello parámetros como la distancia entre dos nodos consecutivos (distancia física real entre dos puntos del entorno que el móvil debe alcanzar a lo largo de su recorrido).

Cuando un plan se va generando de manera que se van añadiendo pasos conforme se avanza hacia la consecución de los objetivos prefijados, se dice que se construye un plan incrementalmente. Los métodos incrementales pueden generar planes añadiendo pasos hacia delante o hacia atrás. En el primer caso, el plan crece desde una situación inicial hacia la situación objetivo, mientras que en el segundo caso, se parte del objetivo para llegar a la situación que se considera inicial. En estos métodos pueden aparecer restricciones en la generación del plan a cada paso que se va añadiendo, haciendo más consistente el resultado final. Los planificadores que generan planes incrementalmente se denominan algoritmos de planificación básicos.

Uno de los recursos que se utiliza en la generación de planes es el de recurrir a intuiciones humanas [Newell y Simon, 1972]. Es decir,

recurrir a estrategias que siguen el conocimiento humano en la resolución habitual de problemas. De entre estas estrategias resaltan dos: divide y vencerás y la abstracción jerárquica [Yang, 1997].

Los planificadores basados en la estrategia del divide y vencerás se caracterizan principalmente por la capacidad de dividir el problema en distintos subproblemas que se abordan separadamente. Esta estrategia se denomina también la descomposición del problema. De esta manera, descomponiendo un problema mayor en problemas menores se puede abordar un proceso de resolución por separado más simple. Así, por ejemplo, si se representa el conjunto de estados mediante un grafo y se recurre a algoritmos de búsqueda en grafos, la subdivisión del problema permitirá analizar sólo parte del grafo haciendo la búsqueda más simple y rápida. Si esto se hace de forma paralela con las distintas partes fruto de la subdivisión, se conseguirá un incremento de la velocidad de resolución del problema.

La combinación de las distintas soluciones de cada subproblema nos dará la solución global. Pero esta combinación no es trivial, puesto que pueden existir partes redundantes que deben ser eliminadas para que la combinación de soluciones dé como resultado una solución eficiente y adecuada. Por otro lado, también se debe aportar la mejor solución de cada subproblema cuando estos presenten soluciones múltiples. Es decir, se debe realizar un proceso de selección de soluciones. Pero ésta no es una característica exclusiva de la descomposición del problema puesto que de abordar la resolución de forma global también se podrían tener diferentes soluciones al final del proceso, debiéndose recurrir a un proceso similar de selección de soluciones.

Cuando un planificador está basado en la abstracción jerárquica, éste se caracteriza por hacer distinción entre diferentes aspectos del problema para abordar la resolución teniendo en cuenta en cada caso estos aspectos. Así, se suelen considerar aspectos globales que permitan convertir el problema en una versión más reducida, para pasar con posterioridad a considerar aspectos más locales que lleven a un refinamiento de la solución. Recurriendo de nuevo al ejemplo de tener un conjunto de estados representados como un grafo, un planificador basado en la abstracción jerárquica abordaría en primer lugar la resolución de un grafo global con los estados más representativos del problema en cuestión. Este grafo global siempre será más reducido, con lo que la resolución de esta parte será más simple y rápida. En un proceso posterior es cuando se considerarán de forma local todos los estados posibles, aumentando el grafo y consiguiendo con su resolución un mayor refinamiento en la solución global.

El problema de planificación puede ser abordado mediante distintas aproximaciones que permiten, según las características de cada una, una buena representación del problema, análisis de posibilidades, proyección en el futuro de las mismas o la simulación de resultados a obtener. Entre las aproximaciones que se encuentran en la bibliografía destacan:

- **El Cálculo de Situaciones**, centrado en la representación del mundo.
- **La Representación en el Espacio de Estados**, una técnica centrada también en representar el mundo a partir de los estados que lo definen. Dentro de esta técnica destacan las representaciones mediante *Grafos*, *Redes de Petri*, el análisis *Means-End* y la representación *STRIPS*.
- **La Lógica de Predicados**, que utiliza predicados relacionados con posibles acciones para expresar sentencias que permitan deducir nuevos hechos.
- **Los Sistemas de Producción**, que establecen reglas que permiten inferir acciones a partir de datos o condiciones.
- **La Lógica Temporal**, que utilizan el tiempo como principal factor para deducir acciones y estados en el futuro.

CÁLCULO DE SITUACIONES

El cálculo de situaciones es una formulación originalmente introducida en [McCarthy y Hayes, 1969]. McCarthy y Hayes se plantean la necesidad de representar el mundo, incluyendo la información y las leyes de cambio necesarias para determinar los estados en los que se encuentra en cada momento. Así, se define el concepto de situación como a una fotografía del mundo en un instante de tiempo considerado. De alguna manera, el concepto de situación es bastante similar al de estado, aunque subyace una connotación más amplia en su definición. De hecho, una situación no es definible en su totalidad, sino que se pueden inferir aspectos de ella a partir de ciertas propiedades o características que sabemos que se cumplen. De estas propiedades y características se podrán deducir, además, nuevas situaciones que se darán en un futuro.

En su forma más común, el cálculo de situaciones deviene en una formulación más restrictiva expuesta en [Green, 1969] que determina que las situaciones sólo pueden construirse a partir de un encadenamiento de acciones que parte de un estado inicial. Esta forma es especialmente apropiada para su uso en planificación. Así, cuando se plantea conseguir un objetivo determinado, se asume que

existe una situación en la que este objetivo se cumplirá. Con ello, esta situación objetivo se alcanzará a partir de una secuencia de acciones o plan que parte de un estado inicial.

El cálculo de situaciones es adecuado para representar situaciones que se mantienen estables a lo largo de un intervalo de tiempo, pero no es lo bastante completa y tiene ciertas carencias como para poder describir, por ejemplo, la duración de las acciones o las características que tienen que ver con condiciones que deben mantenerse a lo largo del tiempo. De hecho a lo largo de estos años, se ha ido completando la formulación buscando una representación más completa del mundo real, como por ejemplo la consideración de eventos concurrentes [McCarthy, 1997], o recurriendo a la formulación original para desarrollar nuevos conceptos como el de estado mental de un robot [McCarthy, 1999], que hace referencia a la necesidad de que el robot sea “consciente” de alguna manera de su propia estructura de cara a enfrentarse a la solución de problemas del mundo real.

Aun así, el cálculo de situaciones ha supuesto el fundamento para establecer representaciones del mundo real en cierta forma basadas en ella. La consideración en su sentido más amplio del espacio de estados es una representación que hereda las consideraciones del cálculo de situaciones.

REPRESENTACIÓN EN EL ESPACIO DE ESTADOS

Anteriormente ya se ha introducido el concepto de estado como una configuración determinada del mundo en un instante determinado. Resulta útil considerar el concepto de espacio de estados como la representación de todos los estados posibles del mundo o entorno en el que se desarrolla una tarea de robot concreta. La solución a un problema determinado consistirá, como también se ha dicho, en pasar de un estado inicial al estado final o solución del problema a través de una secuencia de operadores o acciones.

BÚSQUEDAS EN EL ESPACIO DE ESTADOS. BÚSQUEDA EN GRAFOS

Una representación típica para el espacio de estados es la de un grafo [Harary, 1969], cuyos nodos son los estados posibles y los arcos serían los operadores aplicables. La solución, pues, a un problema dado representado de esta forma vendría dada por una búsqueda en este grafo que comenzara por el nodo que representa el estado inicial y

que, aplicando los operadores correspondientes, acabara en el nodo que supondría el estado final alcanzable.

Existen estrategias de búsqueda en grafos ampliamente conocidas que pueden ser utilizadas para llegar a obtener la solución correspondiente a un problema. Se conocen *métodos de búsqueda ciega* que, en definitiva, son métodos de búsqueda exhaustiva para encontrar caminos desde el nodo de salida (estado inicial del problema) hasta el nodo final (estado final alcanzable del problema). Esta búsqueda exhaustiva se puede reducir utilizando información relativa a la tarea a resolver. Los procedimientos que utilizan esta información se denominan de *búsqueda heurística* o *primera búsqueda buena*, denominándose a la información dependiente de la tarea información heurística. Una forma de considerar esta primera búsqueda buena es la de considerar en cada paso de la búsqueda sólo los nodos que “parezcan más prometedores”. Existen métodos que utilizan una función de evaluación para calcular los “nodos más prometedores”. La elección de esta función de evaluación determina el resultado de la búsqueda.

Uno de los métodos de primera búsqueda buena más conocidos es el *algoritmo A** [Rich, 1983]. Este algoritmo utiliza como función de evaluación de los mejores nodos en la búsqueda la definida en siguiente expresión

Ecuación 2-1

$$f(n) = g(n) + h(n)$$

donde $g(n)$ es una medida del coste de trasladarse desde el nodo inicial hasta el nodo n , y $h(n)$ es una estimación del coste adicional de trasladarse desde el nodo n hasta el nodo objetivo. Esto significa que $f(n)$ representa una estimación del coste necesario para trasladarse desde el nodo inicial al nodo objetivo pasando por el nodo n .

Los algoritmos de técnicas de búsqueda en grafos pueden realizar la búsqueda *hacia adelante*, partiendo de un estado inicial, generando nuevos nodos hasta llegar al objetivo, o *hacia atrás*, razonando desde los estados a alcanzar hasta un estado inicial. En muchas situaciones se utilizan criterios heurísticos o sistemas de producción para recorrer un grafo en busca de la solución más eficiente. En el caso de la búsqueda hacia adelante se considera el término izquierdo de las reglas a aplicar (la condición) comparándola con el estado actual y se emplean los términos derecha (resultados) para la generación de nuevos nodos. En el caso de la búsqueda hacia atrás se comparan los términos derechos con el estado actual y se utilizan los términos de la izquierda para generar nuevos nodos.

Existe la posibilidad de utilizar una estrategia de *búsqueda bidireccional* que aplique ambas búsquedas a la vez hasta que los caminos que se van generando se encuentren en algún lugar.

Un enfoque razonable al resolver un problema dado es el de aplicar una reducción al mismo. La reducción consiste en razonar hacia atrás, estableciendo varios subproblemas hasta llegar a un conjunto de subproblemas con solución final trivial. Para ello, se aplica un *operador reductor* con el que transformar la descripción de un problema dado en descripciones de subproblemas reducidos. Es posible, entonces, aplicar distintos operadores reductores a un mismo problema, y por ello, encontrar distintos conjuntos de subproblemas equivalentes. El resultado final, en todo caso, debe ser el de un conjunto de subproblemas completamente solucionables mediante procesos de búsqueda.

El proceso de reducción también se puede representar como un grafo simple, que tiene como raíz al problema a reducir y cuyos nodos son los subproblemas resultado de aplicar el operador reductor. Puesto que el problema se resuelve solucionando todos los subproblemas resultado de la reducción, los arcos que parten del problema y llegan a los subproblemas son nodos etiquetados como nodos AND. Es posible que alguno de estos subproblemas se haya reducido en otros subproblemas, de forma que resolviendo cualquiera de ellos, se obtenga la resolución de aquel. En este caso, los arcos se etiquetarían como OR.

Un *grafo solución* del problema será el subgrafo de nodos resueltos que implique la resolución del nodo de partida. Se puede asimilar el grafo solución al camino recorrido en la búsqueda realizada en un grafo normal.

Un algoritmo que permite soluciones en un grafo AND/OR [Homem de Mello y Sanderson, 1990], manejando de forma adecuada los arcos AND y OR, es el *algoritmo AO**. Este algoritmo utiliza una estructura que representa el grafo que se ha generado hasta ese momento mediante la búsqueda, asociando a cada nodo una estimación del coste del camino que va desde ese nodo hasta el conjunto de nodos solución. Este coste, en realidad, es una medida de lo “bueno” que es ese nodo. A la vez, se emplea un umbral de coste máximo, de forma que si en algún momento se sobrepasa este umbral, se abandona la búsqueda, considerándose que el coste es demasiado alto para ser práctico.

Una de las recientes aproximaciones a la planificación a partir del uso de grafos se encuentra en la definición del paradigma de planificación

Graphplan [Blum y Furst, 1997]. Graphplan funciona como un planificador de orden total o parcial dependiendo de restricciones temporales o del nivel en el que se actúa, permitiéndose la paralelización de las acciones bajo ciertas condiciones. Bajo estas condiciones y a este nivel, se garantiza encontrar el plan más corto existente.

REDES DE PETRI

Las Redes de Petri [Peterson, 1981] [Silva, 1985] [Murata, 1989] son otro de los formalismos utilizados para representar y formular análisis sobre el espacio de estados de un sistema dado. Las Redes de Petri (RdP) son grafos dirigidos compuestos por dos tipos de nodos (lugares y transiciones) unidos por arcos dirigidos. Estos arcos unen lugares con transiciones y viceversa, pero nunca dos lugares o dos transiciones. Las RdP, al igual que los grafos, pueden ser representadas matricialmente, lo cual permite aplicar algoritmos para su análisis basados en la matemática matricial.

La aplicación de las RdP a la planificación de tareas pasa por asociar lugares y transiciones a estados y acciones representativos de una tarea de robot concreta. De esta manera, se llega a una RdP que, tras su análisis, permite establecer la secuencia correcta de movimientos y acciones que lleva a la resolución de la tarea, alcanzando el subobjetivo que la define.

La ventaja principal frente a los grafos estriba en las diferentes técnicas de análisis existentes en tono a las RdP. Así, la existencia de RdP marcadas permite establecer simulaciones y extender la RdP para analizar posibilidades de acceso a lugares. El marcado permite establecer la sensibilización de las transiciones entre dos lugares, pudiéndose entonces disparar para pasar a un lugar nuevo. Las secuencias de disparo conducen a diferentes marcados alcanzables en una RdP, que a su vez se representan mediante el correspondiente árbol de alcanzabilidad. Todas estas características son especialmente adecuadas para establecer simulaciones de la dinámica de un sistema dado.

Las características de vivacidad (ausencias de bloqueos) o ciclicidad de un sistema se pueden analizar a través de una RdP. Estos análisis son más realistas si se incorpora el factor tiempo a la RdP. Esto se consigue asociando un valor temporal a los lugares, las transiciones o a las propias marcas. Así, una *RdP temporal* [Ghezzi *et al.*, 1991] es aquella en la que se incluye un valor temporal que indica el tiempo

que debe permanecer sensibilizada una transición antes de ser disparada. Cuando este valor temporal aparece en forma de un mínimo y un máximo establecidos a través de una función de tiempo de disparo, se denominan *RdP con tiempo uniforme*.

La aleatoriedad de los sistemas se refleja haciendo que los tiempos de disparo sigan una distribución exponencial. Son las *RdP estocásticas*. Se consigue todavía un acercamiento mayor a la realidad con las *RdP estocásticas generalizadas*, que permiten considerar transiciones inmediatas y transiciones temporizadas en una misma representación.

Otros tipos de RdP como las *coloreadas* [Jensen, 1992] o las de *alto nivel* [Jensen y Rozenberg, 1991] multiplican las posibilidades y características de representación de sistemas complejos, llegando, en el caso de las RdP coloreadas, a hablarse de un lenguaje de modelado y análisis de sistemas distribuidos.

Con todo, las RdP constituyen una herramienta idónea para la representación de un sistema de cara a la simulación de su comportamiento. En un apartado posterior de esta tesis doctoral se muestra la utilización de las RdP para la planificación de las acciones que debe realizar un móvil para alcanzar un lugar concreto del entorno. Este punto se ha realizado con el objetivo de validar la utilización de esta técnica asociado al problema de la planificación de acciones y movimientos, mostrando las ventajas de la simulación de los resultados que se obtienen.

ANÁLISIS *MEANS-END*

Otro de los métodos existentes para realizar el análisis del espacio de estados es la técnica de análisis *means-end* [Newell y Simon, 1963]. Esta técnica se basa en la posibilidad de realizar una búsqueda en distintas direcciones, no sólo adelante o atrás, con el objetivo de resolver las grandes partes del problema en primer lugar y luego volver a resolver los problemas que surgen al relacionar estas partes.

Considerando el espacio de estados y los objetivos alcanzables para la resolución de un problema, y teniendo en cuenta las posibles acciones u operadores que hacen cambiar de estado, el análisis *means-end* se basa en la asunción de un objetivo, el cálculo de la diferencia entre el estado actual y dicho objetivo, y la búsqueda de un operador para reducir dicha diferencia.

Una de las características que se plantean es que, si por cualquier motivo, durante la resolución de un problema, el estado se acerca más o menos al objetivo, separándose del estado al que debería llegar en ese momento, el cálculo de la diferencia hará que se elija el operador más adecuado.

La estructura de datos más importante aplicada en el análisis *means-end* es el *objetivo*. Esta estructura es una descripción de la situación actual, la situación deseada y la historia de los intentos llevados a cabo para cambiar la situación actual a la deseada. Los métodos o procedimientos que se utilizan para alcanzar los objetivos deseados son operadores de reducción del problema que, según su tipo, generan nodos AND (operadores de *transformación y aplicación*) o nodos OR (operador de *reducción*).

La técnica de análisis *means-end* fue introducida en el GPS o Solucionador de Problemas Generales [Newell y Simon, 1963]. Este sistema, fundamentado en la observación de cómo la gente se basaba en esta técnica, partía de un estado inicial A que había que transformar hasta alcanzar un estado final B. El proceso se detenía cuando no se encontraba diferencia entre el nuevo estado transformado a partir de A y el estado final B.

Un gran número de arquitecturas que incluyen módulos de planificación utiliza esta técnica [Langely *et al.*, 1991] [Carbonell *et al.*, 1991] [Kuokka, 1991].

LA REPRESENTACIÓN STRIPS

STRIPS [Fikes y Nilsson, 1971], acrónimo de *Stanford Research Institute Problem Solver*, es una de las aproximaciones más conocidas como representación del mundo a través de estados y como planificador para resolver problemas. Este sistema se desarrolló originalmente para la planificación del robot SHAKEY, y se basa en la descripción de estados como una base de datos. De hecho estos estados y objetivos se pueden obtener a partir de sentencias lógicas. Así, una acción de robot implicará la modificación de un estado, o configuración del mundo, para pasar a otro. Esta característica hace a STRIPS deudor del cálculo de situaciones.

Una de las ventajas de STRIPS es que utiliza una forma de representación de la base de conocimiento bastante eficiente. Este es uno de los problemas con los que se encuentra cualquier método de resolución en general, pues se necesita una base de conocimiento que

permita soportar métodos de interferencia eficientes. La representación propuesta por STRIPS se basa en una *lista de borrado* y una *lista de añadidos*, y utiliza un conjunto de reglas que constan de tres componentes para representar las acciones de robot.

El primer componente es una *precondición* que debe cumplirse para la aplicación de la regla, el segundo componente es una *lista de predicados* que suponen las afirmaciones que deben borrarse de la lista de borrado al aplicar la regla, y el tercer componente es la *lista de afirmaciones* a añadir a la lista de añadidos. Esta representación suele ser suficiente para su aplicación a gran parte de problemas, pero tiene un defecto principal: la falta de representación del tiempo, lo que la hace inviable para plantear un razonamiento temporal.

La metodología de análisis que utiliza STRIPS está basada en el análisis *means-end*, antes descrito. Siempre que se conozca la manera en que un operador modifica un estado y las precondiciones que han de cumplirse para poder aplicar el operador, se podrá emplear el análisis *means-end*. Un operador adecuado será siempre aquel que contenga una lista de fórmulas que posibilite la reducción de la diferencia entre el estado actual y el estado objetivo.

Uno de los problemas que tiene este método de planificación es que no considera la posibilidad de que ocurran acciones simultáneas. STRIPS asume que el mundo cambia sólo como el resultado de una única acción aislada por parte de un agente. De la misma manera, STRIPS también asume que cada acción caracteriza de manera total el cambio que se produce en el mundo cuando dicha acción se ejecuta. Pero no tiene en cuenta los procesos de planificación no lineales.

Estos procesos permiten la ejecución de acciones de manera retardada en el tiempo, de forma que no es posible conocer con anterioridad en qué orden se ejecutarán. Esto es, en un planificador no lineal, un plan puede representarse como la secuencia de acciones A-B-C-D o como la secuencia de acciones A-C-B-D, teniendo en cuenta que las acciones B y C pueden retardarse en el tiempo y por tanto no puede asumirse una secuencia entre ellas. Lógicamente, el estado final al que se llegaría dependería de la característica de las acciones B y C (su posible incompatibilidad) y del orden final en el que se ejecutarán.

A pesar de estos inconvenientes, STRIPS se ha convertido con el tiempo en un modelo que ha sido utilizado en gran número de sistemas de planificación, tanto en su forma original como en versiones mejoradas [Vere y Bickmore, 1990] [Carbonell *et al.*, 1991] [Kuokka, 1991].

LÓGICA DE PREDICADOS

Una de las formas de resolver problemas más utilizadas es la lógica de predicados. El cálculo por predicados de primer orden se emplea para expresar un amplio rango de sentencias, utilizando un formalismo lógico que permite deducir nuevos conocimientos de los antiguos (deducción matemática). Se trata de deducir que una sentencia es correcta probando que procede de sentencias que ya se han demostrado como verdaderas.

La lógica proposicional permite representar el conocimiento simplificando su trato y permitiendo el uso de procedimientos de decisión. Así, se representan hechos del mundo real mediante *proposiciones*, escritas como *fórmulas correctamente formalizadas (wff)*. Existe una serie de reglas de inferencia asociadas para producir *wff* a partir de otras *wff*. Las *wff* derivadas se denominan en la lógica de predicados *teoremas*, y la secuencia de reglas de inferencia utilizadas en la deducción se denomina la *prueba* del teorema.

En [Chang y Lee, 1973] se tiene una buena fuente de aproximación a la lógica de predicados y a la lógica proposicional. Por otro lado, en [Fu *et al.*, 1988] se establece una relación entre la lógica de predicados y la planificación en sistemas robotizados.

La lógica proposicional permite expresar en forma de *wff* afirmaciones del mundo real, como por ejemplo:

el móvil está parado	parado
el móvil está en movimiento	movimiento

De estas proposiciones se pueden deducir otras, por ejemplo:

si el móvil está en movimiento, no está parado
movimiento \rightarrow \sim parado

Pero sin embargo, la lógica proposicional está limitada para expresar afirmaciones más complejas, como por ejemplo:

todos los móviles pueden estropearse

Este tipo de afirmaciones o sentencias necesitan cuantificaciones a menos que se quiera escribir sentencias separadas que expresen las posibilidades de estropearse que tiene cada móvil. Es por eso que se utiliza la lógica de predicados, donde se pueden desarrollar expresiones más completas, basadas también en *wff*.

La lógica de predicados utiliza los *símbolos de predicado*, los *símbolos de función*, los *símbolos de variable* y los *símbolos de constante* como componentes básicos. Con estos símbolos, y mediante el desarrollo de lo que se denominan *fórmulas atómicas*, se pueden expresar relaciones del tipo:

enlugar(movil-A, p1)

en la que se tiene un símbolo de predicado que expresa la relación de que el móvil etiquetado como A está situado en el lugar etiquetado como 1 por medio de una fórmula atómica. En ella, movil-A y p1 son símbolos de constantes.

Utilizando símbolos de variables se tendría una expresión del tipo:

enlugar(x, y)

en la que no se especifican las entidades a las que nos referimos.

Las fórmulas atómicas son la base para el desarrollo de expresiones en la lógica de predicados y se evalúan como verdaderas o falsas según se cumplan o no todas sus sentencias.

Las fórmulas atómicas se combinan entre sí mediante los conectores \wedge (y), \vee (o) y \Rightarrow (implica) para formar *wff* más complejas. A estos conectores se les añade el uso del símbolo \sim (no) para modificar el valor de una *wff* (pasar de verdadero a falso o de falso a verdadero).

El cuantificador universal ($\forall x$) se utiliza cuando una fórmula atómica, $P(x)$, se evalúa como verdadera para todos los valores posibles de x . Si existe al menos un valor de x que permita que la fórmula atómica se evalúe como verdadera, entonces esto se expresa anteponiendo el cuantificador existencial ($\exists x$) a $P(x)$.

Sobre las *wff* se pueden aplicar reglas de inferencia, como el *modus ponens*, para producir nuevas *wff* de los ya conocidos. Las nuevas *wff* obtenidas se consideran *teoremas*, mientras que la secuencia de aplicaciones de reglas de inferencia es lo que se denomina la *prueba* del teorema. En inteligencia artificial, a la resolución de algunos problemas se la considera el hallar la prueba de un teorema dado. Aplicado esto al ámbito de la robótica, dada una tarea concreta para un robot (teorema), la planificación será encontrar la secuencia de acciones que lleven a resolver la tarea concreta, es decir, la prueba del teorema.

SISTEMAS DE PRODUCCIÓN

Los sistemas de programación organizados de tal manera que las acciones que se van tomando dependen de datos o sucesos, algunos imprevistos, se denominan sistemas de inferencia dirigidos por patrones. Estos sistemas han supuesto un avance muy importante en la programación en los últimos años, permitiendo establecer programas que se organizan en módulos que detectan y reaccionan ante patrones.

Los sistemas basados en reglas son un tipo de estos sistemas de inferencia caracterizados por estar compuesto de un conjunto de reglas. Las reglas son módulos dirigidos por patrones formados por un *antecedente* y un *consecuente*. El antecedente, o lado izquierdo, es el responsable de examinar la estructura de datos y compararla con los patrones o plantillas existentes de cara a tomar decisiones que se trasladan en acciones. Dichas acciones vienen expresadas en el consecuente, o lado derecho.

Existe una gran variedad de trabajos relativos a los sistemas basados en reglas y sus aplicaciones. Así por ejemplo, en [Marek *et al.*, 1990] y [Marek *et al.*, 1992] se tienen dos referencias básicas a los sistemas basados en reglas no monótonos. En ellos se estudia la semántica de estos sistemas y su relación con la programación lógica, la lógica de predicados y los sistemas de inferencia en general. En [Stonebreaker, 1992] y [Bailey *et al.*, 1997] se tienen, por otro lado, la aplicación de los sistemas basados en reglas al manejo de bases de datos, utilizándose reglas activas para aumentar las posibilidades de las bases de datos tradicionales.

Cuando las reglas actúan de la manera especificada, incluyéndose la equiparación y catalogación de los datos como parte implícita del sistema, estamos ante lo que se denominan sistemas de producción.

Los sistemas de producción puros, es decir, aquellos en los que cuando el antecedente se evalúa como cierto, se ejecutan las acciones del consecuente, buscando modificar el sistema para acercarse al objetivo final, son los que mayor auge e implantación tienen. Este tipo de sistemas de producción son indicados para la resolución de tareas de robots en los que el entorno puede ser dinámico. Así, el establecimiento de reglas adecuadas puede hacer que los cambios del entorno se evalúen constantemente para tomar las acciones adecuadas de cara a finalizar correctamente un tarea de robot.

Un sistema de producción está compuesto por una *base de hechos*, las *reglas de producción* y una *estrategia de control*.

La base de hechos es el lugar donde se almacena toda la información relativa a los cambios en el sistema. Son las plantillas y patrones en los que se basan las reglas para producir los cambios en el sistema. Trasladado a un sistema en el que un robot debe cumplir determinadas tareas, la base de hechos constituirá un histórico en el que nos basamos para establecer acciones de cara a cumplir las tareas encomendadas, modificándose con nuevas acciones posibles.

Las reglas son la base de los sistemas de producción y cuantifican soluciones parciales o totales de los problemas en pequeñas producciones, según un esquema compuesto por un antecedente y un consecuente.

Ecuación 2-2

$$A \rightarrow C$$

Estas reglas o producciones tienen el significado de:

Ecuación 2-3

SI antecedente **ENTONCES** consecuente

y presentan las siguientes características:

- a) Son independientes unas de otras. Una regla de producción no debe referenciar a otra.
- b) Tienen un carácter aproximado, es decir, están basadas en la comparación de datos (análisis del antecedente) obtenidos de la experiencia, y por tanto hay un cierto factor de incertidumbre.

Las reglas pueden representarse a partir de expresiones similares a la mostrada en la Ecuación 2-3 o utilizando el formalismo de la lógica.

Por último, la estrategia de control es el mecanismo que determina cómo se establece la comparación con la base de hechos y qué reglas disparar como resultado de esta comparación.

Son también muchas las referencias que se pueden encontrar dónde se detallan las características y aplicaciones de los sistemas de producción. En [Newell, 1973] se tiene una referencia general sobre los sistemas de producción, como una primera aproximación a la estructura cognitiva humana. Otros trabajos hacen referencia a los sistemas de producción paralelos [Gupta, 1987] [Kuo y Moldovan,

1992] o los métodos de optimización a partir de la ejecución en paralelo de las reglas [Ishida, 1994].

Como referencias básicas, en [Rich y Knight, 1991] y en [Borrajo *et al.*, 1997] se tiene una buena descripción de los sistemas de producción y la utilización de reglas.

LÓGICA TEMPORAL

Las definiciones de estado, evento o acción, abordadas con anterioridad, están ligadas al factor temporal de una manera indisoluble. Así, cuando nos referimos al concepto de estado, decimos que es la configuración del mundo durante un intervalo de tiempo dado, asumiendo que el mundo es cambiante y que por tanto un estado alude a una configuración temporal.

La resolución a un problema en general lleva asociada un factor temporal. Así, una tarea de robot deberá realizarse dentro de un intervalo temporal delimitado (el tiempo de inicio de la tarea y el tiempo máximo estimado para finalizarla). La consideración del factor tiempo en la definición de problemas y su posterior resolución se hace, por tanto, necesaria; siendo preciso establecer una representación temporal coherente que permita con posterioridad abordar el proceso de resolución de forma óptima.

A lo largo de los últimos años han venido desarrollándose distintas teorías que incluyen definiciones de intenciones, acciones, eventos, estados y en general de la descripción del mundo en la que el factor temporal tiene mucho que ver. De hecho, muchas de estas teorías no son posibles sin la lógica temporal asociada.

En [Allen, 1983], [Allen, 1984] y [Allen y Hayes, 1989] se introduce la descripción de la representación temporal en inteligencia artificial utilizando el intervalo de tiempo como medida básica frente a los puntos de tiempo. La razón del uso de intervalos de tiempo reside en una analogía con el lenguaje hablado. Así, cada vez que se pronuncia una frase del tipo

apagué la luz cuando Juan se iba

se está indicando que se realizó una acción (apagar la luz) en el momento en el que se realizaba otra acción (Juan se iba), que se compone a su vez de otras acciones (Juan coge su abrigo, Juan abre la puerta, Juan sale de casa, etc). Esta situación es muy común e

En [Dean, 1989] se describe el uso de una base de datos de acciones relacionadas con el tiempo para su aplicación en la resolución de problemas en los que el factor temporal tiene un papel importante. Se describe un conjunto de técnicas y operaciones sobre esta base de datos que permiten establecer proposiciones sobre eventos dependientes del tiempo.

Freksa elabora una extensión a la lógica temporal de Allen en [Freksa, 1991]. Esta extensión permite un razonamiento temporal y espacial sobre la base de un conocimiento incompleto o impreciso, tal y como se obtiene de los procesos de inferencia y percepción. La idea principal es establecer una estructura del conocimiento de acuerdo con el vecindario conceptual de las relaciones temporales y espaciales que permita la integración de conocimientos amplios y más refinados. Concretamente, Freksa se basa en la consideración de que relaciones vecinas entre eventos tienen un similar comportamiento en cuanto a los resultados del razonamiento.

En [Goldman y Lang, 1993] se hace uso de la lógica temporal introducida por Allen para hacer un estudio de intenciones en el tiempo. En este trabajo se propone un tratamiento de las intenciones frente a las acciones, partiendo de la definición de intención de Bratman [Bratman, 1987]. En dicho trabajo, Bratman establece a la intención como la fuente de los planes de alto nivel, es decir, como el objetivo a alcanzar a un nivel más alto, como la actitud mental de tener un plan.

Goldman y Lang se sirven de la lógica temporal de Allen, que estiman válida como un sistema de primer orden en el tratamiento de las relaciones temporales entre intervalos de tiempo. En su trabajo, proponen un esquema para el tratamiento de las intenciones que se basa en el establecimiento de unos axiomas y definiciones para el tratamiento de los objetivos como un componente central de la teoría de las intenciones.

En trabajos previos, Cohen y Levesque [Cohen y Levesque, 1990] establecieron una teoría sobre las intenciones relacionadas con los agentes que deben ejecutar un plan. Este trabajo representa una de las bases de la teoría de los agentes. De la misma manera que Goldman y Lang, Cohen y Levesque siguen las definiciones establecidas por Bratman para establecer una serie de características y definiciones para formalizar una intención. Sobre estas definiciones, construyen una lógica sobre los agentes y elaboran unas construcciones entre las que destaca la de *objetivo persistente*. Esta construcción viene a determinar que un agente tiene un objetivo persistente de Φ si:

1. Tiene un objetivo que hace que Φ sea cierto de manera eventual, aunque crea que no lo sea en la actualidad.
2. Antes de desestimar el objetivo Φ , se debe cumplir una de las siguientes condiciones: (1) el agente cree que Φ se ha satisfecho, o (2) el agente cree que Φ nunca será satisfecho.

Hay un margen muy estrecho entre la definición de objetivo persistente y la de intención.

En [Allen y Ferguson, 1994] se recoge la estructura básica de la lógica temporal propuesta por Allen y se la relaciona con eventos y acciones. En este trabajo, Allen y Ferguson establecen una axiomatización de las relaciones temporales que las hace unívocas y que permite que se utilicen de cara a establecer modelos completos de tiempo. Esta axiomatización se establece considerando periodos de tiempo y la relación *precede*, expresada como

$$\textit{precede}(i,j) \equiv \text{el periodo } i \text{ precede al periodo } j$$

Estos son los 5 axiomas que se establecen sobre esta relación:

1. Todo periodo tiene otro periodo que lo precede y otro al que precede.

Ecuación 2-4

$$\forall i, \exists j, k . \textit{precede}(j,i) \wedge \textit{precede}(i,k)$$

2. Varios periodos pueden componer un periodo más largo.

Ecuación 2-5

$$\forall i, j, k, l : \textit{precede}(i, j) \wedge \textit{precede}(j, k) \wedge \textit{precede}(k, l) \rightarrow \\ \exists m : \textit{precede}(i, m) \wedge \textit{precede}(m, l)$$

3. Si un periodo i precede a un periodo j y a un periodo k , entonces cualquier periodo l que preceda a j también precederá a k .

Ecuación 2-6

$$\forall i, j, k, l : \textit{precede}(i, j) \wedge \textit{precede}(i, k) \wedge \textit{precede}(l, j) \rightarrow \textit{precede}(l, k)$$

4. Si dos periodos preceden a un periodo y existe otro periodo que los preceda a ellos dos, entonces ambos periodos son iguales.

Ecuación 2-7

$\forall i, j, k, l:$

$$precede(k, i) \wedge precede(k, j) \wedge precede(i, l) \wedge precede(j, l) \rightarrow i = j$$

5. Para dos pares de periodos tales que i precede a j y k precede a l , entonces, o bien ambos pares tienen la precedencia en el mismo lugar, o bien el lugar en el que i precede a j precede al lugar en el que k precede a l , o viceversa.

Ecuación 2-8

$\forall i, j, k, l: precede(i, j) \wedge precede(k, l) \rightarrow$

$$precede(i, l) \otimes$$

$$(\exists m : precede(k, m) \wedge precede(m, j)) \otimes$$

$$(\exists m : precede(i, m) \wedge precede(m, l))$$

En la representación de los axiomas, al igual que en el resto de representaciones en forma de ecuación relacionadas con la lógica temporal, se han utilizado los símbolos conocidos de las operaciones lógicas \wedge (and), \vee (or), \otimes (or exclusiva); así como los operadores \forall (cualquiera), \exists (existe) y \rightarrow (entonces).

En la siguiente figura se refleja de forma gráfica el significado de cada uno de los axiomas establecidos sobre la relación anterior:

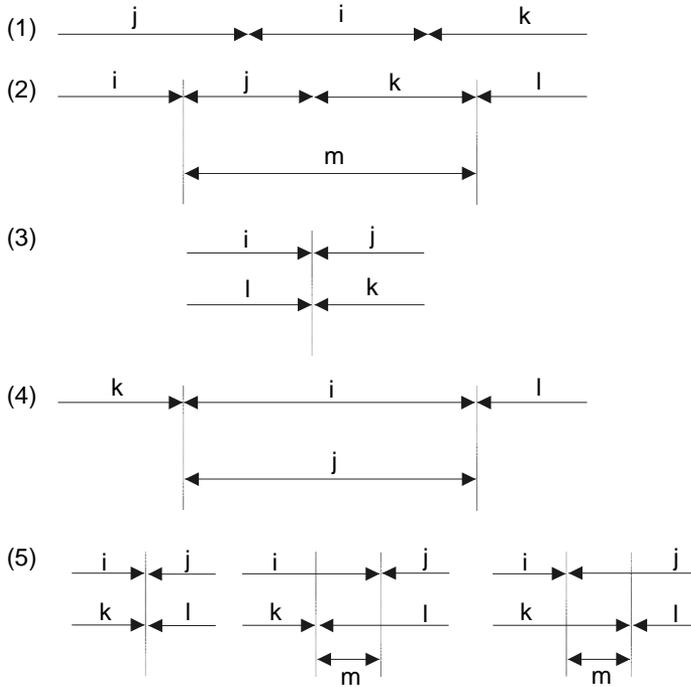


Figura 2-1. Axiomatización de los periodos de tiempo.

Esta axiomatización permite describir todas las posibles relaciones entre dos periodos de tiempo. La Tabla 2-2 muestra gráficamente estas relaciones, ya definidas en la Tabla 2-1, incluyendo las relaciones inversas.

Una relación no incluida en esta tabla pero que resulta muy útil es la relación *disjuntos*. Dos intervalos son disjuntos si no se solapan de ninguna manera. Esta relación se puede expresar a partir de las relaciones básicas como sigue:

Ecuación 2-9

$$disjuntos(i, j) \equiv [anterior(i, j) \vee precede(i, j)] \vee [anterior(j, i) \vee precede(j, i)]$$

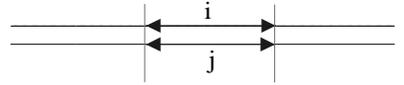
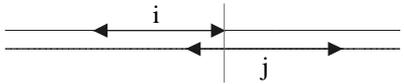
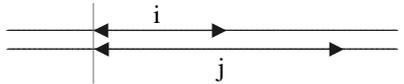
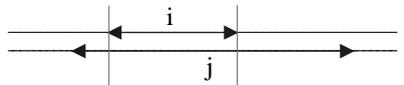
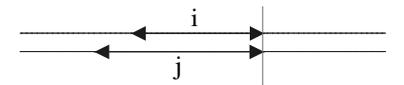
relación	representación gráfica	relación inversa
<i>anterior(i,j)</i>		<i>después(j,i)</i>
<i>precede(i,j)</i>		<i>precedido-por(j,i)</i>
<i>igual(i,j)</i>		<i>igual(j,i)</i>
<i>solapa(i,j)</i>		<i>solapado-por(j,i)</i>
<i>comienza(i,j)</i>		<i>iniciado-por(j,i)</i>
<i>durante(i,j)</i>		<i>contiene(j,i)</i>
<i>finaliza(i,j)</i>		<i>finalizado-por(j,i)</i>

Tabla 2-2. Posibles relaciones entre dos periodos de tiempo.

A partir de estas relaciones se definen los conceptos de intervalo, momento y punto de tiempo. Así, un intervalo será el periodo de tiempo que contenga subperiodos de tiempo; un momento será el periodo de tiempo que no contenga subperiodos, y un punto de tiempo será la construcción que suponga el inicio o el final de un periodo de tiempo.

Hay que distinguir entre momento y punto de tiempo. Mientras que un momento es un periodo de tiempo y como tales pueden establecerse relaciones de precedencia entre un momento y otros

periodos de tiempo, un punto de tiempo no es un periodo y por tanto no pueden establecerse relaciones de precedencia en las que participen.

Estas relaciones permiten establecer un modelo temporal completo. Por ejemplo, esta lógica temporal permite establecer un modelo discreto del tiempo en el que un periodo de tiempo vendrá dado por un par de enteros (I,J), siendo I<J. Así, un momento se corresponderá con un par de enteros del tipo (I,I+1), y los puntos de tiempo se corresponderán con los propios valores enteros.

Allen y Ferguson utilizan estas relaciones temporales básicas para desarrollar una teoría sobre los eventos. Así, para describir un evento en particular, no solamente tienen en cuenta configuraciones del mundo que se deben cumplir antes y después, si no que también consideran los valores temporales de estas configuraciones.

Por ejemplo, el evento de que un robot coja un objeto A y lo apile sobre otro objeto B significará que al final de su ejecución se dará un estado del mundo en que A está sobre B:

Ecuación 2-10

$$APILAR(A,B,t) \rightarrow \exists t' > t : SOBRE(A,B,t')$$

Como se aprecia en la anterior expresión, los eventos y los estados llevan asociados un valor temporal en el que se supone que son ciertos.

Para definir de manera más completa este evento, se incorporan las relaciones temporales anteriores y las precondiciones que deben darse para que el evento se realice. Así, el evento anterior se puede expresar como sigue:

Ecuación 2-11

$$\begin{aligned} APILAR(A,B,t) \rightarrow \exists j,k,l,m,n : \\ LIBRE(A,j) \wedge solapa(j,t) \wedge COGIDO(A,k) \wedge finaliza(k,t) \wedge precede(j,k) \wedge \\ LIBRE(A,l) \wedge precede(t,l) \wedge LIBRE(B,m) \wedge finaliza(t,m) \wedge \\ SOBRE(A,B,n) \wedge precede(t,n) \end{aligned}$$

La representación visual del evento se presenta en la siguiente figura:

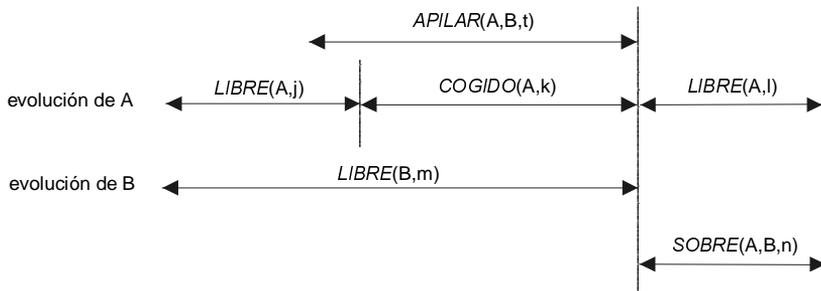


Figura 2-2. Representación visual de las condiciones necesarias para que se realice el evento $APILAR(A,B,t)$.

En [Barber, 1993] se traza un modelo temporal basado en puntos de tiempo (*time-points*) y en duraciones temporales (*temporal duration*) para representar instantes de tiempo y distancias temporales entre dos instantes de tiempo, respectivamente. Sobre este modelo temporal se plantea una métrica a partir de una serie de restricciones que afectan a los puntos de tiempo:

- (BEFORE $tp_i, td_k, [tp_j]$): expresa que tp_i ocurre temporalmente antes que tp_j más una duración temporal td_k .
- (AT $tp_i, td_k, [tp_j]$): expresa que tp_i ocurre temporalmente al tiempo que tp_j más una duración temporal td_k .
- (AFTER $tp_i, td_k, [tp_j]$): expresa que tp_i ocurre temporalmente después que tp_j más una duración temporal td_k .

La duración temporal td puede ser positiva o negativa, tp_j es opcional y representa por defecto la referencia temporal inicial.

A partir de estas relaciones se establecen unas restricciones cuantitativas (métricas y numéricas) y cualitativas (las relaciones algebraicas simbólicas con $td_k=0$, excepto la desigualdad).

El modelo temporal de Barber se utiliza en [Crespo *et al.*, 1994] como base para establecer un sistema de control de procesos en tiempo real basado en una pizarra temporal, como una aplicación del razonamiento temporal sobre intervalos de tiempo. En [Onaíndia, 1997] se formaliza de manera eficiente el modelo de representación y razonamiento temporal en el contexto de su aplicación en sistemas de tiempo real.

PLANIFICACIÓN EN ROBÓTICA

En los apartados anteriores se hace un repaso a una serie de conceptos útiles para plantear la resolución de un problema dado. A partir de estos conceptos, se define la planificación como la estrategia que se sigue para encontrar la solución de un problema determinado.

Hasta este momento, la planificación se ha definido como un proceso más bien genérico, relacionado con cualquier tipo de problemática y que, principalmente, se ha abordado desde el prisma de la inteligencia artificial.

En el campo de la robótica, en el que la programación de los robots industriales supone la elaboración de una estrategia de resolución de una tarea concreta, la planificación tiene una gran importancia. Es por ello que las distintas técnicas de planificación, aunque desarrolladas desde el prisma de la inteligencia artificial, son aplicables a la resolución de tareas de robots.

La robótica móvil tiene un auge importante en estos momentos, debido a que supone un avance en la automatización de las industrias que la robótica, digamos, fija no puede proporcionar. Efectivamente, aunque las factorías de hoy en día ya disponen de una buena implantación de robots estáticos, capaces de realizar las más variadas tareas, la implantación de dispositivos capaces de realizar trabajos que impliquen un desplazamiento es aun incipiente.

Uno de los problemas con que se encuentran los robots móviles en su implantación está derivado, precisamente, de la ventaja que aportan: su capacidad de desplazamiento. Las factorías son entornos dinámicos en los que las posibles rutas de desplazamiento deben compartirse entre operarios, sistemas de transporte no automatizados (carretillas impulsadas por operarios) y otros robots móviles. Es precisamente por ello que gran parte de los esfuerzos en la planificación de robots móviles van encaminados a optimizar los movimientos y acciones que se deben realizar para completar un trabajo encomendado.

En muchas situaciones, las técnicas de planificación se aplican a entornos multirobots, atendiendo a las ventajas que se tienen con la inclusión de más de un robot para la consecución de una tarea. En [Costa *et al.*, 1995] se hace un repaso a algunas de estas técnicas de cara a planificar y controlar las acciones y movimientos en entornos multirobot.

Diversas técnicas se encuentran en la bibliografía centradas en la planificación aplicada a la robótica móvil. Entre las más significativas se encuentran:

- **La Planificación de Movimientos**, centradas en establecer un plan general para que un móvil determinado alcance un objetivo del entorno.
- **La Planificación Dinámica**, que hace referencia a la existencia de otros móviles en el entorno de planificación.
- **La Planificación de Ensamblajes**, dedicada en especial a resolver uno de los típicos problemas de robot, como es el ensamblaje de piezas.
- **La Planificación mediante Sensorización**, que establece el uso de sensores como ayuda a la resolución de un problema de planificación general.

A continuación se revisan estas técnicas, destacando los trabajos que resaltan más en cada una de ellas.

PLANIFICACIÓN DE MOVIMIENTOS

Muchas de las técnicas de planificación que se siguen provienen de los métodos utilizados para un robot, haciendo una extensión para varios robots. En [Latombe, 1991] se tiene una referencia fundamental sobre algoritmos para la planificación de movimientos de robots.

Entre los métodos más utilizados para la planificación de las trayectorias que debe seguir un robot, está la técnica de los *mapas de carreteras (roadmap)*. Esta técnica se basa en la obtención de una red de curvas unidimensionales que conecten el espacio libre por el que ha de moverse el robot y posteriormente realizar una búsqueda de la trayectoria sobre esta red. Existen diferentes técnicas que llevan a la obtención de la red de curvas: uso de un grafo de visibilidad [Lozano-Pérez, 1983], uso de un diagrama de tangencias [Doyle y Jones, 1994] o de métodos de análisis probabilístico [Kavraki y Latombe, 1994].

Otros métodos utilizan la *descomposición celular* o descomposición del espacio libre en un conjunto de figuras geométricas. La trayectoria resultado es una secuencia de estas figuras. Este método de descomposición puede ser exacto [Schwartz y Sharir, 1983] [Brooks y Lozano-Pérez, 1985] [Chazelle, 1987] [Avnaim *et al.*, 1988] [Sleumer y Tschichold-Gürman, 1999] o aproximado [Lozano-Pérez, 1987] [Thrun, 1998].

Los *métodos de optimización* permiten abordar la generación de la trayectoria a partir de una función de coste sobre la distancia a la posición objetivo y a otros objetos del entorno. Entre estos métodos destacan los que se basan en los *campos potenciales* [Khatib, 1986] [Khosla y Volpe, 1988] [Barraquand *et al.*, 1990], la programación dinámica [Barraquand y Ferbach, 1993] o los algoritmos genéticos [Blume *et al.*, 1994].

PLANIFICACIÓN DINÁMICA

Cuando se tienen en cuenta obstáculos móviles en el entorno, el problema de planificación se denomina *planificación dinámica de movimientos de robot* [Latombe, 1991]. En este caso, el tiempo pasa a ser el factor relevante, aparte de las consideraciones espaciales en cuanto a las configuraciones del entorno y del propio robot que se tienen en cuenta en los métodos anteriores. El problema de planificación de movimientos pasa a considerarse el de situar un punto en una configuración espacio-temporal del entorno [Canny y Reif, 1987] [Sutner y Maas, 1988] [Reif y Sharir, 1994] [Berrety *et al.*, 1997].

La consideración del espacio de velocidades del robot es uno de los métodos utilizados para la planificación en entornos dinámicos. El ajuste de velocidades (*velocity tuning*) fue introducido en [Kant y Zucker, 1986]. En este trabajo se proponen dos fases para llevar a cabo la planificación de las actividades de un robot: primero, planificar un camino óptimo que permita evitar los obstáculos estáticos en el entorno de trabajo del robot; segundo, planificar la velocidad adecuada que permita al robot ir sorteando los obstáculos dinámicos del entorno.

Esta aproximación se ha utilizado en trabajos más recientes que consideran una descomposición camino-velocidad para la planificación dinámica de trayectorias de robot [Fraichard y Laugier, 1993] [Fiorini y Schiller, 1995].

En [Mellado, 1996] se tiene una aplicación de la generación de trayectorias con evitación de colisiones para entornos multirobot. En este trabajo se introduce una nueva técnica para el modelado de los sistemas robotizados a partir de la generación de esferoides y *splines* esféricas [Mellado y Tornero, 1993] que resulta de gran utilidad para el cálculo de distancias entre los objetos del entorno. Esta característica proporciona grandes ventajas para el cálculo de colisiones, aspecto fundamental de la planificación de trayectorias. La

planificación de trayectorias se realiza mediante métodos de campos potenciales artificiales y de optimización.

Existen otros enfoques distribuidos para abordar la planificación de los movimientos de un sistema robotizado que actúa en un entorno multirobot. Estos hacen referencia a técnicas que calculan la trayectoria de cada uno de los robots que actúa en el mismo entorno por separado, compatibilizándolas *a posteriori*. Esta compatibilización posterior debe llevar asociada un correcto cálculo de las colisiones que se pueden dar entre trayectorias de distintos robots.

Los métodos que se basan en estudios probabilísticos [Freund y Hoyer, 1988] [Warren, 1990] establecen una jerarquía entre los diferentes sistemas que intervienen, planificando el robot más prioritario primero y supeditando a éste la planificación del resto.

Las técnicas de planificación distribuida basadas en *scheduling* calculan en paralelo las trayectorias para los distintos robots, aplicándose con posterioridad una compatibilización entre las mismas, bien a partir de búsqueda en grafos [Lin y Tsai, 1991]; bien parametrizando las trayectorias con una variable independiente, localizando zonas de colisión y generando curvas que las eviten [O'Donnell y Lozano-Pérez, 1989] [Mohri *et al.*, 1993].

Para aplicaciones en las que varios robots sujetan al mismo tiempo una misma pieza, formando cadenas cinemáticas cerradas, las técnicas de planificación para cadenas cinemáticas abiertas chocan con el aumento considerable de grados de libertad y la complejidad en el cálculo de objetos en el espacio. En este contexto han de aplicarse técnicas que atienden a características propias [Barraquand y Ferbach, 1994] [Koga y Latombe, 1994] [Fisher, 1994], aportando algunas fuertes componentes heurísticas.

PLANIFICACIÓN DE ENSAMBLAJES

Un apartado importante es el dedicado a la planificación de los movimientos necesarios para tareas de ensamblaje. Este tipo de aplicaciones se caracterizan porque el robot debe desplazar su efector final en un entorno limitado con presencia de varias piezas a manipular. Es por ello que resulta muy importante la generación precisa de las trayectorias, en vistas a evitar colisiones no deseadas. Existen distintas técnicas propuestas para la planificación de tareas

de ensamblaje [Lozano-Pérez *et al.*, 1984] [Xiao y Volz, 1989] [Suárez *et al.*, 1995] [Rosell *et al.*, 1999].

Las técnicas basadas en el establecimiento de grafos de cara a encontrar la secuencia de acciones correcta para una tarea de ensamblaje también son bastante comunes. Así, se tienen distintas aproximaciones a las tareas de ensamblaje a partir del establecimiento de grafos AND/OR [Homem de Mello y Sanderson, 1986] [Homem de Mello y Sanderson, 1990] o grafos de precedencia [Naphade *et al.*, 1999a] [Naphade *et al.*, 1999b].

Como se puede apreciar, la mayoría de estas técnicas están centradas en la planificación de las trayectorias que debe seguir un robot para llevar a cabo una tarea. Esta trayectoria puede estar referida tanto a la que sigue el efector final de robot para un manipulador fijo, como al desplazamiento que sigue un robot móvil.

La planificación de las trayectorias de los robots móviles es, de hecho, un campo en el que se siguen numerosos trabajos de investigación. En [Muñoz y Ollero, 1996], donde se propone un método que considera un grafo de visibilidad cinemática para generar la ruta, un algoritmo basado en *beta-splines* para generar la trayectoria y la definición de un perfil de velocidades que se adapten a las restricciones cinemáticas y dinámicas de la trayectoria obtenida.

PLANIFICACIÓN MEDIANTE SENSORIZACIÓN

La inclusión de sensorización en los robots permite utilizar los datos que proporcionan en los procesos de planificación y seguimiento de trayectorias. La planificación de trayectorias basada en los datos de sensorización permite que el robot interactúe y reaccione ante el entorno. Los datos provenientes de los sensores son útiles especialmente para realizar acciones de seguimiento y corrección de la trayectoria planificada.

En [Borenstein *et al.*, 1996] se tiene un completo trabajo de la utilización de sensores en robótica móvil para la obtención de la posición real del robot. La obtención de la posición real del robot permite establecer desviaciones respecto de la posición teórica programada, que se utilizan como una realimentación en posteriores procesos de calibración.

Los datos obtenidos de la sensorización permiten establecer estrategias de planificación *on-line* que generan maniobras de

evitación de obstáculos que aparecen en el espacio de trabajo. El objetivo final de estas técnicas es el de dotar a los robots de un grado de autonomía necesario para que se desenvuelvan en lugares los que se hace necesario un control remoto de los robots, como es el caso de los robots que operan en entornos inaccesibles [Brunner *et al.*, 1993] [Brunner *et al.*, 1995] [Moorehead *et al.*, 1999].

Existen trabajos centrados en la navegación de robots móviles o autónomos a partir del uso de visión artificial. El principal problema asociado a la navegación de robots móviles es, como ya se ha planteado, la estimación de la posición real del mismo durante el seguimiento de una trayectoria prefijada. En ese sentido, el uso de visión artificial permite obtener imágenes del entorno que son utilizadas para realizar el seguimiento de una trayectoria prefijada o para la extracción de marcas naturales o artificiales que servirán para estimar la posición del vehículo. Uno de los campos de aplicación de este tipo de técnicas es en móviles que realizan tareas en entornos agrícolas [Hague y Tillet, 1996] [Sánchez y Marchant, 1997].

Como punto final a este repaso de las técnicas de planificación en robótica existentes, cabe resaltar tendencias novedosas como son el uso de características de comportamiento del mundo animal. En [Möller *et al.*, 1998] se detalla como se aplica a la navegación de un robot móvil el estudio de cierto tipo de hormigas que son capaces de desenvolverse en entornos complejos. En concreto, en este trabajo se integran técnicas de planificación *on-line* de la trayectoria a partir del uso de sensores que se basan en la obtención de datos a partir de luz polarizada, de igual forma que hacen este tipo de hormigas. Estos datos sirven para la generación de un vector que determina el camino para volver a lugares predefinidos a partir de distancias recorridas y de información direccional.

ARQUITECTURAS DE PLANIFICACIÓN

Las técnicas de planificación, y en general los planificadores, se plasman en la implantación de una arquitectura hardware y software adecuada. A nivel hardware, tanto el móvil como los distintos componentes y accesorios que componen el sistema completo se deben elegir para que puedan cumplir los requisitos establecidos, en primer lugar por las actividades a realizar, y en segundo lugar por las características del planificador.

Una arquitectura software, por su parte, hace referencia a la existencia de los distintos módulos y la conexión a nivel de

intercambio de información entre ellos que permitan la ejecución de las distintas etapas de planificación.

Una de las primeras clasificaciones para las arquitecturas de control de los robots móviles se tiene en [Heiserman, 1981]. En este trabajo, se propone una clasificación en lo que se denominan *robots Alfa, Beta y Gamma*, atendiendo a la complejidad del sistema de control. El robot tipo *Alfa* supone un control reflexivo simple. Cuando un robot de este tipo se encuentra con un obstáculo en su camino, el sistema de control simplemente elige una nueva orden de movimiento entre las permitidas. Un robot tipo *Beta* funciona inicialmente igual, pero al escoger una nueva orden de movimiento, la almacena en su memoria para ser usada en condiciones similares del entorno. Por último, un robot tipo *Gamma* tiene un sistema de control más complejo, capaz de predecir el futuro. Posteriormente, en [Rietman, 1994] se amplió esta clasificación introduciendo los robots *delta*, manejados mediante redes neuronales.

Otra clasificación de las arquitecturas de los robots móviles atiende a la filosofía de control utilizada. En esta clasificación se consideran las *arquitecturas jerárquicas*, las *reactivas* y las *basadas en comportamientos* [Brooks, 1986] [Gachet *et al.*, 1992] [Watanabe *et al.*, 1992] [Muñoz *et al.*, 1995]. Las arquitecturas jerárquicas se caracterizan por presentar una estructura vertical, con una separación de funciones en cada uno de los niveles jerárquicos. Las arquitecturas reactivas presentan, por su parte, una estructura horizontal, donde cada elemento supone un comportamiento simple. Los distintos comportamientos actúan en paralelo a partir de la información sensorial, generando respuestas en forma de movimientos. Las arquitecturas basadas en comportamientos presentan, por último, una estructura híbrida entre los dos modelos anteriores.

En [de Lope, 1998] se plantea una arquitectura de control para la navegación y planificación de rutas en robots autónomos. Esta arquitectura de control parte del modelado del entorno a partir de Redes de Petri borrosas y desarrolla un método de navegación basado en un punto de vista reactivo. Así, a partir de la consideración de una serie de procesos a nivel bajo, se generan unos comportamientos que rigen las actividades del robot autónomo. Estas actividades describen el modo de funcionamiento del sistema a largo plazo, desencadenando mecanismos como la navegación libre de colisiones y la planificación eficiente de rutas.

En [Simó, 1997] se presenta una arquitectura para el control de robots móviles basada en principios reactivos. Los principios en los que se basa este trabajo se fundamentan en la existencia de agentes

que dirigen comportamientos reactivos del robot móvil. Así, la arquitectura propuesta plantea una planificación de las tareas en la que se integran distintos algoritmos, mayormente basados en procesos de búsqueda y su aplicación en sistemas de tiempo real.

En [Vandorpe, 1997] se describe una arquitectura de navegación jerárquica aplicada sobre un sistema móvil autónomo. En esta arquitectura jerárquica se sigue una planificación *off-line* de los puntos de paso del robot que utiliza un mapa del entorno y una posterior planificación *on-line* de la trayectoria a seguir a partir de un sistema de sensorización.

El uso de tres robots móviles para la realización de distintos tests de cara a la supervisión y la diagnosis de fallos y su recuperación es la característica principal de la arquitectura implementada en [López, 2000]. Aunque a nivel hardware, los tres móviles son distintos, incorporando diferente sensorización y facilidades de movimiento, a nivel software, todos los móviles presentan la misma arquitectura de navegación, formada por distintas capas funcionales, conformando distintos niveles de abstracción: servocontrol, nivel reactivo, navegación, propiamente dicha, planificación de trayectorias y planificación de tareas.

REPLANIFICACIÓN

El concepto de **replanificación** tiene que ver con la posibilidad de realizar acciones de planificación en tiempo real, a partir de planes establecidos con anterioridad. La mayoría de métodos basados en Inteligencia Artificial para replanificar tareas de robot se basan en los conceptos de eventos y acciones de cara a refinar planes.

Las técnicas de replanificación son generales y pueden ser aplicadas en cualquier ámbito, no solamente en el campo de la robótica. Así, en [Ambite, 1999] se tiene un método denominado planificación por reescritura (*planning by rewriting*) que propone dos fases en la planificación: primero, generar un plan inicial, y segundo, reescribir de forma iterativa el plan inicial hasta encontrar una solución efectiva o llegar al límite de uno de los recursos utilizados.

La reescritura del plan inicial se realiza utilizando un conjunto de reglas declarativas para reescribir términos y grafos. La reescritura de términos significa establecer convergencias entre planes para que sean agrupados de una manera más eficiente, mientras que la

reescritura de grafos implica la sustitución de subgrafos por otros cuando se satisfacen ciertas condiciones en la ejecución de una tarea. Basada en la existencia de niveles de abstracción, en [Washington y Hayes-Roth, 1992] y [Washington, 1994], se introduce una técnica para planificación en tiempo real. Este método se basa en el razonamiento temporal para predecir posibles estados futuros en la ejecución de una tarea y generar planes refinados que permitan llevarla a cabo con eficiencia.

Este método utiliza operadores similares a los introducidos en el planificador STRIPS (anteriormente referido) para representar estados del mundo. A partir de esta representación, se generan intenciones en el futuro que permiten refinar planes incompletos establecidos con anterioridad a cualquier nivel de abstracción.

Por lo que concierne al campo de la robótica, la planificación dinámica, con todas sus vertientes y métodos representa la aproximación más vigente en cuanto a replanificación de tareas se refiere. Así, los métodos relacionados en el apartado anterior suponen en cualquier caso un proceso de refinado de planes previos similar a los propuestos por métodos más generales de Inteligencia Artificial.

El uso de la lógica temporal, detallada en un apartado anterior de este mismo capítulo, permite proyectar una representación de estados y acciones en el futuro a partir del estudio de situaciones pasadas. El razonamiento sobre relaciones temporales del pasado permite establecer una predicción óptima de acciones en el futuro, de manera que se eviten contratiempos o se mejore la ejecución de un plan. El presente trabajo de investigación incluye un apartado dedicado a la replanificación de las trayectorias del móvil a partir de razonamientos de futuro (predicciones) basadas en la lógica temporal.

PLANTEAMIENTO DEL MÉTODO DE PLANIFICACIÓN

La presente tesis doctoral plantea la aplicación de las intuiciones humanas a la planificación de robots móviles, estableciendo una analogía con el proceso de razonamiento del ser humano. Esta aproximación somera al trabajo que aquí se presenta obliga, en primer lugar, a la definición del concepto de **intuición humana**.

Por *intuición* se entiende la asunción inmediata de una idea sin necesidad de recurrir a un proceso de razonamiento. Partiendo de esta premisa, las intuiciones humanas se consideran recursos

inherentes a la persona, de tal manera que componen unas características primarias del razonamiento o conocimiento.

Existen numerosas teorías centradas en los campos de la psicología y de la filosofía relacionadas con los desarrollos y procesos básicos del pensamiento y razonamiento humano. En estas teorías se plantean aspectos relacionados con el aprendizaje automático a partir de procesos básicos de la mente como son el reconocimiento, la captación visual del entorno o la integración de los sentidos.

Abert Bandura formula la llamada Teoría del Aprendizaje Social [Bandura, 1977], en la que plantea que el aprendizaje cognitivo depende de los procesos cognitivos y de las habilidades propias, incidiendo en la influencia del entorno y de las pautas de comportamiento.

Una de las teorías vigentes en la actualidad sobre el desarrollo cognitivo es la planteada por R. Siegler sobre la cognición social o procesamiento de la información [Siegler, 1994]. Según este autor, el desarrollo cognitivo se basa en tomar información del exterior, registrarla de forma simbólica (codificada), combinarla con otras informaciones, almacenarla y enviarla de nuevo al exterior decodificada. Los estudios de Siegler se unen a las teorías vigentes sobre el desarrollo cognitivo y aprendizaje como un proceso de evolución de la mente humana [Piaget, 1926] [Kuhn, 1972].

En cualquier caso, las teorías sobre la cognición se centran en procesos básicos intuitivos que llevan al aprendizaje y al desarrollo del razonamiento humano. Estos procesos intuitivos son la base de actividades como la planificación y resolución de problemas.

Los procesos derivados de intuiciones humanas son la base de los planificadores inteligentes, es decir, de aquellos planificadores cuya principal característica es la de implementar un algoritmo de planificación eficiente y de calidad.

La eficiencia en la planificación hace referencia a la obtención de un plan de manera rápida. La rapidez es aquí un factor importante puesto que en numerosas ocasiones, la planificación se reduce a la búsqueda de una solución a un problema en un espacio de estados muy amplio, con lo que las alternativas posibles de búsqueda se multiplican, retardando la consecución de la solución.

Por su parte, la calidad del plan hace referencia a la obtención de un plan de bajo coste, o al menos, de un coste eficiente. Cuando las alternativas de solución son muchas, la calidad del plan que se

considera como solución final es un factor importante, y el coste del plan (temporal, económico, energético...) determina la calidad del mismo.

En [Yang, 1997] se hace un análisis de las técnicas de planificación inteligentes basadas en los principios de descomposición del problema y de abstracción jerárquica.

En [Vendrell y Mellado, 1998] se hace un planteamiento básico del uso de la descomposición del problema, la jerarquización en niveles y la observación del entorno para la planificación de robots móviles. En este artículo se plantea una arquitectura software de planificación basada en intuiciones humanas y su aplicación para la planificación de robots móviles que se mueven en entornos dinámicos. Este desarrollo supone la base de la metodología de planificación propuesta en esta tesis doctoral.

Las intuiciones humanas que se consideran en este trabajo como base del proceso de planificación son la **descomposición**, la **abstracción**, la **deducción** y la **predicción**. Las dos primeras sugieren un planteamiento del proceso de planificación *a priori*, es decir, suponen la obtención de un plan como resultado a un problema planteado antes de abordarlo. Las dos últimas son intuiciones aplicables durante el proceso de resolución del problema a partir del plan obtenido con anterioridad. Se trata pues de intuiciones que actúan *a posteriori* permitiendo un proceso posterior de replanificación. El siguiente esquema refleja esta actuación:

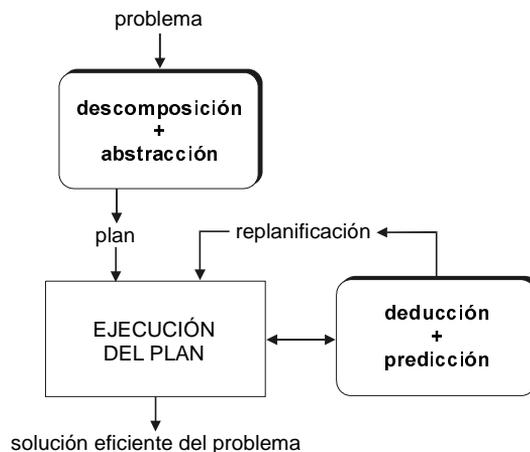


Figura 2-3. Esquema del método de planificación basado en intuiciones humanas.

La descomposición hace referencia a un proceso previo de *divide y vencerás*, ampliamente utilizado en resolución de problemas y en algorítmica. Con este proceso se busca la simplificación en la planificación a partir de su aplicación a partes reducidas del problema de planificación global.

A partir de otro proceso previo, la abstracción, se busca también la simplificación del problema de planificación. De alguna manera, descomposición y abstracción son procesos similares. La diferencia radica en que mientras que la descomposición es un proceso de simplificación cuantitativo, la abstracción plantea una simplificación cualitativa. Es decir, se plantea la aplicación del proceso de planificación sobre diferentes niveles de definición del problema, pero contemplando siempre el total del mismo.

La extracción de datos de la realidad permite plantear un proceso deductivo. Se trata, a diferencia de los procesos anteriores, de un planteamiento en tiempo real que permite la obtención de conclusiones que se utilizan en el proceso de planificación.

Por último, la predicción de estados y situaciones en el tiempo, un proceso inherente a la mente humana, es la base de la planificación de tareas de robots móviles. Este proceso predictivo, planteado en tiempo real, se fundamenta en la consideración del factor temporal en cada una de las situaciones que se exponen y su incertidumbre asociada.

A continuación se referencian de manera más completa las intuiciones humanas base de la metodología de planificación.

DESCOMPOSICIÓN DEL PROBLEMA

De cara a simplificar la planificación, se aborda una **descomposición del problema** en partes. La descomposición del problema permitirá abordar de forma diferenciada aspectos diferentes del mismo. Así, dependiendo de las características de cada parte, se aplicará con mayor o menor facilidad el proceso de planificación.

La descomposición en partes supone una división *vertical* del problema, estableciéndose partes independientes en los que se aborda el proceso de planificación de manera completa.

Uno de los problemas asociados a la descomposición del problema en partes es la posterior combinación de las soluciones resultado de planificar cada parte. La solución final del problema vendrá dada por la combinación eficiente de las soluciones parciales de cada parte.

ABSTRACCIÓN JERÁRQUICA

La **abstracción jerárquica** supone la aplicación de la metodología a distintos niveles de simplificación, comenzando desde un nivel superior en el que se plantea la misión o tarea en su globalidad, hasta llegar al nivel más bajo, en el que se consideran las acciones más elementales que debe realizar el robot.

La abstracción jerárquica supone una simplificación *horizontal* del problema. Esto es, el problema se subdivide en niveles que abarcan todo el problema, y a los que se les aplicará posteriormente el proceso de planificación de manera específica.

OBSERVACIÓN DEL ENTORNO (DEDUCCIÓN)

El proceso de observación del entorno es un proceso intrínseco a la naturaleza humana y que supone la base de cualquier razonamiento. Cuando se plantea un problema en el mundo real, un entorno dinámico, la observación de los distintos agentes del entorno es una de las primeras características básicas para abordar la resolución. El proceso de observación es la base para acudir a otras características como son la deducción de eventos y el aprendizaje de los mismos.

En una arquitectura software asociada a la metodología de planificación, el módulo de observación del entorno juega un papel principal. La extracción de los datos del entorno, la configuración de un histórico de eventos y la deducción de nuevos eventos para agentes que intervienen en el contexto del problema son funciones básicas encomendadas al módulo observador.

ESTIMACIÓN TEMPORAL (PREDICCIÓN)

La estimación temporal es la característica básica propuesta para abordar la planificación de tareas y actividades. En el presente trabajo se plantea el uso de esta intuición humana a un nivel concreto del problema.

La lógica temporal descrita en la sección anterior permitirá considerar una tarea de robot con su duración y su ubicación en el tiempo asociada. Ello nos llevará a la posibilidad de considerar una predicción de las acciones y movimientos de los robots implicados en la resolución de un problema dado. Esta predicción temporal tiene como objeto la optimización de las acciones de los robots controlados.

CONCLUSIONES

La planificación de robots es uno de los principales problemas que se abordan de cara a conseguir sistemas productivos altamente automatizados. En este capítulo se hace referencia a diferentes alternativas que se tienen en cuenta en la planificación de un robot móvil en entornos dinámicos. Las técnicas basadas en Inteligencia Artificial, desarrolladas de manera genérica, son de especial utilidad, describiéndose sistemas completos a partir de la definición de estados, eventos y acciones.

Los sistemas de planificación y replanificación existentes en robótica no suelen abordar una metodología completa, centrándose en la solución de aspectos particulares (planificación de caminos o movimientos, replanificación reactiva a bajo nivel). Este capítulo presenta como principal aportación una metodología global de planificación y replanificación de las actividades de un robot móvil a partir de intuiciones humanas. La descomposición y abstracción permiten configurar un esquema de planificación completo, mientras que la deducción y la predicción, basada en la estimación temporal de las actividades de otros móviles, llevan a la replanificación de las trayectorias del móvil en tiempo real.

Los siguientes capítulos formalizan la metodología de planificación de robots móviles propuesta en este capítulo. El capítulo 3 aborda con detenimiento la planificación de actividades mediante el uso de la descomposición y la abstracción. El capítulo 4 aborda el proceso de deducción, mientras que la predicción, base de la replanificación de las actividades, se aborda de manera específica en el capítulo 5 de la presente tesis doctoral.



Planificación, replanificación y lógica temporal

3 Planificación mediante descomposición y abstracción

Planificación mediante descomposición y abstracción

En el capítulo anterior se han introducido los conceptos de estados, eventos y acciones de cara a describir un problema concreto a solucionar. Así mismo, se ha tratado el concepto de planificación como la estrategia que se sigue para solucionar un problema determinado.

La observación del factor tiempo como una característica asociada a la descripción de un problema nos lleva a considerar, como se ha visto, una lógica temporal. A través de esta lógica temporal se pueden ubicar en el tiempo tareas determinadas y abordar consecuentemente un proceso de estimación temporal de las acciones necesarias para ejecutar esas tareas.

A partir de estos conceptos, se introduce una metodología de planificación y replanificación de actividades de robots basada en intuiciones humanas. En concreto, la presente tesis doctoral propone el uso de los procesos de descomposición, abstracción, deducción y predicción como intuiciones humanas aplicables a la planificación de problemas y replanificación on-line de trayectorias en robótica.

El presente capítulo trata los procesos de descomposición y abstracción como intuiciones humanas aplicables a la resolución de problemas de robot a través de la planificación. La metodología que se presenta propone su uso como procesos previos off-line que simplifican la planificación. En el desarrollo de estos conceptos se introducen criterios y factores relativos a su aplicación en problemas relacionados con la robótica, así como ejemplos de aplicación de cada concepto. Finalmente, se incluyen también ejemplos de planificación para robots articulados, móviles y el empleo de la metodología básica en un proyecto de investigación.

Planificación mediante descomposición y abstracción

DESCOMPOSICIÓN DEL PROBLEMA

La descomposición del problema tiene que ver con la generación de subpartes del problema a partir de una característica que las diferencie. Las partes en que se puede dividir el problema vendrán así definidas por unas condiciones que de una manera general partirán del propio contexto en que se realice la misión encomendada.

La descomposición del problema permite aplicar la metodología de la planificación a partes reducidas del problema, simplificándose con ello la resolución. Para ello, es importante deducir las partes del problema de una manera coherente y robusta que permita extraer soluciones independientes, en el sentido de que no exista una dependencia entre ellas, a la vez que sea posible la combinación final de las distintas soluciones de cara a obtener un resultado final.

Supongamos que se tienen dos mesas sobre las que hay distintos bloques de madera. En este entorno, se plantea apilar de manera ordenada (por ejemplo, de mayor a menor tamaño) los bloques en cada mesa. De manera intuitiva, cualquier persona que se enfrente a este problema tenderá a resolverlo de manera separada en cada mesa.

Este es un ejemplo sencillo de cómo la mente recurre a un proceso intuitivo de descomposición del problema en partes para obtener una solución global: dejar las dos mesas con los bloques apilados de manera ordenada.

En [Yang, 1997] se trata la descomposición del problema en partes y se propone una descomposición global del mismo. Así, si se tiene un problema de planificación representado por

Ecuación 3-1

$$\langle \text{inicio}, \text{objetivo}, \Omega \rangle$$

donde inicio es el estado de partida del problema, objetivo el estado final que se persigue y Ω representa el conjunto de operadores de que se dispone para solucionar el problema.

Se asume que el problema global se descompone en n partes, de forma que cada parte se representará de la forma

Ecuación 3-2

$$\langle \text{inicio}_i, \text{objetivo}_i, \Omega_i \rangle$$

donde inicio_i , objetivo_i y Ω_i representan el estado inicial, el final y el conjunto de operadores, respectivamente, de cada parte en la que se ha subdividido el problema global.

A partir de este supuesto de descomposición, se tiene el siguiente algoritmo para abordar el proceso de planificación:

1. aplicar el proceso de descomposición para obtener las n partes
 $\{\text{parte}_i = \langle \text{inicio}_i, \text{objetivo}_i, \Omega_i \rangle\}$
2. para cada parte i , encontrar el conjunto de soluciones
3. para cada parte i , seleccionar la solución adecuada haciendo solución igual al conjunto de todas las soluciones
4. resolver conflictos entre soluciones
5. combinar las soluciones
6. si los pasos 3 y 4 se cumplen satisfactoriamente, devolver solución como el plan que resuelve el problema
7. si no, si se han considerado todas las combinaciones posibles, devolver fallo; en otro caso, volver al paso 2

Este algoritmo recibe como entrada el estado inicial del problema y el estado final deseado como solución y devolverá un plan que lleve a la resolución del problema. No se contempla, como se aprecia, la posibilidad de volver a descomponer el problema, una vez ya se tiene una primera descomposición. Esto es porque el proceso de descomposición es un proceso a priori que tiene en cuenta las condiciones iniciales planteadas en el problema y por tanto encuentra una descomposición acorde con esas condiciones. Solamente en el caso de cambiar estas condiciones, y esto sólo sucederá en situaciones imprevistas, se podría plantear una nueva descomposición del problema.

La aplicación de este algoritmo teórico se encuentra con una serie de elementos que hay que considerar convenientemente para poderse aplicar con eficiencia. Estos elementos son los siguientes, expuestos en orden de consideración:

1. Un proceso adecuado de descomposición en partes.
2. Un método para encontrar las soluciones para cada parte.
3. La selección conveniente de la solución para cada parte.
4. La resolución de conflictos entre soluciones
5. La combinación eficiente de las soluciones de cada parte para llegar a una solución global.

DESCOMPOSICIÓN EN PARTES

En [Korf, 1985] se aborda el estudio de la complejidad computacional de distintos métodos que llevan a la descomposición de un objetivo global en distintos objetivos parciales. Sus estudios, basados en la aplicación de algoritmos de búsqueda conocidos, demuestran que la descomposición en partes del problema conduce a un solución más eficiente en términos de complejidad computacional, un criterio deseable para una buena descomposición en partes del problema.

La intuición nos lleva a utilizar un proceso de descomposición dirigido a objetivos. Este proceso, que se fundamenta en la consideración de que el objetivo global de un problema está formado por un conjunto de subobjetivos, lleva a dividir el problema en partes según un objetivo parcial a cumplirse.

Las partes en que se divide el problema vienen determinadas por las acciones y estados que las forman. Así, al agrupar estados y acciones fuertemente relacionadas, estamos obteniendo partes distintas de un mismo problema.

Como ejemplo, supongamos que se debe resolver la planificación de un robot móvil dedicado al transporte de piezas entre distintos lugares de una factoría. La intuición nos llevará a considerar 3 partes de este problema que podemos resolver por separado:

1. cargar el bloque del lugar en el que se encuentra
2. transportar el bloque al otro lugar
3. descargar el bloque

Esta división en partes viene determinada por el agrupamiento de estados y acciones fuertemente relacionadas (es habitual que sea bastante similar la planificación que se hace para una carga que la que se hace para una descarga de un objeto), tal y como se indica en la siguiente figura:

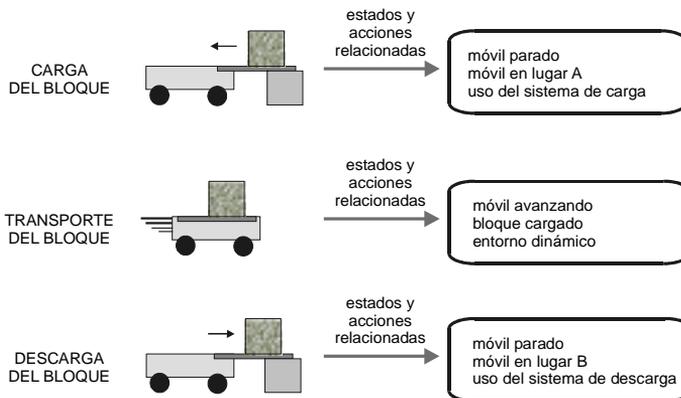


Figura 3-1. Ejemplo de descomposición en partes de un problema.

El problema se reduce a cómo reconocer los estados y acciones que son relevantes para cada parte.

Uno de los factores a tener en cuenta en el proceso de descomposición es la consideración de interacciones entre las partes obtenidas. Son las acciones las que determinan las interacciones entre las partes, cuando la aplicación de una acción nos lleva a estados que no pertenecen a un agrupamiento o parte, o se solapan con otras partes. En estos casos se utilizan procesos de aprendizaje basados en históricos para determinar o estimar posibles interacciones entre objetivos [Yang, 1997].

BÚSQUEDA Y SELECCIÓN DE LA SOLUCIÓN EN CADA PARTE

La resolución del problema en cada parte resultado de la descomposición del problema global atiende a la idiosincrasia del propio problema. Es en este punto donde se deben aplicar los métodos de planificación pertinentes según el problema planteado. Los métodos y técnicas referidos en el capítulo anterior suponen la fuente de referencia para abordar la planificación deseada.

Supongamos una tarea de ensamblaje de un robot articulado, para la que se han considerado las siguientes partes:

1. Situar el efector final del robot en la zona de trabajo en la que están las piezas a ensamblar.
2. Ensamblaje de las piezas.
3. Situar el efector final del robot en un punto de seguridad.

La división en partes atiende a criterios de estado espacial, secuenciación de acciones en el tiempo y operatividad del robot, tal y como se ha expuesto en el apartado anterior.

La planificación que se sigue para cada parte depende del objetivo parcial de cada una de ellas. Así, el objetivo de la primera y tercera parte es el de situar el robot en un punto del espacio, lo cual nos lleva a buscar un método de planificación de trayectorias que atenderá a la optimización del recorrido del efector final, atendiendo a posibles conflictos con el entorno. Considerando el entorno estático, se aplicará para la resolución de estas partes alguna técnica de planificación de movimientos [Latombe, 1991] [Mellado, 1996].

El objetivo planteado para la resolución de la segunda parte es el ensamblaje de las piezas. En este caso se tratan problemas relacionados con el manejo de las piezas y el posicionamiento espacial de las mismas de manera precisa. En estas situaciones se buscan métodos de planificación de las tareas a partir del uso de sensorización adecuada (por ejemplo, sensores de fuerza) que permita reaccionar ante estímulos externos como son el contacto entre las piezas o entre el efector final del robot y la pieza que se está manejando [Mason y Salisbury, 1985] [Chen y Brudick, 1993] [Khatib, 1994] [Sanz *et al.*, 1998].

RESOLUCIÓN DE CONFLICTOS ENTRE SOLUCIONES

La resolución separada de cada parte del problema puede presentar conflictos de interacción entre las partes. Este es un problema típico que se manifiesta en dos formas:

- Cuando la resolución de una parte presupone la disponibilidad de ciertos recursos al tiempo que son requeridos por otras partes.
- Cuando la solución de una parte afecte a cualquiera de las otras partes, variando cualquiera de sus estados.

La existencia de las dos situaciones anteriores dependerá, como resulta obvio, del proceso de separación en partes previo. En la medida que las partes del problema sean más independientes, las

soluciones a cada parte presentarán una menor interacción entre ellas. Los conflictos entre soluciones se asimilan a Problemas de Satisfacción de Restricciones (PSR) [Yang, 1997]. Un PSR se formula como:

1. un conjunto de variables
2. para cada variable, un conjunto de valores o dominio
3. una colección de restricciones relativas a los valores que pueden alcanzar las distintas variables

Las restricciones son esencialmente relaciones entre los dominios de las variables.

Un ejemplo que ilustra un PSR es el siguiente: supongamos que se tiene que pintar una pared que se ha dividido en tres regiones. Para ello se dispone de una serie de colores con los que se puede pintar cada región, y el resultado final debe asignar a cada región un color de manera que sea distinto del de la región adyacente.

Para este ejemplo, las variables son las tres regiones (R1, R2, R3), los dominios son las posibilidades de coloración de cada región y la restricción es el hecho de que cada región debe ser pintada tal que el color sea distinto al de la región adyacente.

La solución a un PSR se centra en la satisfacción de las restricciones que se plantean.

En realidad estos métodos se basan en el estudio de la estructura del plan y las interrelaciones entre operadores. De esta manera, una interrelación que presente una amenaza para el cumplimiento de la planificación en otras partes es la restricción que debe ser satisfecha.

Las soluciones a los conflictos pasan en la mayor parte de casos por un estudio de la estructura del plan considerando todas sus partes y por un correcto secuenciamiento de las resoluciones entre las partes, de forma que se eviten los problemas de compartición de recursos y la modificación de estados de otras partes.

El siguiente ejemplo ilustra muy bien el problema de los conflictos entre soluciones. Supongamos que se debe pintar una escalera de mano y el techo de una habitación. La intuición nos lleva a separar el problema en dos partes: pintar la escalera y pintar el techo. Pero los conflictos aparecen cuando se puede compartir un recurso para la resolución de ambas partes (la brocha) o cuando la ejecución de acciones en una parte pueda modificar condicionantes de la otra parte

(si se pinta la escalera, ésta estará húmeda y no se podrá utilizar para subirse y pintar el techo).

La siguiente figura muestra un grafo simplificado de las acciones, estados y condicionantes (restricciones) del problema global:

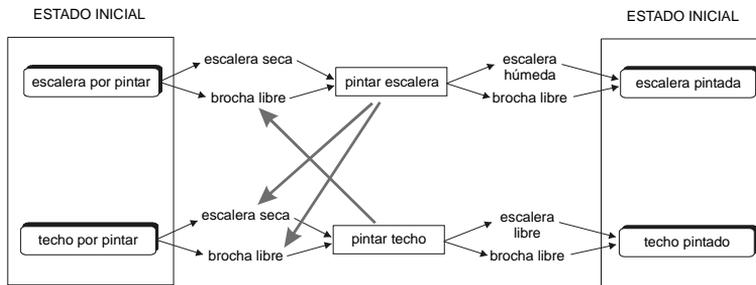


Figura 3-2. Grafo simplificado del problema del pintado de la escalera y el techo.

En el grafo se aprecian las interacciones (flechas gruesas) entre las acciones implicadas en la resolución de una parte y las condiciones que han de cumplirse para la resolución de la otra parte. También se aprecia la implicación de las acciones en los recursos necesarios (la brocha).

Debido a que la solución de ambas partes requiere el uso del mismo recurso, ambas partes deben ser resueltas secuencialmente. Por otra parte, el estudio de los condicionantes del estado final nos indica qué parte debe resolverse antes para evitar las interacciones al plantear una solución global. Después de pintar la escalera, ésta queda húmeda, con lo que se alteran las condiciones que permiten pintar el techo. Sin embargo, después de pintar el techo, la escalera y la brocha quedan libres para poder pintar la escalera.

La correcta secuenciación en el tiempo de las acciones permite evitar conflictos entre la resolución de las partes.

COMBINACIÓN DE SOLUCIONES

La combinación de las soluciones de cada parte de manera efectiva influye en la eficiencia de la solución global. Hay que recordar que la solución de una parte es el plan utilizado para llegar al objetivo parcial que representa a esa parte del problema global. En ese sentido, la combinación de soluciones significa la consideración de

interacciones entre los distintos planes que permiten resolver cada parte del problema. A través de las posibles interacciones entre planes se puede llegar a soluciones combinadas que permitan optimizar acciones que se pueden compartir.

Supongamos que debe realizarse el transporte de una pieza desde el lugar A al lugar A', y el transporte de otra pieza desde el lugar B al lugar B'. Este problema, abordado a partir de la descomposición en partes, nos lleva, por un lado, a resolver el transporte de una pieza de A a A' y el de otra pieza de B a B'. Abordado de forma separada, el problema de transporte se puede resolver utilizando dos robots móviles, uno para el transporte de cada pieza. La siguiente figura muestra un entorno ejemplo en el que aparecen los lugares marcados y la solución por separado (posibles recorridos de los móviles) de ambas tareas de transporte:

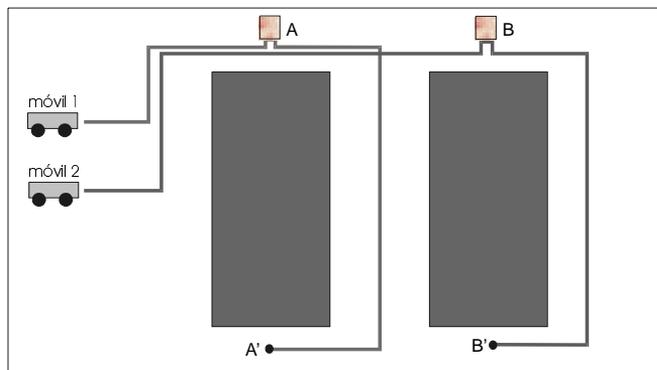


Figura 3-3. Ejemplo de transporte de dos piezas utilizando dos móviles diferentes.

Pero si se dispone de un robot en el que se puedan cargar las dos piezas, la tarea de transporte de los dos piezas se puede optimizar en cuanto al coste global de recursos. La siguiente figura muestra una posible solución a la tarea de transporte utilizando un móvil en el que se pueden cargar las dos piezas:

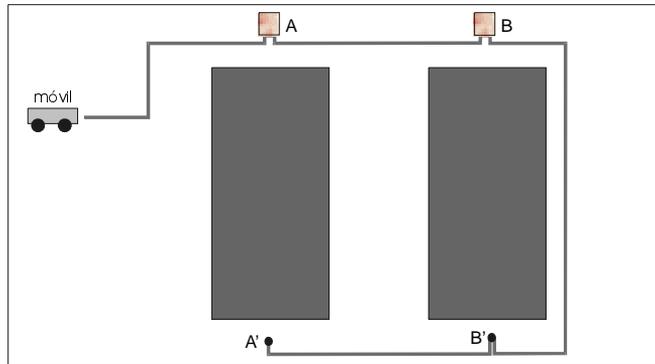


Figura 3-4. Optimización de la tarea de transporte utilizando un mismo móvil para el transporte de las dos piezas.

Como se aprecia en el ejemplo anterior, la optimización de la solución final combinando soluciones parciales se basa en la compartición de un recurso común (el móvil). Además, las propias tareas de transporte permiten esta combinación, puesto que los lugares desde donde deben transportarse las piezas están próximos y permiten un recorrido final óptimo.

De manera formal, la combinación de soluciones se describe como la posibilidad de agrupar un conjunto de pasos consecutivos de distintos planes de forma que el operador o paso que los sustituya consiga el mismo efecto final. Esto es,

Ecuación 3-3

$$\forall s_i, s_j \in \Sigma, \neg \exists s_k \in (\text{Pasos}(\Pi) - \Sigma): s_i \mapsto s_k \mapsto s_j$$

significando que dados dos pasos (s_i, s_j) cualesquiera de la combinación de pasos Σ dentro de un plan Π , no existe ningún paso entre ellos que no pertenezca también a dicha combinación.

Asociado a la combinación de soluciones, se tienen los conceptos de efectos útiles, $\text{Efectos}(\Sigma, \Pi)$, y de precondiciones de red, $\text{Prec-Red}(\Sigma, \Pi)$, de la agrupación [Yang, 1997]. Los efectos útiles hacen referencia a aquellos efectos que produce Σ y que suponen precondiciones para otros pasos dentro del plan Π . Por su parte, las precondiciones de red son las precondiciones de los pasos agrupados en Σ .

De manera gráfica, la siguiente figura muestra el posible agrupamiento de pasos entre las soluciones de dos partes de un mismo problema.

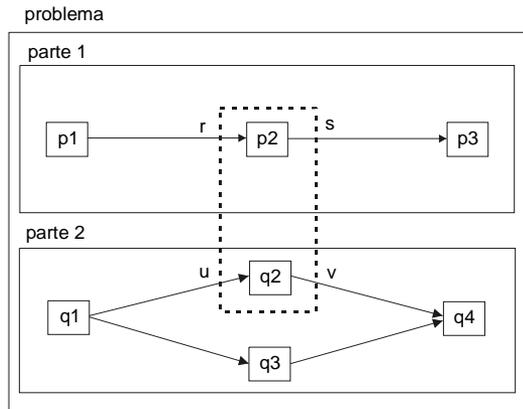


Figura 3-5. Posible agrupación de pasos entre soluciones de dos partes distintas de un mismo problema.

El ejemplo genérico anterior muestra un problema que se ha descompuesto en dos partes y que suponen por tanto dos subproblemas. El plan para resolver el problema global viene formado por los planes correspondientes a los dos subproblemas, $\Pi_1=(p1, p2, p3)$ y $\Pi_2=(q1, q2, q3, q4)$. En este ejemplo se muestra como resulta posible agrupar $\Sigma=\{p2,q2\}$, mientras que no sería posible el agrupamiento de $\{p1, q1, q4\}$.

Atendiendo a las definiciones anteriores, se tiene que $Efectos(\Sigma,\Pi)=\{s,v\}$, mientras que $Prec-Red(\Sigma,\Pi)=\{r,u\}$.

Así pues, el agrupamiento de pasos o combinación de soluciones de partes distintas es posible si se dan una serie de circunstancias que tienen que ver con la secuencialidad de los pasos a agrupar, la compartición de recursos entre soluciones y aspectos relativos al entorno, como la proximidad espacial de los lugares en los que se desarrollan las acciones.

ejemplo de combinación de soluciones

La tarea de transporte anterior sirve como ejemplo ilustrativo de cómo se pueden agrupar pasos entre distintas soluciones de las partes

de un problema global. La siguiente figura muestra un grafo de los pasos a seguir para la resolución del problema global a partir de su descomposición en partes.

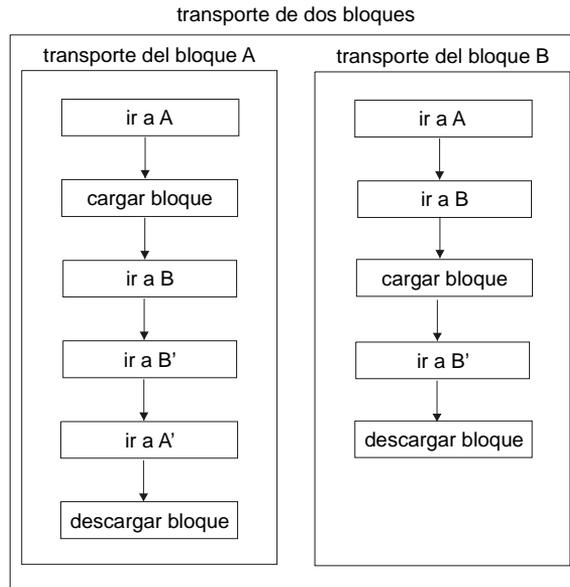


Figura 3-6. Grafo de los pasos a seguir para la solución del problema del transporte de dos bloques.

Aprovechando que existe un recurso que se puede compartir (un robot móvil capaz de transportar ambos bloques a la vez) y que las acciones se desarrollan en un espacio común (el camino que lleva hasta B pasa por A, y el camino que lleva hasta A' pasa por B'), se puede proceder a una combinación de ambas soluciones a partir del agrupamiento de acciones comunes.

Del grafo de acciones anterior se constata que los desplazamientos que son comunes a ambas soluciones se pueden agrupar. Este agrupamiento es posible porque, además de la compartición del recurso y del desarrollo espacial de las acciones (misma ruta de paso), las acciones se desarrollan de manera que existe una secuencialidad que se puede mantener con la agrupación. Todo ello nos lleva a una combinación de las soluciones que hace más efectiva la solución global, tal y como se muestra en la Figura 3-7.

Como se aprecia en dicha figura, en la combinación de soluciones propuesta, los desplazamientos del móvil que se utiliza para el

transporte se consideran acciones comunes al transporte de los dos bloques.

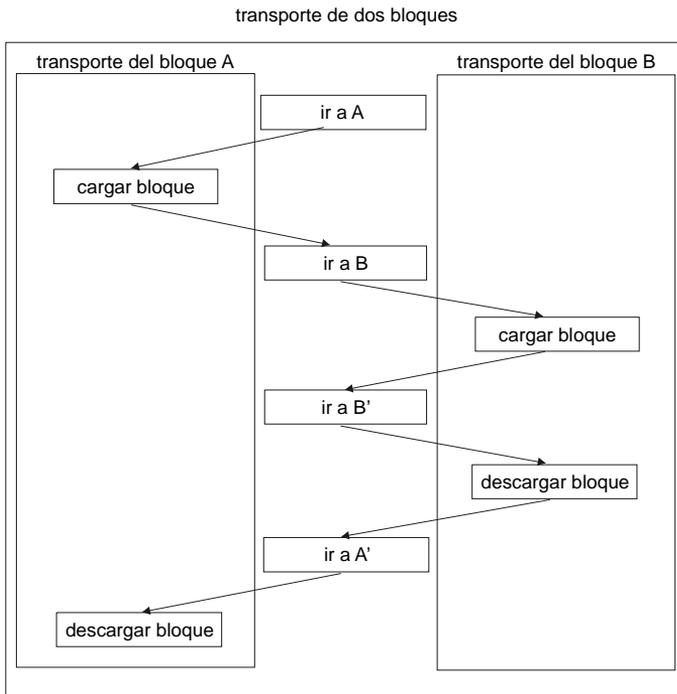


Figura 3-7. Combinación de las soluciones en el problema de planificación del transporte de dos bloques.

La combinación de soluciones y, por extensión, la descomposición en partes deben ceñirse a ciertos parámetros que permitan estimar lo correcto de su aplicación. Por lo que respecta a la descomposición en partes, se debe establecer un criterio que nos permita definir si el número de partes en que se divide el problema global es correcto, y si la propia división es correcta. En cuanto a la combinación de las soluciones de cada parte, habrá que medir si el agrupamiento de pasos entre planes es correcto o, por el contrario, introduce factores que puedan modificar la resolución del problema.

Por lo que respecta al proceso de descomposición en partes, tal y como ya se ha dicho anteriormente, la intención que subyace es la de considerar el conjunto de objetivos a conseguir al final del problema planteado. La consideración de cada uno de estos objetivos por separado puede suponer una parte determinada del problema global.

Pero estos objetivos parciales, como también se ha visto, se pueden agrupar para formar partes más grandes dentro de un problema global. Esta agrupación se hace a partir del estudio de las relaciones entre los objetivos parciales de un problema.

En [Yang, 1997] se plantea un algoritmo para generar partes de un problema determinado a partir del estudio de las interacciones entre los objetivos parciales de un problema global. Yang propone obtener las interacciones entre objetivos a partir de un histórico de planes anteriores. De esta manera, las anteriores interacciones entre dos objetivos contabilizadas en planes anteriores nos darán una idea aproximada de la relación entre ellos.

Finalmente, el algoritmo propone establecer un grafo con todos los objetivos, con pesos en los arcos que unen objetivos que representen las interacciones entre ellos. Un algoritmo posterior recorrerá el grafo buscando agrupar objetivos.

DESCOMPOSICIÓN EN ROBÓTICA

CRITERIOS PARA LA AGRUPACIÓN DE ESTADOS Y ACCIONES

Tal y como ya se ha definido anteriormente, la división en partes de un problema viene determinada por cómo se agrupan sus estados y acciones. En robótica también se sigue este principio de cara a conseguir una correcta descomposición en partes de una actividad determinada. A continuación se presentan distintos criterios que se utilizan para realizar esta agrupación:

- **La operatividad de los agentes que intervienen**

La operatividad que tienen los agentes que intervienen en la resolución de un problema es un criterio primario para establecer una división en partes de un problema. Las posibilidades en cuanto al movimiento y operaciones que pueden realizar los agentes son un indicativo de la existencia de subprocesos distintos dentro de un proceso global.

En el ejemplo expuesto en la Figura 3-1 (transporte de una pieza por parte de un robot móvil), el agente que debe realizar la tarea de transporte es un robot móvil cuya operatividad reside en la posibilidad de desplazamiento que tiene y la existencia de un dispositivo de carga y descarga acoplado. Cuando el robot está parado o cuando está en movimiento se

tienen situaciones distintas en la realización de un proceso global. La alternancia de desplazamientos del robot con paradas es un factor que nos sirve como referencia para dividir un proceso global en partes, al igual que el uso del dispositivo de carga y descarga.

- **Criterios espaciales**

Cuando se aborda una tarea cuya resolución implica el acceso a distintos lugares, resulta apropiado discretizar el espacio y hablar de zonas con características comunes. Se trata de un criterio inmediato al abordar el proceso de descomposición, en el que intervienen factores relacionados con el entorno en la medida en que éste afecta al agente que realiza la tarea.

Por ejemplo, cuando un robot móvil se mueve en una fábrica, si a lo largo de un pasillo no existe la posibilidad de que aparezcan otros móviles, este pasillo formará una zona que delimitará una parte del problema. La posibilidad de que aparezcan otros móviles en zonas distintas de la factoría nos lleva a considerar partes del problema distintas, con distinta idiosincrasia.

- **Criterios temporales**

Se trata de un criterio menos intuitivo pero igualmente útil para abordar una descomposición en partes del problema. Interviene en situaciones en las que la tarea encomendada abarca un tiempo de resolución determinado o está dentro de unos plazos lo suficientemente extensos que permiten pensar en una división temporal o secuenciamiento.

En el ejemplo del transporte de una pieza por parte de un robot móvil, la descomposición en partes propuesta también se ajusta al criterio temporal. Así, para la tarea de transporte objeto del ejemplo, será necesario cargar el bloque como paso previo al transporte, el desplazamiento entre los dos lugares sólo se puede hacer cuando el bloque ya está cargado y, por último, sólo se puede descargar el bloque cuando se ha completado el desplazamiento hasta el lugar de descarga. Como se ve, el criterio temporal es inherente a la división propuesta.

- **El alcance espacio-temporal de las acciones**

Es un criterio relacionado con las acciones que intervienen en el proceso de resolución global del problema. El cálculo de las acciones que se pueden agrupar bajo una misma parte tiene que ver con el objetivo parcial que delimita la parte que se está tratando. Así, las acciones serán relevantes a un objetivo parcial cuando bien permitan conseguir este objetivo parcial, o bien lleven a una precondition o estado previo intermedio a la consecución de ese objetivo.

FACTORES PARA LA DESCOMPOSICIÓN EN PARTES

Consideremos ahora un problema robotizado determinado (P). Este problema tiene asociado un coste temporal (T) que representa un tiempo máximo en el que dicho problema debe resolverse. Al abordar la resolución del problema a partir de la descomposición del mismo en partes, se llega a n posibles descomposiciones

Ecuación 3-4

$$D_1 = \{P_1^1, P_2^1, \dots, P_{m_1}^1\}$$

$$D_2 = \{P_1^2, P_2^2, \dots, P_{m_2}^2\}$$

$$D_3 = \{P_1^3, P_2^3, \dots, P_{m_3}^3\}$$

...

$$D_n = \{P_1^n, P_2^n, \dots, P_{m_n}^n\}$$

Estas descomposiciones son fruto de la consideración de los criterios de operatividad, consideraciones espaciales, temporales y de alcance espacio-temporal anteriormente descritos. El problema estriba en dirimir qué descomposición es la más adecuada para la resolución del problema original P.

Existen distintos factores que tienen una gran importancia a la hora de establecer la conveniencia de elegir una descomposición frente a otra. En concreto, los factores que se consideran para la descomposición de un problema de robot son el **coste temporal**, el **coste energético**, el **número de operaciones** y el **consumo de recursos**.

- **Coste temporal**

Es éste un criterio básico que conduce a la planificación a partir de la descomposición del problema. La resolución de un problema dado a partir de su descomposición en partes no debe suponer un mayor coste temporal.

En las actividades que realiza un robot industrial, el tiempo es un factor crítico que se debe respetar. Los problemas que debe resolver un robot siempre llevan asociado un tiempo de ejecución que se ajusta, por un lado a las necesidades de producción industrial, y por otro, a las propias prestaciones y características del problema a resolver.

El coste temporal asociado a una descomposición en partes de un problema viene representado por

Ecuación 3-5

$$t_i = \sum_{j=1}^m tP_j^i$$

donde t_i representa el coste temporal estimado para la descomposición i , y tP_j^i el coste asociado a la parte j de la descomposición i .

El coste temporal asociado a una parte se puede estimar a partir de las prestaciones del robot (velocidades, aceleraciones, etc.) y de los requisitos de la propia actividad a realizar (tipo de material a manipular, acciones a realizar, etc.). En muchas ocasiones, el mantenimiento de un histórico de las actividades del robot puede conducir a una correcta estimación del coste temporal de las acciones aisladas.

- **Coste energético**

Otro factor crítico es el coste de energía consumido en las acciones que realiza un robot. Por ejemplo, los robots móviles autónomos suelen llevar unas baterías que suministran la energía eléctrica necesaria para accionar los motores. Estas baterías deben cambiarse cuando se supera cierta reserva mínima. El cambio de baterías supone una actividad extra para el móvil que debe desplazarse a un lugar concreto en el que se realice. Es siempre conveniente que esta operación se pueda planificar, teniéndose en cuenta, para poder realizarla en aquellos momentos en que no interrumpan ninguna otra

actividad. Para poder planificar las actividades de recarga de baterías es necesario contemplar el coste energético de la realización de las actividades del robot, minimizándolo para realizar el menor número de recargas posible.

Habitualmente el coste temporal y el coste energético suelen estar asociados, aunque hay acciones que pueden suponer elevado coste temporal sin que tengan asociado un elevado coste energético. Es el caso de las acciones necesarias para manipular un objeto frágil, con un elevado coste temporal provocado por la precisión necesaria de las acciones, frente a desplazamientos de un robot móvil a lo largo de un corredor, con un elevado coste energético asociado.

El coste energético asociado a una descomposición en partes de un problema viene representado por

Ecuación 3-6

$$e_i = \sum_{j=1}^m eP_j^i$$

donde e_i es el coste energético estimado para la descomposición i y eP_j^i representa la estimación hecha para la parte j de esa descomposición.

Una manera de estimar el coste energético es a través de la clasificación del tipo de acciones que realiza el robot para completar una actividad. Esta clasificación debe contrastarse con las características de la actividad que debe realizar el robot y las prestaciones del mismo para deducir el consumo energético.

- **Número de operaciones**

El número de operaciones requerido para realizar una actividad de robot es otro de los factores a tener en cuenta de cara a considerar una descomposición en partes. Es conocido que uno de los criterios que se suelen seguir en la programación de un robot es el ahorro de acciones aisladas. De esta forma, si se pretende resolver un problema cualquiera, es necesario abordar un estudio previo que conduzca a ahorrar acciones de robot. La combinación de soluciones, antes mencionada, permite llegar a una situación en la que se optimiza el número de operaciones de robot

dentro de la resolución de un subobjetivo del problema global.

La estimación del número de operaciones necesarias para realizar una actividad concreta de robot se debe realizar a partir del conocimiento de acciones típicas que conlleven un número establecido de operaciones y de la concreción de las acciones del robot en el entorno de la tarea que está realizando. Así, para una actividad de cogida de una pieza, el robot debe realizar

1. una aproximación al lugar donde está la pieza
2. situarse para que la herramienta coja la pieza de manera adecuada
3. coger la pieza accionando la herramienta
4. retirarse al punto de aproximación

Estas 4 operaciones suponen una actividad típica de cogida de una pieza. Pero esta actividad típica debe concretarse para la tarea encomendada al robot. Así, por ejemplo, es posible que la pieza a coger tenga una geometría que requiera varios movimientos del robot hasta situar la herramienta en la localización adecuada para coger la pieza. Esta característica particular de la tarea incrementaría el número de operaciones estimadas para realizar la actividad.

El número de operaciones estimado para una descomposición en partes de un problema vendría definido por la siguiente expresión

Ecuación 3-7

$$o_i = \sum_{j=1}^m oP_j^i$$

siendo o_i el número de operaciones estimado para la descomposición i , y oP_j^i el número de operaciones estimado para una la parte j de esa descomposición.

- **Consumo de recursos**

Por último, el número de recursos consumidos en una actividad concreta de robot es también un factor que se debe considerar al seleccionar la resolución válida de un problema de robot. Así, en la gran mayoría de ocasiones resulta de gran

importancia que las operaciones necesarias para resolver un problema dado puedan ser resueltas a partir del uso de un menor número de recursos. Este es un factor que depende de la cantidad de recursos disponibles en cada momento y de una previsión de uso de los mismos para posteriores actividades que se puedan solapar en el tiempo.

La expresión que permite estimar el número de recursos a consumir en una descomposición en partes de un problema viene dada por

Ecuación 3-8

$$r_i = \sum_{j=1}^m rP_j^i$$

siendo r_i la estimación de recursos consumidos en la descomposición i , y rP_j^i los recursos consumidos en la parte j de esa descomposición.

El cálculo de rP_j^i se puede estimar a partir de las actividades concretas a realizar en cada parte (subobjetivos a alcanzar), de los recursos disponibles en el sistema en el instante de tiempo en que debe iniciarse la resolución del problema, de las posibilidades de solapamiento en el tiempo con otras actividades o de las propias posibilidades del robot que debe resolver la tarea. De hecho, la combinación de soluciones de cada parte permitirá optimizar los recursos agrupando actividades que pueda resolver un mismo robot en instantes consecutivos de tiempo, por ejemplo.

La combinación de estos cuatro factores permitirá encontrar un criterio objetivo que lleve a la selección de la descomposición en partes apropiada para la resolución del problema de robot planteado. Esta combinación de factores dependerá de las condiciones del propio problema a resolver, así como del entorno en el que se define el problema (posibilidad de actividades simultáneas en la factoría, disponibilidad de recursos, etc.).

De esta manera, se puede llegar a la definición de un índice de bondad que conduzca a elegir la mejor descomposición en partes de un problema

Ecuación 3-9

$$B_i = \alpha t_i + \beta e_i + \gamma o_i + \delta r_i$$

donde α , β , γ y δ son unos pesos que ponderan la importancia de los factores de coste temporal (t_i), coste energético (e_i), número de operaciones (o_i) y recursos requeridos (r_i) para la descomposición en partes.

Supongamos el ejemplo del transporte de dos bloques descrito con anterioridad en sus dos alternativas en la Figura 3-3 y en la Figura 3-4. Estas dos alternativas consideraban el transporte de cada bloque individualmente, utilizando dos robots móviles, o bien, transportar los dos bloques con el mismo robot, aprovechando que un único robot podía transportar ambos bloques. Esta disposición permitía llegar a dos posibles soluciones que consideraban distintas divisiones en partes del problema. La Figura 3-6 y la Figura 3-7 muestran estas dos divisiones. A continuación se analizan cada una de estas alternativas.

La primera alternativa divide el problema en dos partes: el transporte del bloque A con un móvil y el transporte del bloque B con otro móvil. Los distintos factores se pueden estimar de la siguiente manera:

t_1	Ambas partes pueden ser resueltas en paralelo, con lo que el coste temporal global de la solución es menor.
e_1	El consumo energético para ambas partes por separado (eP_1^1 y eP_2^1) resultará menor, aunque su suma conlleva un consumo energético mayor (se realizan operaciones repetidas).
o_1	Con el número de operaciones (oP_1^1 y oP_2^1) sucede lo mismo que con el consumo energético. Su consideración por separado es ventajosa, aunque no la consideración del total de operaciones empleadas.
r_1	El consumo de recursos (r_1) es mayor, utilizándose para resolver el problema dos robots móviles al tiempo, que además pueden tener problemas en cuanto al acceso a lugares comunes.

Tabla 3-1. Estimación de los factores que afectan al índice de bondad B_i para la primera alternativa del problema del transporte de los dos bloques.

En definitiva, esta alternativa resulta ventajosa en cuanto al coste temporal, mientras que en cuanto al número de recursos utilizados es peor.

La segunda alternativa mantiene el mismo número de partes aunque establece una combinación de operaciones a partir del uso de un

único robot para el transporte de ambos bloques. La estimación de los factores que afectan al índice de bondad se puede realizar de la siguiente manera:

t_1	Se resuelven ambas partes de manera combinada, realizando las operaciones de manera secuencial, con lo que el coste temporal global de la solución es mayor.
e_1	El consumo energético para ambas partes por separado (eP_1^1 y eP_2^1) es mayor, puesto que transportar un bloque supone al tiempo transportar el otro bloque, aunque el consumo energético global puede resultar menor (se ahorran operaciones).
o_1	Con el número de operaciones (oP_1^1 y oP_2^1) sucede lo mismo que con el consumo energético. Su consideración por separado es peor, aunque la consideración del total de operaciones empleadas es ventajosa.
r_1	El consumo de recursos (r_1) es menor, utilizándose para resolver el problema un único robot móvil, sin problemas en cuanto a accesos comunes.

Tabla 3-2. Estimación de los factores que afectan al índice de bondad B_i para la segunda alternativa del problema del transporte de los dos bloques.

Así, si se pretende primar el coste temporal frente al resto de factores, se debe utilizar un peso que lo pondere (α) mayor que los otros (β , γ , δ). Por ejemplo, considerar el factor tiempo en un 70% sobre el índice de bondad, y el resto de los factores representados en un 10% cada uno.

Ecuación 3-10

$$\alpha \gg \beta, \gamma, \delta \rightarrow (0.7 \gg 0.1)$$

Esta combinación de pesos aplicada al ejemplo anterior conlleva la selección de la primera alternativa.

Si, por el contrario, el consumo de recursos resulta un factor crítico en el proceso de producción, se utilizará un peso ponderado (δ) mayor frente al resto (α, β, γ). La consideración porcentual respecto del índice de bondad puede ser similar a la anterior.

Ecuación 3-11

$$\delta \gg \alpha, \beta, \gamma \rightarrow (0.7 \gg 0.1)$$

Esta combinación de pesos implicará, en este caso, la selección de la segunda alternativa

En otras ocasiones, el coste energético y el número de operaciones (factores habitualmente muy relacionados entre sí) pueden ser críticos. En este caso se utilizarán pesos ponderados mayores que los otros. Por ejemplo, el coste energético y el número de operaciones pueden representar cada uno un 40% del índice de bondad, mientras que el resto de factores pueden representar un 10%.

Ecuación 3-12

$$\beta, \gamma \gg \alpha, \delta \rightarrow (0.4 \gg 0.1)$$

En resumen, el índice de bondad introducido para estimar una descomposición en partes de un problema robotizado depende de unos pesos, que a su vez dependen de cada tarea en concreto. Un mismo robot puede realizar, así, una sucesión de tareas de manera que el peso de los factores que definen el índice de bondad puede variar dependiendo de las motivaciones de cada una de ellas.

ABSTRACCIÓN JERÁRQUICA

La abstracción es una intuición humana a priori a través de la cual se busca extraer los conceptos importantes del problema. Con ello se busca abordar estos puntos importantes en una primera instancia, de manera que permitan dirigir el proceso de resolución global. Abordar un proceso de planificación, como método de resolución de un problema, a partir de la abstracción jerárquica es una técnica comúnmente aceptada de cara a reducir el coste computacional. De hecho existen distintos sistemas de resolución de problemas basados en la planificación a partir de jerarquías [Sacerdoti, 1974] [Unruh y Rosenbloom, 1989] [Yang *et al.*, 1996].

La siguiente figura muestra el proceso de abstracción jerárquica, a partir de un ejemplo de tres niveles de abstracción.

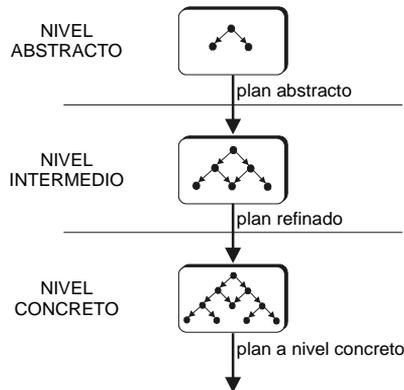


Figura 3-8. Posibles niveles de abstracción en un problema genérico.

El proceso de abstracción puede llevar a considerar desde un solo nivel a múltiples niveles. Es a través de posteriores refinamientos como se van encontrando distintos niveles más concretos hasta llegar a una solución final.

En un proceso de abstracción jerárquica de cara a solucionar un problema determinado se deben cumplir las siguientes propiedades básicas:

- Un nivel abstracto de un problema debe estar bien definido a partir de unas precondiciones que lo describan unívocamente.
- El establecimiento de sucesivos niveles de abstracción a partir del problema concreto se consigue eliminando y/o agrupando operadores y condiciones de niveles más concretos.
- Si se tiene un proceso de abstracción jerárquica cuya solución final al nivel concreto es S_0 , entonces existirá una secuencia de soluciones abstractas $\{S_{n-1}, S_{n-2}, \dots, S_0\}$ para sucesivos niveles de abstracción que conducen a la solución concreta S_0 . Cada solución S_{i-1} al nivel abstracto correspondiente es un refinamiento de la solución abstracta previa S_i .

Knoblock [Knoblock, 1994] establece un método para la generación automática de niveles de abstracción para un problema dado. El algoritmo que propone es independiente del dominio en el que se define el problema y las abstracciones satisfacen la propiedad de monotonicidad ordenada (*ordered monotonicity*), que garantiza que la estructura de abstracción generada para una solución no cambia en posteriores procesos de refinado.

En [Sacerdoti, 1974] se describe un método para generar niveles de abstracción a partir de lo que denomina *valores de criticalidad*. Estos valores de criticalidad vienen a ser un factor que indica la importancia o la prioridad de los factores o características que permiten describir un nivel. De hecho, el método consiste en asignar valores de criticalidad a las condiciones que describen los niveles. De esta manera, un nivel de abstracción i estará definido por condiciones cuya criticalidad sea igual o mayor que i . Con ello, los niveles de abstracción se van creando eliminando precondiciones de criticalidad menor.

Supongamos el problema de tener que ir de un lugar $p1$ a otro $p2$. Para este problema podemos establecer dos niveles de criticalidad en las condiciones que pueden definirlo, el primero relacionado con la ubicación espacial (hay que estar en $p1$ inicialmente) y otro que incorpore además la disponibilidad de un vehículo (coche o moto, por ejemplo) que permita realizar el desplazamiento. La siguiente tabla considera los niveles de criticalidad:

criticalidad	condiciones
1	estar-en($p1$)
0	estar-en($p1$), tener(coche), tener(moto)

Tabla 3-3. Niveles de criticalidad para el problema de ir de $p1$ a $p2$.

En base a estos niveles, se realiza el proceso de abstracción que muestra la siguiente figura:

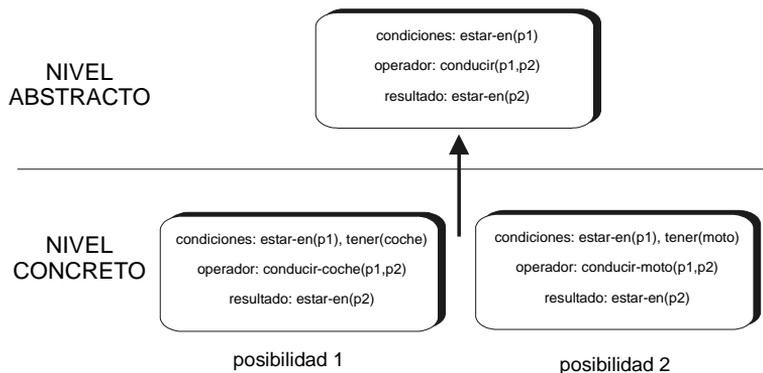


Figura 3-9. Abstracción jerárquica del problema de ir de $p1$ a $p2$.

El ejemplo anterior muestra sólo dos niveles de abstracción. Estos niveles cumplen las características expuestas con anterioridad:

- Los condicionantes definen los niveles de manera unívoca. El nivel concreto viene definido por condicionantes de situación espacial (hace falta estar en p1 para que se inicie la tarea) y de disponibilidad de recursos (hay que tener un vehículo que nos permita desplazarnos de lugar). Por su parte, el nivel abstracto viene descrito únicamente por el condicionante de la ubicación espacial.
- El nivel abstracto se consigue eliminando condicionantes del nivel concreto. Así, el condicionante del nivel abstracto acude sólo a la característica de situación espacial como requisito mayor para acometer la tarea.
- Los operadores de las posibilidades del nivel concreto son soluciones refinadas del operador propuesto en el nivel abstracto. La inclusión de la disponibilidad de vehículo para el desplazamiento significa profundizar más en una solución del problema.

ABSTRACCIÓN EN ROBÓTICA

Una de las carencias que aparecen al revisar la bibliografía y referencias de los distintos trabajos relacionados con la planificación en el ámbito de la robótica es la falta de unificación acerca de los conceptos de actividades y tareas de un robot. La mayoría de trabajos mencionan la planificación de tareas o misiones en robótica como un concepto con connotaciones de planificación a alto nivel, mientras que la planificación de trayectorias o caminos aparecen como conceptos de planificación a más bajo nivel.

Siendo esto así en la mayoría de situaciones, es frecuente encontrar referencias en las que un planificador de misiones o tareas resuelve todo el proceso, partiendo de la definición de los objetivos a alcanzar y devolviendo los puntos geométricos que el robot o el útil del robot debe alcanzar en sus movimientos para alcanzar un determinado objetivo.

En otras ocasiones, los esfuerzos de planificación llegan a conseguir estos mismos resultados pero parten de definiciones de más bajo nivel, en las que únicamente se busca un desplazamiento de un robot móvil a lo largo de un pasillo o el agarre de una pieza con

determinada geometría. En estos casos, la definición no atiende a los objetivos globales a resolver, centrándose únicamente en aspectos particulares a solucionar.

Esta tesis doctoral plantea una propuesta en la que se establecen definiciones de los objetivos a resolver a diferentes niveles, a partir de una abstracción jerárquica del problema de robot global.

Las propiedades mencionadas anteriormente, asociadas al proceso de abstracción jerárquica, llevan a proponer una descomposición del problema de planificación en robótica en los siguientes niveles ordenados de una mayor abstracción a una mayor concretitud:

- **Misiones**
- **Tareas**
- **Movimientos y Acciones**
- **Trayectorias**
- **Órdenes de robot**

Las misiones, que suponen la descripción de la actividad del robot al más alto nivel, vienen determinadas por un planificador de las actividades de la factoría (*scheduler*). Este planificador asigna misiones a los robots a partir de criterios de disponibilidad, ubicación o proximidad espacial y características del robot.

Por debajo, las acciones se planifican de manera inmediata en órdenes propias de robot, mientras que las trayectorias se planifican para obtener los puntos geométricos exactos por los que el robot pasa en sus desplazamientos. Estos desplazamientos son traducibles de manera inmediata en órdenes de robot.

En la siguiente figura se muestra un esquema de los niveles de abstracción propuestos.

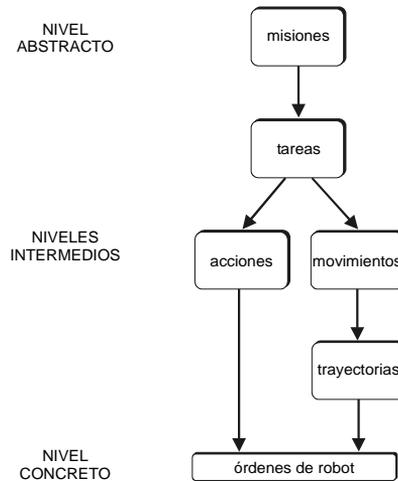


Figura 3-10. Niveles de abstracción en robótica.

Dos son los condicionantes que se consideran básicos para la definición de estos niveles de abstracción:

1. el *entorno* en el que se resuelve el problema.
2. los *agentes* que deben/pueden resolver el problema.

La dependencia o independencia de estos factores condicionará el proceso de planificación en robótica, de manera que según el nivel de dependencia respecto de ellos llevará a establecer los niveles de abstracción propuestos. La siguiente tabla muestra la subdivisión en niveles y cómo se ven afectados por estos dos factores.

niveles de abstracción	condicionantes
misiones	independencia del entorno independencia de los agentes
tareas	dependencia del entorno independencia de los agentes
acciones	dependencia del entorno dependencia de los agentes
movimientos y trayectorias	dependencia del entorno dependencia de los agentes

Tabla 3-4. Niveles de abstracción en la planificación en robótica.

El proceso de planificación se abordará de manera independiente en cada uno de estos niveles.

MISIONES

Una **misión** es la descripción de una operación al más alto nivel de abstracción, asumiéndose que esta descripción conlleva la definición de la operación en su forma más global. La misión es independiente del entorno en el que se debe realizar y del agente que la debe ejecutar.

La definición de la misión

llevar el libro del despacho C-3 al laboratorio de robótica

es una definición en la que no intervienen aspectos del entorno en el que se debe llevar a cabo ni características del agente que la deba desempeñar. La ubicación del despacho C-3 y del laboratorio de robótica no importa para definir esta misión, aunque sí es importante considerar aspectos básicos como que se pueda acceder al despacho C-3, o que este despacho y el laboratorio de robótica estén conectados (se pueda ir de un sitio a otro). Tampoco importa el tipo de transporte que se vaya a utilizar, sea este un vehículo filoguiado, autoguiado, por ruedas, o con patas.

La planificación de misiones lleva a una descomposición en **tareas**, actividades de más bajo nivel en cuya definición sí aparecen aspectos del entorno en el que deben desarrollarse, aunque siguen sin considerar las características del agente que deba realizarlas.

La siguiente figura muestra un entorno global en el que se desempeña la misión descrita anteriormente:

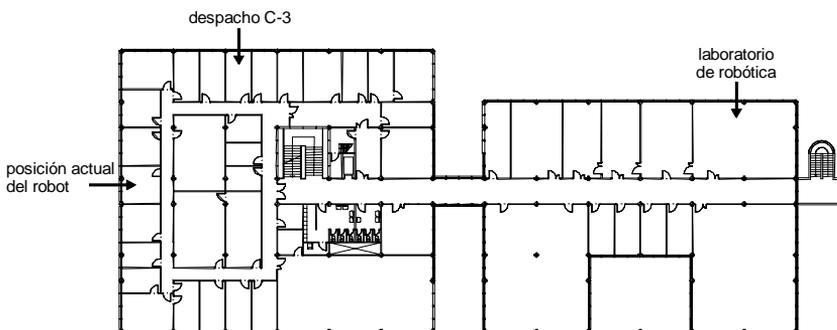


Figura 3-11. Entorno global en el que se desempeña la misión llevar el libro del despacho C-13 al laboratorio de robótica.

La misión se descompone en las siguientes tareas a partir de las características del entorno global:

- salir del despacho en el que está el robot
- ir al despacho C-13
- coger el libro
- salir del despacho C-13
- ir hasta el pasillo principal
- ir al laboratorio de robótica
- dejar el libro en su lugar

Estas tareas se desprenden de la ubicación física en el entorno global del robot, del despacho y del laboratorio (marcados en la figura).

Planificación de Misiones

Tradicionalmente, la planificación de misiones, en las que las actividades de robot vienen descritas al más alto nivel, orientadas al objetivo a conseguir, se enmarcan dentro de los **métodos de planificación basados en Inteligencia Artificial** [Allen *et al.*, 1990]. De esta manera, planificadores expuestos en el capítulo anterior como STRIPS [Fikes y Nilsson, 1971] se muestran apropiados para establecer secuencias de pasos para conseguir el objetivo expuesto en la misión. Habitualmente los planificadores basados en IA suelen depender del dominio y recurren a métodos de búsqueda local que permitan simplificar la planificación [Ambite y Knoblock, 1997] o a reglas heurísticas para la búsqueda de la secuencia óptima de tareas que conduzcan a resolver la misión [Gerevini y Serina, 1999].

En [Brummitt y Stentz, 1996] se plantea la planificación dinámica de varios robots móviles mediante el uso de una gramática adecuada que permite definir expresiones para representar las misiones de robot. El estudio de estas expresiones permite establecer la planificación, considerando un problema en el que se tienen N robots y M objetivos a satisfacer (misiones) por cualquiera de los robots.

Un método que resulta especialmente adecuado para la planificación a alto nivel de misiones es el uso de **sistemas expertos**. Los sistemas expertos, y en concreto los sistemas basados en el conocimiento, permiten tratar de manera eficiente el aspecto temporal para establecer la planificación de las acciones que permiten llevar a cabo una misión [Laffey *et al.*, 1988] [Perkins y Austin, 1990].

En relación con los sistemas basados en el conocimiento aplicados a entornos de tiempo real, los **modelos de pizarra (blackboard)** se muestran como una alternativa ideal para tratar problemas de planificación de elevada complejidad en los que intervienen diferentes recursos [Englemore y Morgan, 1988] [Hsieh y Archibald, 1996] [Xu y Van Brussel, 1997].

En [Mensch *et al.*, 1993] se introduce REAKT, un sistema experto basado en un modelo de pizarra en tiempo real con unas características que lo hace idóneo para la aplicación en tareas de robot. REAKT hace uso de la lógica temporal para establecer predicciones en tiempo real de las acciones del sistema [Crespo *et al.*, 1994]. En un apartado posterior de este capítulo se detalla el uso de este sistema en el desarrollo de un sistema de control para un vehículo autoguiado.

Uno de los métodos que resultan convenientes para la planificación de las misiones en tareas es el uso de **grafos de alto nivel**. Este método se basa en establecer un grafo con los aspectos más relevantes de la misión (subobjetivos a cumplir, estados del problema, acciones relevantes) y aplicar un algoritmo de búsqueda que permita llegar a la secuencia de tareas que permita ejecutar la misión de manera eficiente. La ventaja que se plantea es la de generar un grafo reducido del problema. Esto es posible precisamente por la generación de un nivel de abstracción alto del problema que permite fijarse en los aspectos más relevantes del mismo. Una búsqueda en el grafo conduce a encontrar la secuencia eficiente de tareas que permite cumplir la misión de manera eficiente.

En [Mellado *et al.*, 1998], y más adelante, en un apartado posterior de este capítulo, se muestra la combinación del uso de un sistema basado en modelos de pizarra con la existencia de un grafo de alto nivel para la planificación de misiones de un robot móvil.

TAREAS

Las tareas suponen un nivel intermedio de abstracción, en cuya definición, como se ha comentado, aparece la influencia del entorno en que se deben desempeñar, aunque no de los agentes que deben realizarlas.

La planificación de tareas lleva a la descomposición en **acciones y movimientos**, actividades ambas descritas a un nivel de mayor concreción. Las acciones son actividades que realiza el agente sin que

supongan para él ningún desplazamiento. Así, coger un objeto o realizar una operación de soldadura suponen la actuación de la herramienta del robot pero no un desplazamiento en el entorno de trabajo. Por su parte, los movimientos suponen todo lo contrario, es decir, desplazamientos para llegar a nuevas localizaciones dentro del entorno de trabajo.

Siguiendo con el ejemplo anterior, la tarea ir al despacho C-3 se podrá descomponer en movimientos a un nivel más concreto. En este caso, no existen acciones asociadas a la tarea, sólo movimientos.

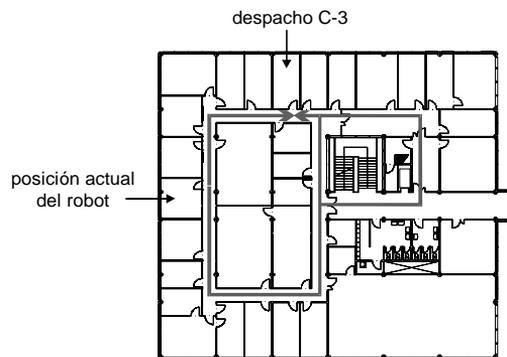


Figura 3-12. Trayectorias posibles para ir al despacho C-3.

La figura anterior muestra las rutas posibles para ir hasta el despacho C-3. La planificación de tareas deberá decidir qué ruta se sigue a partir de criterios relacionados con la optimización de las tareas o restricciones respecto de puntos de paso (dimensiones del móvil, pasillos cortados por eventualidades, etc.).

Sin embargo, la tarea coger el libro se descompondrá en un movimiento y una acción:

- desplazarse hasta donde está el libro
- coger el libro con la herramienta prensil

Planificación de Tareas

Partiendo de la base de que las tareas suponen una descripción más detallada de un objetivo parcial a cumplir dentro de la misión global, la planificación de las mismas permite aplicar en la práctica cualquier técnica de las que se emplean en la planificación de misiones. De

hecho, la mayoría de referencias de planificación en robótica confunden ambas acepciones (misión y tarea) y aplican metodologías similares de manera indistinta.

Dentro de la propuesta de abstracción en distintos niveles que aquí se detalla, los métodos basados en **búsqueda en grafos** son los preferidos. Así, la propuesta pasa por la generación de un grafo más detallado que permita relacionar características inherentes al subobjetivo a cumplir. Estados y acciones relevantes dentro de este subobjetivo son los elementos que configurarán los grafos a este nivel, permitiendo extraer la secuencia de movimientos y acciones correcta para la llegar al subobjetivo que expresa la tarea.

La coexistencia de grafos de alto nivel para la planificación de misiones y de grafos más detallados para la planificación de tareas permite, además, tener una metodología unificada de planificación a distintos niveles. En un apartado posterior de este capítulo se detalla el uso de esta metodología para la planificación de las tareas de un robot móvil.

Las **Redes de Petri (RdP)** suponen una alternativa al uso de grafos (tal y como ya se ha referenciado en un apartado del anterior capítulo). Su principal ventaja estriba en las mejores posibilidades de análisis existentes a través de los distintos tipos y características de las RdP. Uno de los trabajos de investigación en esta tesis doctoral ha sido el de validar el uso de las RdP en el ámbito de la planificación básica de tareas. A continuación se detalla de manera más pormenorizada esta experiencia.

REDES DE PETRI Y DESCOMPOSICIÓN CELULAR

A lo largo del desarrollo de esta tesis doctoral se ha implementado como un trabajo complementario una aplicación software que aborda la planificación de las tareas de un móvil A en un entorno en el que también se mueve otro móvil B. Esta utilidad aplica una planificación basada en la generación automática de la RdP asociada al entorno en el que se mueven ambos móviles.

Una manera de aplicar las RdP para la planificación de tareas es estableciendo zonas del entorno en el que se desarrolla la tarea a partir de un método de descomposición celular [Latombe, 1991]. De esta manera, las zonas del entorno permiten establecer movimientos de paso de una a otra. Bajo este prisma, los lugares coincidirán con cada zona del entorno fruto de la descomposición celular, mientras que las transiciones suponen la acción de pasar de una zona a otra.

Las transiciones llevan asociadas condiciones para establecer la posibilidad de acceso a una zona u otra.

La aplicación desarrollada combina el uso de las RdP con la descomposición celular del entorno, realizando el cálculo de la trayectoria óptima para que el móvil A se desplace desde un lugar de la RdP (origen) a otro lugar (destino), teniendo en cuenta los pesos asociados a las transiciones entre lugares. De esta manera, es posible controlar situaciones en las que un pasillo está bloqueado por alguna eventualidad, asociándole a la transición que permite pasar al lugar que representa este pasillo un peso muy alto. La simulación de los movimientos del móvil B la controla el usuario, de manera que el móvil A va replanificando sus movimientos teniendo en cuenta en cada momento la posición de aquel.

El proceso de planificación y replanificación se realiza mediante la ejecución de un algoritmo básico conocido de búsqueda del camino más corto [Floyd, 1962], que se aplica sobre la representación matricial de la RdP.

La combinación de las RdP con la descomposición celular se puede ver sobre el ejemplo para el cálculo de las trayectorias de la tarea ir al despacho C-13. La siguiente figura muestra una descomposición celular posible:

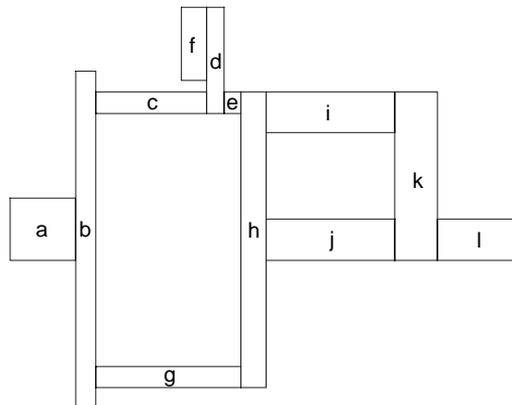


Figura 3-13. Descomposición celular sobre la topología de entorno para la tarea ir al despacho C-13.

La figura anterior muestra una descomposición celular que tiene en cuenta únicamente los pasillos que puede utilizar el móvil en sus desplazamientos por el entorno, el despacho en el que se encuentra y el despacho destino C-13.

La Red de Petri asociada se muestra en la siguiente figura:

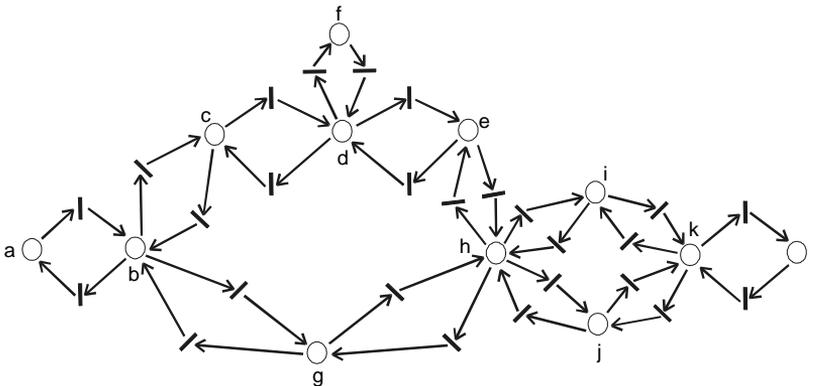


Tabla 3-5. Red de Petri asociada a la descomposición celular para la topología de la tarea ir al despacho C-13.

Un análisis simple de la RdP nos devolverá que las trayectorias posibles para resolver esta tarea. Así, teniendo en cuenta la topología del entorno (despachos, pasillos, puertas) y sin considerar aquellas que impliquen la existencia de bucles en el recorrido, las trayectorias posibles se corresponden con las siguientes secuencias de lugares:

- a-b-c-d-f
- a-b-g-h-e-d-f
- a-b-g-h-j-k-i-h-e-d-f

De las que la primera secuencia resulta ser la más eficiente teniendo en cuenta el tamaño de las celdas resultado de la descomposición celular.

Una vez establecida esta planificación de la trayectoria, la aplicación permite la simulación de la misma, teniendo en cuenta los lugares que va alcanzando el móvil B y replanificando en cada momento la mejor trayectoria para que el móvil A alcance el destino.

ACCIONES

Las acciones definen actividades del robot para las que éste no necesita hacer ningún desplazamiento, bien del útil, para el caso de los robots fijos, o del propio robot, en el caso de los robots móviles. Así, una acción definirá actividades como la cogida o dejada de un objeto, la aplicación de puntos de soldadura, o la carga y descarga de objetos, para lo cual sólo hace falta accionar el útil empleado.

A partir de esta definición, las acciones se pueden asociar directamente a ordenes propias de robot, siendo su planificación inmediata.

MOVIMIENTOS

Anteriormente se ha mencionado que las tareas se dividen en movimientos y acciones. Mientras que las acciones están claras y bien definidas, los movimientos se deben planificar y descomponer a un nivel más concreto para llegar a lo que se conoce como trayectorias. Así, en este nivel de debe clarificar el desplazamiento que debe seguir el útil del robot, en el caso de robots articulados, o el propio robot, en el caso de robots móviles.

Estos desplazamientos configuran una serie de trayectorias a seguir que, obviamente, dependerán de la disposición espacial del entorno (mapa local) y de las posibilidades de movimiento del agente que los va a realizar.

En la propuesta que se presenta en esta tesis doctoral, la planificación de movimientos conduce a la generación de las trayectorias de robot. En realidad una de las características habituales es la de confundir o asimilar la planificación de movimientos con la planificación de trayectorias. Este trabajo pretende establecer una diferencia entre movimientos y trayectorias.

Ambos conceptos vienen definidos a partir de las características del entorno en el que se deben desarrollar y de las posibilidades del agente que los debe desempeñar. Pero mientras que un movimiento define un desplazamiento completo, la trayectoria especifica este desplazamiento definiendo los puntos del espacio por los que debe pasar el robot. Esta es la diferencia fundamental entre ambos conceptos.

Planificación de Trayectorias

Siendo una trayectoria una secuencia de puntos del espacio a alcanzar en un tiempo dado por el robot, la planificación de las trayectorias no será más que la conversión de esta secuencia en un conjunto de órdenes de robot que permitan alcanzar cada punto de la secuencia.

De hecho, la planificación de trayectorias, entendida bajo este prisma, representa la codificación de un programa de robot en un lenguaje concreto, en lo que a los movimientos del robot se refiere.

REPLANIFICACIÓN EN LA ABSTRACCIÓN JERÁRQUICA

El hecho de que se plantee abordar la planificación de una actividad completa de robot a partir de una abstracción en niveles no significa que la resolución sea unidireccional. De hecho, la planificación de cada uno de los niveles no es cerrada en sí misma y puede verse afectada por realimentaciones de niveles inferiores, configurándose un proceso de replanificación.

El planteamiento inicial es *a priori*, con lo que se busca una solución al problema de partida, pero en la medida en que los procesos de planificación de un cierto nivel no se puedan cumplir, habrá que replantear la planificación del nivel superior. Así, se conforma un esquema de abstracción en niveles con realimentación de niveles inferiores. Esta replanificación surgirá desde el nivel inferior (trayectorias y acciones), generando replanificaciones a niveles superiores.

El proceso de replanificación de una actividad de robot siempre debe abordarse al nivel más bajo, en este caso las trayectorias. De esta manera, la planificación global de la actividad de robot no cambia con la excepción del proceso de replanificación en tiempo real de la trayectoria final. Sólo en casos excepcionales y situaciones críticas se plantea el proceso de realimentación con replanificación en niveles superiores, llegando a cambiar la actividad original del robot.

Replanificación de Trayectorias

Cuando una trayectoria no se puede ejecutar es porque se dan situaciones imprevistas que tienen que ver con cambios en la geometría local del entorno en el que se ejecuta la trayectoria o imprevistos relacionados con el propio robot. Los cambios que

afectan a la geometría local del entorno están relacionados con la aparición de posibles obstáculos en la ruta del móvil o de bloqueos temporales en algún pasillo de la factoría, como es el caso de los entornos dinámicos.

La planificación dinámica de movimientos requiere del factor *tiempo* y establece el movimiento de un punto (el robot) entre obstáculos estacionarios en una configuración espacio-tiempo [Latombe, 1991]. Ante este planteamiento, no todos los métodos de planificación de movimientos tradicionales son válidos, debiéndose abordar modificaciones en cualquier caso que tengan en cuenta el factor tiempo.

Cuando la trayectoria inicial prefijada no se puede ejecutar, se debe abordar una replanificación de la misma *on-line*. La replanificación de una trayectoria en tiempo de ejecución se puede abordar mediante arquitecturas de planificación que incluya comportamientos reactivos [Simó, 1997]. En un apartado posterior de este capítulo se menciona la utilización de un nivel de comportamiento reactivo para replanificar un trayectoria en tiempo de ejecución de cara a evitar obstáculos imprevistos [Mellado *et al.*, 1998].

En esta tesis doctoral se plantea la alternativa de utilizar intuiciones humanas, (deducción y predicción) para establecer un conjunto de reglas basadas en la lógica temporal que permitan la replanificación de una trayectoria en tiempo de ejecución. Esta alternativa se detalla en los capítulos posteriores a este.

Replanificación de los Niveles Superiores

Cuando la replanificación en tiempo de ejecución tampoco permite completar una trayectoria, es necesaria la realimentación con el nivel superior, necesiándose abordar de nuevo la planificación de movimientos a partir de los nuevos condicionantes que han surgido.

- *Replanificación de movimientos*: Si la replanificación de trayectorias, a partir de la deducción y predicción, no puede solucionar un problema de ejecución de la trayectoria en curso, se debe plantear una replanificación de los movimientos especificados para llevar a cabo una tarea. Este proceso significará elegir una solución alternativa a la planificación de movimientos previa, generándose una nueva trayectoria para cumplir un movimiento concreto.

- *Replanificación de Acciones:* Se trata de una situación más rara, aunque posible. Cuando una acción (por ejemplo, la carga o la descarga de un objeto) no se puede llevar a cabo, se debe replantear la secuencia de acciones y movimientos que resuelven una tarea y buscar una solución alternativa. En este caso, la imposibilidad de llevar a cabo una acción dependerá de un factor que, con toda seguridad, modificará las condiciones que conllevan la planificación de una tarea. Esto es así porque cuando una acción no se puede llevar a cabo, la razón hay que buscarla en un cambio en la posición de los objetos a coger, en el entorno en el que hay que cogerlos o por causas relacionadas con el utillaje del robot.
- *Replanificación de Tareas:* Es un proceso que se aborda cuando la replanificación de movimientos no soluciona un problema en la ejecución de una actividad de robot. Debido a que las trayectorias y los movimientos están muy ligados (el movimiento “ir hasta el final del pasillo” generará siempre una trayectoria a lo largo del pasillo, que de no poderse ejecutar implicará, en la mayoría de ocasiones, la selección de un movimiento alternativo, no de una trayectoria alternativa), cuando una trayectoria no se pueda ejecutar, se procederá a un proceso de replanificación de tareas.

Para ello, se deberá seleccionar una secuencias de acciones y movimientos alternativa a la original y que también conduzca a la solución de la tarea en curso. En muchos casos, la solución alternativa puede estar calculada de forma previa, con lo que la replanificación no es más que seleccionarla. Pero en otras ocasiones, la imposibilidad de ejecutar una trayectoria llevará a modificar las condiciones que permiten planificar una tarea (cambios en la geometría del entorno que conllevan una nueva topología).

- *Replanificación de Misiones:* La replanificación de misiones supone buscar una nueva secuencia de tareas que permitan llevar a cabo la misión de robot especificada. Al igual que sucede con la replanificación de una tarea, la replanificación puede consistir en la elección de una alternativa que está calculada de forma previa, o por el contrario, habrá que cambiar los condicionantes que se siguen para la planificación de misiones. En este último caso, se abordan cambios importantes en la topología del entorno o bien en las características del propio móvil, incluyendo la no disponibilidad del mismo.

En cualquier caso, la replanificación de una actividad de robot debe emprenderse desde la replanificación de las trayectorias que se ejecutan en tiempo real. Este proceso se aborda en esta tesis doctoral a partir de las intuiciones de deducción y predicción y se describe en los capítulos posteriores.

COMBINACIÓN DE LA DESCOMPOSICIÓN EN PARTES Y DE LA ABSTRACCIÓN EN NIVELES

La descomposición en partes del problema y la consideración de distintos niveles de abstracción son conceptos compatibles entre sí. No solamente compatibles, sino deseables a la hora de abordar la resolución de un problema general, y en particular, las actividades que debe desempeñar un sistema robotizado.

Mientras que la descomposición en partes considera una división del problema en vistas a la resolución de objetivos parciales que lleven a la resolución global, la abstracción jerárquica implica una división en niveles de un problema, sea parcial o global. De alguna manera, la división en partes supone una división vertical, mientras que los niveles de abstracción implican una división horizontal del problema.

En la Figura 3-16 se describe un esquema que combina ambos procesos de planificación.

En los apartados posteriores de este mismo capítulo se muestran distintos ejemplos de la planificación de actividades de un robot a partir de la combinación de la descomposición del problema y la aplicación de una abstracción jerárquica. Así, se muestran un ejemplo genérico para una actividad de un robot articulado y otro para un robot móvil. Por último, se muestra la aplicación del método en un caso real.

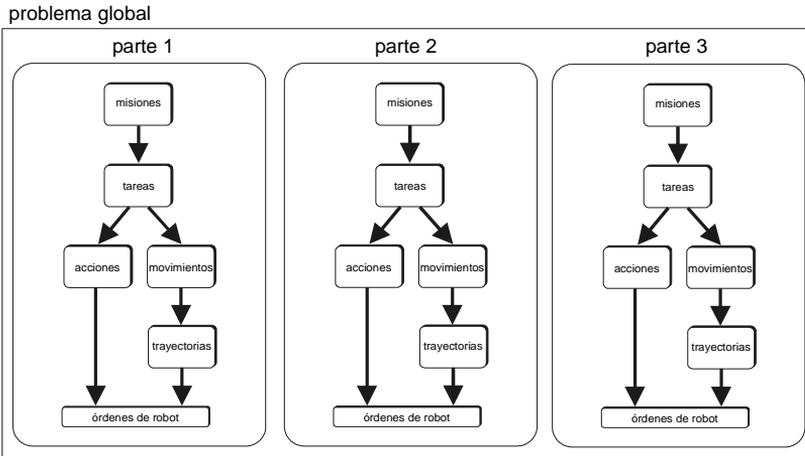


Figura 3-16. Esquema de la combinación de la división en partes y abstracción jerárquica en la resolución de un problema para un sistema robotizado.

EJEMPLO DE APLICACIÓN A UN ROBOT ARTICULADO

El proceso de descomposición y abstracción es aplicable a la planificación de un trabajo asignado a un robot articulado, es decir, un brazo-robot con un cierto número de grados de libertad, situado en un lugar fijo de la factoría y destinado a hacer operaciones (soldadura, manejo de materiales, etc.) en un entorno concreto y limitado.

Consideremos el siguiente ejemplo:

Un robot articulado debe ensamblar dos piezas y posteriormente colocar el objeto resultante en una caja

En la siguiente figura se presenta un entorno ejemplo en el que se encuentra el robot articulado para desempeñar la actividad propuesta, así como la ubicación espacial de las piezas a ensamblar y la caja en la que hay que colocar el objeto final. De igual manera, la figura muestra el recorrido que deberá realizar el útil del robot articulado para resolver el proceso de ensamblaje en una mesa y del posterior embalaje en la otra mesa.

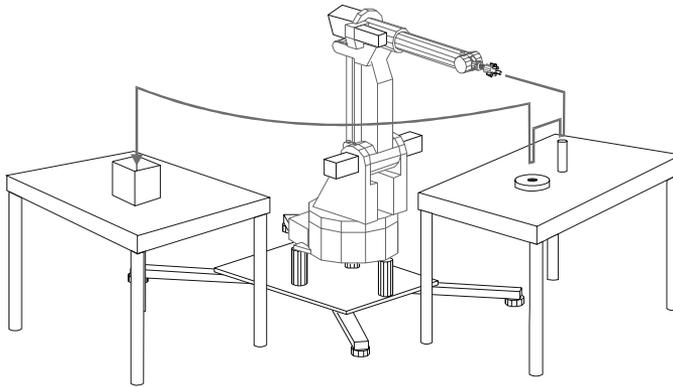


Figura 3-17. Planteamiento del problema de ensamblaje de dos piezas y posterior embalaje del objeto resultante.

Descomposición en Partes del Problema

Los criterios que se deben seguir para establecer distintas partes para abordar un problema dado son:

- La operatividad de los agentes que intervienen
- Criterios espaciales
- Criterios temporales
- Alcance espacio-temporal de las acciones

Estos criterios se pueden aplicar al considerar el proceso ejemplo que nos ocupa de la siguiente manera:

1. Operatividad del robot articulado

El robot articulado se utiliza en este proceso para tareas básicas de manipulación de piezas. Su operatividad reside, pues, en el uso de una herramienta tipo pinza adecuada que permita coger las piezas que se deben ensamblar y posteriormente colocar en la caja. Tanto en la fase de ensamblaje como en la de embalaje, el robot deberá manejar el mismo tipo de piezas, por lo que la operatividad del mismo no es un factor determinante, en este caso, para establecer partes distintas.

2. *Distribución espacial*

La distribución del entorno del robot establece el ensamblaje de las piezas en una mesa y el embalaje en la otra mesa. Esta disposición es típica de situaciones industriales reales, en las que se permite alimentar o retirar piezas mientras el robot realiza uno de los procesos, sin necesidad de detenerlo.

En este caso, realización del ensamblaje y el embalaje en lugares distintos permiten establecerlos como partes distintas del problema.

3. *Distribución temporal*

El ejemplo que estamos tratando implica que el robot articulado debe realizar en primer lugar el proceso de ensamblaje de las piezas, para posteriormente colocar en la caja el objeto resultado del ensamblaje. Se trata de dos procesos que se deben resolver de manera secuencial en el tiempo, por lo que son susceptibles de formar partes separadas del proceso global.

4. *Alcance espacio-temporal de las acciones*

Este criterio alude a la consecución de objetivos parciales de un mismo proceso global. En este caso, el ensamblaje y el embalaje son procesos típicos en las labores que desempeña un sistema robotizado en la industria, con idiosincrasia y problemática propia.

Por lo que respecta al ensamblaje, se deben resolver factores relativos al agarre de las piezas, evitación de colisiones en los movimientos cercanos al resto de piezas o posicionamiento preciso en el espacio del útil del robot. En cuanto al proceso de embalaje, su principal característica es la del manejo de un objeto mayor para colocarlo en el interior de una caja, por lo que también aparecen factores a resolver relativos al posicionamiento preciso del útil en el espacio y evitación de colisiones con la propia caja.

La resolución de estas acciones permite conseguir los objetivos parciales que plantean los procesos de ensamblaje y embalaje por separado.

Vemos pues que la consideración de estos criterios ante el ejemplo que nos ocupa nos lleva a distinguir dos partes claras en este problema: *el proceso de ensamblaje y el proceso de embalaje*

Cada parte, entonces, se aborda por separado, planificando la resolución del problema en procesos separados. El siguiente esquema muestra la resolución de la actividad planteada a partir de la resolución de cada una de las partes:

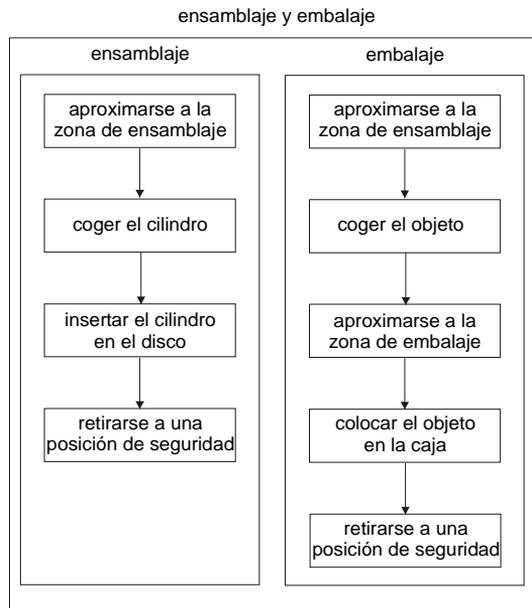


Figura 3-18. Esquema de la descomposición en partes del proceso de ensamblaje y embalaje por parte de un robot articulado.

Es en este punto cuando se consideran los posibles conflictos en la resolución de las ambas partes y la combinación de las soluciones que se proponen en cada parte. En realidad se trata de la consideración de los estados, condicionantes y operadores que definen cada parte. Este estudio debe llevar a agrupar operadores y estados con el fin de optimizar la realización de la tarea final.

Así, no será necesario retirarse a la localización de seguridad una vez acabado el proceso de ensamblaje para luego, al iniciarse el proceso de embalaje acudir de nuevo a coger el objeto ya montado de la mesa donde se ensambló. La consideración de la secuencialidad de los procesos nos permite eliminar estas acciones con el fin de ahorrar pasos en la resolución del proceso global.

El siguiente esquema muestra el resultado de la consideración de conflictos y combinación de las soluciones (acciones) de las dos partes resultado de la descomposición:

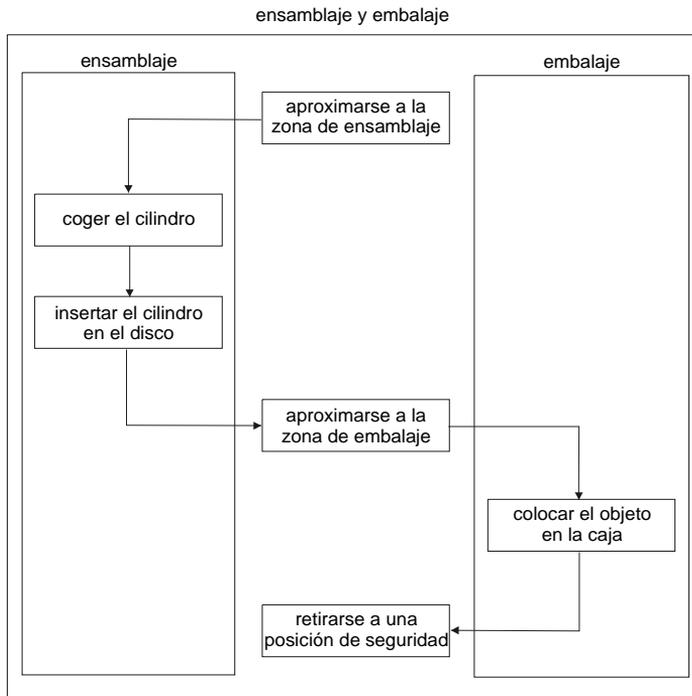


Figura 3-19. Combinación de soluciones para las dos partes resultados de la descomposición del proceso de ensamblaje de dos piezas y posterior embalaje por parte de un robot articulado.

Abstracción Jerárquica del Problema

Los criterios que se utilizan para establecer los distintos niveles de abstracción son el entorno y el agente. La consideración de estos criterios lleva a la propuesta de los niveles de misión, tarea, acción y trayectoria para los procesos que realiza un sistema robotizado.

La **misión** es la descripción a más alto nivel del proceso global. Considerando las dos partes en las que se ha descompuesto el problema, estas serían las misiones a considerar en cada parte:

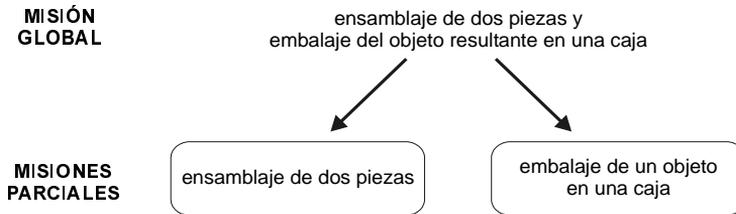


Figura 3-20. Abstracción en misiones del problema.

La descripción de las misiones parciales se corresponde con las partes que se consideran del problema. Tanto la descripción del problema global como las misiones que describen las partes del mismo son independientes del entorno en el que se vayan a realizar y del agente que deba realizarlas.

Las planificación de las misiones permite obtener una serie de tareas a desempeñar. La definición de estas tareas está relacionada con la consideración del entorno en el que se deben realizar las misiones. Dicho entorno contempla una zona para realizar ensamblajes y otra para el embalaje (Figura 3-17). Estas tareas, por tanto, son dependientes del entorno, aunque siguen definiéndose de manera independiente del agente que las va a desempeñar. El siguiente esquema añade al anterior la abstracción en tareas del problema:

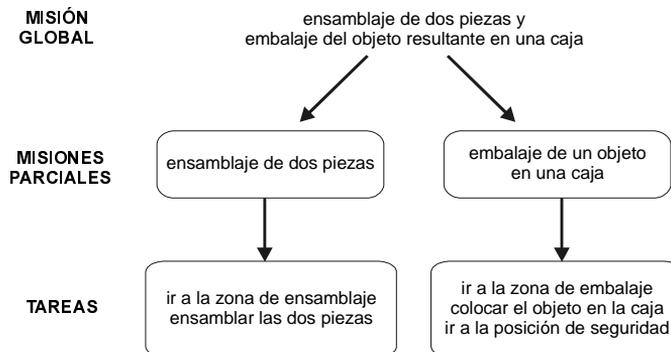


Figura 3-21. Abstracción en misiones y tareas del problema.

Las tareas se planifican llegando al nivel más concreto que describe el problema. Para ello, es necesario considerar las características del agente que debe desempeñarlas, es decir, del robot articulado. El

robot articulado utiliza una pina para manejar las piezas de cara a su ensamblaje y embalaje. La pinza permite asir el cilindro y realizar el ensamblaje del mismo en el disco dándole un cuarto de vuelta. Posteriormente, el objeto final se manipula cogiéndolo por el mismo cilindro ya montado.

La planificación de tareas lleva a la definición de **trayectorias** y **acciones** del robot articulado. Las trayectorias supondrán desplazamientos del útil del robot en el espacio, posicionándose en sucesivas localizaciones espaciales. Las acciones suponen, en este caso, el accionamiento del útil, es decir, de la pinza con la que coge el objeto. Se incluye en este último apartado el ensamblaje del cilindro.

El siguiente esquema muestra la concreción en trayectorias y acciones del problema.

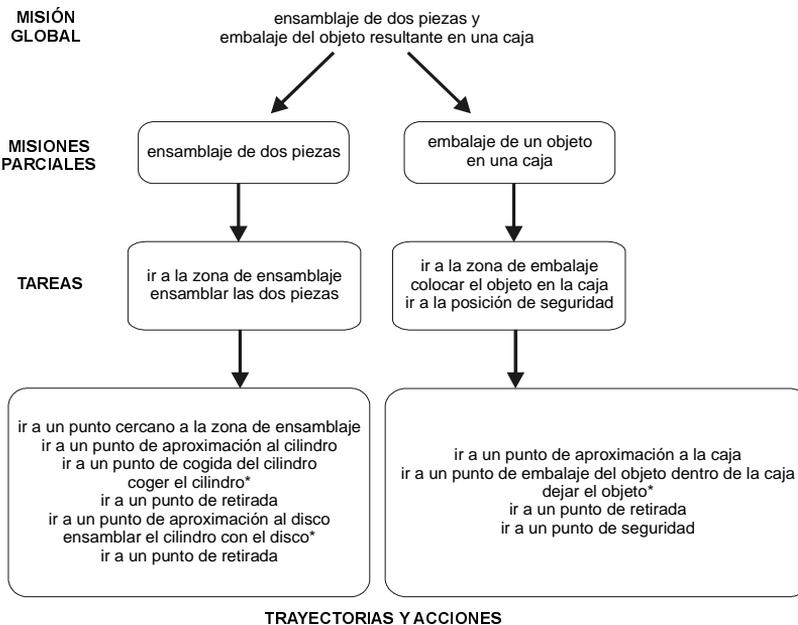


Figura 3-22. Concreción final del problema en trayectorias y acciones.

Las definiciones marcadas con un asterisco (*) indican acciones, es decir, procesos básicos que no implican desplazamiento del robot articulado. Estas acciones se resuelven a partir de comandos básicos como los que permiten realizar la apertura y cierre de la pinza.

El resto de definiciones configuran las trayectorias que debe realizar el útil del robot articulado. Estas trayectorias se planifican en este nivel concreto generándose los caminos geométricos exactos por los que debe pasar el útil del robot, considerándose aspectos como la evitación de colisiones con objetos del entorno.

EJEMPLO DE APLICACIÓN A UN ROBOT MÓVIL

La descomposición en partes y abstracción jerárquica se puede aplicar igualmente a las actividades que desempeña un robot móvil. Consideremos para ello el problema del transporte de dos piezas por parte de un robot móvil, descrito en la Figura 3-4. En este problema, el robot móvil debe transportar dos piezas localizadas en los lugares denotados como A y B hasta las nuevas localizaciones A' y B', respectivamente.

Descomposición en Partes

La consideración de los objetivos que deben cumplirse en la actividad encomendada (transporte de dos piezas) llevan, de manera inmediata, a dividir el problema en dos partes:

transporte de una pieza de A a A'
transporte de una pieza de B a B'

Esta división nos lleva a considerar el esquema propuesto en la Figura 3-7 como descomposición del problema de transporte.

Sin embargo, si aplicamos los criterios establecidos para la división del problema en partes, la división final puede ser distinta. La siguiente tabla muestra el estudio de la aplicación de los criterios al problema del transporte de dos piezas:

critério	descripción	estudio particular
<i>operatividad</i>	Posibilidades de movimiento y operacionales del robot móvil	El robot móvil puede desplazarse entre todas las localizaciones definidas, pudiendo transportar dos piezas al mismo tiempo. El proceso propuesto supone hacer operaciones de carga y descarga para cada pieza.
<i>espacial</i>	Ubicación espacial de las diferentes actividades	Las piezas deben cargarse en lugares próximos y descargarse en lugares también próximos entre sí.
<i>temporal</i>	Secuenciación temporal de las acciones	La posibilidad de transportar las dos piezas al mismo tiempo debido a que el camino a recorrer coincide obliga a esta secuencia de actividades: <ol style="list-style-type: none"> 1. cargar pieza de A 2. cargar pieza de B 3. descargar pieza en B' 4. descargar pieza en A'
<i>alcance de las acciones</i>	Consideración de objetivos parciales	Los desplazamientos del robot móvil y las actividades de carga y descarga se deben alternar de cara a la resolución del problema global.

Tabla 3-6. Estudio de los criterios que se siguen para establecer la división en partes al problema del transporte de dos piezas.

Este estudio nos lleva la división del proceso en 2 partes:

carga de piezas de A y B
descarga de piezas en B' y A'

Se podría haber llegado a considerar cuatro partes distintas, considerando las cargas y las descargas de cada pieza por separado, pero el hecho de que sea un problema en el que los procesos siguen una secuencialidad para su resolución hace que se puedan optimizar movimientos del robot móvil entre posiciones de carga y entre posiciones de descarga. Esta es la razón por la que la división atiende

a procesos de carga de piezas por un lado y a procesos de descarga por otro.

El siguiente esquema muestra la división en partes final del problema:

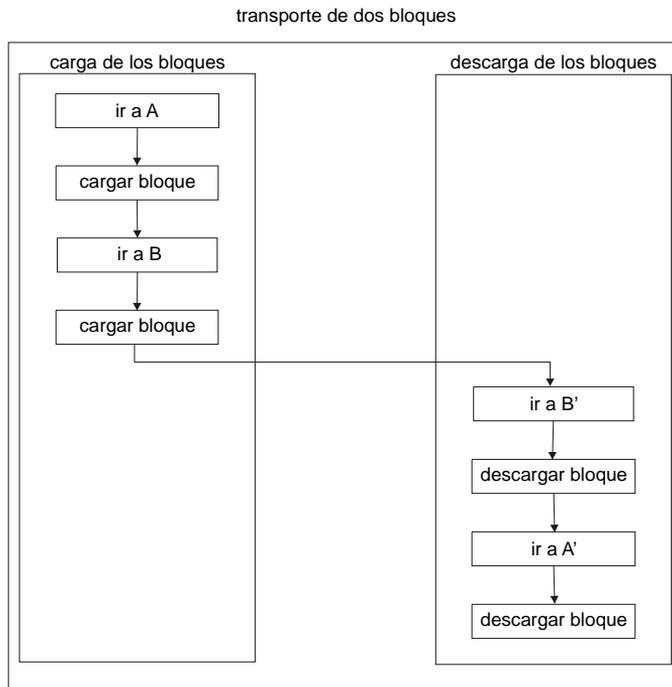


Figura 3-23. Descomposición en partes del problema del transporte de dos bloques por parte de un robot móvil.

En cualquier caso, ambas propuestas de descomposición en partes son válidas, aunque esta última estructura mejora los procesos a realizar, aprovechando la secuencialidad de la actividad global.

Abstracción Jerárquica

El siguiente esquema muestra, a partir de las dos partes en las que se descompone el problema, las misiones que las describen:

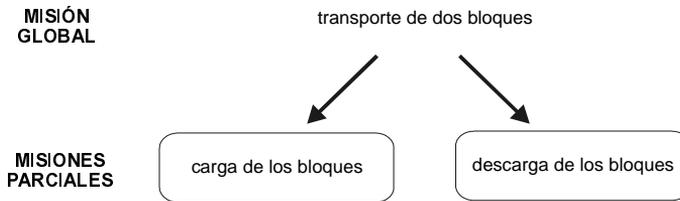


Figura 3-24. Esquema que muestra las misiones que describen las dos partes del problema.

Estas dos misiones, independientes en su descripción del entorno en el que se deben realizar y del agente que las debe realizar, se planifican para obtener las tareas correspondientes.

La planificación de las misiones es abordada a partir de un grafo de alto nivel del entorno. Es por ello que las tareas resultado de esta planificación vienen definidas teniendo en cuenta el entorno global en el que deben desempeñarse. El siguiente esquema muestra el resultado de la planificación de misiones:

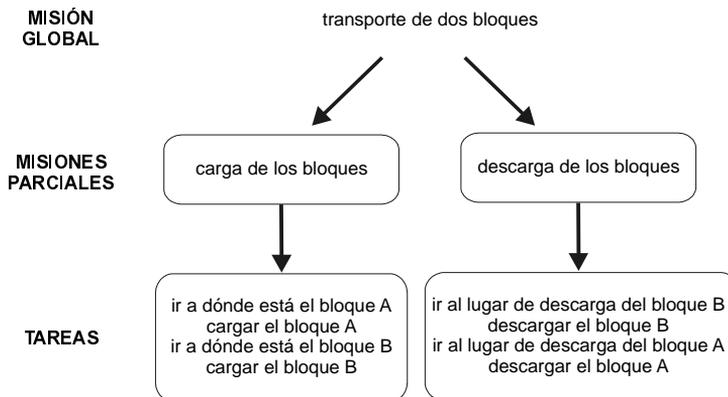


Figura 3-25. Esquema que muestra la generación de las tareas como resultado de la planificación de misiones en ambas partes.

En el esquema anterior se aprecia como las tareas dependen de las ubicaciones espaciales de los bloques para definir desplazamientos a los lugares de carga y descarga.

Las planificación de tareas permitirá obtener acciones y trayectorias como descripciones concretas de las actividades que debe realizar el móvil para llevar a cabo la carga y descarga de los bloques. El siguiente esquema muestra el resultado de esta planificación:

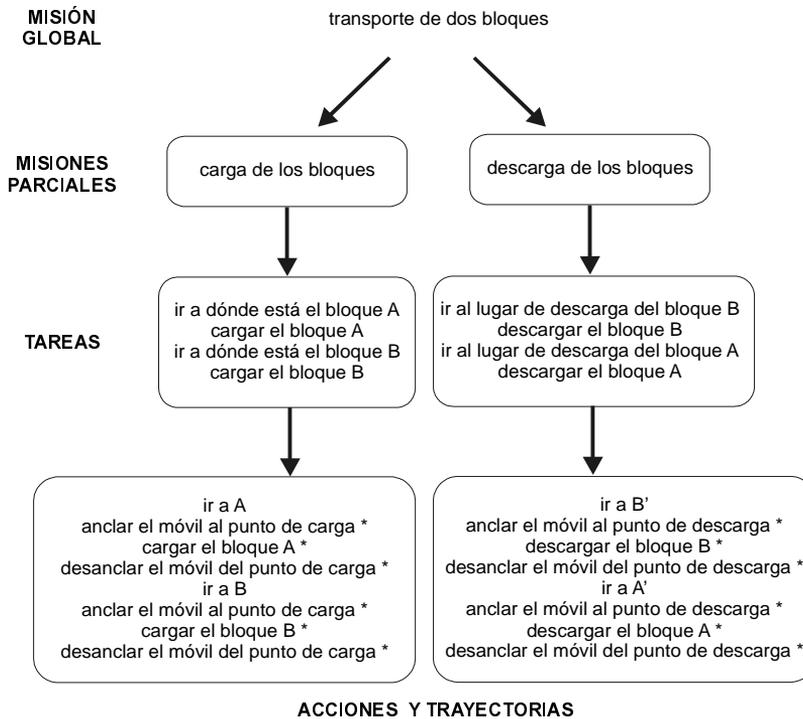


Figura 3-26. Esquema final de la abstracción en niveles del problema del transporte de dos bloques.

En el nivel concreto en el que se definen las acciones y las trayectorias, las actividades marcadas con asterisco (*) representan acciones, mientras que el resto son trayectorias.

EL PROYECTO RETRARO

Durante los años 1995 a 1997, el autor de esta tesis doctoral estuvo trabajando como investigador en el proyecto ESPRIT "Use of the Expert System Platform REAKT for Transport Robot Guidance

(RETRARO)³. El objetivo principal de este proyecto se centró en uso del sistema experto en tiempo real REAKT⁴ [Mensch *et al.*, 1993] para el control de un vehículo autoguiado autónomo en un entorno industrial real.

En concreto, el trabajo del proyecto se centró en la empresa Schoeller Bregenz (SB), en Bregenz, Austria. La principal actividad de esta empresa es la producción de hilaturas. Para esta actividad, en la empresa se realiza el transporte de unas cajas con bobinas entre las diferentes unidades de procesamiento de la lana. Inicialmente, para el transporte de material en la empresa se hacía uso de un sistema de cuatro vehículos filoguiados (VFGs) caracterizado por:

- Los VFGs se mueven a lo largo de sus rutas en la empresa guiados por un sistema de cables inductivos instalados en el suelo.
- Hay un ordenador central que coordina todas las actividades de los VFGs.
- Los VFGs se comunican con el ordenador central mediante un módem infrarrojos provisto de una interfaz RS-232.
- Los VFGs tienen un sistema de sensorización especial dedicado al seguimiento de los cables inductivos en el suelo.
- Los diferentes caminos se dividen en segmentos, limitados por sensores de infrarrojos y lectores de códigos de barras.

Este sistema, que funcionaba correctamente y permitía las actividades de transporte entre distintos puntos de la empresa de manera correcta, tenía un claro problema de flexibilidad de cara a futuros cambios en la distribución de planta de la empresa.

El objetivo principal del proyecto RETRARO consistía en desarrollar un sistema que permitiera abordar nuevas situaciones en la empresa que se tradujeran en nuevas actividades de transporte. El sistema debía considerar, además, situaciones imprevistas que se pudieran producir a lo largo del transporte de las cajas, para lo cual se debía dotar de una mayor autonomía de los vehículos y de un sistema de planificación. Este sistema de planificación debería considerar prioridades, longitudes de rutas de transporte, zonas conflictivas y el manejo de recursos comunes.

³ ESPRIT IV Project N° 20.788 “Use of the Expert System Platform REAKT for Transport Robot Guidance (RETRARO)”. 1995 a 1997.

⁴ ESPRIT II Project N° 4561 “REAKT: Environment and Methodology for Real-Time Knowledge Based Systems. 1990 a 1993.

El resultado del trabajo de investigación en este proyecto se tradujo en la actualización de uno de los vehículos de la empresa, como primer prototipo de un vehículo autoguiado (VAG) con flexibilidad suficiente como para abordar las actividades y características antes descritas.

Como principal aportación del autor de esta tesis al proyecto, se aplicó la génesis de la metodología de planificación [Vendrell y Mellado, 1998] que se presenta en esta tesis doctoral. Así, se aplicó una planificación de las tareas del VAG basada en una estructura jerárquica básica (misiones, tareas, movimientos) y en una descomposición en partes atendiendo a zonas distintas del entorno. El resultado final se plasma en una arquitectura software adecuada que refleja los distintos módulos que permiten la planificación jerárquica del robot móvil.

A lo largo de este apartado y en los siguientes capítulos se describen los resultados aplicados en el proyecto RETRARO y su relación con la metodología final que se plantea en la tesis.

Planteamiento del Problema RETRARO

La siguiente figura muestra la distribución en planta inicial de la empresa SB.

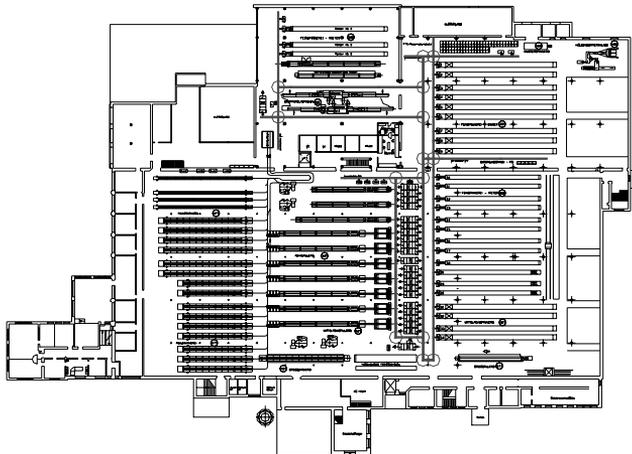


Figura 3-27. Disposición en planta inicial de la empresa SB. En la figura se muestra la ruta que siguen los vehículos filoguiados.

En esta disposición en planta se muestran los caminos que deben seguir los VFGs para llevar a cabo las actividades de transporte entre las diferentes unidades de la empresa.

La ampliación del sistema debía considerar nuevas actividades de transporte a desempeñarse en zonas nuevas de la empresa. Para ello, se utilizó uno de los VFGs existentes, cambiando su arquitectura para que pudiera realizar actividades de forma autónoma. Es decir, se reconvirtió un VFG en un VAG. Las nuevas características del VAG permitirían:

- Considerar nuevas rutas de transporte sin necesidad de la instalación de cable inductivo en el suelo.
- Seguir caminos alternativos en la ejecución de una actividad de transporte.
- Considerar eventualidades en la ejecución de una actividad (recarga de baterías, pasillos bloqueados...).
- Ejecutar las actividades de transporte propias en un entorno dinámico en el que coexisten otros móviles (los otros VFGs), dando prioridad en zonas de paso compartidas a las actividades de transporte ajenas.

La siguiente figura muestra la distribución en planta de la empresa SB considerando las nuevas rutas de transporte del VAG.

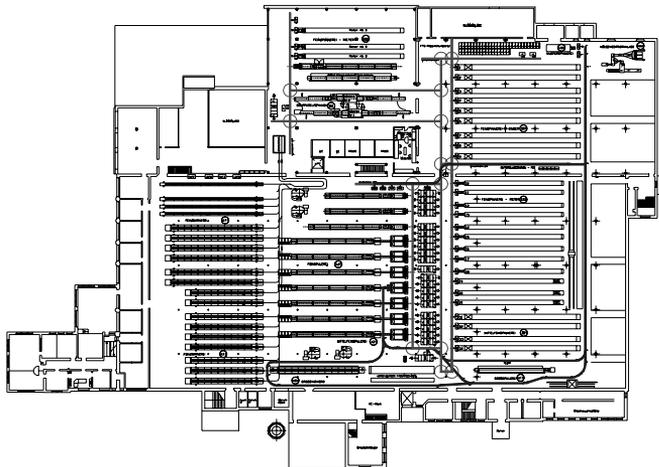


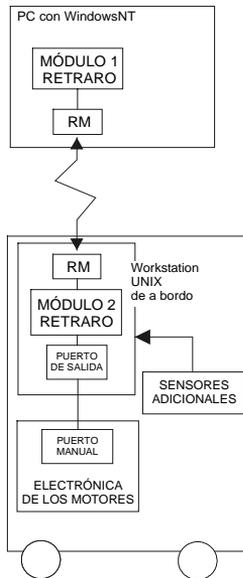
Figura 3-28. Distribución en planta de la empresa SB considerando las rutas del VAG.

El desarrollo del proyecto RETRARO llevó a implementar un sistema de control para el vehículo reconvertido en VAG. En esta reconversión, se implementó una nueva arquitectura hardware para el móvil con las siguientes características [Mellado *et al.*, 1999]:

- Un ordenador central (un PC con WindowsNT) para el control del VAG. Este ordenador central mantiene una parte de la arquitectura software, dedicada a la planificación de misiones, la coordinación con el resto de móviles del entorno (los 3 VFGs), la supervisión del sistema y el interfaz con el usuario. La comunicación con el VAG se realiza mediante una conexión radio módem que se garantiza en las localizaciones de inicio y fin de las actividades.
- Un ordenador de a bordo instalado sobre el VAG (una *workstation* portátil con un sistema operativo UNIX). Este ordenador se encarga de mantener la información que proporcionan los sensores instalados en el VAG de cara a facilitar la navegación del móvil y la evitación de obstáculos. Así mismo, es en este a parte dónde se realiza la planificación de bajo nivel (tareas y trayectorias), junto con la inclusión de un módulo que permite un comportamiento reactivo del vehículo.
- El nivel servo del móvil se mantiene, utilizándose el modo de operación manual que permite la transferencia directa de comandos de control al vehículo en caso de situaciones de emergencia.

La siguiente figura muestra el esquema de esta arquitectura hardware:

Ordenador Central



VFG reconvertido en VAG

Figura 3-29. Arquitectura hardware del VAG del proyecto RETRARO.

El nuevo sistema de control del VAG incluía la implementación de una nueva arquitectura software que se detalla en un capítulo posterior.

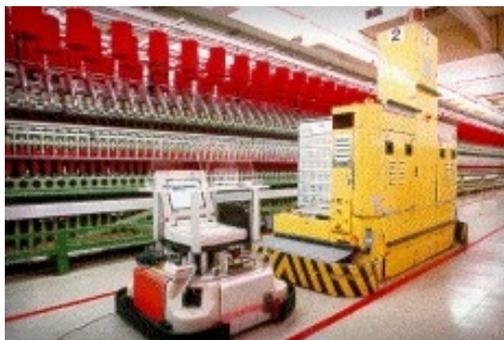


Figura 3-30. VFG de la empresa SB reconvertido en VAG dentro del proyecto RETRARO.

Descomposición en Partes

La planificación de las actividades del VAG debe seguir la premisa básica de dar prioridad siempre a las actividades que realizan los otros móviles del entorno. Esta característica se hace crítica cuando el VAG debe atravesar una zona que es compartida con los tres VFGs. La siguiente figura muestra con detalle las rutas que puede seguir el VAG.

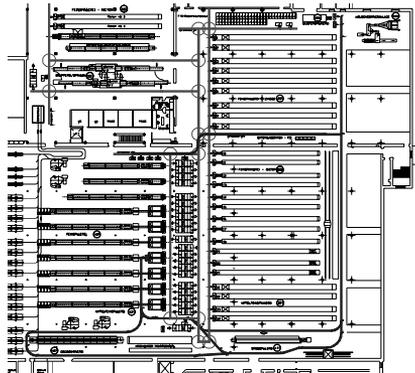


Figura 3-31. Detalle de las rutas que debe seguir el VAG en la empresa SB.

Se aprecia en la figura anterior que existen dos zonas en la que se cruzan las trayectorias de los VFGs y las del VAG. También se pueden ver en la parte inferior de la figura la existencia de rutas alternativas del VAG que evitan cruzar por una de las zonas compartidas con los VFGs. Sólo se debe recurrir a rutas alternativas en casos extremos (por ejemplo, cuando la espera a que se libere una zona compartida ocupada por un VFG sea demasiado grande e impida cumplir los tiempos requeridos para la ejecución de una actividad de robot), puesto que atraviesan corredores con alto tráfico de operarios, lo que hace menos deseable esta trayectoria.

La existencia de una zona compartida entre el VAG y los VFGs se utilizó como factor determinante para realizar una descomposición de la planificación de las actividades del VAG en las siguientes partes: una zona de desplazamientos libres del VAG y dos zonas compartidas con los VFGs.

Si se atiende a los criterios expuestos anteriormente para la descomposición en partes de un problema, se tiene el siguiente estudio:

critério	descripción	estudio particular
<i>operatividad</i>	Posibilidades de movimiento y operacionales del robot móvil	El VAG puede desplazarse con total libertad por los pasillos de la empresa, pudiendo realizar acciones de carga y descarga de cajas con bobinas de hilos. Estas acciones pueden realizarse en cualquier punto de una línea de trabajo, con lo que la alternancia entre desplazamientos del móvil y acciones de carga y/o descarga es total. La operatividad, pues, no es un factor determinante para la descomposición en partes del problema.
<i>espacial</i>	Ubicación espacial de las diferentes actividades	Aunque las cajas pueden cargarse y descargarse en lugares distintos de la empresa, este criterio, atendiendo a las características de las zonas que atraviesa el VAG es determinante para establecer una zona de movimiento libre y zonas compartidas con el resto de móviles.
<i>temporal</i>	Secuenciación temporal de las acciones	No hay secuencialidad clara en las operaciones que realiza el VAG. La alternancia entre cargas, descargas y desplazamientos para distintas actividades no permite establecer una descomposición en base a la secuencialidad de las operaciones.
<i>alcance de las acciones</i>	Consideración de objetivos parciales	Los desplazamientos del robot móvil y las actividades de carga y descarga se deben alternar de cara a la resolución del problema global.

Tabla 3-7. Estudio de los criterios que se siguen para establecer la división en partes en la planificación del proyecto RETRARO.

Como se aprecia, la aplicación de los criterios lleva a considerar la descomposición en tres partes distintas que se corresponden con zonas de la empresa SB. La siguiente figura muestra esta división espacial:

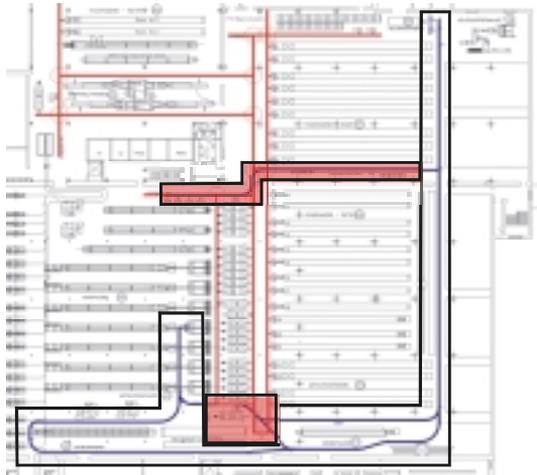


Figura 3-32. Descomposición en zonas de cara a la planificación de las actividades del VAG del proyecto RETRARO. En sombreado se muestran la zonas compartidas con los VFGs, mientras que la zona de libre circulación del VAG se muestra con borde resaltado.

Abstracción Jerárquica

La planificación de las actividades que debe realizar el VAG del proyecto RETRARO lleva asociada una abstracción jerárquica que se ajusta a la propuesta en la presente tesis doctoral. El flujo de información que se establece es el siguiente [PSADD RETRARO, 1997] [Garbajosa *et al.*, 1997] [Mellado *et al.*, 1997] [Mellado *et al.*, 1998] [Mellado *et al.*, 1999]:

1. Se generan tareas de transporte (*transport tasks*) a alto nivel.
2. A partir de los requerimientos de la tarea de transporte se decide qué partes se desarrollan en las zonas de navegación libre y cuáles se desarrollan en la zona compartida. En base a esta decisión, se generan las tareas de movimiento (*motion tasks*).
3. A un nivel intermedio, las tareas de movimiento se descomponen en acciones de movimiento (*motion actions*) y

acciones auxiliares (*auxiliary actions*) y de anclaje (*docking actions*).

4. Las acciones auxiliares y de anclaje se descomponen en acciones de carga y descarga y en acciones de anclaje.
5. Las acciones de movimiento se planifican para obtener comandos de movimiento al nivel más bajo posible.

Este flujo de información se corresponde con la propuesta de la metodología, particularizando nombres a las actividades propias de la empresa y del agente que las desempeña, el VAG. De hecho, las razones que se siguen para establecer este flujo de información están basadas en la existencia de los niveles de abstracción que propugna la metodología.

El siguiente esquema muestra la correspondencia entre la información establecida en RETRARO y los niveles de abstracción propuestos:

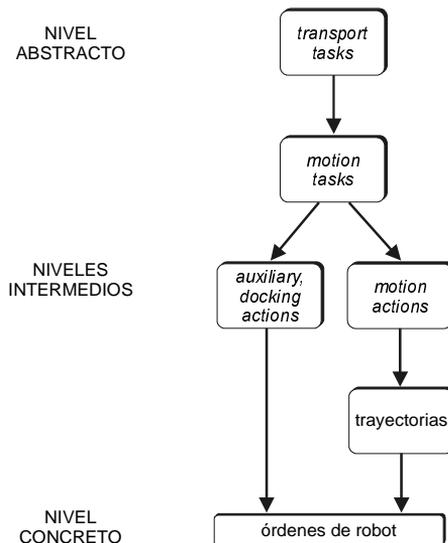


Figura 3-33. Niveles de abstracción en RETRARO.

A continuación se definen estos niveles según los criterios de la abstracción jerárquica y se describe cómo se aborda la planificación en cada nivel.

- Tareas de transporte (*Transport tasks*):

Es el más alto nivel de abstracción y se corresponde con las misiones. A este nivel de abstracción, las tareas de transporte vienen definidas de la siguiente manera:

Lot XX from Position YY to Position ZZ

siendo Position YY y Position ZZ cualquiera de las localizaciones de carga y descarga de la empresa.

Como se aprecia, la definición de las misiones no viene determinada por las características del agente que la debe realizar ni por la disposición del entorno. De hecho, en principio cualquier móvil puede realizar la misión. Será un planificador de alto nivel (*scheduler*) el que se encargue de asignar las misiones a los móviles según una prioridad establecida.

La planificación de las misiones consistirá en dividir la tarea de transporte en tareas de movimiento (*motion tasks*) a realizarse en las diferentes zonas en las que actúa el VAG. De esta manera, una tarea de transporte como la de arriba quedaría dividida en las siguientes tareas de movimiento:

Lot XX from Position YY to TRPI

Lot XX from TRPI to TRPJ

Lot XX from TRPJ to Position ZZ

Siendo TRPI y TRPJ puntos que intercomunican las zonas de libre navegación con la zona compartida.

La planificación de las misiones se hace a un alto nivel, atendiendo a los requisitos de la tarea de transporte y a una primera aproximación a la topología del entorno, que considera las distintas zonas en las que se divide el mismo.

- Tareas de Movimiento (*Motion Tasks*):

Las tareas de movimiento suponen uno de los niveles intermedios de abstracción, y describen los objetivos a cumplirse en cada zona en las que se divide el entorno. De alguna manera son también misiones parciales para cada

parte del problema. De hecho, en la aplicación particular de la metodología al proyecto, la definición de la misión para cada zona es una tarea de movimiento. Esta definición no viene influenciada por las características del VAG, estando asociada a la zona en la que se debe realizar.

La planificación de las tareas de movimiento se realiza a partir de grafos más detallados que incluyen puntos de paso del móvil y que reflejan la topología básica del entorno. La aplicación de algoritmos de búsqueda en este grafo junto con la consideración de los objetivos parciales que debe cubrirse nos lleva a dividir la tarea de movimiento en acciones de movimiento (*Motion Actions*), acciones auxiliares (*Auxiliary Actions*) y acciones de anclaje (*Docking Actions*).

- Acciones de Movimiento (*Motion Actions*):

Se corresponden con los movimientos, es decir, los desplazamientos que debe realizar el VAG en una zona. La definición de las acciones de movimiento se realiza a partir de las posibilidades de movimiento del móvil y del conocimiento de la geometría concreta del entorno. Para ello, se dispone de un grafo detallado del entorno, que incluye todos los puntos de paso que se consideran a lo largo del entorno.

Un algoritmo eficiente de búsqueda en el grafo permite planificar estas acciones de movimiento y generar las trayectorias concretas que definen el desplazamiento del móvil.

- Acciones Auxiliares (*Auxiliary Actions*) y de Anclaje (*Docking Actions*):

En ambos casos, tanto las acciones auxiliares que se requieren para accionar el sistema automático de carga y descarga de cajas desde el VAG, como las acciones para el anclaje del VAG a la máquina sobre la que hay que cargar o descargar una caja, se corresponden con acciones propiamente dichas. Se trata en todo caso de actividades que no suponen desplazamiento del VAG, descritas a partir de las características operativas del propio VAG.

- Trayectorias:

Las *Motion Actions* se planifican generándose las trayectorias concretas que debe seguir el VAG. Para ello, se utiliza un grafo que refleja con detalle la geometría espacial del entorno. Esta planificación supone llegar al nivel más bajo de definición, el de los comandos de robot u órdenes directas, que en este caso reflejaran acciones básicas de avance del móvil, a partir de la trayectoria generada.

CONCLUSIONES

Este capítulo ha introducido una metodología para la planificación de las actividades que debe realizar un robot, y cuya principal innovación se centra en la aplicación de intuiciones humanas para la resolución del problema de planificación. Las intuiciones humanas se utilizan en Inteligencia Artificial para establecer algoritmos de planificación generales avanzados, aunque su aplicación en robótica es escasa. El planificador fruto de este trabajo de investigación está basado en las intuiciones humanas de **abstracción** y **descomposición** del problema.

La descomposición de la actividad del robot permite considerar distintas partes del problema de forma separada, obteniéndose soluciones parciales que se combinan para generar una solución global del problema. Así, como principal aportación se han definido unos criterios para la descomposición en partes del problema (*operatividad de los agentes, criterios espaciales, criterios temporales y alcance espacio-temporal de las acciones*) y unos factores (*coste temporal, coste energético, número de operaciones y consumo de recursos*) que permiten decidir la conveniencia de la descomposición en partes, estableciéndose un índice de bondad (B_i) que pondera estos factores.

La abstracción jerárquica, por su parte, conlleva la generación de distintos niveles de concepción de la actividad robotizada a planificar. De esta manera, se resuelve el problema de planificación en cada uno de estos niveles, obteniéndose como resultado las entradas para los niveles inferiores. En el presente trabajo se introducen y establecen los niveles de *misiones, tareas, movimientos, acciones, trayectorias y órdenes de robot*. Estos niveles quedan perfectamente definidos respecto de su dependencia o independencia del entorno y del robot.

La descomposición en partes y la jerarquización son conceptos compatibles entre sí que se pueden combinar para la planificación de actividades de robot complejas.

Ambos conceptos se han aplicado en el presente trabajo de investigación a ejemplos de actividades realizadas tanto por robots articulados, como móviles. Los ejemplos descritos muestran la aplicabilidad del método de planificación en cada caso.

Como una utilidad generada en paralelo al desarrollo de este trabajo de investigación, se ha implementado una aplicación para validación del uso de las Redes de Petri en la planificación de tareas de un robot móvil basada en la descomposición celular del entorno. La herramienta software desarrollada considera, al realizar la planificación de las tareas del robot móvil, la existencia de zonas bloqueadas eventualmente y las actividades que realiza otro móvil en el entorno. Para obtener en cada momento los movimientos adecuados del robot móvil, se ha empleado un algoritmo de búsqueda del camino más corto que opera sobre la representación matricial de la Red de Petri.

Las bases de esta metodología de planificación de robots han sido desarrolladas y se han aplicado dentro de un proyecto europeo de investigación, como principal aportación del autor de esta tesis doctoral. En este contexto, se ha desarrollado la arquitectura hardware y software que mantiene esta metodología de planificación, reconvirtiendo uno de los vehículos filoguiados existentes en una empresa textil en un vehículo autoguiado cuyas actividades vienen determinadas por el método de planificación aquí expuesto.



4 *Replanificación: El proceso deductivo*

Replanificación: El proceso deductivo

Este capítulo presenta el proceso de deducción de acciones y tareas que realizan otros agentes en el entorno como primer paso dentro del proceso general de replanificación de las trayectorias que sigue un robot al ejecutar sus actividades.

En primer lugar, se definen y clasifican los distintos tipos de agentes que intervienen en las diferentes actividades de un proceso productivo. Esta clasificación permite tener en cuenta la operatividad de cada agente y las posibilidades de replanificación de sus trayectorias. A continuación, se formaliza la definición de una trayectoria como una secuencia de puntos del entorno y tiempos de paso asociados. La consideración de los intervalos temporales, a partir de una incertidumbre en el tiempo de paso, es clave para abordar con posterioridad el razonamiento temporal sobre las trayectorias de los distintos agentes.

El proceso deductivo se estructura a través de una observación del entorno, sensorizando de manera adecuada el entorno de trabajo. La información obtenida permite el reconocimiento de las trayectorias que siguen los distintos agentes. Con este fin, se introduce un método básico de reconocimiento y aprendizaje de trayectorias de los mismos.

El desarrollo de este capítulo incluye los algoritmos básicos que definen el proceso de replanificación global y el proceso deductivo en particular.

Replanificación: El proceso deductivo

EL PROCESO DE REPLANIFICACIÓN

En muchas ocasiones, cuando se requiere la resolución de un problema determinado, existe una serie de condicionantes, ajenos de alguna manera al problema, que interfieren en su resolución y que en sí mismos no forman parte del problema. Así, por ejemplo, ante la tarea de ir a comprar el periódico, cuya solución óptima en principio sería la de encontrar la ruta más corta para ir al kiosco y volver, factores como el tráfico, el tiempo que hace o incluso la cantidad de gasolina que queda en el coche afectan directamente a su solución. De hecho, si hay mucho tráfico en alguna calle, con atascos, siempre será mejor elegir una ruta alternativa que, aunque sea más larga, nos lleve al kiosco en menor tiempo; si está lloviendo, es probable que aumente el número de coches en las calles y se ralentice el tráfico, lo cual afectará a la duración de la tarea; y, por último, si el depósito del coche está en reserva, es posible que se deba parar en una gasolinera próxima para repostar, lo cual afecta sustancialmente a la tarea en cuestión.

En resumen, existe en gran número de situaciones una serie de factores, ajenos en cierto sentido a un problema concreto, como la consecución de una tarea determinada, que afectan a la resolución del mismo, alterando el resultado o generando una solución menos eficiente de la que cabría pensar. Esta influencia externa introduce la necesidad de replanificar en tiempo real la solución para un problema dado.

Cuando estos factores aparecen, hay que tenerlos en cuenta como parte del problema a la hora de proceder a su resolución. La dificultad de tenerlos en cuenta radica en el hecho de que algunos de estos factores no son controlables directamente. Esto significa que el agente encargado de resolver el problema, al no poder actuar sobre estos factores (por ejemplo, la lluvia) debe tenerlos en cuenta a partir de un proceso de observación y actuar en consecuencia (si llueve demasiado, esperar o buscar una ruta alternativa para evitar atascos). Es decir, debe realizar un proceso deductivo basado en la extracción de datos del entorno.

El proceso deductivo, abordado durante la ejecución de un plan, permite obtener datos de factores no controlados. Estos factores llevan a la modificación del plan inicial, introduciendo cambios que permitan ejecutarlo de manera eficaz. En el ejemplo anterior, la actividad comprar el periódico se planifica a priori generando un plan que permite llevarla a cabo. Este plan se modificará durante su

ejecución teniendo en cuenta factores externos como el tráfico actual en una calle dada.

En un proceso de producción industrial, compuesto por una sucesión de actividades en el tiempo, pueden estar involucrados distintos agentes (personas, máquinas, sistemas de transporte...) que realizan estas actividades de forma independiente o coordinada. Cuando un agente realiza una actividad simple concreta, se considera que el agente actúa de forma independiente, mientras que cuando se requiere más de un agente para completar una actividad, éstos actúan de forma coordinada. Esta asunción (actividades diferentes, agentes diferentes en un mismo proceso de producción) lleva a la configuración de un entorno complejo, considerando al entorno como el lugar en el que se desarrolla el proceso de producción, es decir, la factoría o empresa en el que se ubican los recursos necesarios para el proceso de producción.

Ante entornos complejos, como los que se deducen de las consideraciones anteriores, cuando un agente determinado lleva a cabo su actividad, debe tener en cuenta el comportamiento de otros agentes. Esta consideración nos lleva al planteamiento del principio, en el que aparece una serie de factores que, no siendo propios de un problema concreto, afecta al desarrollo y resolución del mismo y que deben tenerse en cuenta tanto para proporcionar una solución eficiente al problema, como para no interferir en el desarrollo de las actividades propias de estos agentes externos.

La consecuencia de estas observaciones es que es necesario contar con una replanificación de las actividades de acuerdo con las acciones de agentes externos que actúan en el mismo entorno. El siguiente algoritmo esboza el proceso de la replanificación de trayectorias que se plantea en esta tesis doctoral:

```
replanificar-trayectoria;  
  si (actividad-externa-es-desconocida) entonces  
    observar-y-deducir;  
  fin-si;  
predecir;  
fin;
```

Figura 4-1. Algoritmo básico para la replanificación de trayectorias.

Como se aprecia, el proceso de replanificación está basado en dos principios básicos:

1. Observar y deducir.
2. Predecir.

En este capítulo se aborda el primero de los principios, la deducción de las actividades que realizan los agentes que participan en el proceso productivo global. El siguiente capítulo describe el segundo de los principios, la predicción de las acciones y movimientos y su ajuste de cara a evitar imprevistos y optimizar su ejecución.

El proceso deductivo abordado en la metodología de replanificación se basa en la observación de las actividades de los agentes externos que comparten el entorno de trabajo con robot que se pretende controlar. Se trata por tanto de un planteamiento *on line*, es decir, durante la ejecución de las actividades planificadas para el robot controlado. Este proceso permite deducir actividades que realizan el resto de agentes del entorno.

Por otro lado, otro aspecto importante del proceso deductivo es que se produce a nivel de las trayectorias que cubren los móviles. Esto es, la observación de las actividades de los agentes externos permitirá obtener datos acerca de los desplazamientos y tiempos empleados, lo cual nos llevará a reconocer trayectorias de los mismos. Actuando a este nivel, los datos recogidos en el proceso deductivo afectarán a través de un proceso predictivo a las propias trayectorias del robot controlado.

Es por ello que es necesario, como paso previo, la correcta clasificación de los agentes que pueden intervenir en el mismo entorno de producción, así como la correcta formalización de las trayectorias que siguen al ejecutar sus actividades.

CLASIFICACIÓN DE LOS AGENTES

Es necesario establecer el posible comportamiento de los agentes con los que un robot se puede encontrar en su entorno antes de pasar a definir el propio proceso deductivo. El comportamiento nos va a permitir establecer una clasificación de los agentes que se desenvuelven en un entorno compartido.

Debido a que el proceso deductivo actúa reconociendo actividades de los agentes que comparten el entorno con el robot controlado, y a que

las actividades se reconocen a través de los movimientos que estos agentes externos realizan, a partir de este momento se utilizará el término de agente o móvil de manera indiferente para referirse a los agentes que actúan en un entorno común.

Se consideran tres tipos de agentes:

1. *Agentes no controlados con comportamiento fijo.*

Son los que realizan una serie de actividades de manera sistemática y con frecuencia fija, pero sobre los que no se puede actuar para su replanificación. Un ejemplo típico es el de un vehículo filoguiado, que tiene una ruta perfectamente determinada por el cable inductivo que le sirve para el seguimiento de sus desplazamientos. Este tipo de agentes ejecuta sus trayectorias de una manera perfectamente definida en cuanto a su duración y frecuencia.

2. *Agentes no controlados con comportamiento aleatorio.*

Se trata de aquellos cuyo comportamiento es imprevisible en cuanto a la trayectoria que siguen en cada momento, y sobre los que tampoco se puede actuar para su replanificación. Es el caso de aquellos móviles que se mueven en el mismo espacio que el robot que se desea controlar, pero de los que no sabemos cuál será el siguiente movimiento que van a realizar. En este grupo de agentes se sitúan, por ejemplo, los vehículos conducidos de manera manual utilizados en actividades de transporte en una factoría.

3. *Agentes con comportamiento controlado*

Se trata de los agentes de los que se conoce de antemano las actividades que van a realizar y sobre los que se puede actuar de cara a una eventual replanificación. En un sistema con varios agentes, un agente con comportamiento controlado es aquel del que se conoce siempre la trayectoria que sigue en un momento determinado y las que va a ejecutar en el futuro.

Esta clasificación determina el modo de actuación del proceso deductivo. Así, según el tipo de agente a observar, el proceso deductivo realiza una serie de acciones u otras.

A efectos de replanificación, el factor a tener en cuenta es su controlabilidad, es decir, si es posible actuar sobre ellos durante la ejecución de sus actividades con el fin de optimizar sus movimientos y

evitar futuros imprevistos. Desde este punto de vista, la presente tesis doctoral considera por un lado los agentes o móviles controlados, como objeto de la replanificación, y los móviles no controlados, como agentes externos al móvil controlado, que agrupan a los dos primeros tipos de la clasificación anterior

LAS TRAYECTORIAS DE LOS MÓVILES

Tal y como se ha introducido al principio de este capítulo, el proceso deductivo actúa a nivel de las trayectorias de los móviles. Esto es, el reconocimiento de desplazamientos y tiempos empleados a partir de un proceso de observación lleva a deducir las trayectorias que están ejecutando los móviles observados. Para abordar, entonces, el proceso deductivo es necesario formalizar el concepto de trayectoria.

Supongamos un entorno por el que se mueve un móvil realizando una serie de actividades. El proceso de deducción toma como referencia la observación del móvil a lo largo de una serie de puntos del entorno. A cada uno de estos puntos de paso lo llamaremos **punto característico**, y denotará cualquier punto del entorno que se pretenda observar y del que se suponga que el móvil va a realizar una acción característica (cambio de dirección en el desplazamiento, paradas, cargas o descargas, etc.). Entre dos puntos característicos se tendrá siempre un camino directo, no pudiendo existir este enlace entre dos puntos característicos separados por obstáculos que impidan su conexión.

A partir de la consideración de los puntos característicos, el entorno se puede considerar como un grafo no dirigido E , definido como sigue

Ecuación 4-1

$$E = (P, A)$$

donde P (vértices o nodos) son los puntos característicos del entorno y A (arcos) son los caminos directos que unen los diferentes puntos característicos. Esta es una definición bien conocida [Aho *et al.*, 1983] que se ajusta a la estructura de datos que se manejan en este trabajo.

Así, al definir el entorno como un grafo no dirigido, cada arco del conjunto de arcos A será un par de puntos característicos de P no ordenado, es decir, si (a,b) es un arco no dirigido, entonces $(a,b) \equiv (b,a)$. Esto significa que cualquier camino directo se puede recorrer en ambos sentidos.

Dentro del entorno (grafo), dos puntos característicos (vértices) serán *adyacentes* cuando exista un camino directo (arco) que los una. A partir de esta definición del entorno, un **camino** será una secuencia de puntos característicos

Ecuación 4-2

$$\{p_1, p_2, \dots, p_n\}$$

tal que el par (p_i, p_{i+1}) es un arco entre los puntos característicos indicados.

Supongamos un entorno básico sobre el que se disponen una serie de puntos característicos:

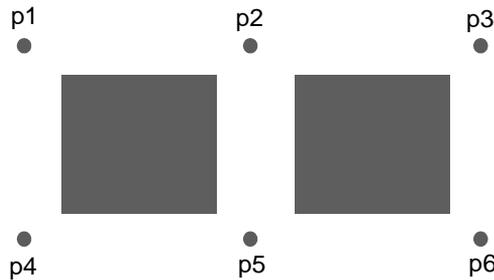


Figura 4-2. Entorno ejemplo básico con los puntos característicos.

Los puntos característicos dispuestos en el entorno cumplen los requisitos antes mencionados de estar conectados por caminos directos. En la disposición de los puntos característicos en este entorno se aprecia como el propio entorno y la disposición de los puntos característicos en él impiden la conexión directa entre algunos de ellos, como por ejemplo entre p2 y p4, teniéndose que realizar esta conexión a través de otro punto característico, en este caso, p5 ó p1.

El siguiente grafo $E=(P,A)$ aclara la conexión entre los puntos característicos:

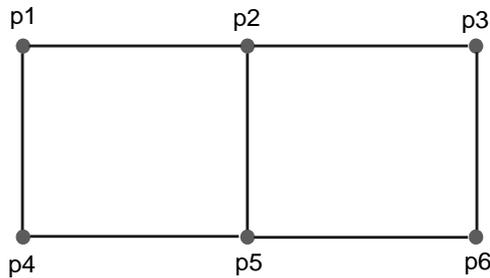


Figura 4-3. Grafo de los puntos característicos.

siendo

Ecuación 4-3

$$P = \{p1, p2, p3, p4, p5, p6\}$$

$$A = \{(p1, p2), (p2, p3), (p1, p4), (p2, p5), (p3, p6), (p4, p5), (p5, p6)\}$$

Un posible camino en este grafo es el siguiente:

Ecuación 4-4

$$\{p1, p2, p5, p6\}$$

La configuración del entorno como un grafo permite realizar búsquedas en el mismo de cara a encontrar los caminos óptimos que lleven de un punto característico a otro. La búsqueda de estos caminos óptimos se basará principalmente en características de coste temporal del paso de un punto a su adyacente. Existen multitud de algoritmos que permiten la búsqueda de caminos óptimos en un grafo [Aho *et al.*, 1983].

Atendiendo a la continuidad en el espacio de las configuraciones alcanzadas por un móvil a lo largo de sus desplazamientos por un entorno dado, un camino puede definirse como [Latombe, 1991]

Ecuación 4-5

$$f : [0,1] \rightarrow C$$

es decir, una función continua f que define las configuraciones alcanzadas por el móvil dentro de la totalidad de configuraciones posibles en el entorno en cuestión, C . Así, se tendrá que la configuración de partida vendrá definida por $f(0)$, mientras que la de llegada será $f(1)$.

Comparando esta definición con la anterior, vemos que el conjunto de puntos característicos del entorno (P) estará dentro del total de configuraciones posibles a alcanzar por el móvil en el entorno

Ecuación 4-6

$$P \subset C$$

Para hacer una búsqueda del camino óptimo basada en el coste temporal hay que introducir el factor tiempo en la definición anterior. De hecho, tal y como se plantea en el siguiente capítulo, el proceso de predicción posterior al de deducción se basa en la estimación temporal de los movimientos de los agentes observados, lo que hace imprescindible tratar con el factor “tiempo”, no incluido en la definición anterior de camino.

En [Latombe, 1991] se introduce el concepto de **trayectoria** como una función continua q definida en $[0, T]$ de forma que $q(0)$ se corresponde con la configuración inicial y $q(T)$ con la configuración objetivo de un camino dado, siendo T el tiempo empleado en recorrer el camino asociado. Es decir, el concepto de trayectoria está ligado a la concreción en el tiempo de un camino dado. Así, cada $q(t)$ alcanzado en una trayectoria se corresponderá con una configuración del camino f .

Es interesante relacionar el concepto de trayectoria con el concepto de camino desde el punto de vista del tratamiento de los puntos característicos de un entorno. Para ello, es necesario considerar el **coste temporal** de un camino.

Se asume que recorrer un camino directo entre dos puntos característicos p_i y p_{i+1} requiere un coste temporal cuyo tiempo medio es $t_{i,i+1}$. Para un móvil determinado, el coste temporal en recorrer un camino completo no sólo estriba en el coste de desplazarse a lo largo de la secuencia de puntos característicos que lo forman, sino que hay que tener en cuenta las posibles acciones que el móvil realiza en un punto característico. Estas acciones (cargas, descargas, cambios de dirección, etc.) requieren un consumo de tiempo que denotaremos como tiempo consumido en un punto característico t_i . Así, el coste temporal, c , de un camino $\{p_1, p_2, \dots, p_n\}$ vendrá dado por la siguiente expresión

Ecuación 4-7

$$c = t_1 + \sum_{i=1}^{i=n-1} (t_{i,i+1} + t_{i+1})$$

En este punto es necesario introducir el concepto de **incertidumbre temporal** considerando un intervalo temporal en vez de un valor temporal medio. Este concepto nos permite contemplar tiempos mínimos y máximos en la estimación de un coste temporal determinado, bien sea del desplazamiento del móvil a lo largo de un camino directo entre dos puntos característicos, bien para la consideración del tiempo consumido en un punto característico determinado.

La incertidumbre temporal vendrá asociada a distintas características del entorno y del propio móvil. Así, la velocidad del móvil y su rango de velocidades alcanzables o las eventualidades que puedan darse (evitar un obstáculo, detenerse para dejar pasar otro móvil, etc.) son factores que afectan a la consideración de la incertidumbre temporal.

Los intervalos temporales asociados al desplazamiento hasta un punto característico y al consumo de tiempo en él se definen como

Ecuación 4-8

$$\begin{aligned} t_{i,i+1-\text{mín}} &= t_{i,i+1} - \Delta_{i,i+1} \\ t_{i,i+1-\text{máx}} &= t_{i,i+1} + \Delta_{i,i+1} \end{aligned} \rightarrow [t_{i,i+1-\text{mín}}, t_{i,i+1-\text{máx}}]$$

$$\begin{aligned} t_{i-\text{mín}} &= t_i - \delta_i \\ t_{i-\text{máx}} &= t_i + \delta_i \end{aligned} \rightarrow [t_{i-\text{mín}}, t_{i-\text{máx}}]$$

siendo $\Delta_{i,i+1}$ y δ_i las incertidumbres asociadas al desplazamiento de p_i a p_{i+1} y al tiempo consumido por el móvil en p_i , respectivamente.

Así, los costes temporales mínimo y máximo asociados a un camino quedarán definidos como sigue

Ecuación 4-9

$$c_{\text{mín}} = t_{1-\text{mín}} + \sum_{i=1}^{i=n-1} (t_{i,i+1-\text{mín}} + t_{i+1-\text{mín}})$$

$$c_{\text{máx}} = t_{1-\text{máx}} + \sum_{i=1}^{i=n-1} (t_{i,i+1-\text{máx}} + t_{i+1-\text{máx}})$$

Considerando, pues, intervalos temporales y retomando el concepto de trayectoria como la concreción temporal de un camino, ésta quedará redefinida como una función q aplicada sobre el intervalo temporal denotado por $[t_{\text{inicio}}, (t_{\text{inicio}} + c_{\text{máx}})]$. Esto es, sobre el intervalo definido por el tiempo estimado para el inicio de la trayectoria (t_{inicio}) y el resultado de sumarle a este tiempo de inicio el coste máximo estimado.

La asociación de intervalos temporales a los puntos característicos de un camino a partir de las definiciones de la Ecuación 4-2 y de la Ecuación 4-8 permite llegar a la siguiente definición de trayectoria:

Ecuación 4-10

$$T = \{p_1, (p_2, t_2), \dots, (p_n, t_n)\}$$

en la que cada punto del camino p_i lleva asociado un intervalo temporal t_i que indica el tiempo en el que se debe alcanzar ese punto durante la ejecución de la trayectoria. Este intervalo temporal asociado a cada punto incluye el coste del desplazamiento hasta ese punto y el posible tiempo consumido en el punto anterior.

Es de notar que el primer punto no lleva asociado intervalo temporal ninguno, puesto que, siendo el punto de inicio de la trayectoria, su único valor temporal asociado será el tiempo de inicio de la misma. Cuando este tiempo se confirme, se podrá hacer un ajuste en el tiempo de la trayectoria de la siguiente manera:

Ecuación 4-11

$$T = \left\{ p_1, (p_2, t_2), \dots, (p_n, t_n) \right\}$$

$$t_i = [t_{i-\min}, t_{i-\max}]$$

tiempo de inicio = t_0

↓

$$T = \left\{ (p_1, t_0), (p_2, [t_0 + t_{2-\min}, t_0 + t_{2-\max}]), \dots, (p_n, [t_0 + t_{2-\min} + \dots + t_{n-\min}, t_0 + t_{2-\max} + \dots + t_{n-\max}]) \right\}$$

Las actividades que se reconocen a partir del proceso de deducción vendrán denotadas por un camino y su coste temporal asociado, y en última instancia por la trayectoria. Así, dada una trayectoria para un móvil, se podrá realizar una estimación de su coste temporal total, de los tiempos de paso y del consumo en los puntos característicos que forman la secuencia del camino que la define. La concreción en el tiempo de las actividades del móvil, es decir, la trayectoria, es la que permitirá realizar el posterior proceso de predicción, que a su vez llevará a la replanificación en tiempo real de las actividades del móvil controlado.

Como ya se ha visto, una trayectoria está asociada a la tarea que realiza un móvil, siendo un nivel más concreto de descripción. La planificación de una tarea lleva a obtener una trayectoria concreta.

Según se consideren las tareas de un móvil, se tienen dos supuestos distintos:

- a) *Una tarea de un móvil viene definida por un camino que empieza en el mismo punto en que acabó la anterior tarea realizada.*

Este es el caso ideal en el que no existen la necesidad de que se genere una tarea que suponga una “trayectoria de enlace” entre dos tareas consecutivas. De esta forma, las trayectorias asociadas a cada una de las tareas empezarán en el punto característico en el que finalizó la trayectoria asociada a la tarea previa del móvil que la realiza.

Este supuesto elimina la posibilidad de que aparezcan puntos de paso del móvil no identificables con ninguna tarea programada por el planificador de alto nivel y que por tanto no puedan ser reconocidos.

- b) *Cuando acaba una tarea, se realiza un desplazamiento hasta el punto de inicio de la tarea siguiente.*

Se trata del caso en el que la trayectoria asociada a la tarea previa acaba en un punto característico p_i y la trayectoria asociada a la tarea siguiente debe empezar en el punto p_j . En este caso, para comenzar la segunda tarea, el móvil debe desplazarse desde el punto p_i hasta el punto p_j . Pero esta situación se reduce a la anterior si consideramos la trayectoria recorrida entre ambas tareas como la trayectoria asociada a una nueva tarea. Así, ésta pasa a considerarse como una tarea que empieza donde acabó la primera y acaba en el mismo punto en el que se inicia la segunda.

Como se aprecia, los dos supuestos conducen a la existencia de tareas siempre enlazadas, y por tanto que el móvil está continuamente realizando los desplazamientos correspondientes a las trayectorias asociadas.

EL PROCESO DEDUCTIVO

Las actividades que realiza un móvil se deducen a partir de las trayectorias que éste sigue. Cuando las trayectorias de un móvil no son conocidas a priori, se debe iniciar un proceso deductivo que conduzca a su conocimiento. De las trayectorias que sigue un móvil

no controlado en el mismo entorno, se hará con posterioridad una predicción de sus actividades que permita una planificación óptima de las actividades propias.

La base del proceso de deducción es la observación del entorno, que lleva a obtener información de los móviles no controlados en forma de las trayectorias que siguen. A partir de esta información, las actividades de estos móviles no controlados se reconocen como secuencias de puntos del entorno asociados a tiempos de paso, del mismo modo en que se definen las trayectorias.

El proceso deductivo se establece a partir de dos pasos:

1. Obtención de información de movimientos y acciones del móvil no controlado (observación del entorno).
2. Deducción de la trayectoria del móvil no controlado.

El siguiente algoritmo describe el proceso deductivo:

```
observar-y-deducir;  
  s=vacía;  
  repetir-siempre  
    obtener(p,t);  
    s=añadir(p,t);  
    coincidencias = comparar-puntos(p) ^  
                    comparar-tiempos(t);  
    si (coincidencias=1) entonces  
      (* secuencia reconocida *)  
      devuelve(s);  
      s=vacía;  
    fin-si;  
    si (coincidencias=0) entonces  
      (* nueva secuencia a reconocer *)  
      nueva-secuencia(s);  
    fin-si;  
  fin-repetir-siempre;  
fin;
```

Figura 4-4. Algoritmo del proceso deductivo.

Este algoritmo se aplica en la deducción de las actividades que realiza cada uno de los agentes externos (móviles no controlados) en el entorno.

OBSERVACIÓN DEL ENTORNO

Este primer punto del proceso deductivo viene reflejado en el algoritmo *observar-y-deducir* por la función *obtener(p,t)*, encargada de obtener información relativa a los movimientos del móvil no controlado.

En un proceso de producción industrial (personas, sistemas de transporte más o menos automatizados, vehículos autoguiados...), los agentes o dispositivos que intervienen se mueven habitualmente a lo largo de una red de pasillos y corredores que limita su libertad. Cuando ocurre esta situación se pueden dar colisiones y conflictos de acceso entre los diferentes móviles. Estas colisiones pueden ser evitadas si se obtiene la posición de los móviles por medio de algún tipo de sensorización adecuada (barreras ópticas en un conjunto establecido de puntos de paso, sensores ópticos o magnéticos dispuestos en el suelo, o incluso sistemas basados en GPS). Mediante el uso de esta sensorización, el sistema de control realiza una observación del entorno sin interferir en las actividades de los dispositivos observados.

El proceso deductivo se basa en la observación del entorno mediante el uso de sensorización adecuada. Esta observación consiste en la obtención de puntos y tiempos de paso de un móvil a lo largo del seguimiento de sus trayectorias. Esta serie de puntos de paso (a partir de este momento, cuando se mencione la expresión *puntos de paso* se asumirá que denotan puntos y tiempos de paso) es la información básica necesaria para poder estimar a partir de ella las actividades que realiza un móvil. Más que las actividades que realiza un móvil, a este nivel de concreción, lo que importa es establecer el comportamiento del mismo, entendiendo como comportamiento las trayectorias que suele realizar, el momento en que las realiza y los tiempos que emplea para ello. Esta información nos debe ayudar a caracterizar plenamente una trayectoria, de cara a poder realizar en un futuro la planificación temporal de las acciones de otros móviles.

La obtención de la secuencia de puntos de paso se puede realizar con cualquier tipo de sensorización. Así, existen diversos métodos para obtener puntos de paso de un móvil:

- *Sensores de infrarrojos*: Utilizando sensores infrarrojos de bajo coste, instalando un emisor en los puntos de paso del entorno, por ejemplo en el suelo del entorno de trabajo del móvil, y un receptor en la parte baja del propio móvil.

- *Sensores ópticos*: Se pueden utilizar sensores fotoeléctricos para identificar puntos de paso de un móvil, como las barreras fotoeléctricas o los códigos de barras. El uso de códigos de barras aporta la ventaja de poder identificar mediante la lectura del código correspondiente el móvil concreto que pasa por este punto, en el caso de tener más de un móvil en el entorno.
- *GPS*: Este sistema de sensorización para obtener la posición en todo momento de un móvil es adecuado en espacios abiertos, mientras que en entornos cerrados se muestra menos preciso, requiriendo la instalación de módulos repetidores.

En todos los casos, la obtención de la serie de puntos de paso para un móvil con comportamiento no controlado no presenta ninguna injerencia en la realización de sus propias tareas.

Conforme se van obteniendo datos relativos a un móvil no controlado, se va construyendo una secuencia de puntos y tiempos de paso asociados. En el algoritmo anterior, es la función $s = \text{añadir}(p, t)$ la que se encarga de este proceso.

DEDUCCIÓN DE LAS TRAYECTORIAS

Conforme se va obteniendo una secuencia de puntos de paso con sus correspondientes tiempos asociados (p, t) , ésta se va comparando con las trayectorias conocidas para deducir la que sigue el móvil no controlado en la actualidad. Esta es la base del proceso de deducción, planteado en dos niveles distintos y realizado por las funciones `comparar-puntos` y `comparar-tiempos`.

1. Reconocimiento del camino (puntos de paso).

Se reconoce en primer lugar el camino geométrico mediante la comparación de los puntos de paso del móvil con los puntos de paso establecidos que componen el total de trayectorias conocidas. Este primer paso lo realiza la función `comparar-puntos`. En realidad, si separamos los puntos de paso de los tiempos de paso, este proceso se reduce a encontrar una cadena de caracteres (puntos de paso) dentro de una cadena mayor (el camino asociado a una trayectoria conocida). Esta búsqueda de una subcadena dentro de una

cadena mayor es un problema muy caracterizado en algorítmica [Wirth, 1987].

2. *Reconocimiento final de la trayectoria (tiempos de paso).*

Se compara el tiempo de paso para un punto dado con el intervalo temporales establecido para el mismo punto en la posible trayectoria a reconocer. Este segundo paso lo realiza la función `comparar-tiempos`.

Así, el reconocimiento de una trayectoria se basa en el establecimiento de comparaciones a dos niveles (puntos y tiempos de paso). La primera comparación reconoce un camino geométrico, mientras que la segunda lleva al reconocimiento final de una trayectoria conocida. Si la primera comparación es negativa, se abandona el reconocimiento de la actividad en cuestión, en cambio, si es positiva, se hace la comparación del segundo nivel.

Ambas funciones, `comparar-puntos` y `comparar-tiempos`, devuelven como resultado el número de comparaciones exitosas. En el primer caso, la función `comparar-puntos` devuelve el número de caminos geométricos que van coincidiendo con los conocidos, mientras que la segunda función devuelve las trayectorias finales que coinciden a partir de los caminos reconocidos en primera instancia.

Cuando el valor que devuelve la segunda función es 1, ello significa que se ha reconocido la trayectoria que está realizando el agente sobre el que se aplica este proceso deductivo. En el caso de que no se reconozca la trayectoria que realiza este agente externo, se debe considerar como una nueva trayectoria y debe ser añadida al histórico de trayectorias conocidas. Este proceso de reconocimiento de una nueva trayectoria y añadirla al histórico, lo realiza la función `nueva-secuencia`.

Esta forma de actuar implica la existencia de un histórico que debe actualizarse con la inclusión de nuevas trayectorias reconocidas. El mantenimiento de esta base de datos se aborda en un punto posterior de este mismo capítulo.

A continuación se describe el funcionamiento del proceso de reconocimiento de trayectorias a partir de un ejemplo concreto. Supongamos que se tiene el siguiente entorno ejemplo con 20 puntos característicos distintos:

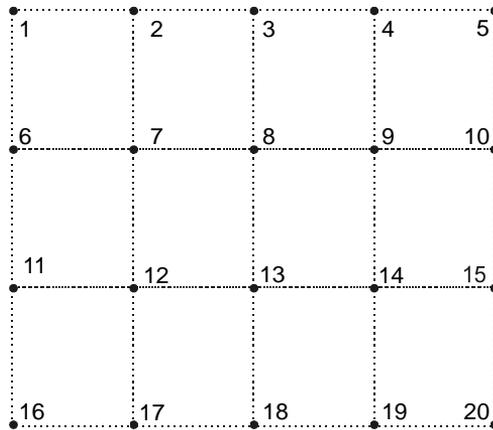


Figura 4-5. Entorno ejemplo con 20 puntos característicos y sus conexiones.

Supongamos también que para este entorno, se tienen las siguientes trayectorias conocidas:

Ecuación 4-12

$$a = \{p2, (p7, [2,4]), (p12, [2,4]), (p17, [2,4])\}$$

$$b = \{p2, (p7, [2,4]), (p6, [2,4]), (p11, [2,4]), (p12, [2,4]), (p13, [2,4]), (p14, [2,4]), (p15, [2,4])\}$$

$$c = \{p2, (p7, [3,6]), (p12, [3,6]), (p11, [3,6])\}$$

$$d = \{p17, (p18, [4,8]), (p19, [4,8]), (p20, [4,8])\}$$

Estas trayectorias están descritas de acuerdo con la expresión definida en la Ecuación 4-10, y se pueden ver en la siguiente figura:

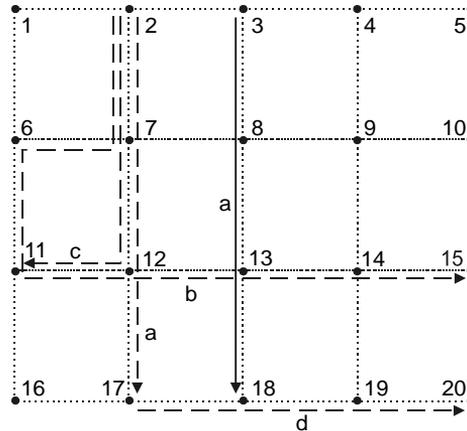


Figura 4-6. Entorno ejemplo con las posibles trayectorias marcadas. En trazo continuo aparece la trayectoria del móvil controlado, mientras que en trazo discontinuo se representan las posibles trayectorias del móvil no controlado.

Se inicia entonces un proceso deductivo sobre las actividades que realiza un agente externo de manera que la función $\text{obtener}(p, t)$ obtiene como resultado la pareja compuesta por el punto p_2 con tiempo de paso $t=0$. Asumiendo este tiempo de paso como el tiempo de inicio de una trayectoria, las trayectorias conocidas se pueden ajustar en el tiempo a partir de las expresiones de la ecuación Ecuación 4-11 de la siguiente manera:

Ecuación 4-13

$$\begin{aligned}
 a &= \{p_2, (p_7, [2,4]), (p_{12}, [4,8]), (p_{17}, [6,12])\} \\
 b &= \{p_2, (p_7, [2,4]), (p_6, [4,8]), (p_{11}, [6,12]), (p_{12}, [8,16]), (p_{13}, [10,20]), \\
 &\quad (p_{14}, [12,24]), (p_{15}, [14,28])\} \\
 c &= \{p_2, (p_7, [3,6]), (p_{12}, [6,12]), (p_{11}, [9,18])\} \\
 d &= \{p_{17}, (p_{18}, [4,8]), (p_{19}, [8,16]), (p_{20}, [12,24])\}
 \end{aligned}$$

Estas trayectorias ajustadas en el tiempo muestran periodos de tiempo asociados a cada punto, indicando que se estima que el punto correspondiente se alcanzará entre los tiempos mínimo y máximo indicados en el intervalo temporal.

La siguiente tabla muestra una traza para un ejemplo de funcionamiento del algoritmo *observar-y-deducir* a la hora de deducir la trayectoria que sigue un agente externo:

obtener (p, t)	comparar-puntos (p)	comparar-tiempos (t)	resultados	trayectoria reconocida
(p2,0)	coinc.=(a,b,c) nº coinc.=3	coinc.=(a,b,c) nº coinc.=3	posibles=(a,b,c) descart.=(d)	ninguna
(p7,2.5)	coinc.=(a,b,c) nº coinc.=3	coinc.=(a,b) nº coinc.=2	posibles=(a,b) descart.=(c,d)	ninguna
(p6,5)	coinc.=(b) nº coinc.=1	coinc.=(b) nº coinc.=1	posibles=(b) descart.=(a,c,d)	b

Tabla 4-1. Traza del funcionamiento del algoritmo observar-y-deducir para la deducción de una trayectoria de un agente externo.

RECONOCIMIENTO DE NUEVAS TRAYECTORIAS

Cuando no se consigue reconocer ninguna trayectoria de las contenidas en el histórico, a partir de la información que se obtiene de la observación del entorno, es posible que nos encontremos ante una nueva trayectoria no reconocida hasta este momento. En este caso, el sistema debe actuar reconociendo la actividad como nueva e incluyéndola en el histórico de trayectorias. Se trata, en definitiva, de un proceso de aprendizaje de nuevas trayectorias.

El aprendizaje de los puntos de paso para una actividad determinada es un proceso difícil de abordar. Dejando aparte los tiempos de paso, y considerando sólo los puntos, el proceso consiste en considerar una serie global de puntos de paso, resultado de la ejecución de una actividad determinada de un agente externo a lo largo de un tiempo determinado, y deducir de esta serie global la secuencias de puntos de paso que representan las trayectorias que va realizando dicho agente. Este proceso no es en absoluto trivial puesto que unido al hecho de que hay un desconocimiento previo de las actividades que puede realizar el móvil, se tiene el grave inconveniente de éstas pueden ser de duración variable, con lo que pueden aparecer secuencias de puntos de paso de longitudes diferentes. El problema consiste, pues, en encontrar secuencias de puntos de paso de distintas longitudes en una sucesión de puntos de paso de longitud mayor (la serie global de puntos de paso que se va obteniendo en la observación del entorno).

En realidad, el problema se podría enmarcar dentro del ámbito del reconocimiento de patrones, ampliamente tratado en aplicaciones como el reconocimiento de caracteres, las redes neuronales o el reconocimiento de imágenes en visión artificial [Fu, 1977] [Bunke y Sanfeliu, 1990] [Traven, 1993] [Sergeyev y Chernov, 1997].

Un método simple para reconocer nuevas trayectorias es considerar que se inicia una nueva trayectoria cada vez que se repite un carácter en la serie global de puntos de paso. El siguiente algoritmo básico describe este método:

```
nueva-secuencia(s);  
  repetir  
    obtener(p,t);  
    s=añadir(p,t);  
  hasta (p-repetido);  
  añade(s);  
  s=vacia;  
  s=añadir(p,t);  
fin;
```

Figura 4-7. Algoritmo básico para reconocer nuevas trayectorias.

El algoritmo obtiene los sucesivos puntos de paso (*p*) por los que circula el móvil construyendo la secuencia (*s*) a reconocer. Cuando se detecta por segunda vez el paso del móvil por un punto anteriormente reconocido (*p-repetido*), se considera la secuencia completa. Este último punto reconocido se considera el último de la secuencia reconocida y el primero de la nueva secuencia a reconocer .

La siguiente figura refleja gráficamente este proceso:

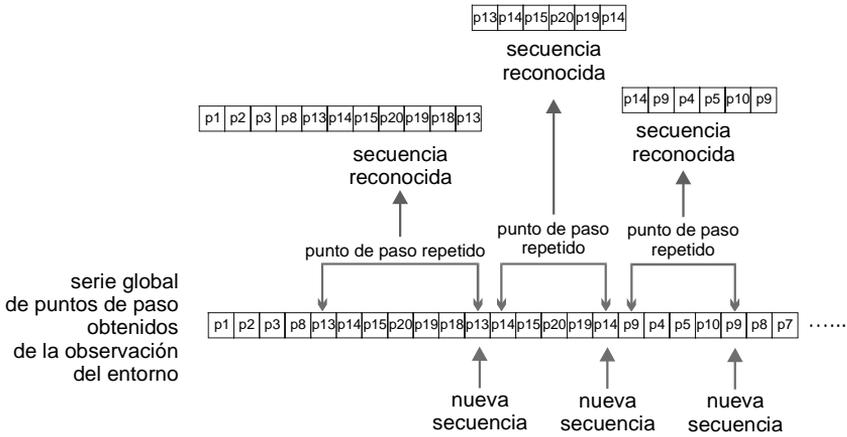


Figura 4-8. Proceso de reconocimiento de nuevas trayectorias como secuencias de puntos de paso, considerando que se inicia una nueva secuencia al pasar por segunda vez por un punto de paso dado.

Este proceso, de gran sencillez, reconoce de manera completa una trayectoria cuando ésta no supone el paso del robot móvil por un mismo punto del entorno dos veces. En cambio, cuando una trayectoria implica pasar por un mismo punto más de una vez, reconoce una secuencia de puntos de paso dentro de una trayectoria completa. Por ejemplo, la trayectoria representada por el siguiente camino (obviando los tiempos de paso asociados):

Ecuación 4-14

$$t1 = \{p1, p2, p3, p4, p9, p8, p3, p2, p1\}$$

sería reconocida a partir de las siguientes secuencias de puntos de paso:

Ecuación 4-15

$$s1 = \{p1, p2, p3, p4, p9, p8, p3\}$$

$$s2 = \{p3, p2, p1\}$$

Afinando más todavía, el algoritmo básico de reconocimiento llevaría a incluir total o parcialmente en la segunda secuencia la siguiente trayectoria que se realizara. Por ejemplo, supongamos que el agente externo realiza primero la actividad especificada como la trayectoria t1 de la Ecuación 4-14 y luego la actividad descrita por la siguiente trayectoria t2 (obviando en ambos casos los tiempos de paso):

Ecuación 4-16

$$t2 = \{p1, p6, p7, p2, p1\}$$

En este caso, la serie global de puntos de paso sería como sigue

Ecuación 4-17

$$p1, p2, p3, p4, p9, p8, p3, p2, p1, p6, p7, p2, p1$$

y el proceso descrito reconocería las siguientes secuencias:

Ecuación 4-18

$$s1 = \{p1, p2, p3, p4, p9, p8, p3\}$$

$$s2 = \{p3, p2, p1, p6, p7, p2\}$$

$$s3 = \{p2, p1\}$$

Así, en vez de reconocerse las trayectorias completas que sigue un móvil, se reconocen partes de las mismas o secuencias que pueden incluir más de una trayectoria. Pero este aspecto no supone un inconveniente porque el algoritmo si que reconoce secuencias de puntos de paso perfectamente aisladas, lo que es suficiente para utilizarlas en el proceso posterior de predicción.

Al tiempo que se va reconociendo una secuencia de puntos que sigue un agente externo al realizar una actividad, el algoritmo va asociando tiempos a cada uno de los puntos de paso para configurar la trayectoria que representa dicha actividad. Este proceso queda descrito en la siguiente figura:

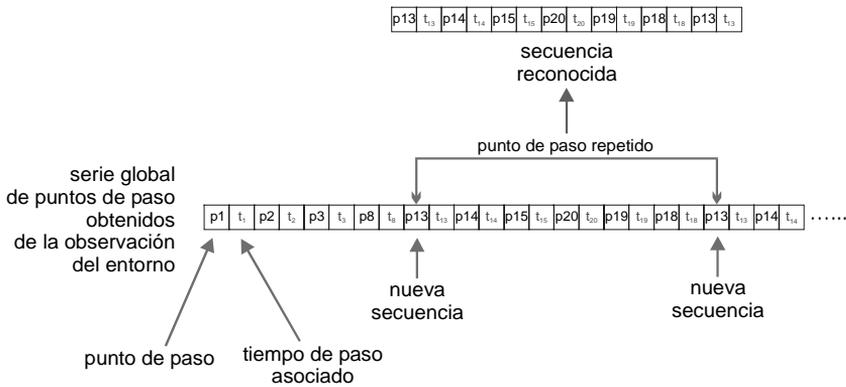


Figura 4-9. Proceso de reconocimiento completo de una secuencia de puntos de paso con sus tiempos asociados.

Puesto que en la definición de trayectoria de la Ecuación 4-10 aparecen intervalos temporales asociados a cada punto y no un único valor temporal, la secuencia reconocida debe transformarse, convirtiendo los tiempos de paso en intervalos temporales. Una manera sencilla de obtener el intervalo temporal asociado a cada punto de una secuencia es estableciendo un factor de incertidumbre arbitrario fijo o bien hacerlo depender de la distancia entre dos puntos consecutivos. Esta última alternativa resulta bastante realista, haciendo que a mayor distancia entre dos puntos, la incertidumbre en cuanto al tiempo empleado para el desplazamiento sea mayor.

En cualquier caso, partiendo de una secuencia de puntos y tiempos de paso obtenida como

Ecuación 4-19

$$\{(p1, t_1), (p2, t_2), \dots, (pn, t_n)\}$$

se calculan los tiempos empleados en el desplazamiento entre dos puntos de la siguiente manera

Ecuación 4-20

$$\begin{aligned} (p1 \rightarrow p2) &\Rightarrow t_{1,2} = t_2 - t_1 \\ (p2 \rightarrow p3) &\Rightarrow t_{2,3} = t_3 - t_2 \\ &\dots \\ (p(n-1) \rightarrow pn) &\Rightarrow t_{(n-1),n} = t_n - t_{n-1} \end{aligned}$$

quedando la trayectoria final reconocida de la siguiente forma

Ecuación 4-21

$$T = \{p1, (p2, [t_{1,2} - \Delta_{1,2}, t_{1,2} + \Delta_{1,2}]), \dots, (pn, [t_{(n-1),n} - \Delta_{(n-1),n}, t_{(n-1),n} + \Delta_{(n-1),n}])\}$$

siendo los valores $\Delta_{i,j}$ las incertidumbres asociadas al desplazamiento entre los puntos p_i y p_j de la secuencia original obtenida.

En un caso real, el móvil realiza un número limitado de actividades, representado por un número también limitado de trayectorias. El procedimiento visto para el reconocimiento de nuevas trayectorias reconocería un número también limitado de secuencias, mayor en principio que el número real de trayectorias.

Supongamos el caso más general, aquel en el que todos los puntos de paso en un entorno determinado están conectados entre sí. El número total de las posibles secuencias a reconocer de una longitud determinada viene dado por la consideración de las *variaciones sin repetición* de m elementos tomados de n en n , siendo m el número total de puntos de paso del entorno, y n la longitud de la secuencia de puntos de paso a reconocer:

Ecuación 4-22

$$V_{m,n} = m \times (m-1) \times (m-2) \times \dots \times (m-n+1) \quad n \leq m$$

Supongamos un entorno en el que existan 20 posibles puntos de paso ($m=20$). Suponiendo el caso más general en que todos los puntos de paso están conectados entre sí, y con una longitud de secuencia mínima (2 puntos de paso, $n=2$), el número total de secuencias posibles a reconocer sería

Ecuación 4-23

$$V_{20,2} = 20 \times \dots \times (20 - 2 + 1) = 20 \times 19 = 380$$

Si las actividades del agente externo no incluyen puntos de paso repetidos, la longitud máxima de la secuencia a reconocer sería $n=m=20$. En este caso, el número total de secuencias posibles vendría dado por

Ecuación 4-24

$$V_{20,20} = 20 \times \dots \times (20 - 20 + 1) = 20 \times 19 \times \dots \times 1 = 2,43 \times 10^{18}$$

Se puede ver que éste sería el peor caso, en el que el número total de secuencias posibles combinando el orden de las posibles tareas sería $m!$. Si a esto le añadimos la característica de que las secuencias a

reconocer pueden tener longitud variable, el número total de posibles secuencias (N_{sec}) viene dado por la siguiente expresión

$$\begin{aligned} N_{sec} &= m + [m \cdot (m-1)] + [m \cdot (m-1) \cdot (m-2)] + \dots + \\ &+ [m \cdot (m-1) \cdot (m-2) \cdot \dots \cdot (m-m+1)] = \\ &= \sum_{i=1}^m [m \cdot (m-1) \cdot (m-2) \cdot \dots \cdot (m-i+1)] \end{aligned}$$

Ecuación 4-25

que podemos considerar como acotada por

$$N_{sec} = m \cdot m!$$

Ecuación 4-26

Esta última expresión nos daría el número máximo de secuencias que se pueden reconocer en un entorno con m puntos de paso.

En entornos industriales reales, un robot realiza un número reducido de actividades, lo cual hace factible el reconocimiento de secuencias a través de un método simple como el descrito. Sólo secuencias de puntos de paso que representen actividades imprevistas pueden afectar a este proceso, aunque de una manera poco significativa.

MANTENIMIENTO DEL HISTÓRICO DE TRAYECTORIAS

El proceso de reconocimiento de trayectorias se apoya, como se ha dicho, en la existencia de un histórico de trayectorias conocidas, al que se recurre para establecer las comparaciones que nos permiten deducir el comportamiento de un agente externo. El histórico se crea a partir de todas las trayectorias reconocidas como secuencias de puntos del entorno con intervalos temporales asociados. Cuando aparece una nueva secuencia, bien por medios directos, bien como resultado del proceso de aprendizaje, ésta debe añadirse al histórico.

Aparte de la propia descripción de la trayectoria según la expresión de la Ecuación 4-10, en el histórico se debe guardar una *etiqueta* que permita su identificación junto con un valor que indique el número total de veces que se ha reconocido esta trayectoria a lo largo del proceso de deducción.

Este último valor es crucial para el mantenimiento del histórico, de manera que nos va a servir para discriminar entre trayectorias posibles, susceptibles de que sean ejecutadas en cualquier momento por cualquier móvil, y trayectorias cuya ejecución ha resultado ser anómala, y que por tanto son improbables que vuelvan a darse. Se

trata del caso en el que durante la ejecución de una trayectoria, el móvil ha sufrido algún imprevisto (evitación de una colisión, teniendo que cambiar su camino geométrica, por ejemplo) que le hubiera ocasionado algún tipo de retardo y que haya hecho que los tiempos de paso por un determinado punto cambiaran, saliéndose fuera del intervalo preestablecido (actividades con idéntico camino pero distintos tiempos de paso). Esta trayectoria no se correspondería con ninguna de las registradas y pasaría a considerarse como una nueva, almacenándose por tanto en el histórico. Con la improbabilidad de que volviera a darse una ejecución similar (mismo inconveniente, mismo retardo), conforme aumentara el número de trayectorias ejecutadas, el valor asociado a esta trayectoria anómala disminuiría, y por debajo de un cierto umbral, podría finalmente eliminarse del histórico de trayectorias.

CONCLUSIONES

El conocimiento de las actividades que realiza un móvil externo es imprescindible para abordar la replanificación de las trayectorias de un robot móvil. Cuando estas actividades no son conocidas de antemano, es preciso, pues, iniciar un proceso de deducción que permita ir conociendo en tiempo real las trayectorias que sigue un agente externo.

En este capítulo se aborda este proceso a partir de los algoritmos básicos de deducción, principal aportación dentro del trabajo de investigación. Así, el algoritmo *observar-y-deducir*, introduce un método básico para reconocer trayectorias a partir de un histórico de las actividades conocidas que realiza un móvil externo. La comparación de puntos y tiempos que se van obteniendo mediante la observación del entorno, lleva a establecer cuál es la trayectoria que recorre en un momento determinado un móvil externo.

Cuando no se puede reconocer dicha trayectoria, debido que no está contemplada en el histórico, el algoritmo *nueva-secuencia* la reconoce en forma de una secuencia de puntos y tiempos de paso y la añade al histórico.

Cada vez que se reconoce una secuencia, el histórico se actualiza a partir de un índice que identifica el número de veces que se ejecuta una trayectoria. Cuando este índice permanece bajo o no se actualiza a lo largo de un periodo de tiempo determinado, se considera que la secuencia asociada se ejecutó como fruto de una eventualidad

(evitación de una colisión, cambio de dirección por estar un pasillo bloqueado, etc.) y se elimina del histórico.

El algoritmo *observar-y-deducir* se encuentra dentro del algoritmo básico de replanificación *replanificar-trayectoria*, también introducido en este capítulo y que se completa con el algoritmo de predicción que se describirá en el siguiente capítulo.

✱

5 *Replanificación: El proceso predictivo*

Replanificación: El proceso predictivo

En este capítulo se presenta el segundo principio del proceso de replanificación: la predicción. En realidad se trata del punto más importante de la replanificación, una vez se conocen las distintas trayectorias que siguen todos los móviles del entorno del proceso productivo. El método que se presenta se asienta sobre el razonamiento temporal realizado entre las trayectorias estimadas que siguen los móviles, aplicándose un ajuste de velocidades sobre el móvil que se pretende replanificar.

Bajo estas premisas, este capítulo presenta el procedimiento básico de predicción a partir de los correspondientes algoritmos. Estos algoritmos se asientan en unas reglas de comportamiento que dependen del tipo de interacción posible entre las trayectorias (punto compartido, segmento compartido en el mismo sentido, segmento compartido en sentido contrario).

Un estudio completo de los distintos caso posibles muestra la aplicación de las reglas de comportamiento y la replanificación correspondiente de las trayectorias de cara a evitar posibles las interacciones.

Replanificación: El proceso predictivo

EL PROCESO PREDICTIVO

Una vez se conoce la trayectoria que ejecuta el agente externo que comparte el entorno con el móvil planificado, empieza un proceso predictivo que conduce a la replanificación sobre la trayectoria original del móvil controlado. La misión del este proceso predictivo es determinar nuevas acciones a realizar por parte del móvil que se controla, coordinando en lo posible, la trayectoria propia con la del agente externo.

Así, la replanificación se basa en la predicción de acciones futuras en forma de tiempos de paso por puntos característicos del entorno. Estas acciones futuras se basarán en el establecimiento de intervalos temporales en los que el agente externo podrá encontrarse en determinado punto del entorno.

El proceso predictivo se basa en la lógica temporal ya mencionada en el capítulo 2 de esta tesis doctoral, y más concretamente, en las relaciones temporales definidas en [Allen y Ferguson, 1994] y en los trabajos relativos al razonamiento temporal expuestos en [Crespo *et al.*, 1994] y [Onaindía, 1997]. El razonamiento temporal parte de la idea de que toda variable lleva asociada un intervalo temporal definido por los tiempos de inicio y fin de dicho intervalo. Una variable con valor de duración cero es aquella cuyo intervalo temporal asociado tiene igual tiempo de inicio que de finalización. En general, dentro de este intervalo temporal se dan distintos valores para la variable que se está controlando, cumpliéndose el esquema de la siguiente figura:

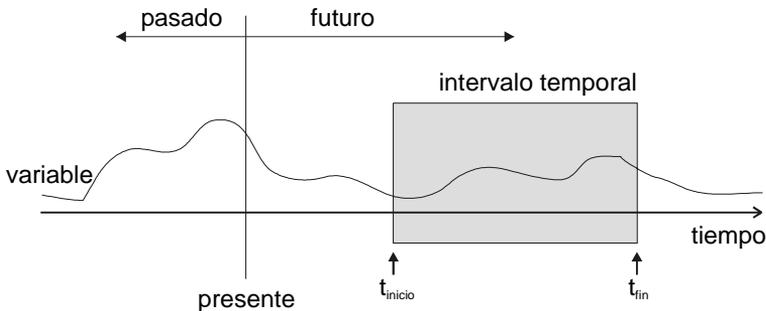


Figura 5-1. Intervalo temporal asociado a una variable.

Los intervalos temporales vienen determinados por una incertidumbre asociada al valor que toma la variable. Esta incertidumbre, a su vez, viene definida bien por el valor en sí de la propia variable (es el caso en el que los valores llevan asociado un factor de fiabilidad) o por la ocurrencia temporal de distintos valores (el caso en el que el valor que alcanza una variable depende de factores externos). La incertidumbre determinará la duración del intervalo temporal.

La consideración de intervalos temporales asociados a las variables hace necesario trabajar con predicciones, es decir, valores futuros estimados para una variable determinada dentro de su intervalo temporal. Estas predicciones permiten planificar las acciones de futuro necesarias para evitar posibles problemas en situaciones en las que los valores de ciertas variables dependen de otras.

La estimación temporal, aplicada a la replanificación de trayectorias de móviles, permite calcular las actividades de los móviles condicionadas a la información de que se dispone acerca de actividades que realizan otros agentes. Las acciones o trayectorias de los móviles se pueden decidir con antelación utilizando valores futuros (por ejemplo, estimaciones de acciones en forma de tiempos de paso por zonas compartidas) y razonando a partir de los mismos. Cuando las funciones que se estiman para un agente determinado se cumplen, se pueden confirmar actividades futuras del propio móvil, así como nuevas suposiciones hechas para el agente, y con ello realizar nuevas estimaciones.

Así, para una actividad determinada, los tiempos de paso del agente que la realiza por los puntos que la configuran llevan asociados intervalos temporales, tal y como se ha visto en la descripción de las trayectorias en el capítulo anterior. En esta descripción, la incertidumbre asociada a los intervalos temporales depende (en gran medida) de la velocidad del móvil que realiza la tarea y de la distancia entre dos puntos consecutivos de la trayectoria.

En este capítulo se describe el proceso predictivo que permite estimar futuras acciones y replanificar las propias de cara a optimizar su ejecución, evitando posibles colisiones. A través de unas reglas de actuación deducidas de la lógica temporal, se estudian los casos básicos, demostrando su aplicación práctica como técnica de replanificación de trayectorias.

El proceso predictivo se basa en el estudio de las relaciones temporales entre la trayectoria del móvil a replanificar y la del agente que actúa en el mismo entorno. Los resultados de este razonamiento

temporal se utilizan para replanificar la trayectoria del móvil mediante el ajuste de velocidades.

La clasificación de los agentes hecha en el capítulo anterior sirve para considerar **móviles controlados** a los agentes con comportamiento controlado, mientras que los agentes no controlados con comportamiento fijo y los que tienen un comportamiento aleatorio se pueden englobar bajo la denominación única de **móviles no controlados**. El proceso predictivo siempre asume que las trayectorias que siguen los móviles no controlados tienen una mayor prioridad, con lo que la replanificación de la trayectoria únicamente se aplica sobre el móvil controlado.

Estas premisas llevan a la elaboración de un algoritmo que resume a *grosso modo* los pasos que sigue el proceso predictivo y que se presenta a continuación:

```
predecir;  
  comparar-secuencias(s1,s2);  
  si (nº-de-interacciones=0) entonces  
    devuelve(nulo);  
  si no  
    para (cada-interacción) hacer  
      repetir  
        para (cada-observación)  
          ajustar(s1,s2);  
        fin-para;  
      hasta (pasar-la-interacción);  
    fin-para;  
  fin-si;  
fin;
```

Figura 5-2. Algoritmo del proceso predictivo.

El algoritmo del proceso predictivo está basado en dos pasos principales:

1. El establecimiento de posibles interacciones entre las actividades del móvil controlado y las del móvil no controlado, realizado por la función *comparar-secuencias*. Esta función se encarga de hacer una primera comparación, detectando de manera global si existen interacciones entre ambas trayectorias. Si no hay interacciones, no cabe ninguna

replanificación, mientras que si se detecta alguna interacción, se debe aplicar la replanificación según el tipo.

2. La replanificación de la trayectoria del móvil controlado a partir del ajuste de velocidades, implementado en la función `ajustar`. Esta función se llama cada vez que se produce una observación del entorno, esto es, cada vez que se confirma el paso de uno de los dos móviles por un punto característico del entorno. Así, a través de la confirmación de los tiempos de paso, la función ajusta los nuevos intervalos, establece las nuevas interacciones y confirma o revoca la replanificación de la trayectoria del móvil controlado mediante el ajuste de velocidades pertinente.

El algoritmo se lanza cuando se programa una actividad para un móvil controlado, ejecutándose en paralelo con la trayectoria que sigue. El primer paso (la comparación inicial de las trayectorias o secuencias que siguen los móviles del entorno) se realiza sólo una vez, al principio, estableciéndose las posibles interacciones.

La observación del entorno es el evento que espera el algoritmo para hacer un ajuste de velocidades a partir de la estimación temporal de las secuencias actualizadas en se instante de tiempo. Así, cada vez que se actualiza el paso de uno de los móviles del entorno por un punto sensorizado, se genera el evento que lanza la función `ajustar`, actualizándose el razonamiento temporal y aplicándose, si cabe, la replanificación de la trayectoria del móvil controlado.

PREDICIÓN DE INTERACCIONES

El establecimiento de interacciones a nivel de puntos de paso entre la trayectoria o secuencia que sigue el móvil controlado (s_1) y la que sigue un móvil no controlado en el mismo entorno (s_2) es el primer paso que realiza en proceso predictivo. La función `comparar-secuencias` se encarga de realizar este proceso a partir de los tipos posibles de interacciones entre dos trayectorias. Estos son:

1. Punto compartido.

Es el caso en el que dos trayectorias comparten un único punto característico del entorno. Esto significa en la práctica, que ambas trayectorias se cruzan en un punto característico del entorno.

Esta característica se define a partir de la siguiente expresión

Ecuación 5-1

$$s1, s2 \in S : \\ \exists pi: (pi \in s1) \wedge (pi \in s2)$$

donde S es el conjunto de todas las secuencias (trayectorias) posibles, s1 y s2 son las secuencias a comparar, y pi es un punto compartido en ambas definiciones.

2. Segmento compartido en el mismo sentido.

Se trata del caso en el que dos trayectorias comparten una sub-secuencia de puntos característicos del entorno en el mismo orden de definición. Esta situación representa en la práctica que ambos móviles comparten una parte del camino recorrido en el mismo sentido de avance.

Esta característica se representa mediante la siguiente expresión:

Ecuación 5-2

$$s1, s2 \in S : \\ \exists s_s = \{pi, pj, \dots, pk\}: (s_s \subset s1) \wedge (s_s \subset s2)$$

siendo $s_s = \{pi, pj, \dots, pk\}$ la sub-secuencia compartida en ambas secuencias (s1 y s2) en el mismo sentido de avance.

3. Segmento compartido en sentido opuesto.

El caso en el que se comparte una misma sub-secuencia de puntos pero definidas en orden inverso. Esta situación representa en la práctica que ambos móviles comparten una parte del camino recorrido en sentido de avance opuesto.

La siguiente expresión representa esta interacción:

Ecuación 5-3

$$s1, s2 \in S : \\ \exists \left[(s_s = \{pi, pj, \dots, pk\}) \wedge (\overline{s_s} = \{pk, \dots, pj, pi\}) \right]: (s_s \subset s1) \wedge (\overline{s_s} \subset s2)$$

donde $s_s = \{pi, pj, \dots, pk\}$ y $\overline{s_s} = \{pk, \dots, pj, pi\}$ son las sub-secuencias en s1 y s2, respectivamente.

Estas tres condiciones se deben comprobar ordenadamente, de manera que siempre se buscan en primer lugar interacciones del tipo

de segmento compartido, bien en el mismo sentido, bien en sentido contrario. De no encontrarse este tipo de interacciones, se buscan puntos compartidos entre secuencias. Esto se debe hacer así puesto que un punto compartido está incluido en la definición del segmento compartido, con lo que hay que evitar la confusión de detectar un interacciones del tipo punto compartido dentro de un segmento compartido.

Así, el algoritmo que define la función `comparar-secuencias` será como sigue:

```
comparar-secuencias(s1,s2);
  n°-de-interacciones=0;
  interacciones=vacío;
  para (s1,s2) hacer
    si (condición2) entonces
      interacción=segmento-mismo-sentido;
      añadir(interacción,interacciones);
      incrementar(n°-de-interacciones);
      eliminar(interacción,s1,s2);
    fin-si;
  fin-para;
  para (s1,s2) hacer
    si (condición3) entonces
      interacción=segmento-sentido-contrario;
      añadir(interacción,interacciones);
      incrementar(n°-de-interacciones);
      eliminar(interacción,s1,s2);
    fin-si;
  fin-para;
  para (s1,s2) hacer
    si (condición1) entonces
      interacción=punto;
      añadir(interacción,interacciones);
      incrementar(n°-de-interacciones);
      eliminar(interacción,s1,s2);
    fin-si;
  fin-para;
  devuelve(interacciones,n°-de-interacciones);
fin;
```

Figura 5-3. Función `comparar-secuencias`.

En el algoritmo anterior se aprecia como se recorren las secuencias, buscando en primer lugar segmentos compartidos en el mismo sentido (condición2), en segundo lugar segmentos compartidos en sentido contrario (condición3), y por último, puntos compartidos (condición1). Cada vez que se encuentra una interacción de cualquier tipo, se añade el tipo de interacción y su definición a un vector de interacciones (interacciones) y se incrementa una variable que guarda el número total de interacciones encontradas (nº-de-interacciones). Para evitar confusiones en las siguientes búsquedas de interacciones, se eliminan de las secuencias los puntos que definen la interacción encontrada.

El tipo de interacción dictará con posterioridad las actuaciones a realizar en el segundo paso, es decir, en la replanificación de la trayectoria del móvil controlado mediante el ajuste de velocidades.

REPLANIFICACIÓN POR AJUSTE DE VELOCIDADES

La función `ajustar` estudia las relaciones temporales en las trayectorias entre las que se han detectado interacciones cada vez que se realiza una nueva observación del entorno. Este estudio permitirá establecer las acciones de replanificación necesarias para que la ejecución de ambas trayectorias resulte eficiente, evitándose las posibles colisiones. La replanificación de la trayectoria del móvil controlado se realiza a través de un ajuste de velocidades, que se desprende del razonamiento temporal entre los intervalos de tiempo asociados a las interacciones entre ambas trayectorias.

El algoritmo básico de la función `ajustar` es como sigue:

```
ajustar(s1,s2);  
  ajustar-tiempos(s1,s2);  
  caso (interacción)  
    punto: regla1(s1,s2);  
    segmento-mismo-sentido: regla2(s1,s2);  
    segmento-sentido-contrario: regla3(s1,s2);  
  fin-caso;  
fin;
```

Figura 5-4. Algoritmo básico de la función `ajustar`.

La función actúa de la siguiente manera:

1. Reajusta los tiempos de los intervalos temporales para ambas secuencias, una vez confirmados los tiempos de paso en la última observación del entorno.
2. Según el tipo de interacción aplica unas reglas de replanificación a través del ajuste de velocidades. Estas reglas establecen las relaciones temporales existentes para la interacción y replanifican la trayectoria del móvil controlado, tal y como se detalla más adelante.

El ajuste final de velocidades depende de las estimaciones de futuro respecto de las posibles interacciones. Así, la función estima interacciones y deduce nuevas velocidades para el móvil controlado. Conforme avanza el tiempo, las estimaciones se van confirmando, al tiempo que se mejora la deducción de las nuevas velocidades. Finalmente, el ajuste de velocidades se aplica en un punto cercano a la zona de interacción de forma que el móvil pueda reaccionar, replanificando su trayectoria para evitar la posible colisión.

El ajuste de velocidades depende del tipo de interacción prevista, desprendiéndose su valor de los cálculos temporales realizados. Para cada caso, se aplican unas reglas que se deducen de las reglas generales del razonamiento temporal, visto en el segundo capítulo de esta tesis doctoral.

ESTUDIO DE LOS CASOS

Partiendo de los posibles tipos de interacción entre dos trayectorias en un mismo entorno (la que sigue el móvil controlado y la que sigue un móvil no controlado en el mismo entorno), la función *ajustar* aplica unas reglas generales que permiten determinar un ajuste de velocidades para el móvil controlado. A través de este ajuste de velocidades, el móvil controlado adelanta o retrasa el paso por las zonas que determinan la interacción posible, evitando probables colisiones con el móvil no controlado.

EL CASO DEL PUNTO COMPARTIDO

Se asume que p_i es un punto característico del entorno compartido en la definición de dos secuencias (trayectorias) s_1 y s_2 , de acuerdo con la condición expresada en la Ecuación 5-1. Este punto compartido tendrá asociados unos intervalos temporales en cada secuencia. Esta situación queda definida en la siguiente expresión:

Ecuación 5-4

$$s1 \Rightarrow pi \Rightarrow t_1 = [t_{1-\min}, t_{1-\max}] = [t_{i-\min}^{s1}, t_{i-\max}^{s1}]$$

$$s2 \Rightarrow pi \Rightarrow t_2 = [t_{2-\min}, t_{2-\max}] = [t_{i-\min}^{s2}, t_{i-\max}^{s2}]$$

donde t_1 y t_2 son los intervalos temporales asociados a pi en $s1$ y $s2$, respectivamente.

La función de replanificación $regla1(s1, s2)$ actuaría según estas definiciones, replanificando la trayectoria del móvil controlado a partir de un ajuste de velocidades según las siguientes reglas:

$\exists pi : \{ (pi, t_1) \in s1 \wedge (pi, t_2) \in s2 \};$

1. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=CIERTO)]
→ adelantar($t=t_{1-\max}-t_{2-\min}$)
2. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=FALSO)]
→ retrasar($mnc-en-pi+1$)

Figura 5-5. Reglas de replanificación para el caso del punto compartido.

En estas reglas, la función *interacción-temporal* se encarga de estudiar las relaciones temporales entre los intervalos de tiempo que se establecen según el tipo de interacción. Esta función se basa en la siguiente expresión:

[precede(t_1, t_2) \vee precedido-por(t_1, t_2) \vee igual(t_1, t_2)
 \vee solapa(t_1, t_2) \vee solapado-por(t_1, t_2) \vee
comienza(t_1, t_2) \vee iniciado-por(t_1, t_2) \vee
durante(t_1, t_2) \vee contiene(t_1, t_2) \vee finaliza(t_1, t_2) \vee
finalizado-por(t_1, t_2)]
→ interacción-temporal=CIERTO

Figura 5-6. Regla básica para el estudio de las posibles interacciones temporales entre dos intervalos temporales estimados.

Las reglas anteriores exponen un ajuste de velocidades para el caso de que exista cualquier tipo de interacción entre ambos intervalos de tiempo. La comprobación de las relaciones temporales entre ambos intervalos de tiempo es la base que permite deducir las acciones de replanificación. Pero este ajuste de velocidades se aplica comprobando la posibilidad de que el móvil controlado pueda adelantarse o no. Esto se realiza a través de la función `viabile-adelantar`. Sólo cuando las prestaciones del móvil y las características de la actividad que realiza lo permitan, será posible adelantar las acciones y movimientos del móvil controlado.

Así, la primera regla comprueba las interacciones temporales y el posible adelanto de la trayectoria del móvil controlado. Esto último se realiza a través de la función `viabile-adelantar`, que se evalúa como cierta o falsa dependiendo del tiempo que se quiera adelantar el móvil controlado. Esta función comprueba las posibilidades de adelanto del móvil en cuanto a sus prestaciones (velocidades máximas) y de la actividad que realiza (velocidad media y máxima velocidad permitida al realizarla). Si se permite el adelanto de la trayectoria, éste se realiza a través de la función `adelantar(t=t1-máx-t2-mín)`, de una manera gradual en el tiempo, distribuyendo el tiempo de adelanto en los tramos entre dos puntos que hay que recorrer hasta el punto compartido.

Este adelanto permite asegurar que el móvil pasará por el punto compartido en su peor tiempo antes de que lo haga el otro móvil con su mejor tiempo, evitándose la posible colisión.

La segunda regla comprueba que no se puede adelantar al móvil controlado y calcula un retraso de manera que cuando llegue al punto compartido, el otro móvil ya haya pasado. Este retraso se aplica haciendo que el móvil controlado no avance hasta el punto compartido hasta que se reciba la confirmación de que el otro móvil no haya llegado al punto siguiente. Así, existirá siempre entre ambos móviles una distancia de seguridad compuesta por el tramo que separa el punto compartido con el siguiente punto de la trayectoria del móvil no controlado. La función `retrasar(mnc-en-pi+1)` es la que implementa este retraso.

Estas reglas básicas de replanificación se pueden mejorar añadiendo a los tiempos de avance del móvil un tiempo de seguridad (t_s) que asegure que se mantiene una cierta distancia entre los móviles. Este tiempo es especialmente útil en la consideración de las relaciones temporales “precede” y “precedido-por”, en las que el tiempo

calculado para adelantar será nulo a menos que se considere este tiempo de seguridad.

Con todo, la función que calcula el adelanto del móvil controlado quedaría como sigue:

$$\text{adelantar}(t=t_{1-\text{máx}}-(t_{2-\text{mín}}-t_s)) \equiv \text{adelantar}(t=(t_{1-\text{máx}}-t_{2-\text{mín}})+t_s)$$

Figura 5-7. Función para el cálculo del tiempo de adelanto considerando el tiempo de seguridad.

ejemplo de aplicación

Supongamos el siguiente entorno (similar al mostrado en la Figura 4.6 del capítulo anterior):

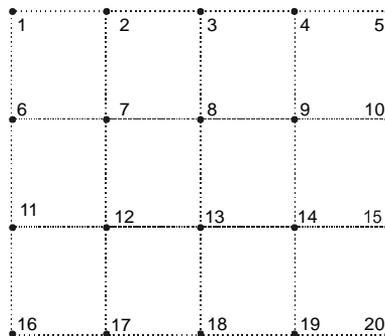


Figura 5-8. Entorno ejemplo.

Se considera que un móvil controlado debe ejecutar en este entorno una trayectoria a, mientras que otro móvil no controlado va a ejecutar una trayectoria b, definidas ambas como sigue:

Ecuación 5-5

$$a = \{p3, (p8, [2,3]), (p13, [2,3]), (p18, [2,3])\}$$

$$b = \{p2, (p7, [2,4]), (p6, [2,4]), (p11, [2,4]), (p12, [2,4]), (p13, [2,4]), (p14, [2,4]), (p15, [2,4])\}$$

La siguiente figura muestra ambas trayectorias sobre el entorno ejemplo:

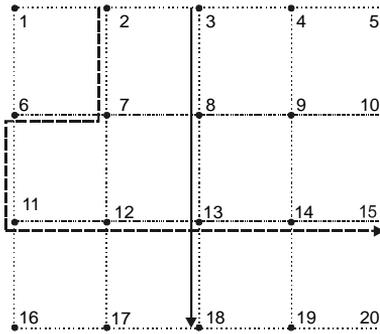


Figura 5-9. Representación de la trayectoria a (en línea continua) y la trayectoria b sobre el entorno ejemplo para el caso del punto compartido.

Supongamos también, que la trayectoria que sigue el móvil no controlado comienza a ejecutarse con $t=0$, mientras que la trayectoria que debe seguir el móvil controlado empezará su ejecución con $t=6$. Con estos tiempos de inicio confirmados, el ajuste temporal de ambas trayectorias es como sigue:

Ecuación 5-6

$$a = \{(p3,6), (p8,[8,9]), (p13,[10,12]), (p18,[12,15])\}$$

$$b = \{(p2,0), (p7,[2,4]), (p6,[4,8]), (p11,[6,12]), (p12,[8,16]), (p13,[10,20]), (p14,[12,24]), (p15,[14,28])\}$$

Para ajustarse todavía más al ejemplo del capítulo anterior, en el que se deducía la trayectoria b como la ejecutada por el móvil no controlado (Tabla 4.1), se asume también en este caso que en el instante $t=5$ se reconoce dicha trayectoria. Esta situación lleva a un nuevo ajuste temporal en el que la trayectoria b se ajusta a los tiempos cumplidos de la siguiente manera:

Ecuación 5-7

$$a = \{(p3,6), (p8,[8,9]), (p13,[10,12]), (p18,[12,15])\}$$

$$b = \{(p2,0), (p7,2.5), (p6,5), (p11,[7,9]), (p12,[9,13]), (p13,[11,17]), (p14,[13,21]), (p15,[15,25])\}$$

En un primer momento, las interacciones entre ambas trayectorias y su tipo se detectan a partir de un proceso de comparación espacio-temporal implementado en la función `comparar-secuencias`. Así, las condiciones expuestas en la Ecuación 5-1, Ecuación 5-2 y la Ecuación 5-3 se comprueban para deducir el tipo de interacción entre ambas trayectorias. La siguiente figura muestra el ajuste temporal, los intervalos de tiempo estimados y como se detecta la interacción entre ambas trayectorias:

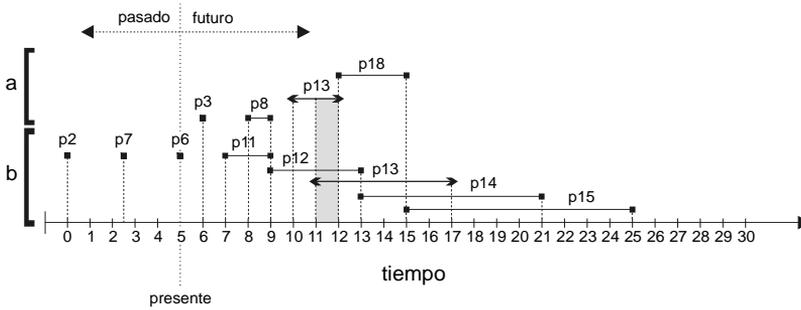


Figura 5-10. Comparativa de la estimación espacio-temporal entre las secuencias a y b. En la comparativa se aprecia el ajuste temporal, los intervalos estimados en el futuro y la interacción detectada (en gris) en el punto compartido p13. Se han marcado los intervalos temporales asociados al punto compartido con punta de flecha (\longleftrightarrow), frente al resto de intervalos que no suponen ningún tipo de interacción, marcados con punta cuadrada (■—■).

Una vez detectada la posible interacción, la replanificación de la trayectoria del móvil controlado se realiza mediante la función `ajustar` a partir de las reglas especificadas para el tipo de interacción (Figura 5-5).

El razonamiento temporal estudiado muestra que existe una relación de “solape” entre la trayectoria a y la b.

Ecuación 5-8

$$\begin{aligned}
 a \rightarrow (p13, [10, 12]) &\approx (p13, [t_{13-\min}^a, t_{13-\max}^a]) = (p13, t_{13}^a) \\
 b \rightarrow (p13, [11, 17]) &\approx (p13, [t_{13-\min}^b, t_{13-\max}^b]) = (p13, t_{13}^b) \rightarrow \\
 (t_{13-\min}^a < t_{13-\min}^b < t_{13-\max}^a < t_{13-\max}^b) &\rightarrow \text{solapa}(t_{13}^a, t_{13}^b)
 \end{aligned}$$

En el caso de que se pueda aplicar la primera regla, se debe calcular una acción de adelanto del móvil controlado de manera que se asegure que éste pasa por el punto compartido antes de que lo haga el móvil no controlado. Considerando un tiempo de seguridad $t_s=0.5$, el tiempo mínimo de adelanto se calcula a través de la función adelantar como sigue:

Ecuación 5-9

$$\text{adelantar}(t = (t_{13-\max}^a - t_{13-\min}^b) + t_s) \Rightarrow \text{adelantar}(1.5)$$

La aplicación de este adelanto se hace gradualmente, distribuyéndolo entre los tramos entre puntos que debe recorrer el móvil controlado hasta llegar al punto compartido p13. Puesto que hay dos tramos a recorrer antes de llegar a p13 (de p3 a p8, y de p8 a p13), se debe aplicar un adelanto en cada tramo de 0.75 unidades de tiempo sobre la trayectoria original. De esta manera, la trayectoria a quedará redefinida como sigue:

Ecuación 5-10

$$a = \{(p3, (p8, [1.25, 2.25])), (p13, [1.25, 2.25]), (p18, [2, 3])\}$$

y el nuevo reajuste en el tiempo será:

Ecuación 5-11

$$a = \{(p3, 6), (p8, [7.25, 8.25]), (p13, [8.5, 10.5]), (p18, [10.5, 13.5])\}$$

Así, de forma gráfica, la comparativa entre la trayectoria a reajustada y la trayectoria b es como sigue:

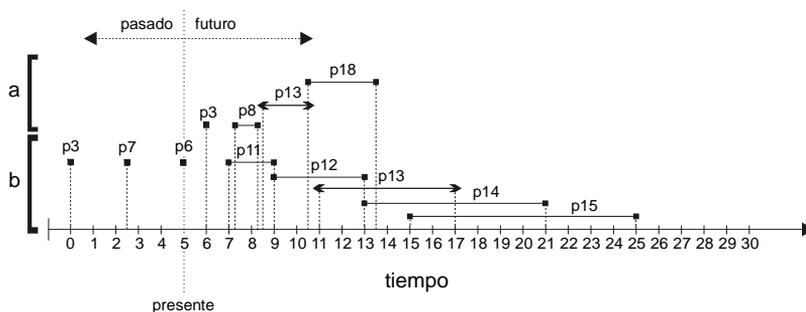


Figura 5-11. Comparativa de la estimación espacio-temporal de las secuencias a y b después de reajustar la trayectoria a adelantándola.

De la comparativa de las estimaciones de la figura anterior se desprende que no va a haber conflicto en el punto p13 con el reajuste de velocidades aplicado. La interacción entre ambos móviles en el punto compartido se ha evitado, pasando el móvil controlado antes por dicho punto y manteniéndose el intervalo de seguridad especificado.

Cuando la primera regla no se puede aplicar por las restricciones de la actividad del móvil (velocidad máxima de la actividad) o por las propias características del móvil, entonces se cumple la segunda regla, que calcula un retraso para evitar el conflicto en el punto compartido. Como ya se ha especificado, a diferencia del adelanto, que se aplica de manera inmediata, el retraso sólo se debe calcular con los tiempos de paso por los puntos previos al punto compartido ya confirmados. Esto es así para poder aplicar un retraso óptimo, ajustado en lo posible a la última observación del entorno realizada antes de llegar a la interacción.

Así, sobre los planteamientos originales de la Ecuación 5-7 y la Figura 5-10, se asume que el móvil no controlado sigue con la misma progresión y continúa empleando 2.5 unidades de tiempo en recorrer cada tramo entre dos puntos; y el móvil controlado emplea también la media estimada, 2.5 unidades de tiempo. Esta suposición se plasma en la siguiente tabla:

tiempo	0	2.5	5	6	7.5	8.5
trayectoria a	--	--	--	p3	--	p8
trayectoria b	p2	p7	p6	--	p11	--

Tabla 5-1. Tiempos de paso por los puntos previos al punto compartido entre ambas trayectorias.

Esta asunción establece que en $t=8.5$, el móvil controlado ha llegado al punto previo al compartido. Es en este momento cuando se debe aplicar el retraso sobre el móvil no controlado, deteniéndolo hasta que se confirme que el otro móvil ha pasado por el punto compartido, esto es, cuando se tenga constancia de que ha llegado al siguiente punto de su trayectoria.

En este punto, ambas trayectorias se ajustan en el tiempo de la siguiente manera:

Ecuación 5-12

$$a = \{(p3,6), (p8,8.5), (p13,[10.5,11.5]), (p18,[12.5,14.5])\}$$

$$b = \{(p2,0), (p7,2.5), (p6,5), (p11,7.5), (p12,[9.5,11.5]), (p13,[11.5,15.5]), (p14,[13.5,19.5]), (p15,[15.5,23.5])\}$$

Este ajuste se plasma en la siguiente comparativa espacio-temporal:

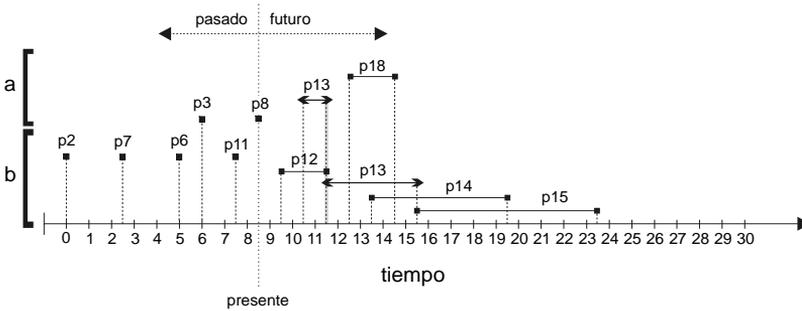


Figura 5-12. Comparativa de la estimación espacio-temporal de las trayectorias a y b con los tiempos confirmados para los puntos previos al punto compartido.

De donde se desprende la siguiente relación temporal:

Ecuación 5-13

$$a \rightarrow (p13, [10.5, 11.5]) \approx (p13, [t_{13-min}^a, t_{13-max}^a]) = (p13, t_{13}^a)$$

$$b \rightarrow (p13, [11.5, 15.5]) \approx (p13, [t_{13-min}^b, t_{13-max}^b]) = (p13, t_{13}^b)$$

$$(t_{13-max}^a = t_{13-max}^b) \rightarrow \text{precede}(t_{13}^a, t_{13}^b)$$

Esta relación temporal, que indica el riesgo de una posible colisión por la relación de “precedencia”, conduce a aplicar el retraso mencionado hasta confirmarse que el móvil no controlado ha pasado por el punto compartido.

Asumiendo la misma progresión en ambos móviles, entonces, el móvil no controlado llegará a p14 (el siguiente punto en su trayectoria al punto compartido) con t=15, según la siguiente tabla de tiempos:

tiempo	0	2.5	5	6	7.5	8.5	10	12.5	15
trayectoria a	--	--	--	p3	--	p8	--	--	--
trayectoria b	p2	p7	p6	--	p11	--	p12	p13	p14

Tabla 5-2. Tiempos de paso hasta llegar el móvil no controlado al punto p14.

Es en este momento cuando el móvil controlado debe reiniciar su marcha, ya con la seguridad de que no se va a producir ninguna colisión. Con todo, las trayectorias de ambos móviles quedarán reajustadas en el tiempo como sigue:

Ecuación 5-14

$$a = \{(p3,6), (p8,8.5), (p13,[17,18]), (p18,[19,21])\}$$

$$b = \{(p2,0), (p7,2.5), (p6,5), (p11,7.5), (p12,10), (p13,12.5), (p14,15), (p15,[17,19])\}$$

Mostrando la comparativa de este último ajuste la ausencia de colisión:

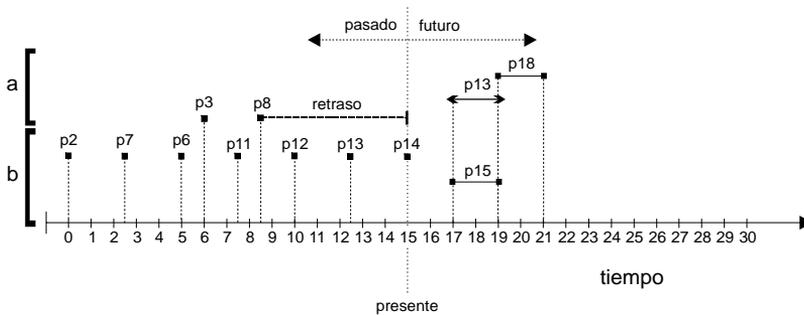


Figura 5-13. Comparativa entre ambas trayectorias, reajustadas retrasando el móvil controlado para hacer que pase después del otro móvil por el punto compartido.

EL CASO DEL SEGMENTO COMPARTIDO EN EL MISMO SENTIDO

De acuerdo con la definición de la Ecuación 5-2, un segmento compartido por dos trayectorias es una sub-secuencia ordenada de puntos que aparece en las dos secuencias que representan a ambas trayectorias. En una situación real, un segmento compartido

representa un pasillo o corredor de una factoría por el que deben pasar ambos móviles, en este caso en el mismo sentido de la marcha.

De cara a evitar una posible colisión en el segmento compartido, las acciones de replanificación deben asegurar que el móvil controlado pasa antes o después por el segmento, pero nunca al mismo tiempo que el otro móvil. El adelanto permite que el móvil controlado salga del segmento compartido antes de que entre en él el otro móvil. Por su parte, el retraso asegura que el móvil controlado pase por el segmento compartido después que el otro móvil, manteniendo una cierta distancia que asegure la ausencia de colisión en cada uno de los puntos que conforman el segmento.

La primera posibilidad de replanificación (adelantar el móvil controlado para que pase antes) considera todo el segmento como un único punto, de manera que ambos móviles no pueden permanecer al mismo tiempo dentro de él. Bajo esta premisa, se puede calcular un intervalo temporal en cada trayectoria que abarque todo el segmento de la siguiente manera:

Ecuación 5-15

$$s1 \Rightarrow s_s = \{p_i, p_j, \dots, p_k\} \Rightarrow t_1 = [t_{1-\min}, t_{1-\max}] = [t_{i-\min}^{s1}, t_{k-\max}^{s1}]$$

$$s2 \Rightarrow s_s = \{p_i, p_j, \dots, p_k\} \Rightarrow t_2 = [t_{2-\min}, t_{2-\max}] = [t_{i-\min}^{s2}, t_{k-\max}^{s2}]$$

donde t_1 y t_2 son los intervalos asociados al segmento compartido, s_s , en $s1$ y $s2$, respectivamente.

Cuando el móvil controlado no se puede adelantar y hay que retrasarlo para que pase después del móvil no controlado, se debe controlar que pase por detrás en cada uno de los puntos del segmento controlado. Esto se consigue haciendo que, una vez el móvil controlado haya alcanzado el punto previo al segmento compartido, éste se mueva al siguiente punto sólo cuando se confirme que el móvil no controlado va dos puntos por delante dentro del segmento compartido. Este procedimiento asegurará que el móvil controlado vaya por detrás del no controlado, manteniéndose un tramo entre ambos móviles como distancia de seguridad para evitar la colisión por alcance.

A partir de estas dos posibilidades, la función `regla2(s1,s2)` actuaría según las dos reglas de replanificación básicas que siguen:

$\exists s_s = \{p_i, \dots, p_k\} : \{(s_s, t_1) \subset s_1 \wedge (s_s, t_2) \subset s_2\};$

1. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=CIERTO)]
→ adelantar($t = (t_{1-\text{máx}} - t_{2-\text{mín}}) + t_s$)
 2. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=FALSO)]
→ retrasar(mnc-en- p_{j+1}), $\forall p_j \in s_s$
-

Figura 5-14. Reglas de replanificación para el caso del segmento compartido en el mismo sentido.

Nótese que la primera regla se aplica comparando los intervalos estimados para el global del segmento (t_1 y t_2), mientras que en la segunda regla, si no se permite el adelanto sobre la estimación global del segmento (intervalos t_1 y t_2), se aplica un retraso sobre el móvil controlado de manera que este va accediendo a los puntos p_j del segmento compartido sólo cuando el otro móvil confirma que va alcanzando los puntos que van por delante del segmento (p_{j+1}).

De esta manera, la primera regla, comprueba que el robot pueda ser adelantado en caso de que exista una probable colisión calculada mediante la estimación temporal y aplica el adelanto correspondiente con la función $\text{adelantar}(t = (t_{1-\text{máx}} - t_{2-\text{mín}}) + t_s)$. Esta regla es similar a la primera regla del caso anterior.

La segunda regla, por su parte, controla que el móvil controlado pase después del otro por cada punto del segmento. Para ello se hace mover al móvil controlado sólo cuando se comprueba que el no controlado ha alcanzado el punto siguiente. La función $\text{retrasar}(\text{mnc-en-}p_{j+1})$ es la que aplica un retraso sobre el móvil controlado al acceder a un punto p_j del segmento compartido hasta que se comprueba que el móvil no controlado ha alcanzado el punto p_{j+1} de su trayectoria. Este retraso garantiza un margen de seguridad de un tramo entre ambos móviles.

Estas reglas de replanificación se pueden mejorar si se considera que el móvil no controlado también puede circular por detrás del controlado a lo largo del segmento compartido, manteniendo una cierta distancia de seguridad de manera que se asegure la ausencia de colisión. Se trata de una situación real en la que existen pasillos o corredores en el entorno industrial lo suficientemente largos como

para permitir que circulen dos móviles al mismo tiempo, separados una cierta distancia.

Esta posibilidad se puede abordar de manera bastante sencilla considerando los distintos tramos entre dos puntos consecutivos que forman el segmento compartido. De esta manera, si se controla que ambos móviles no estén en un mismo tramo al mismo tiempo, se podrá permitir el acceso simultáneo al segmento compartido sin que se produzca ninguna colisión. Para conseguirlo, se calculan intervalos temporales únicos para cada tramo y se establece el razonamiento temporal entre ellos. La siguiente expresión muestra cómo se extraen los intervalos temporales asociados a los tramos, dependiendo de la definición de cada trayectoria.

Ecuación 5-16

$$\{p_i, p_j, \dots, p_k\} \equiv \{p_i, p_j\} + \dots + \{p(k-1), p_k\} = s_1 + \dots + s_{(k-i)} \rightarrow$$

$$\rightarrow \forall s_i = \{p_i, p_j\} \Rightarrow \begin{cases} s1 \rightarrow t_3 = [t_{3-\min}, t_{3-\max}] = [t_{i-\min}^{s1}, t_{j-\max}^{s1}] \\ s2 \rightarrow t_4 = [t_{4-\min}, t_{4-\max}] = [t_{i-\min}^{s2}, t_{j-\max}^{s2}] \end{cases}$$

Es importante resaltar que el adelanto sobre cada tramo se aplica sólo si haciendo una estimación de los adelantos en cada tramo se comprueba que todos se van a poder aplicar sobre el móvil controlado. Es decir, en primer lugar se debe hacer una estimación temporal sobre el primer tramo, deducir el adelanto necesario y hacer una nueva estimación del ajuste temporal de la trayectoria. Después se hace una nueva estimación temporal sobre el siguiente tramo, se deduce el correspondiente adelanto y se ajusta de nuevo la trayectoria. Así sucesivamente hasta el último de los tramos. Sólo en el caso de que todos los sucesivos adelantos en cada tramo sean aplicables al móvil controlado (dependiendo de sus prestaciones y de las características de la actividad), se comenzará a aplicar de forma efectiva los adelantos.

Para que estas estimaciones sean válidas y los adelantos deducidos aseguren la ausencia de colisión por alcance sobre el móvil no controlado, se deben hacer siempre considerando los tiempos mínimos de éste último.

Así, las reglas anteriores quedan mejoradas de la siguiente manera:

$$\exists s_s = \{p_i, \dots, p_k\} = (s_1 + \dots + s_{(k-i)}), \quad s_i = \{p_i, p_{i+1}\} : \\ \{(s_s, t_1) \subset s_1 \wedge (s_s, t_2) \subset s_2\} \wedge \{(s_i, t_3) \subset s_s \subset s_1 \wedge \\ (s_i, t_4) \subset s_s \subset s_2\};$$

1. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=CIERTO)]
→ adelantar($t = (t_{1-\text{máx}} - t_{2-\text{mín}}) + t_s$)
2. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=FALSO)] \wedge
[interacción-temporal(t_3, t_4) \wedge
(viable-adelantar=CIERTO), $\forall s_i$]
→ adelantar($t = (t_{3-\text{máx}} - t_{4-\text{mín}}) + t_s$)
3. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=FALSO)] \wedge
[interacción-temporal(t_3, t_4) \wedge
($\exists s_i$: viable-adelantar=FALSO)]
→ retrasar($\text{mnc-en-pj}+1$), $\forall p_j \in s_s$

Figura 5-15. Reglas de replanificación mejoradas para el caso del segmento compartido en el mismo sentido.

De esta manera, la primera regla intenta aplicar siempre una replanificación por la que el móvil controlado pase en primer lugar por el segmento compartido, antes de que el otro móvil llegue a él, proporcionando una mayor seguridad ante posibles colisiones. La segunda regla busca que el móvil controlado pase también antes por el segmento compartido, pero permitiendo que el otro móvil vaya por detrás, circulando ambos al mismo tiempo por el segmento. La tercera regla, por su parte, hace que el móvil controlado se retrase circulando por detrás del otro móvil.

ejemplo de aplicación

Consideremos sobre el entorno ejemplo de la Figura 5-8 que un móvil controlado va a realizar una trayectoria a y un móvil no controlado realiza una trayectoria b definidas como sigue:

Ecuación 5-17

$$a = \{p3, (p8, [2,3]), (p13, [2,3]), (p14, [2,3]), (p15, [2,3])\}$$

$$b = \{p2, (p7, [2,4]), (p6, [2,4]), (p11, [2,4]), (p12, [2,4]), (p13, [2,4]),$$

$$(p14, [2,4]), (p15, [2,4])\}$$

Como se aprecia, la trayectoria b está definida como en el ejemplo anterior, mientras que la trayectoria que sigue el móvil controlado es distinta. Ambas trayectorias se aprecian en la siguiente figura:

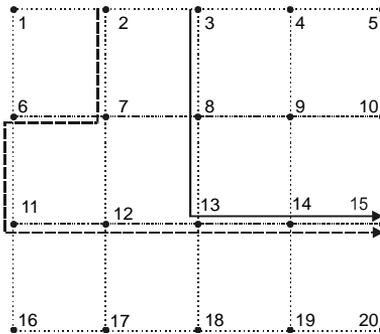


Figura 5-16. Representación de la trayectoria a (línea continua) y la trayectoria b sobre el entorno ejemplo para el caso del segmento compartido en el mismo sentido de avance.

Considerando que la trayectoria b se inicia con $t=0$ y que la trayectoria a se inicia con $t=4$, se tendrá el siguiente ajuste temporal:

Ecuación 5-18

$$a = \{(p3, 4), (p8, [6,7]), (p13, [8,10]), (p14, [10,13]), (p15, [12,16])\}$$

$$b = \{(p2, 0), (p7, [2,4]), (p6, [4,8]), (p11, [6,12]), (p12, [8,16]), (p13, [10,20]),$$

$$(p14, [12,24]), (p15, [14,28])\}$$

Supongamos también, igual que en el ejemplo del caso anterior, que se deduce que el móvil no controlado está ejecutando la trayectoria b en el instante de tiempo $t=5$. De esta manera, el ajuste temporal para ambas trayectorias en ese momento se recalculará como sigue:

Ecuación 5-19

$$a = \{(p3, 4), (p8, [6,7]), (p13, [8,10]), (p14, [10,13]), (p15, [12,16])\}$$

$$b = \{(p2, 0), (p7, 2.5), (p6, 5), (p11, [7,9]), (p12, [9,13]), (p13, [11,17]),$$

$$(p14, [13,21]), (p15, [15,25])\}$$

En un primer momento, las interacciones entre ambas trayectorias se detectan a partir de la función *comparar-secuencias*, que implementa el proceso de comparación espacio-temporal entre las estimaciones realizadas para ambas trayectorias, tal y como muestra la siguiente figura:

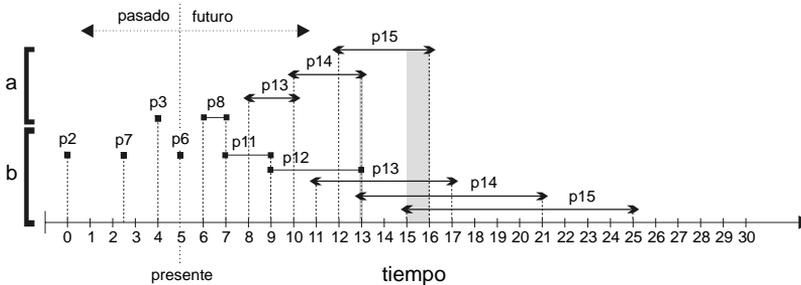


Figura 5-17. Comparativa de la estimación espacio-temporal de las trayectorias a y b para el caso del segmento compartido en el mismo sentido de avance.

Como se aprecia en la figura, entre ambas trayectorias se detecta un segmento compartido en el mismo sentido de avance, definido por los puntos del entorno {p13, p14, p15} (se cumple la condición expresada en la Ecuación 5-2).

El ajuste de velocidades se realiza mediante la función *ajustar*, aplicando las reglas definidas para este tipo de interacción. Así, con la primera regla, que considera un intervalo único para todo el segmento compartido, el razonamiento temporal mostraría el siguiente resultado:

Ecuación 5-20

$$\begin{aligned}
 a &\rightarrow \{(p13, [8, 10]), (p14, [10, 13]), (p15, [13, 16])\} \Rightarrow t_{13-15}^a \\
 b &\rightarrow \{(p13, [11, 17]), (p14, [13, 21]), (p15, [15, 25])\} \Rightarrow t_{13-15}^b \\
 t_{13-15}^a &= [8, 16] = [t_{13-\min}^a, t_{15-\max}^a] \\
 t_{13-15}^b &= [11, 25] = [t_{13-\min}^b, t_{15-\max}^b] \\
 (t_{13-\min}^a < t_{13-\min}^b < t_{15-\max}^a < t_{15-\max}^b) &\rightarrow \text{solapa}(t_{13-15}^a, t_{13-15}^b)
 \end{aligned}$$

Esta relación de “solape” entre los intervalos considerados para todo el segmento, conducen a aplicar una acción de adelanto sobre el móvil controlado deducida a partir de la función *adelantar* y considerando un tiempo de seguridad $t_s=0.5$, como sigue:

Ecuación 5-21

$$\text{adelantar}(t = (t_{15-\text{máx}}^a - t_{13-\text{mín}}^b) + t_s) \Rightarrow \text{adelantar}(5.5)$$

Este adelanto se debe aplicar gradualmente hasta sobrepasar el último punto del segmento compartido, es decir, en los tramos que se recorren hasta llegar al segmento compartido y a lo largo del propio segmento. Esto supone aplicar un adelanto en tres tramos: de p8 a p13, de p13 a p14, y de p14 a p15, considerando que el móvil controlado ya ha salido del punto p3 y que por tanto en el tramo de p3 a p8 no se puede aplicar el adelanto correspondiente. Así, el adelanto a aplicar debe ser de 1.833 unidades de tiempo.

Si se tiene en cuenta que la trayectoria a está definida de forma que el tiempo empleado en recorrer un tramo oscila entre un mínimo de 2 y un máximo de 3, resulta impensable que se pueda aplicar este adelanto en cada tramo (supone reducir más de un 90% el tiempo mínimo y más de un 60% el tiempo máximo).

Por su parte, la segunda regla intenta aplicar un adelanto considerando el segmento compartido en los tramos que lo componen. Así, el segmento queda definido por los siguientes tramos y sus intervalos temporales asociados:

Ecuación 5-22

$$\{p13, p14, p15\} = \{p13, p14\} + \{p14, p15\} = s_{13-14} + s_{14-15} \rightarrow$$

$$a \Rightarrow \begin{cases} s_{13-14} \rightarrow t_{13-14}^a = [t_{13-\text{mín}}^a, t_{14-\text{máx}}^a] = [8, 13] \\ s_{14-15} \rightarrow t_{14-15}^a = [t_{14-\text{mín}}^a, t_{15-\text{máx}}^a] = [10, 16] \end{cases}$$

$$b \Rightarrow \begin{cases} s_{13-14} \rightarrow t_{13-14}^b = [t_{13-\text{mín}}^b, t_{14-\text{máx}}^b] = [11, 21] \\ s_{14-15} \rightarrow t_{14-15}^b = [t_{14-\text{mín}}^b, t_{15-\text{máx}}^b] = [13, 25] \end{cases}$$

La representación gráfica de la comparación entre estos intervalos temporales asociados a los tramos del segmento compartido es como sigue:

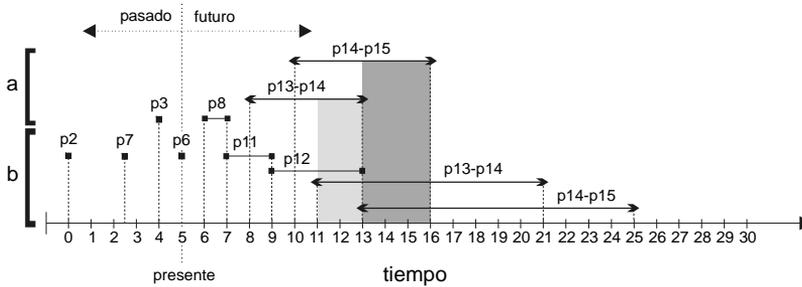


Figura 5-18. Comparativa de la estimación espacio-temporal considerando los tramos en el caso del segmento compartido en el mismo sentido de avance.

De la figura anterior se desprenden unas relaciones de “solape” entre los intervalos asociados a los tramos del segmento compartido (reflejadas en distintos tonos de gris). Las siguientes expresiones muestran formalmente estas relaciones fruto del razonamiento temporal:

Ecuación 5-23

$$s_{13-14} \rightarrow \left| \begin{array}{l} a \rightarrow t_{13-14}^a = [t_{13-\min}^a, t_{14-\max}^a] = [8, 13] \\ b \rightarrow t_{13-14}^b = [t_{13-\min}^b, t_{14-\max}^b] = [11, 21] \end{array} \right| \rightarrow$$

$$(t_{13-\min}^a < t_{13-\min}^b < t_{14-\max}^a < t_{14-\max}^b) \rightarrow \text{solapa}(t_{13-14}^a, t_{13-14}^b)$$

$$s_{14-15} \rightarrow \left| \begin{array}{l} a \rightarrow t_{14-15}^a = [t_{14-\min}^a, t_{15-\max}^a] = [10, 16] \\ b \rightarrow t_{14-15}^b = [t_{14-\min}^b, t_{15-\max}^b] = [13, 25] \end{array} \right| \rightarrow$$

$$(t_{14-\min}^a < t_{14-\min}^b < t_{15-\max}^a < t_{15-\max}^b) \rightarrow \text{solapa}(t_{14-15}^a, t_{14-15}^b)$$

A partir de este resultado, se estima el ajuste temporal y se calculan los adelantos a aplicar para cada tramo del segmento compartido. Finalmente, sólo en el caso de que se pueda aplicar el adelanto por cada tramo, se empezará a aplicar.

El valor temporal del adelanto para evitar la colisión en el primer tramo, asegurando que el móvil controlado pasa antes, y considerando un tiempo de seguridad $t_s=0.5$, se calcula a partir de la función adelantar de la siguiente manera:

Ecuación 5-24

$$s_{13-14} \Rightarrow \text{adelantar}(t = (t_{14-\max}^a - t_{13-\min}^b) + t_s) \Rightarrow \text{adelantar}(2.5)$$

Este adelanto se debe aplicar proporcionalmente hasta sobrepasar el primer tramo, esto es, en el tramo que va de p8 a p13 y en el que va de p13 a p14. Ello nos lleva a aplicar un adelanto de 1.25 unidades de tiempo por tramo. Aunque el adelanto sigue siendo muy grande para aplicarse en relación a los tiempos especificados para recorrer cada tramo (más de un 60% sobre el tiempo mínimo y algo más de un 40% sobre el tiempo máximo), se puede considerar factible. De esta manera, se plantea un nuevo ajuste en el tiempo de la trayectoria a como se muestra en la siguiente expresión:

Ecuación 5-25

$$a = \{(p3,4), (p8,[6,7]), (p13,[6.75,8.75]), (p14,[7.5,10.5]), (p15,[9.5,13.5])\}$$

Ello conduce a recalcular de la siguiente manera los intervalos temporales asociados a los tramos en que se descompone el segmento compartido para la trayectoria a:

Ecuación 5-26

$$a \Rightarrow \begin{cases} s_{13-14} \rightarrow t_{13-14}^a = [t_{13-\min}^a, t_{14-\max}^a] = [6.75, 10.5] \\ s_{14-15} \rightarrow t_{14-15}^a = [t_{14-\min}^a, t_{15-\max}^a] = [7.5, 13.5] \end{cases}$$

La siguiente figura muestra la comparativa entre ambas trayectorias después de aplicar el adelanto:

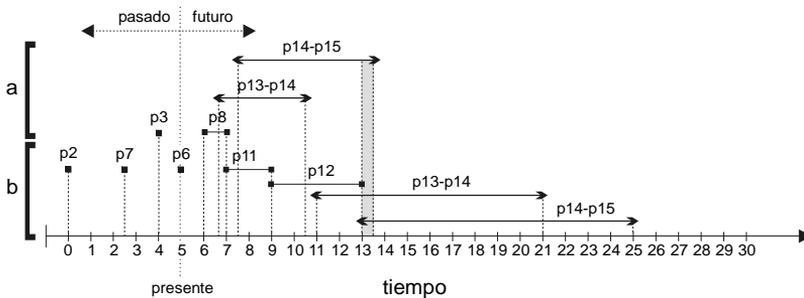


Figura 5-19. Comparativa de la estimación espacio-temporal entre ambas trayectorias después de aplicar el adelanto sobre la trayectoria a relativo al primer tramo del segmento compartido.

En la figura anterior se aprecia como el adelanto a aplicar evitaría la posible colisión en el primer tramo del segmento compartido, manteniéndose una probable colisión en el segundo tramo (marcada en gris), aunque reducida con respecto a la estimación original.

Una nueva estimación temporal sobre el segundo tramo lleva a deducir una relación de “solape” expresada formalmente como sigue:

Ecuación 5-27

$$s_{14-15} \rightarrow \left| \begin{array}{l} a \rightarrow t_{14-15}^a = [t_{14-\min}^a, t_{15-\max}^a] = [7.5, 13.5] \\ b \rightarrow t_{14-15}^b = [t_{14-\min}^b, t_{15-\max}^b] = [13, 25] \end{array} \right| \rightarrow \\ (t_{14-\min}^a < t_{14-\min}^b < t_{15-\max}^a < t_{15-\max}^b) \rightarrow \text{solapa}(t_{13-14}^a, t_{13-14}^b)$$

De nuevo la función adelantar, considerando un tiempo de seguridad $t_s=0.5$, permite calcular el ajuste de velocidades necesario para evitar la probable colisión en este tramo:

Ecuación 5-28

$$s_{14-15} \Rightarrow \text{adelantar}(t = (t_{15-\max}^a - t_{14-\min}^b) + t_s) \Rightarrow \text{adelantar}(1)$$

Este adelanto se debe aplicar al recorrer el mismo tramo que va de p14 a p15, de manera que se asegure que se sale de este tramo antes de que entre en él el otro móvil. Aplicar este adelanto en el tramo supone una reducción de un 50% sobre el tiempo mínimo originalmente establecido para ir de un punto a otro, y una reducción de un 33.33% sobre el tiempo máximo, adelanto que puede considerarse factible. Así, se plantea un nuevo ajuste temporal sobre la trayectoria a como sigue:

Ecuación 5-29

$$a = \{(p3,4), (p8, [6,7]), (p13, [6.75,8.75]), (p14, [7.5, 10.5]), \\ (p15, [8.5, 12.5])\}$$

Con lo que el intervalo asociado al último tramo será:

Ecuación 5-30

$$a \Rightarrow s_{14-15} \rightarrow t_{14-15}^a = [t_{14-\min}^a, t_{15-\max}^a] = [7.5, 12.5]$$

La representación gráfica para esta nueva estimación temporal de las dos trayectorias será:

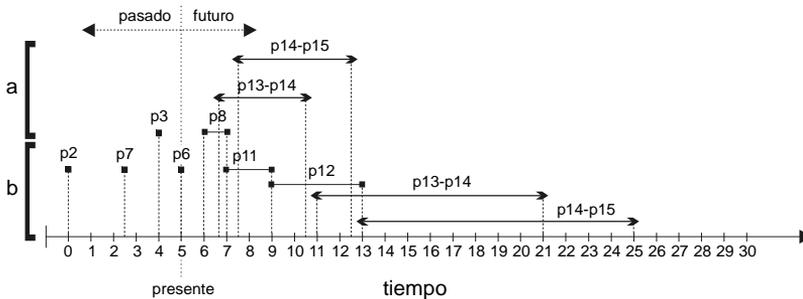


Figura 5-20. Comparativa de la estimación espacio-temporal entre ambas trayectorias después de aplicar el adelanto sobre la trayectoria a relativo al segundo y último tramo del segmento compartido.

En esta última figura se aprecia que el riesgo de colisión desaparece si se aplican los adelantos calculados para cada tramo. Estos adelantos sólo se pueden aplicar dependiendo, como ya se ha dicho, de las prestaciones del móvil y de las características de la actividad que éste realiza. En cualquier caso, los adelantos calculados por tramos son más factibles de aplicar que un adelanto que evite la colisión en la totalidad del segmento compartido.

Por último, la tercera regla hace que el móvil controlado pase después que el móvil no controlado por el segmento compartido cuando no es posible aplicar ningún adelanto. Así, partiendo del ajuste temporal original (Ecuación 5-19) y de la comparativa entre ambas trayectorias (Figura 5-17), el ajuste de velocidades se calcula mediante la función *ajustar*, para aplicar el retraso hasta que se confirme que el otro móvil llega al siguiente punto, calculado a su vez por la función *retrasar*($mnc-en-p_{j+1}$). La aplicación de esta regla hace que el móvil controlado pase después del no controlado por cada uno de los puntos del segmento compartido.

De manera similar al ejemplo del caso anterior, la regla se aplica cuando el móvil controlado llega al punto previo al segmento compartido. Entonces se generan los retrasos pertinentes para que sólo cuando el otro móvil llegue al punto siguiente al que se debe mover el controlado, se le dé permiso para efectuar el movimiento.

Así, suponiendo que el móvil no controlado recorre los tramos entre puntos en 2.5 unidades de tiempo (siguiendo el planteamiento original), y que el móvil controlado lo hace empleando 3 unidades de

tiempo, se tendrá la siguiente tabla de tiempos de paso por los puntos anteriores al primer punto del segmento compartido:

tiempo	0	2.5	4	5	7
trayectoria a	--	--	p3	--	p8
trayectoria b	p2	p7	--	p6	--

Tabla 5-3. Tiempos de paso por los puntos previos al punto compartido entre ambas trayectorias.

Al igual que en el caso anterior, esta suposición establece que el móvil controlado llega al punto previo al segmento compartido con $t=7$, ajustándose las trayectorias en este momento como sigue:

Ecuación 5-31

$$a = \{(p3,4), (p8,7), (p13,[9,10]), (p14,[11,13]), (p15,[13,16])\}$$

$$b = \{(p2,0), (p7,2.5), (p6,5), (p11,[7,9]), (p12,[9,13]), (p13,[11,17]), (p14,[13,21]), (p15,[15,25])\}$$

Plasmándose en la siguiente comparativa espacio-temporal entre ambas trayectorias:

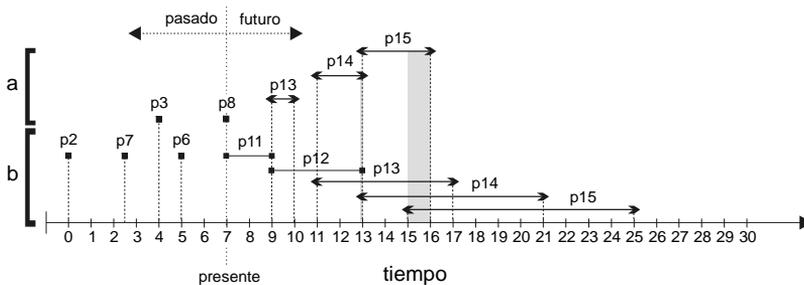


Figura 5-21. Comparativa de la estimación espacio-temporal entre ambas trayectorias cuando el móvil controlado ya ha alcanzado el punto previo al segmento compartido.

La figura muestra que no hay riesgo de colisión en el primer punto del segmento compartido, al que llegaría antes el móvil controlado. Pero se muestra la posible colisión en los siguientes puntos debido a la diferencia de velocidades entre ambos móviles. Puesto que se ha comprobado que el móvil controlado no se puede adelantar, no se puede correr el riesgo de dejar que entre en primer lugar en el segmento compartido y que luego se presente la posible colisión, con lo que se procede a retrasar su paso por el segmento compartido.

Por lo tanto, en este momento el móvil controlado se retrasa hasta confirmarse que otro móvil ha alcanzado el segundo punto del segmento compartido (p14). Si se continúa con la misma progresión, la tabla de tiempos de paso se actualizará como sigue:

tiempo	0	2.5	4	5	7	7.5	10	12.5	15
trayectoria a	--	--	p3	--	p8	--	--	--	--
trayectoria b	p2	p7	--	p6	--	p11	p12	p13	p14

Tabla 5-4. Tabla de tiempos de paso para ambos móviles hasta confirmarse que el no controlado llega al segundo punto del segmento compartido (p14).

En este momento (t=15), el móvil controlado ya puede avanzar hasta el primer punto del segmento compartido (p13), reajustándose ambas trayectorias como sigue:

Ecuación 5-32

$$a = \{(p3,4), (p8,7), (p13,[17,18]), (p14,[19,21]), (p15,[21,24])\}$$

$$b = \{(p2,0), (p7,2.5), (p6,5), (p11,7.5), (p12,10), (p13,12.5), (p14,15), (p15,[17,19])\}$$

La comparación de la estimación espacio-temporal para este último ajuste de tiempos queda como sigue:

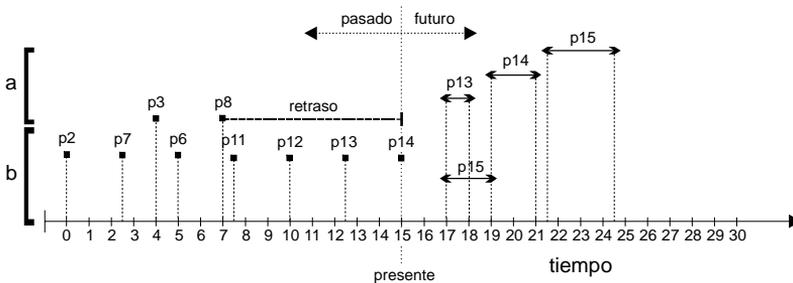


Figura 5-22. Comparativa de la estimación espacio-temporal después de aplicar el retraso sobre el móvil controlado para hacer que pase después por el primer punto del segmento compartido.

Siguiendo con la progresión de ambas trayectorias hasta confirmar el paso del móvil no controlado por el siguiente punto (p15) y del móvil

controlado por el primer punto del segmento compartido (p13), la tabla de tiempos se actualiza como sigue:

tiempo	0	2.5	4	5	7	7.5	10	12.5	15	17.5	18
trayectoria a	--	--	p3	--	p8	--	--	--	--	--	p13
trayectoria b	p2	p7	--	p6	--	p11	p12	p13	p14	p15	--

Tabla 5-5. Tabla de tiempos de paso hasta confirmarse la llegada del móvil controlado al primer punto del segmento compartido (p13) y la del móvil no controlado al siguiente punto (p15), último de su trayectoria.

Ajustándose en el tiempo las trayectorias de la siguiente manera:

Ecuación 5-33

$$a = \{(p3,4), (p8,7), (p13,18), (p14,[20,21]), (p15,[22,24])\}$$

$$b = \{(p2,0), (p7,2.5), (p6,5), (p11,7.5), (p12,10), (p13,12.5), (p14,15), (p15,17.5)\}$$

Quedando, por último, la comparativa espacio-temporal así:

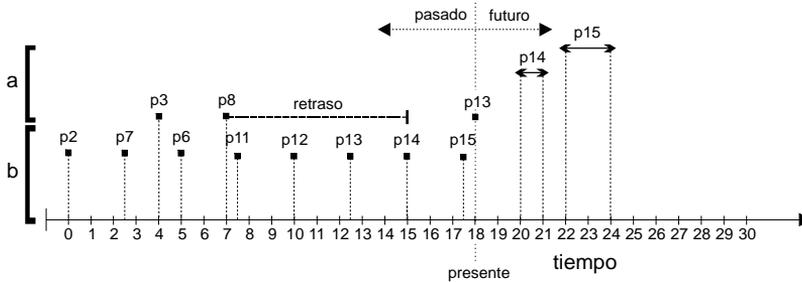


Figura 5-23. Comparativa final de la estimación espacio-temporal para el caso del segmento compartido en el mismo sentido de avance.

Se aprecia que antes de que el móvil controlado llegue a p13, el otro móvil ya ha finalizado su tarea, con lo que no haría falta retrasarlo más. En este caso, y debido a la corta longitud del segmento compartido, con un solo retraso aplicado al inicio del mismo, es suficiente para asegurar que el móvil controlado pasa después del no controlado, evitándose cualquier riesgo de colisión.

EL CASO DEL SEGMENTO COMPARTIDO EN SENTIDO CONTRARIO

Una variante del caso anterior es aquella situación en la que ambos móviles recorren trayectorias que comparten un segmento pero en sentido de avance opuesto. El equivalente a una situación real es un entorno industrial en el que se tiene un pasillo por el que circulan dos móviles en sentido opuesto. Esta situación se corresponde con la definición de la Ecuación 5-3, donde se muestra como una sub-secuencia de una trayectoria aparece en orden inverso en la otra trayectoria.

Las acciones de replanificación que aseguren la ausencia de colisión a lo largo del segmento compartido son más simples que el caso anterior, puesto que en esta situación nunca podrán coexistir ambos móviles al mismo tiempo en el segmento. En realidad se trata de que siempre pase un móvil antes que otro por el segmento compartido, lo cual nos lleva a una situación similar a la del punto compartido. De hecho, las reglas de replanificación a aplicar son las mismas, pero considerando intervalos de tiempo que abarquen todo el segmento compartido para ambas trayectorias. Estos intervalos de tiempo se calculan como sigue:

Ecuación 5-34

$$\begin{aligned} s1 \Rightarrow \overline{s_s} = \{p_i, p_j, \dots, p_k\} &\Rightarrow t_1 = [t_{1-\min}, t_{1-\max}] = [t_{i-\min}^{s1}, t_{k-\max}^{s1}] \\ s2 \Rightarrow \overline{s_s} = \{p_k, \dots, p_j, p_i\} &\Rightarrow t_2 = [t_{2-\min}, t_{2-\max}] = [t_{k-\min}^{s2}, t_{i-\max}^{s2}] \end{aligned}$$

Así, si es posible, se adelantará el móvil controlado para pasar antes por el segmento compartido. En caso de que no pudiera adelantarse el móvil, se plantea su retraso a la entrada del segmento, hasta que se confirme que el otro móvil ha alcanzado el primer punto fuera del segmento compartido en su trayectoria.

Este planteamiento se plasma en la función $regla3(s1, s2)$ que actúa según las siguientes reglas de replanificación básicas:

$$\exists [s_s = \{p_i, \dots, p_k\} \wedge \overline{s_s} = \{p_k, \dots, p_i\}] :$$

$$\{(s_s, t_1) \subset s1 \wedge (\overline{s_s}, t_2) \subset s2\};$$

1. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=CIERTO)]
→ adelantar($t = (t_{1-\text{máx}} - t_{2-\text{mín}}) - t_s$)
2. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=FALSO)]
→ retrasar(mnc-en-pi-1)

Figura 5-24. Reglas de replanificación para el caso del segmento compartido en sentido contrario.

Nótese como ambas reglas se aplican teniendo en cuenta los intervalos temporales t_1 y t_2 , que abarcan todo el segmento compartido, calculados según la expresión de la Ecuación 5-34.

Ambas reglas son similares a las del punto compartido, expuestas en la Figura 5-5. Así, la primera regla intenta adelantar el móvil controlado para que pase en primer lugar, a partir de la relación temporal existente entre ambos intervalos. Por su parte, la segunda regla aplica un retraso para que el móvil controlado pase después, hasta confirmarse la salida del otro móvil del segmento compartido. Esta última acción se realiza a través de la función `retrasar(mnc-en-pi-1)`, en la que se denota el punto siguiente al segmento compartido para la trayectoria que sigue el móvil no controlado como `pi-1`.

ejemplo de aplicación

De nuevo se considera el entorno ejemplo de la Figura 5-8, en el que se tiene un móvil controlado que va a realizar una trayectoria a y un móvil no controlado que sigue una trayectoria b. Ambas trayectorias quedan definidas en la siguiente expresión:

Ecuación 5-35

$$a = \{p3, (p8, [2,3]), (p13, [2,3]), (p12, [2,3]), (p11, [2,3]), (p16, [2,3])\}$$

$$b = \{p2, (p7, [2,4]), (p6, [2,4]), (p11, [2,4]), (p12, [2,4]), (p13, [2,4]), (p14, [2,4]), (p15, [2,4])\}$$

La trayectoria b está definida como en los ejemplos anteriores, mientras que la trayectoria a es distinta. En la siguiente figura se aprecian ambos recorridos:

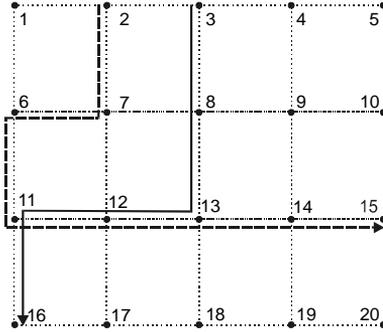


Figura 5-25. Representación de la trayectoria a (línea continua) y la trayectoria b sobre el entorno ejemplo para el caso del segmento compartido en sentido de avance contrario.

Considerando que la trayectoria b se inicia en $t=0$ y que la trayectoria a tiene previsto iniciarse en $t=4$, se tendrá el siguiente ajuste temporal:

Ecuación 5-36

$$a = \{(p3,4), (p8,[6,7]), (p13,[8,10]), (p12,[10,13]), (p11,[12,16]), (p16,[14,19])\}$$

$$b = \{(p2,0), (p7,[2,4]), (p6,[4,8]), (p11,[6,12]), (p12,[8,16]), (p13,[10,20]), (p14,[12,24]), (p15,[14,28])\}$$

Si, igual que en los ejemplos anteriores, se deduce en $t=5$ que el móvil no controlado está recorriendo la trayectoria b, para este instante de tiempo se tendrá el siguiente ajuste temporal, con algunos tiempos de paso ya confirmados:

Ecuación 5-37

$$a = \{(p3,4), (p8,[6,7]), (p13,[8,10]), (p12,[10,13]), (p11,[12,16]), (p16,[14,19])\}$$

$$b = \{(p2,0), (p7,2.5), (p6,5), (p11,[7,9]), (p12,[9,13]), (p13,[11,17]), (p14,[13,21]), (p15,[15,25])\}$$

En un primer momento, las interacciones entre ambas trayectorias se calculan a partir de la función `comparar-secuencias`, igual que en

los ejemplos anteriores. Esta función implementa el proceso de comparación espacio-temporal que permite detectar, para este ejemplo, un segmento compartido en sentido opuesto. Esta comparativa se refleja en la siguiente figura:

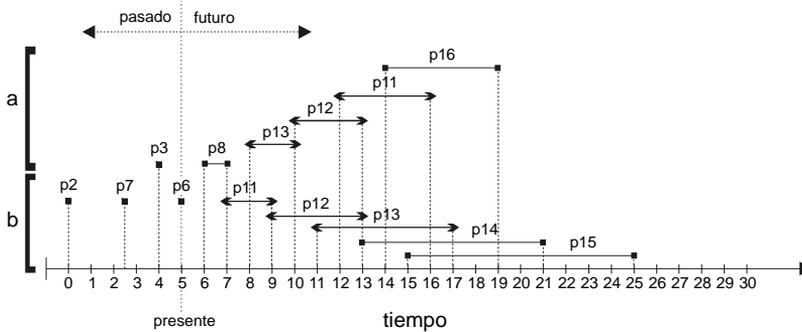


Figura 5-26. Comparativa de la estimación espacio-temporal de las trayectorias a y b para el caso del segmento compartido en sentido de avance contrario.

Ambas trayectorias comparten un segmento de la misma recorrido en sentido opuesto. Mientras que la trayectoria a recorre el tramo formado por el segmento {p13, p12, p11}, la trayectoria b recorre este mismo tramo en sentido opuesto, es decir, el segmento {p11, p12, p13}, cumpliéndose la condición de la Ecuación 5-3.

El ajuste de velocidades, realizado a través de la función *ajustar*, considera para este caso intervalos temporales que comprenden todo el segmento. Calculados estos intervalos para ambas trayectorias, el razonamiento temporal entre ellos muestra el siguiente resultado:

Ecuación 5-38

$$\begin{aligned}
 & a \rightarrow \{(p13, [8, 10]), (p12, [10, 13]), (p11, [12, 16])\} \Rightarrow t_{13-11}^a \} \rightarrow \\
 & b \rightarrow \{(p11, [7, 9]), (p12, [9, 13]), (p13, [11, 17])\} \Rightarrow t_{11-13}^b \} \rightarrow \\
 & \quad \left. \begin{aligned}
 t_{13-11}^a &= [8, 16] = [t_{13-\min}^a, t_{11-\max}^a] \\
 t_{11-13}^b &= [7, 17] = [t_{11-\min}^b, t_{13-\max}^b]
 \end{aligned} \right\} \rightarrow \\
 & (t_{11-\min}^b < t_{13-\min}^a < t_{11-\max}^a < t_{13-\max}^b) \rightarrow \text{durante}(t_{13-15}^a, t_{13-15}^b)
 \end{aligned}$$

La relación temporal deducida se puede ver en la siguiente figura que representa los intervalos temporales de la totalidad del segmento compartido para ambas trayectorias:

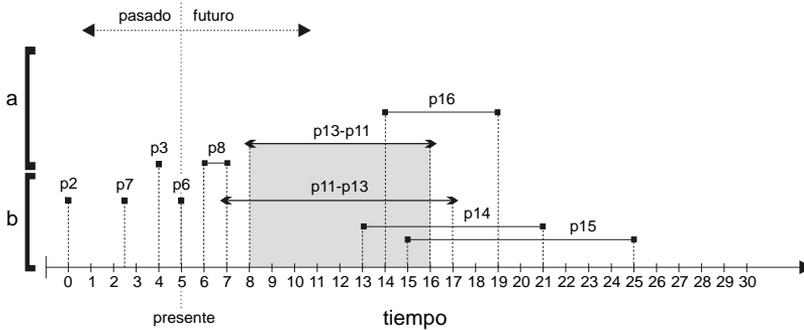


Figura 5-27. Comparativa de la estimación espacio-temporal de las trayectorias a y b para el caso del segmento compartido en sentido de avance contrario. En gris se muestra la interacción entre los intervalos temporales asociados a la totalidad del segmento compartido para ambas trayectorias.

Aplicando la primera regla, esta relación entre los intervalos considerados para todo el segmento conduce a una replanificación sobre el móvil no controlado adelantándolo para que pase en primer lugar por el segmento compartido. Así, la función adelantar calcula este adelanto, considerando, igual que en anteriores ocasiones, un tiempo de seguridad $t_s=0.5$:

Ecuación 5-39

$$\text{adelantar}(t = (t_{11-\text{máx}}^a - t_{11-\text{mín}}^b) + t_s) \Rightarrow \text{adelantar}(9.5)$$

Al igual que en los ejemplos anteriores, el adelanto se debe aplicar gradualmente en los tramos que van hasta la salida del móvil controlado del segmento compartido. Considerando que en $t=5$ el móvil controlado ya ha salido del primer punto y que por tanto el adelanto ya no puede ser aplicado en el primer tramo, el adelanto deberá aplicarse en los siguientes tramos: {p8, p13}, {p13, p12}, {p12, p11}. Lo cual supone un adelanto de 3.166 unidades de tiempo por tramo.

Si se tiene en cuenta que la trayectoria a está definida tal que el tiempo mínimo en recorrer un tramo es de 2 y el máximo es de 3, el adelanto resulta imposible de aplicar.

El análisis, por su parte, de la segunda regla conduce al cálculo de un retraso para que el móvil controlado circule por el segmento

compartido después que el no controlado. Así, partiendo del ajuste temporal original para ambas trayectorias (Ecuación 5-37) y de la comparativa entre ellas (Figura 5-26), el ajuste de velocidades se realiza mediante la función ajusta, calculándose a su vez un retraso a partir de la función retrasar(mnc-en-pi-1).

Igual que en anteriores situaciones, es cuando el móvil controlado llega al punto anterior al segmento compartido cuando se calcula el retraso a aplicar. Si se supone que el móvil no controlado recorre los tramos entre puntos en 2.5 unidades de tiempo (progresión del planteamiento original), y que el móvil controlado lo hace en 3 unidades de tiempo, se tendrá la siguiente tabla de tiempos de paso:

tiempo	0	2.5	4	5	7
trayectoria a	--	--	p3	--	p8
trayectoria b	p2	p7	--	p6	--

Tabla 5-6. Tiempos de paso por los puntos previos al punto compartido entre ambas trayectorias.

Esta suposición lleva al siguiente ajuste de tiempos para ambas trayectorias:

Ecuación 5-40

$$\begin{aligned}
 a &= \{(p3,4), (p8,7), (p13,[9,10]), (p12,[11,13]), (p11,[13,16]), \\
 &\quad (p16,[15,19])\} \\
 b &= \{(p2,0), (p7,2.5), (p6,5), (p11,[7,9]), (p12,[9,13]), (p13,[11,17]), \\
 &\quad (p14,[13,21]), (p15,[15,25])\}
 \end{aligned}$$

Que se refleja en la siguientes comparativas espacio-temporales:

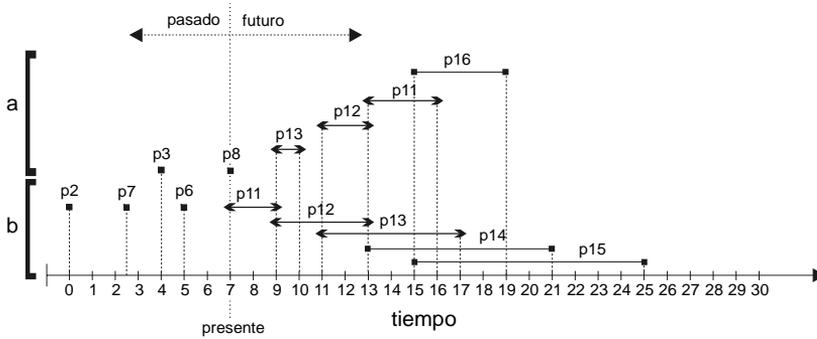


Figura 5-28. Comparativa de la estimación espacio-temporal entre ambas trayectorias cuando el móvil controlado ya ha alcanzado el punto previo al segmento compartido.

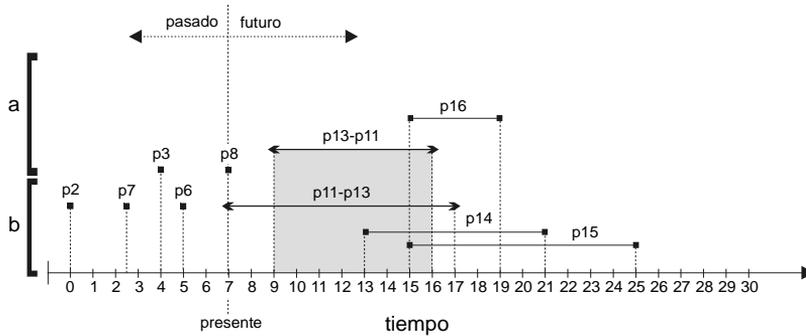


Figura 5-29. Comparativa de la estimación espacio-temporal entre ambas trayectorias cuando el móvil controlado ya ha alcanzado el punto previo al segmento compartido. La figura muestra la relación temporal entre los intervalos asociados a todo el segmento compartido.

Se aprecia en la figura anterior que se mantiene la relación temporal “durante”, deducida a partir del siguiente razonamiento temporal actualizado:

Ecuación 5-41

$$\begin{aligned}
 & \left. \begin{aligned}
 a &\rightarrow \{(p13, [9, 10]), (p12, [11, 13]), (p11, [13, 16])\} \Rightarrow t_{13-11}^a \\
 b &\rightarrow \{(p11, [7, 9]), (p12, [9, 13]), (p13, [11, 17])\} \Rightarrow t_{11-13}^b
 \end{aligned} \right\} \rightarrow \\
 & \left. \begin{aligned}
 t_{13-11}^a &= [9, 16] = [t_{13-\min}^a, t_{11-\max}^a] \\
 t_{11-13}^b &= [7, 17] = [t_{11-\min}^b, t_{13-\max}^b]
 \end{aligned} \right\} \rightarrow \\
 & (t_{11-\min}^b < t_{13-\min}^a < t_{11-\max}^a < t_{13-\max}^b) \rightarrow \text{durante}(t_{13-15}^a, t_{13-15}^b)
 \end{aligned}$$

La segunda regla aplica un retraso sobre el móvil controlado hasta que se confirme la salida del móvil no controlado del segmento compartido, es decir, hasta que éste haya alcanzado el punto siguiente en su trayectoria al último que conforma el segmento compartido. Considerando la misma progresión, la tabla de tiempos de paso se actualizaría como sigue:

tiempo	0	2.5	4	5	7	7.5	10	12.5	15
trayectoria a	--	--	p3	--	p8	--	--	--	--
trayectoria b	p2	p7	--	p6	--	p11	p12	p13	p14

Tabla 5-7. Tabla de tiempos de paso para ambos móviles hasta confirmarse que el no controlado llega al punto siguiente al último que conforma el segmento compartido.

En este momento, el móvil controlado ya puede avanzar hacia el primer punto del segmento compartido (p13) con lo que las trayectorias se reajustarían según la siguiente expresión:

Ecuación 5-42

$$\begin{aligned}
 a &= \{(p3,4), (p8,7), (p13,[17,18]), (p12,[19,21]), (p11,[21,24]), \\
 & \quad (p16,[23,27])\} \\
 b &= \{(p2,0), (p7,2.5), (p6,5), (p11,7.5), (p12,10), (p13,12.5), \\
 & \quad (p14,15), (p15,[17,19])\}
 \end{aligned}$$

Quedando la comparación espacio-temporal según la siguiente figura:

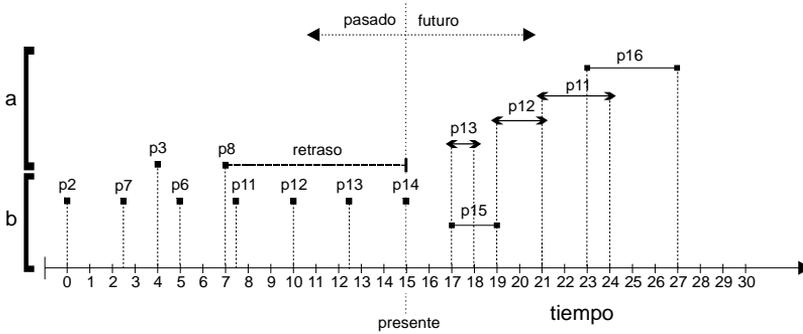


Figura 5-30. Comparativa de la estimación espacio-temporal después de aplicar el retraso sobre el móvil controlado para hacer que pase después por el segmento compartido.

En esta última figura se aprecia como el retraso aplicado permite que el móvil no controlado pase por el segmento compartido antes que lo haga el móvil controlado, evitándose cualquier posible colisión en dicho segmento.

REPLANIFICACIÓN DE MÚLTIPLES MÓVILES

EL PLANTEAMIENTO BÁSICO

Las reglas de replanificación definidas en el proceso predictivo expuesto en los puntos anteriores son aplicables en una situación en la que se tiene un robot móvil moviéndose en un entorno dinámico y en el que existe un móvil no controlado realizando sus propias tareas. Se trata de reglas de replanificación básicas basadas en un proceso de predicción de movimientos y acciones de ambos móviles.

Cuando en el entorno se encuentran varios móviles, algunos de ellos controlables y otros no controlables, realizando actividades diferentes, las reglas de replanificación anteriores son mejorables, de cara a la coordinación de las actividades de todos los móviles y a una correcta secuenciación del resultado final.

Existen las siguientes alternativas a considerar:

1. Cada móvil controlado lleva su propio sistema de planificación y replanificación basado en la metodología expuesta en la tesis.

2. Existe un único sistema de control centralizado para todos los móviles controlados.

A continuación se abordan estas dos posibilidades, esbozando una propuesta de replanificación de las actividades de los móviles controlados en un entorno compartido, de cara a evitar posibles interacciones, coordinar sus acciones y optimizar el global de actividades del sistema.

UN SISTEMA DE CONTROL PARA CADA MÓVIL

Se trata de la primera alternativa a considerar, en la que cada móvil realizando actividades en el entorno compartido tiene su propio sistema de control. Los sistemas de control no intercambian información entre ellos, con lo que las posibles replanificaciones se hacen desconociendo los ajustes del resto de móviles.

Los móviles controlados incorporan un sistema de control basado en los principios expuestos en esta tesis doctoral. Es decir, un sistema de replanificación de actividades a partir de la deducción y la predicción. En este caso, las reglas anteriores se aplicarían individualmente a cada móvil, considerando al resto como móviles no controlados cuyas actividades no deben entorpecerse.

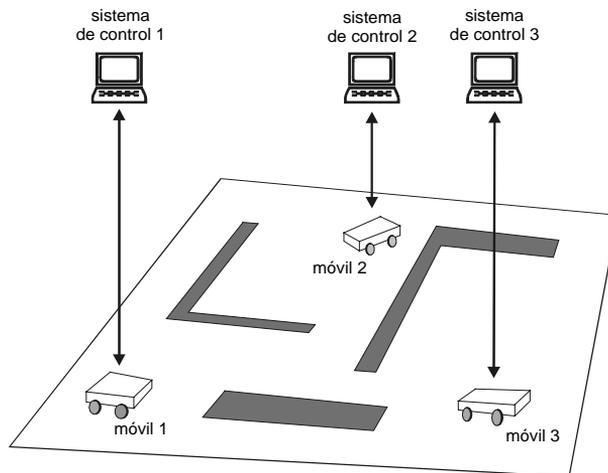


Figura 5-31. Esquema representativo de la existencia de un sistema de control para la replanificación de las actividades para cada móvil del entorno.

Así, el algoritmo básico de predicción expresado en la Figura 5-2 se ejecutaría en cada móvil controlado, actuando de manera independiente de la siguiente manera:

- En primer lugar se comparan las secuencias que representan las actividades del móvil y las del resto de móviles del entorno. La comparativa devuelve el número de interacciones existentes entre las secuencias. Esto es, las interacciones a nivel de puntos o segmentos en común entre la trayectoria que sigue el móvil controlado y cada una de las trayectorias que ejecutan el resto de móviles del entorno.
- Para cada interacción, por orden de aparición en la trayectoria del móvil controlado, se aplicaría el ajuste de velocidades a través de la función ajustar. Esta función aplicaría las reglas anteriores, adelantando o retrasando al móvil frente a los otros móviles.

Puesto que el resto de los móviles controlados estarían a la vez aplicando el mismo algoritmo, reajustando sus velocidades de manera independiente, y ya que no hay intercambio de información y cada móvil desconoce las acciones de replanificación del resto, el resultado final no sería previsible. Así, si un móvil hace una estimación espacio-temporal para una interacción con otro móvil, las acciones de replanificación calculadas podrían no ser válidas debido a que el otro móvil también estaría calculando sus propias acciones de replanificación. Esta actitud conduciría a movimientos imprevisibles por parte de todos los móviles, originándose situaciones de bloqueo a lo largo de la ejecución de las actividades.

UN SISTEMA DE CONTROL CENTRALIZADO PARA TODOS LOS MÓVILES CONTROLADOS

El problema de descoordinación anterior se soluciona si se tiene un único sistema de control para todos los móviles controlados. De esta manera, los móviles conocen las acciones de replanificación del resto, pudiendo adaptarse a las mismas y evitándose descoordinaciones que conlleven bloqueos.

La replanificación de las actividades de los móviles se centrará en 1) la deducción de las trayectorias que siguen los móviles no controlados del entorno, y 2) el proceso de predicción de los movimientos y acciones de todos los móviles y los consiguientes ajustes.

La siguiente figura muestra un esquema representativo de esta alternativa:

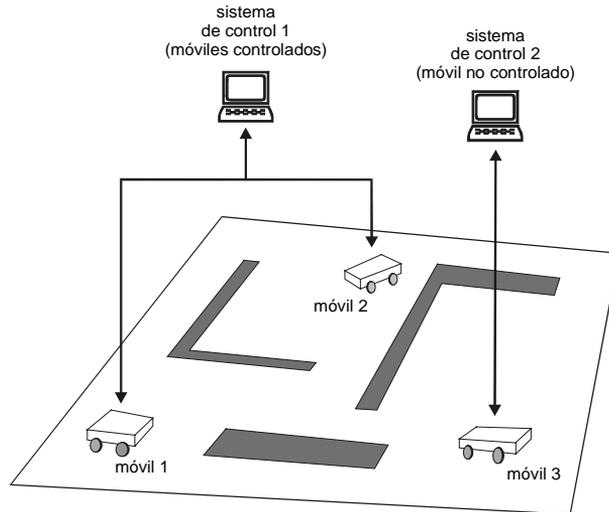


Figura 5-32. Esquema representativo de la existencia de un sistema centralizado para la replanificación de los móviles controlados.

La ventaja que presenta esta configuración estriba en poder replanificar las actividades de los móviles controlados de manera coordinada, conociendo los reajustes que se efectúan en cada móvil. De esta manera, el funcionamiento básico del proceso de replanificación de las actividades de los móviles controlados del entorno es el siguiente:

1. Se deducen las trayectorias que siguen los móviles no controlados.
2. Se comparan las secuencias que representan las trayectorias que siguen en un momento dado todos los móviles.
3. Las trayectorias que siguen los móviles que no se controlan se respetan, dejándolas sin replanificar.
4. Los móviles controlados se ordenan según la prioridad de la actividad que están realizando en ese momento.
5. En primer lugar se replanifican las actividades del móvil más prioritario. A partir de este momento, este móvil se considera como no controlado, replanificándose las actividades de los otros móviles sin interferir en sus movimientos.
6. El resto de móviles se van replanificando teniendo en cuenta los reajustes que han ido efectuando los más prioritarios.

Este funcionamiento básico, igual que se planteó en los casos básicos anteriores, se repite a cada observación del entorno para todas las interacciones calculadas. De esta manera, el algoritmo principal del proceso predictivo, expresado en la Figura 4.1 del capítulo cuarto, sería el mismo.

Por su parte, el proceso deductivo tendría el siguiente algoritmo:

```
observar-y-deducir;
  S=vacío;
  repetir-siempre
    obtener(p,t,m);
    S.m=añadir(p,t);
    coincidencias = comparar-puntos(p) ^
                    comparar-tiempos(t);
    si (coincidencias=1) entonces
      (* secuencia reconocida *)
      devuelve(S.m);
      S.m=vacía;
    fin-si;
    si (coincidencias=0) entonces
      (* nueva secuencia a reconocer *)
      nueva-secuencia(S.m);
    fin-si;
  fin-repetir-siempre;
fin;
```

Figura 5-33. Algoritmo del proceso deductivo en un sistema centralizado para la replanificación de los móviles controlados.

Como se aprecia en el anterior algoritmo, y en contraposición al algoritmo original de la Figura 4.4 del capítulo anterior, se utiliza un vector de secuencias (S) en el que se van almacenando las distintas secuencias reconocidas de los móviles no controlados. Así, la observación del entorno consiste en la obtención de puntos (p) y tiempos de paso (t) para un móvil no controlado determinado (m).

La identificación del móvil que confirma el paso por un punto del entorno y el tiempo asociado, sirve de índice del vector para ir construyendo la secuencia correspondiente (S.m). La deducción de esta secuencia es similar a como ya se planteó en el capítulo anterior. El algoritmo se compone, en fin, de un bucle que se repite siempre en el que se van obteniendo puntos y tiempos de paso asociados a un

móvil, deduciendo las secuencias para cada móvil en cada momento, y devolviéndose las secuencias reconocidas.

El reconocimiento de nuevas trayectorias sería similar al esbozado en el capítulo anterior (algoritmo de la Figura 4.7), con la única salvedad de tener que asociar la trayectoria a un móvil no controlado para su identificación. En un último extremo, las trayectorias almacenadas en la BBDD de Trayectorias no necesitan tener asociadas el identificador del móvil que las realiza. Es en el momento de deducir la trayectoria concreta que sigue un móvil cuando se le asocia este identificador de cara a implementar el proceso predictivo.

Por su parte, el algoritmo básico del proceso predictivo será como sigue:

```
predecir;  
  comparar-secuencias(s1, ..., sn);  
  si (nº-de-interacciones=0) entonces  
    devuelve(nulo);  
  si no  
    para (cada-interacción) hacer  
      secuencias-  
      ordenadas=ordena(interacción.secuencias);  
      repetir  
        para (cada-observación)  
          ajustar(interacción.tipo,  
                 secuencias-ordenadas);  
        fin-para;  
      hasta (pasar-la-interacción);  
    fin-para;  
  fin-si;  
fin;
```

Figura 5-34. Algoritmo del proceso predictivo en un sistema centralizado para la replanificación de los móviles controlados.

Como se aprecia, la estructura del algoritmo del proceso predictivo es similar el expresado en la Figura 5-2. La diferencia estriba en que:

1. La función `comparar-secuencias` tiene en cuenta las secuencias que representan a todas las trayectorias que siguen todos los móviles del entorno.

2. La función *ajustar* realiza la planificación de las actividades de las secuencias involucradas en cada interacción de manera ordenada. Para cada interacción se ordenan las secuencias involucradas (*interacción.secuencias*) según la prioridad de las actividades que representan, dejándose aparte las no controladas. Cuando dos secuencias tienen la misma prioridad, se considera en primer lugar la que lleve más porcentaje de trayectoria recorrida en el momento de la ordenación.

La ordenación de las secuencias implicadas en una interacción se propone a partir de las prioridades asociadas a cada una. En primer lugar se considera una **ordenación por prioridades estática**, es decir, se atiende a la prioridad de ejecución de la actividad que viene determinada por el propio *scheduler* del proceso de producción. De esta manera, las secuencias que representen actividades más prioritarias en su ejecución siempre deben ser respetadas, y no se replanificarán frente a secuencias menos prioritarias, que son las que deben ser reajustadas.

Cuando dos secuencias involucradas en una misma interacción tienen la misma prioridad asignada, entonces se considera una **ordenación por prioridades dinámica** para establecer un orden entre ambas. Esta prioridad dinámica atiende a factores como el porcentaje de secuencia ya ejecutado en el momento de realizarse la ordenación, o la máxima velocidad ajustable posible. De esta manera, la secuencia con más tramos recorridos en el momento de la ordenación será prioritaria frente a la otra, debido a que dispone de un menor margen para reajustarse. De igual manera, cuando en una secuencia el margen para reajustar la velocidad de ejecución sea menor (bien por limitaciones de la propia actividad o del móvil que la ejecuta), se considerará prioritaria frente a la otra.

Puesto que el proceso de predicción viene determinado por las interacciones estimadas entre las secuencias, será necesario tener en cuenta los distintos tipos de interacciones, igual que sucedía en el estudio del caso básico entre un móvil controlado y otro no controlado. Las condiciones a partir de las cuales se reconocen los diferentes tipos de interacción varían de las expresadas anteriormente en la Ecuación 5-1, Ecuación 5-2 y en la Ecuación 5-3. Las nuevas condiciones a evaluar son las siguientes:

1. Punto compartido (condición1).

Ecuación 5-43

$$s_1, \dots, s_n \in S : \\ \exists p_i : (p_i \in s_i) \wedge \dots \wedge (p_i \in s_j) \quad s_i, \dots, s_j \in \{s_1, \dots, s_n\}$$

2. Segmento compartido en el mismo sentido (condición2).

Ecuación 5-44

$$s_1, \dots, s_n \in S : \\ \exists s_s = \{p_i, p_j, \dots, p_k\} : (s_s \subset s_i) \wedge \dots \wedge (s_s \subset s_j) \quad s_i, \dots, s_j \in \{s_1, \dots, s_n\}$$

3. Segmento compartido en sentido contrario (condición3).

Ecuación 5-45

$$s_1, \dots, s_n \in S : \\ \exists \left[(s_s = \{p_i, p_j, \dots, p_k\}) \wedge (\overline{s_s} = \{p_k, \dots, p_j, p_i\}) \right] : \\ (s_s \subset s_i) \wedge \left\{ \overline{s_s} \subset s_j \right\} \wedge \dots \wedge \left\{ \overline{s_s} \subset s_k \right\} \quad s_i, \{s_j, \dots, s_k\} \in \{s_1, \dots, s_n\}$$

Estas condiciones se evalúan en la función `comparar-secuencias` (Figura 5-35), que detecta en un primer momento las posibles interacciones.

Este algoritmo funciona de manera similar al expresado en la Figura 5-3, con la diferencia de que se tienen en cuenta todas las secuencias en ejecución en ese momento. Además, al reconocerse una interacción posible (*interacción*), no sólo se debe guardar el tipo de la misma, sino que se debe establecer a qué secuencias afecta la interacción (*secuencias*). Esta información se guarda para cada interacción encontrada en un vector denominado *interacciones*. Al finalizar, la función devuelve este vector y el número total de interacciones reconocidas, guardado en la variable `nº-de-interacciones`.

```
comparar-secuencias(s1,...,sn);
n°-de-interacciones=0;
interacciones=vacío;
para (s1,...,sn) hacer
    si (condición2) entonces
        interacción={segmento-mismo-sentido,
                      secuencias};
        añadir(interacción,interacciones);
        incrementar(n°-de-interacciones);
        eliminar(interacción,s1,...,sn);
    fin-si;
fin-para;
para (s1,...,sn) hacer
    si (condición3) entonces
        interacción={segmento-sentido-contrario,
                      secuencias};
        añadir(interacción,interacciones);
        incrementar(n°-de-interacciones);
        eliminar(interacción,s1,...,sn);
    fin-si;
fin-para;
para (s1,...,sn) hacer
    si (condición1) entonces
        interacción={punto,secuencias};
        añadir(interacción,interacciones);
        incrementar(n°-de-interacciones);
        eliminar(interacción,s1,...,sn);
    fin-si;
fin-para;
devuelve(interacciones,n°-de-interacciones);
fin;
```

Figura 5-35. Función `compara-secuencias` en un sistema centralizado para la replanificación de los móviles controlados.

Una vez establecidas inicialmente las posibles interacciones, el funcionamiento básico del proceso predictivo se realiza mediante la función `ajustar`, aplicando los ajustes en la velocidad de los móviles controlados de cara a evitar los conflictos que se puedan producir. El algoritmo de esta función se muestra en la siguiente figura:

```
ajustar(interacción.tipo,secuencias-ordenadas);
ajusta-tiempos(secuencias-ordenadas);
caso (interacción.tipo)
  punto:
    para (cada-secuencia-ordenada) hacer
      obtener-información;
      si (secuencia-más-prioritaria)
        NADA;
      sino
        regla1(secuencias-ordenadas);
      fin-si;
    fin-para;
  segmento-mismo-sentido:
    para (cada-secuencia-ordenada) hacer
      obtener-información;
      si (secuencia-más-prioritaria)
        NADA;
      sino
        regla2(secuencias-ordenadas);
      fin-si;
    fin-para;
  segmento-sentido-contrario:
    para (cada-secuencia-ordenada) hacer
      obtener-información;
      si (secuencia-más-prioritaria)
        NADA;
      sino
        regla3(secuencias-ordenadas);
      fin-si;
    fin-para;
  fin-caso;
fin;
```

Figura 5-36. Función ajustar en un sistema centralizado para la replanificación de los móviles controlados.

Para cada interacción, se replanifica cada secuencia de manera ordenada. La replanificación viene determinada por un ajuste de velocidades que se realiza a través de unas reglas para cada tipo de interacción, similares a las vistas en la Figura 5-5, la Figura 5-15 y la Figura 5-24. En los casos anteriores, las reglas sólo tenían en cuenta dos secuencias, la del móvil controlado y la del móvil no controlado, ajustándose la velocidad del móvil controlado. En este caso, las reglas

deben tener en cuenta la secuencia del móvil a replanificar y todas las secuencias que representan a las trayectorias más prioritarias que la que se pretende replanificar. Si la secuencia a replanificar es la más prioritaria, no se replanifica.

Replanificar a partir de las secuencias más prioritarias, sin interferir en las mismas, es lo mismo que considerar a estas secuencias como representativas de móviles no controlados. La diferencia estriba en que hay que considerar que puede haber más de una secuencia con mayor prioridad, debiendo tenerse en cuenta todos los intervalos de tiempo asociados. Estos intervalos se pueden englobar en uno sólo, teniendo en cuenta los tiempos más restrictivos para establecer sus límites (mínimo de los tiempos mínimos y máximo de los tiempos máximos).

Así pues, las reglas a aplicar para cada caso serían como siguen:

1. Punto compartido (regla1).

Considerando la secuencia a replanificar (si), y el conjunto de secuencias más prioritarias ({sj,...,sk}), los intervalos temporales se calculan a partir de la siguiente expresión:

Ecuación 5-46

$$\begin{aligned}
 si \Rightarrow pi \Rightarrow t_1 &= [t_{1-\min}, t_{1-\max}] = [t_{1-\min}^{si}, t_{1-\max}^{si}] \\
 \{sj, \dots, sk\} \Rightarrow pi \Rightarrow t_2 &= [t_{2-\min}, t_{2-\max}] = [\min(t_{1-\min}^{sj}, \dots, t_{1-\min}^{sk}), \max(t_{1-\max}^{sj}, \dots, t_{1-\max}^{sk})]
 \end{aligned}$$

Con lo que las reglas de ajuste de velocidades serían:

$$\exists pi : [(pi \in si ; t_1) \wedge (pi \in \{sj, \dots, sk\} ; t_2)] ;$$

1. [interacción-temporal(t_1, t_2) \wedge (viable-adelantar=CIERTO)]
 \rightarrow adelantar($t = t_{1-\max} - t_{2-\min}$)
 2. [interacción-temporal(t_1, t_2) \wedge (viable-adelantar=FALSO)]
 \rightarrow retrasar($\forall mnc - en - pi + 1$)
-

Figura 5-37. Reglas de actuación para el caso del punto compartido, para la replanificación de los móviles controlados con un sistema centralizado.

La interacción temporal se calcula en este caso de la misma manera como se hacía en los casos anteriores, a partir de la definición de la Figura 5-6.

Cabe apreciar que al adelantar el móvil controlado se tiene en cuenta el intervalo temporal que abarca a todos los móviles con secuencias más prioritarias, siendo el cálculo del tiempo de adelanto similar a cómo se hacía con anterioridad. Por su parte, cuando se debe retrasar el móvil controlado, ahora hay que tener en cuenta que todos los móviles con mayor prioridad (considerados como no controlados) hayan llegado al punto siguiente al de la interacción en su secuencia correspondiente ($\forall mnc-en-pi+1$).

2. Segmento compartido en el mismo sentido (regla2).

Atendiendo a las definiciones del segmento compartido en la misma dirección (Ecuación 5-44), cuando se considera todo el segmento para intentar aplicar un adelanto sobre el móvil controlado de manera que este pase y abandone el segmento antes que el resto de móviles, los intervalos temporales asociados a la secuencia del móvil controlado a replanificar (si) y, de manera global, a las secuencias de los móviles con mayor prioridad ($\{sj, \dots, sk\}$), se definen como sigue:

Ecuación 5-47

$$\begin{aligned}
 si \Rightarrow s_s = \{pi, pj, \dots, pk\} \Rightarrow t_1 &= [t_{1-\min}, t_{1-\max}] = [t_{i-\min}^{si}, t_{k-\max}^{si}] \\
 \{sj, \dots, sk\} \Rightarrow s_s = \{pi, pj, \dots, pk\} \Rightarrow t_2 &= [t_{2-\min}, t_{2-\max}] = \\
 &= [\min(t_{i-\min}^{sj}, \dots, t_{i-\min}^{sk}), \max(t_{k-\max}^{sj}, \dots, t_{k-\max}^{sk})]
 \end{aligned}$$

Siendo s_s el segmento compartido, t_1 el intervalo temporal asociado a si para s_s , y t_2 el intervalo temporal que abarca a todos los intervalos asociados a $\{sj, \dots, sk\}$ para s_s .

En este caso se tiene en cuenta la posibilidad de que el móvil controlado avance por el segmento compartido detrás del resto de móviles, a una cierta distancia de seguridad. Esta distancia de seguridad se establece teniendo en cuenta los tramos que componen el segmento compartido y sus intervalos temporales asociados:

Ecuación 5-48

$$s_s = \{p_i, p_j, \dots, p_k\} \equiv \{p_i, p_j\} + \dots + \{p(k-1), p_k\} = s_1 + \dots + s_{(k-i)} \rightarrow$$

$$\rightarrow \forall s_i = \{p_i, p_j\} \Rightarrow \begin{cases} s_i \rightarrow t_3 = [t_{3-\min}, t_{3-\max}] = [t_{i-\min}^{s_i}, t_{j-\max}^{s_i}] \\ \{s_j, \dots, s_k\} \rightarrow t_4 = [t_{4-\min}, t_{4-\max}] = \\ [\min(t_{i-\min}^{s_j}, \dots, t_{i-\min}^{s_k}), \max(t_{j-\max}^{s_j}, \dots, t_{j-\max}^{s_k})] \end{cases}$$

Con estas definiciones de los intervalos asociados a la totalidad del segmento compartido y a cada uno de los tramos de dicho segmento, se establecen las siguientes reglas de ajuste de velocidades:

$$\exists s_s = \{p_i, \dots, p_k\} = (s_1 + \dots + s_{(k-i)}), \quad s_i = \{p_i, p_{i+1}\} :$$

$$\{(s_s \subset s_i; t_1) \wedge (s_s \subset \{s_j, \dots, s_k\}; t_2)\} \wedge$$

$$\{(s_i \subset s_s \subset s_i; t_3) \wedge (s_i \subset s_s \subset \{s_j, \dots, s_k\}; t_4)\};$$

1. [interacción-temporal(t_1, t_2) \wedge (viable-adelantar=CIERTO)]
 \rightarrow adelantar($t = (t_{1-\max} - t_{2-\min}) + t_s$)
 2. [interacción-temporal(t_1, t_2) \wedge (viable-adelantar=FALSO)] \wedge
 [interacción-temporal(t_3, t_4) \wedge (viable-adelantar=CIERTO), $\forall s_i$]
 \rightarrow adelantar($t = (t_{3-\max} - t_{4-\min}) + t_s$)
 3. [interacción-temporal(t_1, t_2) \wedge (viable-adelantar=FALSO)] \wedge
 [interacción-temporal(t_3, t_4) \wedge ($\exists s_i$: viable-adelantar=FALSO)]
 \rightarrow retrasar($\forall mnc$ -en- p_j+1), $\forall p_j \in s_s$
-

Figura 5-38. Reglas de actuación para el caso del segmento compartido en el mismo sentido, para la replanificación de los móviles controlados con un sistema centralizado.

3. Segmento compartido en sentido contrario (regla3).

A partir de la definición de la Ecuación 5-45, en la que se establece la condición de segmento compartido en distinto sentido, se establecen los siguientes intervalos temporales asociados a la secuencia del móvil controlado a replanificar (s_i) y, de manera global, a las secuencias de los móviles con mayor prioridad ($\{s_j, \dots, s_k\}$):

Ecuación 5-49

$$\begin{aligned}
 si \Rightarrow s_s = \{pi, pj, \dots, pk\} \Rightarrow t_1 = [t_{1-\min}, t_{1-\max}] &= [t_{i-\min}^{si}, t_{k-\max}^{si}] \\
 \{sj, \dots, sk\} \Rightarrow \overline{s_s} = \{pk, \dots, pj, pi\} \Rightarrow t_2 = [t_{2-\min}, t_{2-\max}] &= \\
 [\min(t_{k-\min}^{sj}, \dots, t_{k-\min}^{sk}), \max(t_{i-\max}^{sj}, \dots, t_{i-\max}^{sk})] &
 \end{aligned}$$

Siendo s_s el segmento compartido en el sentido de avance del móvil controlado, $\overline{s_s}$ el mismo segmento pero en sentido de avance de los móviles más prioritarios, t_1 el intervalo de tiempo asociado a si para s_s , y t_2 el intervalo temporal que abarca a todos los intervalos asociados a $\{sj, \dots, sk\}$ para $\overline{s_s}$.

Con estos intervalos definidos, las reglas de ajuste de velocidades se establecen como siguen:

$$\exists [s_s = \{pi, \dots, pk\} \wedge \overline{s_s} = \{pk, \dots, pi\}] :$$

$$[(s_s \subset si; t_1) \wedge (\overline{s_s} \subset \{sj, \dots, sk\}; t_2)] :$$

1. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=CIERTO)]
→ adelantar($t = (t_{1-\max} - t_{2-\min}) - t_s$)
 2. [interacción-temporal(t_1, t_2) \wedge
(viable-adelantar=FALSO)]
→ retrasar($\forall mnc - en - pi - 1$)
-

Figura 5-39. Reglas de replanificación para el caso del segmento compartido en el sentido contrario, para la replanificación de los móviles controlados con un sistema centralizado.

ALTERNATIVAS Y CONSIDERACIONES

En esta apartado se comentan algunas alternativas y consideraciones que se pueden tener en cuenta en la replanificación de múltiples móviles. Estas alternativas complementan el planteamiento básico de replanificación presentado en los apartados anteriores. En concreto, se tienen en cuenta las siguientes consideraciones en la replanificación de múltiples móviles:

1. Replanificación solidaria.

El planteamiento básico de replanificación recurre a una ordenación de las secuencias implicadas por prioridades, de manera que son los móviles que ejecutan secuencias con menor prioridad los que deben reajustar sus secuencias adaptándose a las actividades de los más prioritarios. Este planteamiento, que permite el reajuste de las secuencias de una manera simple y segura, supone que algunos móviles deban, bien adelantarse forzando al máximo sus velocidades, bien retrasarse en exceso para evitar una interacción; mientras que los móviles más prioritarios no modifican para nada sus secuencias.

Se podrían evitar estas situaciones a partir de un planteamiento de replanificación *solidario*, en el que todos los móviles reajustaran sus secuencias para evitar grandes desfases en los tiempos finales de ejecución de las mismas. Así, si en el planteamiento anterior un móvil debería retrasarse un cierto tiempo para evitar una interacción, esperando que otros dos móviles pasen por un punto compartido, se puede plantear que los dos móviles más prioritarios adelanten sus secuencias para que el retraso que debe hacer el menos prioritario no sea excesivo. Este tipo de planteamiento permitiría que el proceso de producción global no se desajustara en exceso, habiendo secuencias que se adelanten o retrasen en demasía.

2. Interacciones solapadas.

Uno de los problemas que pueden aparecer al considerar múltiples móviles controlados y no controlados en un mismo entorno es el de las interacciones solapadas. Esto es, que entre unas secuencias se comparta un tipo de interacción (por ejemplo, un segmento compartido) y que, al mismo tiempo, esta interacción se solape con otro tipo de interacción con otras secuencias (por ejemplo, que el segmento compartido esté incluido en otro segmento mayor, compartido con otras secuencias).

La siguiente figura muestra un ejemplo posible de esta situación:

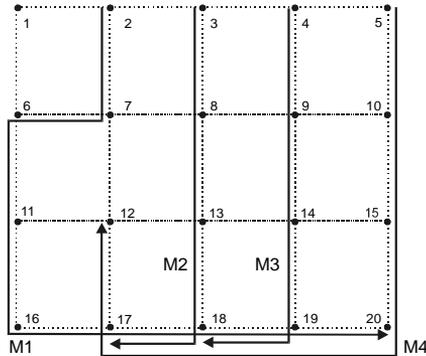


Figura 5-40. Entorno ejemplo con secuencias con interacciones solapadas.

En esta figura se muestran cuatro secuencias en el entorno ejemplo ya visto anteriormente. Estas secuencias están definidas por las siguientes secuencias de puntos:

Ecuación 5-50

$$\begin{aligned}
 M1 &= \{p2, p7, p6, p11, p16, p17, p18, p19, p20\} \\
 M2 &= \{p3, p8, p13, p18, p17\} \\
 M3 &= \{p4, p9, p14, p19, p18\} \\
 M4 &= \{p5, p10, p15, p20, p19, p18, p17, p12\}
 \end{aligned}$$

Las interacciones que se tienen son las siguientes:

- M1 y M2 comparten un segmento en dirección opuesta ($\{p17, p18\}$ en M1 y $\{p18, p17\}$ en M2).
- M1 y M3 comparten un segmento en dirección opuesta ($\{p18, p19\}$ en M1 y $\{p19, p18\}$ en M3).
- M1 y M4 comparten un segmento en dirección opuesta ($\{p17, p18, p19, p20\}$ en M1 y $\{p20, p19, p18, p17\}$ en M4).
- M2 y M3 comparten un punto (p18).
- M2 y M4 comparten un segmento en el mismo sentido ($\{p18, p17\}$).
- M3 y M4 comparten un segmento en el mismo sentido ($\{p19, p18\}$).

Como se aprecia, estas interacciones se solapan entre sí (el segmento mayor contiene al resto de segmentos en los que se detecta interacción).

La resolución de este tipo de conflictos es abordable con el planteamiento básico de replanificación anterior, y resulta más complicada cuando se intenta una replanificación *solidaria*. Esto es así porque los ajustes *solidarios* entre dos secuencias que comparten una interacción pueden conducir a desajustes mayores entre una de estas secuencias y una tercera secuencia con otro interacción solapada.

Así, abordando la replanificación desde el punto de vista del planteamiento original (replanificación por prioridades), considerando todas las interacciones solapadas como pertenecientes al mismo grupo, la replanificación es posible.

Para el ejemplo anterior, si suponemos que las secuencias involucradas se ordenan según sus prioridades de la siguiente manera:

Ecuación 5-51

$$M1 > M2 > M3 > M4$$

la replanificación sería como sigue:

1. M1 mantendría la secuencia intacta, siendo la de mayor prioridad no se replanificaría.
2. M2 se replanificaría adaptándose a M1, lo cual implicaría un adelanto o retraso en su ejecución.
3. M3 se replanificaría de acuerdo a M1 y a M2. En este caso, se tendría en cuenta la ejecución de M1 en el segmento compartido y el paso de M2 por el punto que comparten.
4. Por último, M4 debería replanificarse según la ejecución de M1 del segmento compartido y las de M2 y M3 por los segmentos compartidos respectivos.

3. Techos de prioridades.

Una consideración necesaria que debe hacerse al plantear una replanificación por prioridades es el establecimiento de un *techo de prioridad* para una interacción concreta. Esto es, forzar a que cuando un móvil esté ejecutando su secuencia pasando por una zona compartida por otros móviles, adquiera momentáneamente la mayor prioridad de los móviles que van a circular por esta zona. Esta característica implicaría que en ningún caso se podría aplicar una regla de replanificación que implicara el reajuste de un móvil a partir de una interacción que esté ejecutando en ese momento.

Supongamos el siguiente ejemplo:

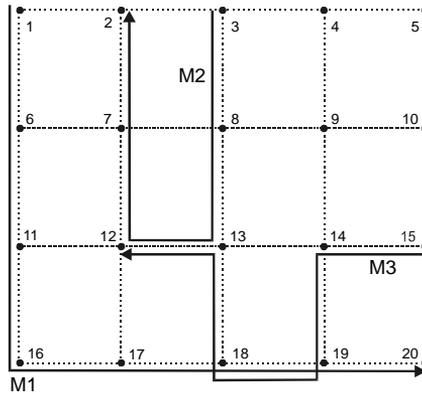


Figura 5-41. Entorno ejemplo con tres secuencias y dos interacciones que no se solapan entre ellas.

En la figura anterior aparecen tres secuencias definidas como sigue:

Ecuación 5-52

$$\begin{aligned}
 M1 &= \{p1, p6, p11, p16, p17, p18, p19, p20\} \\
 M2 &= \{p3, p8, p13, p12, p7, p2\} \\
 M3 &= \{p15, p14, p19, p18, p13, p12\}
 \end{aligned}$$

De las cuales se deducen las siguiente interacciones:

- M1 y M3 comparten un segmento en sentido opuesto ($\{p18, p19\}$ en M1 y $\{p19, p18\}$ en M3).
- M2 y M3 comparten un segmento en el mismo sentido ($\{p13, p12\}$).

Supongamos la secuencia que recorre M1 tenga asignada una mayor prioridad que la que recorre M3, y que ésta, a su vez, tenga una mayor prioridad que la que recorre M2. Para esta situación, la replanificación atendiendo a las prioridades y por el orden en que ocurren las interacciones, sería como sigue:

1. M3 se replanificaría ajustando su secuencia para adaptarse a la de M1.
2. M2 debería replanificarse posteriormente adaptándose a la ejecución de la secuencia que recorre M3, y por tanto a los reajustes ya hechos con respecto a M1.

Supóngase que al replanificar la secuencia que recorre M3 frente a M1, no es posible adelantarla y se deba retrasar M3 hasta que M1 llegue a p20 (el siguiente punto en su secuencia al segmento compartido). Mientras M3 permanece retrasado, M2 puede estar recorriendo su secuencia, siendo posible que éste entre en el segmento compartido cuando M3 vuelva a reanudar su marcha. En este momento debería replanificarse la segunda interacción, en la que M2 tiene una menor prioridad que M3, pero aquí surge el problema de que el móvil con la menor prioridad ya ha accedido a la zona de interacción, siendo imposible su replanificación.

En estas situaciones, la solución es asignar a todo móvil que esté recorriendo una zona de interacción con otros móviles el máximo de las prioridades de los móviles involucrados, es decir, un techo de prioridad. De esta manera al replanificarse las secuencias involucradas de manera ordenada, si hay algún móvil que ya esté dentro de la zona compartida será el de mayor prioridad y por tanto no se replanificará.

CONCLUSIONES

A lo largo de este capítulo se ha presentado un método de replanificación de las trayectorias que siguen los robots móviles en un entorno dinámico. El método se complementa con un proceso de deducción de las trayectorias que siguen otros móviles en el mismo entorno, detallado en el capítulo anterior.

La replanificación de las trayectorias de los móviles controlados se plantea a partir del razonamiento temporal aplicado a las secuencias que representan a las distintas trayectorias que siguen los móviles del entorno. Este razonamiento temporal, base del proceso predictivo, permite la estimación de acciones de futuro sobre las trayectorias, permitiendo considerar posibles interacciones entre ellas.

El estudio de las diferentes interacciones entre secuencias permite abordar los distintos casos que se pueden dar entre dos móviles que recorren sendas trayectorias en un mismo entorno. Así, en este capítulo se han formalizado y definido los condicionantes de los casos básicos (punto compartido, segmento compartido en el mismo sentido de avance y segmento compartido en sentido contrario).

De la consideración de cada caso se desprende una posible replanificación de la trayectoria que recorre el móvil controlado, que se plasma en un reajuste de su velocidad, adelantándose o

retrasándose el móvil controlado para evitar la colisión con el otro móvil. Este reajuste de velocidades se aplica a través de unas reglas de comportamiento básico, formalizadas a partir de las características de cada caso. De igual manera, para aplicar el razonamiento temporal en cada caso, se han formalizado y definido los intervalos temporales asociados a cada interacción, tanto para el móvil controlado, como para el otro móvil. Se han incluido ejemplos detallados que ilustran la aplicación del método de replanificación para cada caso.

Finalmente, se ha considerado la aplicación del método cuando en un mismo entorno aparecen múltiples móviles controlados y no controlados. En esta situación, se han tenido en cuenta dos alternativas: sistemas de control individuales o un mismo sistema de control centralizado. El planteamiento básico de replanificación se ha adaptado finalmente a un sistema centralizado de control que permitiera coordinar la replanificación de más de un móvil controlado. La propuesta se basa en la aplicación de una replanificación a partir de las prioridades asignadas a cada secuencia. En este apartado final, se consideran alternativas de replanificación y consideraciones que pueden conllevar problemas en el reajuste de las secuencias, indicando la solución y comentando las posibilidades de cada caso.

Tanto para el caso básico de dos móviles, como para el caso de múltiples móviles, se aportan los algoritmos y funciones que implementan el método propuesto.



Replanificación: El proceso predictivo

6 *Arquitectura software*

Arquitectura software

Este capítulo se dedica a presentar una propuesta de una arquitectura software para un robot móvil que soporte la metodología de planificación y replanificación expuesta a lo largo de esta tesis doctoral. En primer lugar, se referencian distintas arquitecturas de control conocidas, incidiendo en sus características en lo que respecta a la planificación de robots. Posteriormente se detallan y especifican los módulos que componen la arquitectura. Los algoritmos y funciones expuestas en los capítulos anteriores como parte de la metodología se asocian a cada módulo de la arquitectura, especificando las dependencias entre ellos y el flujo de los datos.

Finalmente, se presenta una primera aproximación práctica a esta arquitectura, implementada en el contexto del proyecto RETRARO, ya referenciado en anteriores capítulos. Esta primera aproximación supone el desarrollo básico de los procesos de descomposición y abstracción como principales características de la planificación de actividades de robot.

Arquitectura software

ARQUITECTURAS SOFTWARE DE CONTROL

El software y el hardware implicados en el control de un robot configuran la arquitectura del mismo. La arquitectura software está estrechamente relacionada con el hardware, de manera que se puede considerar como una abstracción del mismo en forma de un código de control. Una arquitectura software para el control de un robot viene determinada a partir de una serie de módulos de código con comunicación entre ellos.

La gran mayoría de arquitecturas existentes están diseñadas con la intención de resolver problemas concretos, incorporando a nivel físico los componentes necesarios para llevar a cabo las funciones previstas por el robot. El desarrollo de los módulos de código que se corresponden con las características físicas del robot completa una arquitectura de control.

Durante años, se han venido aplicando arquitecturas genéricas basadas en principios de la Inteligencia Artificial al control de robots. Pero la falta de aplicabilidad de este tipo de arquitecturas de propósito general ha hecho que se cuestionen, desarrollándose, como antes se ha comentado, arquitecturas basadas en la resolución o aplicabilidad de principios más concretos.

Es habitual encontrar en la bibliografía referencias a arquitecturas desarrolladas para robots móviles. Se trata de un campo de acción más amplio que el de los robots articulados, donde no existen características atractivas de resolver como es la autonomía de los propios robots. De esta manera, se desarrollan arquitecturas que pretenden abordar el comportamiento de los robots móviles a lo largo de la ejecución de una actividad concreta a través de su interacción con el entorno, algo que se manifiesta de manera clara en los robots móviles autónomos.

En [Simó, 1997] se tiene una buena aproximación a distintas arquitecturas para robots móviles, incidiendo en el comportamiento reactivo de las mismas. En este trabajo se presenta, además, una propuesta de arquitectura de control para robots móviles basada en motivaciones. Así, se centra la atención en el proceso reactivo dentro de la ejecución de las actividades de un robot móvil, haciéndose especial hincapié en las restricciones de tiempo real del problema.

De nuevo aparece un repaso a las arquitecturas de robots móviles en [Muñoz, 1998]. En este caso, se hace especial énfasis las

clasificaciones de las arquitecturas en cuanto a su filosofía y se propone una arquitectura de robots móviles basada en comportamientos. La arquitectura propuesta tiene como objetivos principales el abordar las tareas de navegación del móvil bajo diferentes tipos de funcionamiento: tele-operación, semiautónomo y autónomo.

Existen otras aproximaciones distintas a la arquitectura de un robot móvil que se basan en características concretas. Así, por ejemplo, en [Schoner *et al.*, 1995] se establecen los principios de las arquitecturas de control de robots móviles a través de la teoría de sistemas dinámicos aplicada a la robótica basada en el comportamiento. Más recientemente, en [Beccari *et al.*, 1998] se presenta una librería de funciones en tiempo real que facilita el desarrollo de arquitecturas de control que inciden en los procesos reactivos.

Por otro lado, en lo que respecta a sistemas de tele-robótica, en [Graves, 1995] se tiene una propuesta de arquitectura generalizada. Esta arquitectura genérica se basa en la fusión de decisiones que provienen de múltiples fuentes (factores humanos, reactivos, deliberativos) para establecer diferentes modos de operación (compartido, tele-operación, supervisor) a partir de diferentes eventos y situaciones.

En [López, 2000] se tiene un estudio de la relación entre sistemas de supervisión y detección de fallos, su recuperación y las arquitecturas de navegación de un robot móvil. En este trabajo se abordan los distintos tipos de arquitecturas existentes, sus características, y se ubica en ellos la existencia de un módulo supervisor encargado de la diagnosis del sistema. En este contexto, se presenta una metodología de diagnosis y recuperación de fallos que se aplica a distintas arquitecturas de robots móviles existentes.

En este capítulo se presenta una arquitectura de control software para un robot móvil acorde con la metodología de planificación y replanificación de robots detallada a lo largo de los capítulos anteriores. El objetivo de este capítulo no es el de introducir ni describir una nueva arquitectura de control para robots móviles, sino el de transcribir la metodología de planificación y replanificación en una arquitectura genérica, acorde a las características del método. De esta manera, la arquitectura que aquí se presenta cumple los siguientes requisitos:

- Se refleja la abstracción jerárquica que se realiza sobre el problema a resolver, existiendo unos módulos principales encargados de la resolución por niveles del problema.

- La descomposición del problema se asume a partir del primer estadio de planificación (planificación de misiones).
- La replanificación de las actividades se asumen a bajo nivel, insertándose un módulo de replanificación de trayectorias.

A continuación se detalla la composición de la arquitectura propuesta.

ARQUITECTURA PROPUESTA

Tal y como se adelanta en el punto anterior, una de las características principales de la arquitectura que se propone es su paralelismo con la abstracción jerárquica definida en la metodología. Así, los principales niveles establecidos para la planificación de las actividades de un robot (misiones, tareas, movimientos, trayectorias) tienen su propio módulo de planificación, comunicados con el resto de componentes de la arquitectura.

La Figura 6-1 muestra un esquema de la arquitectura software propuesta.

La arquitectura que se propone tiene, a primera vista, dos principales características:

1. Paralelismo con la jerarquización en niveles.
2. Replanificación a nivel de trayectorias.

La estructura principal de planificación está definida por los módulos abordan el proceso de planificación en cada nivel. Estos módulos se comunican en cascada entre sí, obteniendo datos del módulo superior y devolviendo resultados al módulo inferior. Existe una realimentación “hacia arriba” entre los módulos para solucionar situaciones en las que no se pueda planificar un nivel determinado.

La replanificación de trayectorias se refleja en la arquitectura, como principal aportación de la metodología, con la inclusión del módulo DEDUCTIVO+PREDICTIVO. Este módulo recibe datos de sensorización del entorno, tanto del robot controlado como de los robots no controlados, detectando las trayectorias que siguen los distintos móviles y generando nuevos parámetros para reajustar la trayectoria del móvil controlado.

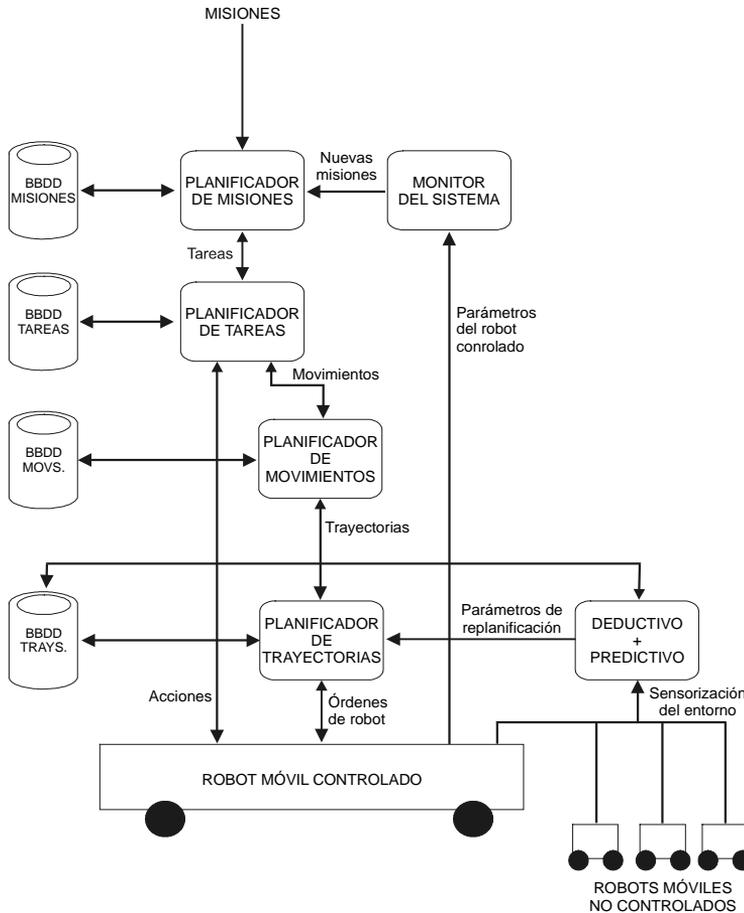


Figura 6-1. Arquitectura software propuesta.

Complementando a estas dos características principales, se incorporan las bases de datos que contienen información sobre misiones, tareas, movimientos y trayectorias ya ejecutadas. Se trata de un histórico sobre las actividades de los robots en el entorno ya introducido en los capítulos anteriores, que juega un papel importante tanto para la planificación de cada uno de los niveles, como en la replanificación de las trayectorias.

A continuación se detallan los módulos que componen la arquitectura y sus principales características.

PLANIFICADOR DE MISIONES

El planificador de misiones tiene como principal objetivo la generación de las tareas de robot a partir de la misión encomendada. Esta misión viene determinada por el planificador global del proceso de producción, que asigna misiones a cada móvil, coordinando las actividades del proceso productivo y utilizando como patrón el proceso de descomposición detallado en la metodología de planificación.

De igual manera, este módulo recibe también como entrada nuevas misiones que vienen determinadas por el Monitor del Sistema, distintas a las que provienen del planificador global. Así, cualquier eventualidad, como un nivel bajo de batería, queda determinada por el Monitor del Sistema, que genera la misión correspondiente para que sea resuelta en el momento adecuado, dentro de la secuencia global de misiones. El establecimiento de prioridades en las misiones permite decidir el orden de ejecución de las mismas.

Con todo, la planificación de misiones supone el nivel más alto de abstracción, y el objetivo es el de generar la secuencia de tareas necesarias para su resolución, con independencia del robot y del entorno en que se debe resolver.

La siguiente tabla resume las características de este módulo:

Planificador de Misiones	
entradas	<ul style="list-style-type: none"> • Misiones asignadas por el planificador global • Misiones asignadas por el Monitor del Sistema • Datos de la BBDD de Misiones • Restricciones del nivel inferior (realimentaciones)
salidas	<ul style="list-style-type: none"> • Tareas que resuelven la misión • Nuevos datos para la BBDD de Misiones
métodos de planificación sugeridos	<ul style="list-style-type: none"> • Sistemas expertos • Grafos de Alto Nivel

Tabla 6-1. Características del Planificador de Misiones

PLANIFICADOR DE TAREAS

Las tareas se planifican generándose los movimientos y acciones que debe llevar a cabo el robot para resolverlas. Se trata, como ya se avanzó en su momento, de una planificación de nivel intermedio que depende del entorno en el que se realizan las tareas. Las acciones son actividades básicas que no generan desplazamiento del robot y que se envían de manera directa al móvil controlado para su ejecución a bajo nivel. Por su parte, los movimientos deben ser planificados a un nivel más concreto.

Tanto la imposibilidad de ejecutar una acción como un movimiento, supondrá la realimentación desde niveles inferiores de las restricciones que lo impiden. A partir de aquí se establece el proceso de replanificación “hacia arriba” común a todos los módulos.

De igual manera, cuando no es posible planificar una tarea descomponiéndola en las acciones y movimientos pertinentes, se devuelve al módulo superior (planificador de misiones) el conjunto de restricciones que así lo indican, de cara a que se establezca una nueva planificación de las misiones.

La siguiente tabla resume las características del módulo:

Planificador de Tareas	
entradas	<ul style="list-style-type: none"> • Tareas resultado de la planificación de la misión • Datos de la BBDD de Tareas • Restricciones del nivel inferior (realimentaciones)
salidas	<ul style="list-style-type: none"> • Acciones y Movimientos que resuelven la tarea • Nuevos datos para la BBDD de Tareas • Restricciones al nivel superior (realimentaciones)
métodos de planificación sugeridos	<ul style="list-style-type: none"> • Búsquedas en Grafos • Análisis mediante Redes de Petri

Tabla 6-2. Características del Planificador de Tareas.

PLANIFICADOR DE MOVIMIENTOS

La planificación de movimientos, ampliamente estudiada en robótica, genera las trayectorias concretas que debe seguir el móvil para resolver los desplazamientos. En esta arquitectura, el módulo planificador de movimientos se encarga de generar las trayectorias concretas que debe seguir el robot móvil (ver ejemplos detallados en las Figuras 3.15 y 3.16 del capítulo tercero).

Las entradas a este módulo son los movimientos necesarios para realizar una tarea, provenientes del módulo superior, y las salidas serán la especificación de esos movimientos en trayectorias concretas que incluyan coordenadas del espacio (puntos concretos) y tiempos de paso asociados. Por otro lado, la realimentación desde el nivel inferior y hacia el nivel superior de restricciones que indiquen la imposibilidad de planificación suponen datos para la replanificación, tal y como sucede en los módulos ya detallados.

El resumen de características del Planificador de Movimientos se muestra en la siguiente tabla:

Planificador de Movimientos	
entradas	<ul style="list-style-type: none"> • Movimientos resultado de la planificación de las tareas • Datos de la BBDD de Movimientos • Restricciones del nivel inferior (realimentaciones)
salidas	<ul style="list-style-type: none"> • Trayectorias concretas a seguir por el robot móvil. • Nuevos datos para la BBDD de Movimientos • Restricciones al nivel superior (realimentaciones)
métodos de planificación sugeridos	<ul style="list-style-type: none"> • <i>Roadmap</i>, descomposición celular, campos potenciales. • Planificación dinámica....

Tabla 6-3. Características del Planificador de Movimientos.

PLANIFICADOR DE TRAYECTORIAS

El objetivo básico de este módulo es el de generar las órdenes de robot básicas para ejecutar una trayectoria. El proceso de planificación recibe como entradas las trayectorias del nivel superior,

generándose los comandos de robot a partir de las características y especificaciones del robot a controlar. Cuando no es posible la ejecución de la trayectoria u ocurre cualquier eventualidad que lo imposibilite, se devuelven las restricciones pertinentes hacia el nivel superior, como realimentaciones del planificador.

La planificación de trayectorias se completa con un *módulo reactivo*, ubicado sobre el propio robot, capaz de reaccionar ante imprevistos (por ejemplo, un obstáculo no previsto en un corredor) a partir de datos de sensorización del propio móvil. Este nivel reactivo, ajeno a la planificación de trayectorias, intenta evitar los imprevistos (por ejemplo, rodeando el obstáculo detectado) sin tener que replanificar la trayectoria programada. Cuando esto no es posible, se devuelven desde el propio móvil las restricciones detectadas al Planificador de Trayectorias para intentar una planificación alternativa. Se trata del proceso de realimentación “hacia arriba” ya detallado en anteriores módulos, también incluido en este nivel.

La metodología incluye un proceso alternativo de replanificación a partir del conocimiento de las trayectorias que siguen otros móviles del entorno. Esto se refleja en este nivel con la entrada de nuevos parámetros de replanificación al Planificador de Trayectorias. Estos datos provienen del módulo Deductivo + Predictivo, que se detalla más adelante. En este caso no se trata de buscar una planificación alternativa a la trayectoria, sino de reajustar los parámetros de la trayectoria a partir, principalmente, de nuevos valores de velocidades.

La siguiente tabla resume este módulo:

Planificador de Trayectorias	
entradas	<ul style="list-style-type: none"> • Trayectorias resultado de la planificación de los Movimientos • Datos de la BBDD de Trayectorias • Restricciones del nivel reactivo (realimentaciones)
salidas	<ul style="list-style-type: none"> • Trayectorias concretas a seguir por el robot móvil. • Nuevos datos para la BBDD de Movimientos • Restricciones al nivel superior (realimentaciones)
métodos de planificación sugeridos	<ul style="list-style-type: none"> • Conversión en órdenes de robot (programación) • Ajuste de velocidades (replanificación)

Tabla 6-4. Características del Planificador de Trayectorias.

MONITOR DEL SISTEMA

El Monitor del Sistema está encargado de realizar el control sobre los parámetros de funcionamiento principales del robot móvil. Así, este módulo se encarga de chequear el estado de las baterías del robot móvil o de controlar el buen funcionamiento del utillaje incluido en el móvil (por ejemplo, los sistemas de carga y descarga de objetos). En cualquier caso, el monitor del sistema, aparte de informar sobre los valores de estado del robot, genera nuevas misiones para resolver situaciones cotidianas o no relacionadas con el robot móvil, fuera de las actividades propias del proceso productivo global.

La entrada, pues, a este módulo son los valores de los parámetros principales del robot móvil (sensorización propioceptiva), mientras que las salidas serán nuevas misiones que permitan resolver eventualidades relacionadas con los valores del sistema leídos. Así por ejemplo, ante la lectura de un nivel bajo de las baterías del robot móvil, se debe generar una misión para recarga o cambio de las mismas, con la prioridad adecuada para que sea ejecutada dentro del proceso productivo.

Monitor del Sistema	
entradas	<ul style="list-style-type: none"> • Valores de los parámetros del sistema o variables de estado del robot móvil (sensorización propioceptiva).
salidas	<ul style="list-style-type: none"> • Nuevas misiones para resolver eventualidades propias del sistema.
actividades	<ul style="list-style-type: none"> • Deducción y extracción de misiones a partir de valores del sistema.

Tabla 6-5. Características del Monitor del Sistema.

DEDUCTIVO+PREDICTIVO

El módulo Deductivo+Predictivo se encarga de todo el proceso de replanificación *on line* de las trayectorias. Se trata éste de un proceso realizado a bajo nivel y en el que se incluyen todas las funcionalidades descritas en los capítulos anteriores. Así pues, el objetivo principal de este módulo es el de generar nuevos parámetros (velocidades y requisitos temporales) para el reajuste de las trayectorias del móvil controlado en tiempo real. Estos parámetros se extraen a partir de la información de sensorización del entorno, conforme se van confirmando las trayectorias en ejecución de cada móvil. Tal y como ya se ha descrito, un proceso de deducción de las trayectorias de los

móviles no controlados y el posterior razonamiento temporal de las acciones y desplazamientos futuros de los móviles, permitirá, a partir de unas reglas de comportamiento, recalcular los valores de velocidad de las trayectorias y por tanto su replanificación.

De cara a que este proceso se pueda realizar *on line* y cumpla los requisitos de tiempo real, tanto este módulo como el Planificador de Trayectorias deben estar ubicados sobre el propio móvil controlado. De igual manera, debe existir una copia total o parcial en memoria caché de la BBDD de Trayectorias, de cara a que los accesos a la base de datos sean rápidos y eficientes.

Es en este módulo donde se ubican las funciones y algoritmos incluidos en los capítulos anteriores. Así, el algoritmo básico de replanificación, *replanificar-trayectoria* es el programa principal que se encarga de lanzar todo el proceso, llamando a su vez a las funciones encargadas de la deducción de las trayectorias externas (*observar-y-deducir*) y de la predicción de interacciones (*predecir*).

El siguiente esquema muestra la relación entre estas funciones principales del proceso de replanificación:

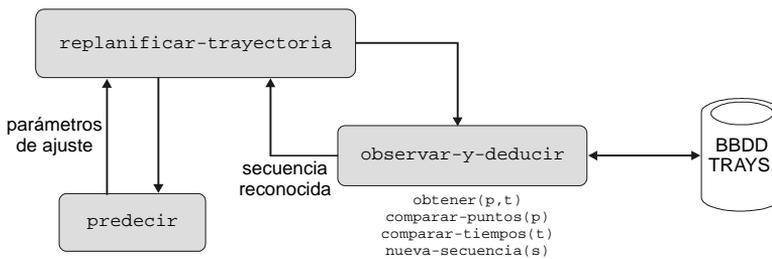


Figura 6-2. Relación entre las funciones y algoritmos principales de replanificación.

Por su parte, el proceso predictivo se basa en la función *predecir*, que a su vez realiza dos funciones básicas: 1) comparar, mediante la función *comparar-secuencias*, las trayectorias del móvil controlado y del móvil no controlado a partir de las secuencias que las representan y 2) calcular nuevos parámetros para la replanificación de la trayectoria que sigue el móvil controlado mediante la función *ajustar*.

La siguiente figura muestra un esquema con la relación entre estos procedimientos:

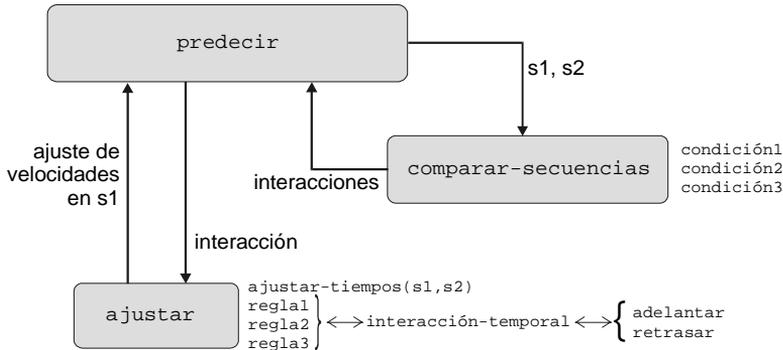


Figura 6-3. Relaciones y funciones del proceso predictivo.

Con todo, la siguiente tabla resume las características del módulo Deductivo+Predictivo:

Deductivo+Predictivo	
entradas	<ul style="list-style-type: none"> Valores relativos a puntos y tiempos de paso de los móviles (sensorización exteroceptiva).
salidas	<ul style="list-style-type: none"> Nuevas parámetros para el reajuste de la trayectoria que sigue el móvil controlado (ajuste de velocidades).
actividades	<ul style="list-style-type: none"> Deducción de las trayectorias de móviles externos. Ajuste temporal de las trayectorias. Razonamiento temporal en el futuro (predicción). Ajuste de velocidades.

Tabla 6-6. Características del módulo Deductivo+Predictivo.

LA ARQUITECTURA RETRARO

Una primera aproximación a la arquitectura propuesta, centrada en el proceso de planificación, se aplicó en el desarrollo del proyecto de investigación RETRARO. Como ya se ha detallado en capítulos anteriores, el principal objetivo de este proyecto de investigación se

centró en la mejora del proceso de producción de una empresa de tratamiento de la lana para introducir un sistema de transporte autoguiado convirtiendo para ello uno de los 4 vehículos filoguiados existentes en la empresa. Las características principales del proyecto (móviles, entorno, zonas, etc.) ya se han descrito anteriormente y se pueden encontrar en [Garbajosa *et al.*, 1997] [Mellado *et al.*, 1998] [Mellado *et al.*, 1999].

La característica principal en cuanto al funcionamiento del sistema RETRARO se centra en la existencia de distintos sectores de la empresa: un sector por donde se desplazan los 3 vehículos filoguiados (VFGs) existentes y otro sector de navegación libre por donde se mueve el nuevo vehículo autoguiado (VAG). Existen zonas compartidas entre ambos sectores, con la posible existencia de interacciones entre el VAG y los VFGs. Siempre se considera a los VFGs con mayor prioridad frente al VAG, debiendo esperarse este último a que la zona compartida esté libre. Para controlar esta característica se instalaron en el acceso a las zonas compartidas unas barreras ópticas, gestionadas por un módulo de control de tráfico.

Para poder cumplir con todas las características del nuevo sistema, se diseñó una arquitectura de control integrando todas las funcionalidades del sistema. La siguiente figura muestra la arquitectura hardware final del sistema RETRARO:

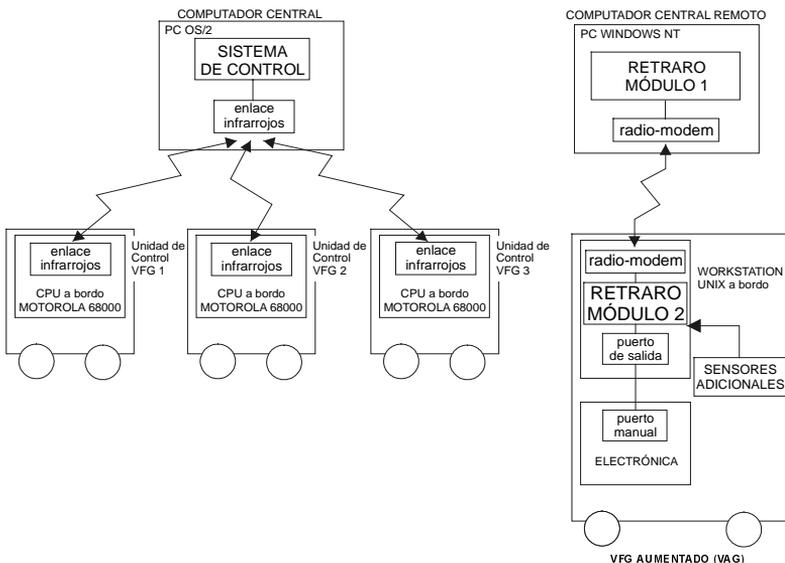


Figura 6-4. Arquitectura hardware RETRARO.

En esta figura se aprecia cómo los VFGs mantienen un ordenador central remoto basado en el sistema operativo OS/2 que se comunica mediante un enlace de infrarrojos con cada uno de los tres vehículos. En este ordenador central se ubica el sistema de control de los VFGs. Éstos tienen instalada a bordo una Unidad de Control propia basada en un microprocesador Motorola 68000.

Por su parte, la conversión del cuarto VFG en VAG supone disponer de un ordenador central remoto basado en el sistema operativo Windows NT. En este ordenador se dispone el RETRARO Módulo 1, que se comunica con el RETRARO Módulo 2, ubicado en la *workstation* UNIX de a bordo del nuevo VAG. La comunicación entre el PC remoto y el VAG se establece mediante un radio-modem. Las órdenes finales para el VAG se pasan hacia la electrónica del móvil a través de los consiguientes puertos de comunicación. Por último, el nuevo VAG dispone de un sistema de sensorización adicional que envía parámetros al RETARO Módulo 2.

La arquitectura software del sistema se muestra en la Figura 6-5.

Esta figura muestra los dos módulos básicos que componen la arquitectura software del sistema: RETRARO MÓDULO 1 (RM1) y RETRARO MÓDULO 2 (RM2). El primero de ellos se ubica en el ordenador central, mientras que el segundo se instala directamente sobre la *workstation* UNIX de a bordo. La arquitectura software refleja la incorporación del sistema existente sin modificaciones en una nueva arquitectura que proporciona también componentes relacionados con la navegación libre.

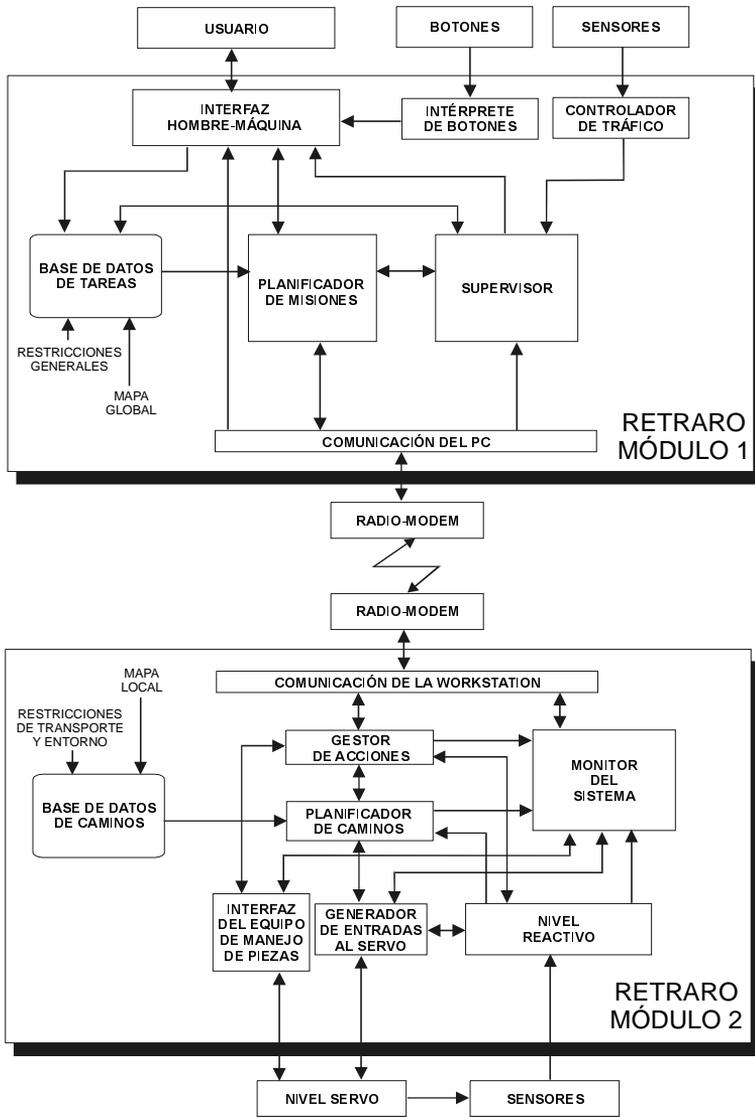


Figura 6-5. Arquitectura software RETRARO.

Los módulos relacionados con la metodología de planificación expuesta en esta tesis doctoral son los siguientes:

- **Planificador de misiones:** Recibe los requerimientos de las tareas a realizar por el VAG y decide qué parte de la tarea se realiza en las zonas de navegación libre y cuál en las zonas de cruce con los caminos de los VFGs (descomposición). Esta acción significa que se divide la tarea en lo que se denominan tareas de movimiento (*motion tasks*). Estas tareas de movimiento implican, bien movimientos sólo en las zonas de navegación libre, bien movimientos sólo en las zonas compartidas.
- **Supervisor:** Controla y permite el acceso a las zonas compartidas del VAG, según la información que proporciona el Controlador de Tráfico. De igual manera, este módulo controla el estado de las baterías del móvil y los datos que le proporciona el Monitor del Sistema, de cara a generar nuevas tareas para el Planificador de Misiones que supongan un cambio de baterías u otro tipo de servicios.
- **Gestor de Acciones:** Se encarga de generar tres posibles tipos de acciones de cara a cumplir satisfactoriamente una tarea de movimiento:
 - *Acciones de Movimiento* para desplazar el VAG, que son enviadas al Planificador de Caminos.
 - *Acciones Auxiliares* para la carga y descarga de piezas, que son enviadas a la Interfaz del Equipo de Manejo de Piezas.
 - *Acciones de Anclaje* para el correcto acople del VAG a las máquinas de los procesos de producción, que son enviadas al Nivel Reactivo.
- **Planificador de Caminos:** Se encarga de generar los comandos de movimiento básicos a partir de las acciones de movimiento y enviarlos al Generador de Entradas al Servo. Los comandos de movimiento se generan a partir de factores que se obtienen de la Base de Datos de Caminos, de cara a obtener el camino óptimo.

En RETRARO existe además un módulo dedicado al control de tráfico, encargado, como se ha dicho, de controlar el acceso del VAG a las zonas compartidas. Esto se consigue a partir de una sensorización del entorno mediante barreras ópticas, situadas en los accesos a las zonas compartidas. Mediante esta sensorización, es posible detectar si hay algún VFG en una zona compartida o en las inmediaciones,

retrasándose el acceso del VAG a la zona compartida para evitar una posible colisión.

Estos módulos principales se encargan de realizar la planificación de las actividades encomendadas al VAG del sistema RETRARO y guardan un paralelismo claro con la arquitectura de planificación propuesta en la metodología objeto de este trabajo. A continuación se detalla esta relación:

- El Planificador de Misiones y el Gestor de Acciones de RETRARO se corresponden con el Planificador de Misiones y el Planificador de Tareas introducidos en esta metodología. Sus objetivos respectivos son similares, aunque con la particularización de la aplicación a un caso concreto, como supone el sistema RETRARO.
- El Planificador de Caminos de RETRARO se corresponde con el Planificador de Movimientos y el Planificador de Trayectorias de la metodología. Sus funciones son las mismas. En el caso RETRARO, las particularidades del sistema de producción no requerían hacer distinción entre movimientos y trayectorias, aunándose ambos conceptos bajo el nombre de caminos. La BBDD de caminos de RETRARO, por tanto, se corresponde con las BBDD de movimientos y de trayectorias detalladas en la metodología global.
- El Monitor del Sistema de RETRARO y la parte del Supervisor encargada del control de los parámetros internos del móvil se corresponden con el Monitor del Sistema de la metodología.

A partir de esta relación se establece un paralelismo claro entre la arquitectura de planificación de RETRARO y la expuesta en la metodología. Pero este paralelismo no sólo se establece a partir de las relaciones entre los módulos, sino además por la filosofía de trabajo utilizada. RETRARO considera zonas distintas de navegación y descompone las actividades del VAG según se realicen en estas zonas, lo que se corresponde con el proceso de descomposición expuesto en la metodología y ya detallado en el capítulo correspondiente, precisamente con el ejemplo RETRARO. Por otro lado, la consideración de misiones, tareas de movimiento, acciones y caminos de RETRARO es similar a la abstracción jerárquica especificada en la metodología de planificación de la tesis doctoral.

CONCLUSIONES

En este capítulo se ha presentado la descripción de una arquitectura de control para un robot móvil que refleja la metodología de planificación y replanificación descrita a lo largo de la tesis doctoral. Se trata de una arquitectura software genérica cuyos módulos contienen las funcionalidades establecidas en la metodología, a partir de las características principales de la misma. Así, la planificación de las actividades del móvil se aborda mediante la descomposición de las actividades y una jerarquía de niveles. Estos procesos principales se reflejan directamente en la estructura modular en vertical de la arquitectura. Bajo este prisma, la arquitectura que se propone se puede clasificar como una arquitectura jerárquica.

Por otro lado, la replanificación de las trayectorias del móvil en tiempo real se aborda mediante un módulo que implementa los procesos deductivo y predictivo. Este módulo, que rompe de alguna manera la estructura vertical de la arquitectura, funciona a partir de las entradas del sistema de sensorización del entorno, generando nuevos parámetros de ajuste de las trayectorias que se siguen. Si se atiende a esta característica, la replanificación se plantea a partir de una “sub-arquitectura” reactiva.

Otra aportación realizada se refleja en la arquitectura del sistema RETRARO, como una primera aproximación a la metodología de planificación descrita. En este capítulo se describe la arquitectura implementada dentro de este proyecto de investigación, estableciéndose el paralelismo y la relación con la arquitectura genérica introducida. De esta manera se puede ver como los principios de descomposición en partes de las tareas del móvil (en el caso de RETRARO, a partir de las zonas del sistema) y la abstracción jerárquica se reflejan y aplican en la arquitectura software.



Arquitectura software

7 Conclusiones

Conclusiones

La planificación de robots es un problema ampliamente abordado desde distintos prismas. Desde el punto de vista de la Inteligencia Artificial, el control de sistemas o las matemáticas, son muchas y diversas las soluciones aportadas al problema de conseguir que un robot cumpla unos objetivos determinados al realizar una actividad.

La mayoría de las soluciones planteadas a la hora de resolver una actividad de un robot están centradas en aspectos muy particulares del problema robotizado, o proponen soluciones a un determinado nivel de concreción. Así, por ejemplo, existen diversas técnicas aplicadas en el campo de la planificación de caminos o arquitecturas de control definidas para sistemas robotizados concretos. Por otro lado, los planificadores generales resultan en ocasiones soluciones demasiado amplias y de difícil conexión con un sistema robotizado específico.

CONTRIBUCIONES

El presente trabajo introduce una metodología completa de planificación de las actividades de un robot basada en intuiciones humanas. Esto es, las acciones de planificación están guiadas por conceptos familiares al ser humano que se aplican de manera cotidiana en la resolución de un problema. Así, la *descomposición en partes* y la *abstracción en niveles* del problema se proponen como conceptos a combinar para definir una arquitectura genérica de planificación de robots. De igual manera, la *deducción* y la *predicción de futuras acciones* son los principios en los que se basa la replanificación *on line* de las trayectorias que sigue un robot en la ejecución de sus actividades.

En ese sentido, las aportaciones principales de esta tesis doctoral se centran en los siguientes puntos:

- *Análisis de distintos métodos de planificación de robots, contrastándolos con el planteamiento de la tesis a través de la validación de la metodología en experiencias previas (proyecto de investigación RETRARO).*
- *Formalización de una metodología de planificación de actividades de robot basada en la combinación de la descomposición y la abstracción del problema.*
- *Formalización y aplicación de la replanificación de trayectorias de robot a partir de la deducción y predicción de acciones futuras mediante el ajuste de velocidades.*

PLANIFICACIÓN

La tesis doctoral aporta un modelo de planificación de robots que combina la descomposición del problema en partes, y la jerarquización del mismo en distintos niveles. Por lo que respecta a la descomposición del problema, se propone la utilización de unos criterios (*operatividad de los agentes, criterios espaciales, criterios temporales y alcance espacio-temporal de las acciones*) que permiten establecer una correcta descomposición en partes del problema. Dichos criterios son de consideración particular al problema a resolver, abordándose su análisis de forma diferente según las características del mismo. De igual manera, se introducen unos factores (*coste temporal, coste energético, número de operaciones y consumo de recursos*) que permiten a su vez obtener un índice de bondad (B_i) sobre la descomposición en partes elegida.

La abstracción del problema en niveles es el otro principio sobre el que se asienta el método de planificación. El presente trabajo propugna la generación de distintos niveles de abstracción, planificando cada nivel de manera separada. Así, se proponen unos niveles de abstracción (*misiones, tareas, movimientos, acciones, trayectorias y órdenes de robot*) de aplicación general en problemas robotizados. Estos niveles se establecen a partir de la dependencia o independencia con respecto del robot que ejecuta la actividad planteada y del entorno en el que se debe ejecutar. Así, el nivel más abstracto (*misión*) describe una actividad con independencia del robot que la va a ejecutar y del entorno dónde se ejecutará, mientras que el nivel más concreto (*orden de robot*) describe una acción u orden de robot a partir de las propias características y funcionalidades del robot y del entorno local en el que se ejecuta esa acción.

La combinación de descomposición y abstracción definen la metodología de planificación final. Mientras que la descomposición implica una estructura vertical del problema, la abstracción en niveles induce a pensar en el problema “horizontalmente”. Ambos conceptos se plasman en una arquitectura software de planificación que se presenta en la parte final de este documento. Esta arquitectura refleja de manera especial la jerarquía en niveles establecida para las actividades robotizadas con la existencia de distintos módulos de planificación que se corresponden con los niveles de abstracción propuestos.

En el trabajo presentado se incluyen ejemplos básicos de la aplicación del método de planificación. De igual manera, se describen las

aportaciones realizadas en el contexto del proyecto de investigación RETRARO, en el cual se aplicó el proceso de planificación a un vehículo autoguiado basado en los principios de descomposición y abstracción. El trabajo realizado por el autor en este proyecto de investigación constituyó la génesis de la metodología completa que se presenta ahora, publicándose las experiencias en diversos artículos de investigación y presentándose los resultados de la planificación sobre una maqueta desarrollada para el uso del robot móvil miniatura Khepera (6th Interfaces'97).

REPLANIFICACIÓN

El otro gran punto que conforma la tesis doctoral es la replanificación de trayectorias de robot. El método plantea la replanificación *on line* de las actividades de robot al nivel de las trayectorias que sigue. Los conceptos sobre los que se basa el método son la deducción de actividades externas y la predicción de acciones futuras. Este planteamiento se refleja de manera especial en robots móviles, debido a su característica de desplazamiento en el entorno, aunque es aplicable a robots articulados, a partir del desplazamiento del efector final por el espacio.

Para establecer las acciones de reajuste de una trayectoria se plantea un razonamiento temporal sobre todas las trayectorias en ejecución en el mismo entorno. Esta actividad implica el conocimiento de las trayectorias de los móviles ajenos al sistema de control y ello se consigue a partir de un proceso de deducción basado en la observación del entorno. El uso de un sistema de sensorización adecuado permite conocer el paso de los móviles por puntos característicos y los tiempos asociados. Así, es posible reconocer la trayectoria que sigue un móvil a partir de la secuencia (puntos y tiempos de paso) que la representa. En el desarrollo de este trabajo se formalizan los conceptos de trayectoria y secuencia, incluyéndose intervalos temporales que reflejan la incertidumbre en el paso de un móvil por un punto del espacio.

El reajuste final de la trayectoria que sigue un móvil controlado se concreta, como ya se ha dicho, a partir de un proceso de razonamiento sobre el futuro de las secuencias que representan a las distintas trayectorias en ejecución. El razonamiento temporal permite establecer posibles interacciones (*puntos compartidos, segmentos en mismo sentido o segmentos en sentido contrario*) entre secuencias a nivel de espacio y tiempo. Una interacción indica la posibilidad de colisión de dos móviles en la ejecución de sus actividades.

La replanificación final de la trayectoria que sigue el móvil controlado se realiza a través de un ajuste de velocidades de la misma. Este ajuste se aplica mediante unas reglas de comportamiento definidas según el tipo de interacción detectado. De esta manera, el móvil controlado ajusta su velocidad para evitar la interacción con el otro móvil del entorno. Estas acciones se deciden en el futuro, aplicándose el adelanto proporcionalmente en el tiempo hasta sobrepasar el punto de interacción, o retrasándose hasta detectar que el otro móvil ya ha pasado por la zona compartida.

En el trabajo desarrollado se incluyen los algoritmos básicos del método de replanificación (*replanificar-trayectoria*), del proceso deductivo (*observar-y-deducir*) y del proceso predictivo (*predecir*). En la arquitectura software que se incluye en el último capítulo de la tesis doctoral aparece reflejada la replanificación de trayectorias con la inclusión del módulo correspondiente.

Por último, el proceso de replanificación se plantea para el caso de tener múltiples móviles controlados y no controlados en un mismo entorno. Los principios de deducción y predicción se aplican también en este caso, proponiéndose una resolución a partir de las prioridades asignadas a cada trayectoria en ejecución. Así, móviles con actividades que tengan asignada una prioridad menor deben replanificar sus trayectorias para adaptarse a las que siguen otros móviles con mayor prioridad. Esta replanificación se aplica a partir de la ordenación de las secuencias que representan a las distintas trayectorias implicadas en una interacción según su prioridad (estática, dinámica). Al igual que en el caso básico de dos móviles (uno controlado y otro no controlado), se proponen los algoritmos básicos del método de replanificación con las modificaciones y características propuestas para múltiples móviles.

Este punto final incluye también la revisión de unas alternativas y consideraciones sobre el método de replanificación. Así, se revisan y proponen situaciones de mejora de la replanificación (replanificación solidaria) y se comentan posibles problemas (interacciones solapadas, techos de prioridad).

TRABAJOS FUTUROS

En toda labor de investigación aparecen nuevos puntos y bifurcaciones cuya consideración queda fuera de la frontera temporal del propio desarrollo del trabajo o suponen alternativas no supuestas en principio dentro del ámbito del trabajo previsto. Estas alternativas

acaban configurando nuevos puntos de investigación que se desprenden del estudio acometido. La presente tesis doctoral no supone una excepción a esta regla. Así, a lo largo de su elaboración han aparecido hitos que permiten establecer temas de investigaciones futuras.

Los trabajos futuros que se desprenden de esta tesis doctoral se centran en la *automatización del proceso de planificación*, el *reconocimiento y aprendizaje de trayectorias* y en la *mejora de la replanificación de trayectorias*.

En el método de planificación expuesto se introducen parámetros y criterios que permiten abordar tanto la descomposición en partes como la abstracción en niveles del problema. Aunque la metodología de planificación permite particularizar el estudio de cada actividad de robot y su posterior aplicación, en ocasiones puede resultar complicado el establecimiento de las partes del problema o la elección de una técnica concreta de planificación. La inclusión de sistemas expertos en un primer estadio de aplicación del método, clasificando las actividades de robot, permitiría establecer las partes de un problema de manera correcta y la elección de una técnica concreta de resolución en cada nivel de abstracción.

En el presente trabajo de investigación se ha propuesto una técnica simple de reconocimiento de trayectorias de los móviles del entorno. La aplicación de algoritmos para el reconocimiento de trayectorias (tanto ya conocidas como nuevas) basados en principios como el reconocimiento de patrones puede ser un buen camino a seguir en la mejora del proceso deductivo.

Las alternativas y consideraciones estudiadas en el apartado dedicado a la replanificación de trayectorias centran los trabajos futuros en cuanto a la mejora de la misma. Un proceso de replanificación de trayectorias de múltiples móviles *solidario*, ya introducido en ese punto, es el camino de mejora deseable frente al método propuesto. Este nuevo frente de mejora de la replanificación debería considerar el proceso global de producción para establecer criterios en el reajuste de velocidades del total de móviles controlados involucrados en una interacción. El objetivo final sería la replanificación de las trayectorias optimizando los tiempos de producción totales. Esta optimización resulta complicada de aplicar en situaciones de solape de interacciones. La inclusión de algoritmos de clasificación que consideren los tipos de interacción y las prioridades de ejecución de las trayectorias involucradas en interacciones solapadas es un camino abierto en la mejora del proceso de replanificación.

La estrecha relación entre la planificación de un robot y la programación final del mismo sugiere el establecimiento de una línea de trabajo en este sentido. El autor de esta tesis doctoral ha estado trabajando en el campo de la programación de robots en los últimos años y participando en distintos proyectos de investigación relacionados, de entre los que cabe resaltar los siguientes:

- Planificación de Movimientos para Vehículos Móviles Autónomos en Entornos Industriales Basados en Percepción Sensorial (Proyecto CICYT N° TAP95-1086-Co2-01. 1995-1998)
- Control Remoto para un Sistema Multi-Robot con Sensorización Específica mediante el Diseño de una Arquitectura de Red de Altas Prestaciones usando Tecnologías Multimedia (Proyecto de la Consellería de Cultura, Educación y Ciencia, Ref. GV97-TI-04-37. 1998-1999)
- Inverse Kinematics and Robot Language Programming for Generis (Proyecto con la Comisión Europea, Contract Number 13787-1998-03 F1EP ISP ES. 1998-1999)
- Troquelado de Piezas mediante la Generación Automática de Programas de Robots (Proyecto con la empresa Sommer Allibert Industria-Lignotock S.A., Enero-Mayo 2000).
- Realidad Virtual aplicada a la Tele-Robótica. (Proyecto coordinado del Plan Incentivo a la Investigación de la UPV, 2000-2001).

Del trabajo de investigación en este campo, se han obtenido resultados que se plasman en diversos artículos publicados ([Vendrell *et al.*, 1991] [Vendrell *et al.*, 1992a] [Vendrell *et al.*, 1992b] [Mellado *et al.*, 1992] [Vendrell *et al.*, 1993] [Mellado y Vendrell, 1994] [Mellado y Vendrell, 1995] [Navarro *et al.*, 1996] [Sánchez *et al.*, 1996] [Navarro *et al.*, 1997] [Vendrell *et al.*, 1999] [Sánchez *et al.*, 2000]).

La línea de investigación propuesta en planificación-programación de robots tiene como objetivo principal aunar los resultados del proceso de planificación con la codificación de los mismos en forma de secuencias de acciones eficientes para la resolución de los problemas dados, configurándose con ello un proceso de programación automática de robots.

✱

Bibliografía y referencias

Bibliografía y referencias

[Aho *et al.*, 1983]

Data Structures and Algorithms

A.V. Aho, J.E. Hopcroft y J.D. Ullman
Addison-Wesley. 1983.

[Allen, 1983]

Maintaining Knowledge about Temporal Intervals

J.F. Allen

Communications of the ACM, Vol. 26, N^o 11,º pp. 832-843.
Noviembre, 1983.

[Allen, 1984]

Towards a General Theory of Action and Time

J.F. Allen

Artificial Intelligence, 23, pp. 123-154. 1984.

[Allen y Hayes, 1989]

Moments and Points in an Interval-Based Temporal Logic

J.F. Allen y P.J. Hayes

Computational Intelligence, 5 (4), pp. 225-238. 1989.

[Allen *et al.*, 1990]

Readings in Planning

J.F. Allen, J. Hendler y A. Tate

Morgan Kaufmann Publishers. 1990

[Allen y Ferguson, 1994]

Actions and Events in Temporal Logic

J.F. Allen y G. Ferguson

Technical Report, Computer Science Department, The University of Rochester. Julio, 1994.

[Ambite y Knoblock, 1997]

Planning by Rewriting: Efficiently Generating High-Quality Plans

J.L. Ambite y C.A. Knoblock

Proceedings of the 14th National Conference on Artificial Intelligence.
Rhode Island, 1997.

<http://www.isi.edu/~ambite/97-aaai-pbr.ps>

[Ambite, 1999]

Planning by Rewriting

J.L. Ambite

Tesis Doctoral. University of Southern California. 1999.

<http://www.isi.edu/~ambite/thesis.pdf>

[Avnaim *et al.*, 1988]

A Practical Exact Motion Planning Algorithm for Polygonal Objects amidst Polygonal Obstacles

F. Avnaim, J.-D. Boissonnat y B. Faverjon
Proceedings of the IEEE Symposium on Robotics and Automation,
pp 1656-1661. 1988.

[Bailey *et al.*, 1997]

Structural Issues in Active Rule Systems

J. Bailey, G. Dong y K. Ramamohanarao
Proceedings of the Sixth International Conference on Database
Theory, LNCS 1186, pp. 203-214, Delphi, Greece. 1997.

[Bandura, 1977]

Social Learning Theory

A. Bandura
Englewood Cliffs, N.J.: Prentice-Hall. 1977.

[Barber, 1993]

A Metric Time-Point and Duration-Based Temporal Model

F.A. Barber
SIGART-ACM Bulletin, Vol. 4, n^o 3, pp. 30-49. 1993.

[Barraquand *et al.*, 1990]

Robot Motion Planning with many Degrees of Freedom and Dynamic Constraints

J. Barraquand, B. Langlois y J.C. Latombe
Robotics Research, 5, pp. 435-444. 1990.

[Barraquand y Ferbach, 1993]

Path Planning through Variational Dynamic Programming

J. Barraquand y P. Ferbach
Research report 33, Paris Research Laboratory, Paris. Septiembre,
1993.

[Barraquand y Ferbach, 1994]

A Penalty Function Method for Constrained Motion Planning

J. Barraquand y P. Ferbach
Proceedings of the 1994 IEEE International Conference on Robotics
and Automation, pp. 1235-1242, San Diego, California. 1994.

[Beccari *et al.*, 1998]

A Real-Time Library for the Design of Hybrid Robot Control Architectures

G. Beccari, S. Caselli, M. Reggiani, F. Zanichelli
IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'98,
Victoria, BC, Canada. 1998.

[Berrety *et al.*, 1997]

Dynamic Motion Planning in Low Obstacle Density Environments

R.P Berrety, M. Overmars y A.F. van der Stappen
Proceedings of the Workshop on Algorithms and Data Structures
(WADS'97). 1997.

[Blum y Furst, 1997]

Fast Planning Through Planning Graph Analysis

A.L. Blum y M.L. Furst
Artificial Intelligence, 90, pp 281-300. 1997.

[Blume *et al.*, 1994]

Robot Trajectory Planning with Collision Avoidance using Genetic Algorithms and Simulation

C. Blume, S. Krisch y W. Jacob
Proceedings of the 25th International Symposium on Industrial
robots, pp. 169-175, Hannover. 1994

[Borenstein, 1996]

Where Am I? Sensors and Methods for Autonomous Mobile Robot Positioning

J. Borenstein, H.R. Everett y L. Feng
Technical Report, Department of Mechanical Engineering and
Applied Mechanics, The University of Michigan. 1996
<http://www-personal.engin.umich.edu/~johannb/position.htm>

[Borrajo *et al.*, 1997]

Inteligencia Artificial. Métodos y Técnicas

D. Borrajo, N. Juristo, V. Martínez y J. Pazos
Ed. Centro de Estudios Ramón Areces, Madrid. 1997.

[Bratman, 1987]

Intention, Plans, and Practical Reason

M.E. Bratman
Harvard University Press, Cambridge. 1987.

[Brooks y Lozano-Pérez, 1985]

A Subdivision Algorithm in Configuration Space for Findpath with Rotation

R. Brooks y T. Lozano-Pérez

IEEE Transactions on Systems, Man, and Cybernetics, 15 (2), pp. 224-233. 1985.

[Brooks, 1986]

A Robust Layered Control System for a Mobile Robot

R.A. Brooks

IEEE Journal of Robotics and Automation, Vol. 2, n° 1, pp. 14-23. 1986.

[Brummit y Stentz, 1996]

Dynamic Mission Planning for Multiple Mobile Robots

B.L. Brumitt y A. Stentz

Proceedings of the IEEE International Conference on Robotics and Automation, N° 3, pp. 2396-2401. 1996.

[Brunner *et al.*, 1993]

Multisensory Shared Autonomy and Tele-Sensor-Programming Key Issues in the Space Robot Experiment ROTEX

B. Brunner, G. Hirzinger, K. Landzettel y J. Heindl

Proceedings of the IEEE/RSJ International Conference on Intelligent Robotics and Systems, pp. 2123-2139, Yokohama, Japón. 1993.

[Brunner *et al.*, 1995]

Tele Sensor Programming – A Task-Directed Programming Approach for Sensor-Based Space Robots

B. Brunner, K. Landzettel, B.M. Steinmetz y G. Hirzinger

Proceedings of the 7th International Conference on Advanced Robotics, Sant Feliu de Guixols, España. 1995.

[Bunke y Sanfeliu, 1990]

Syntactic and Structural Pattern Recognition: Theory and Applications

H. Bunke y A. Sanfeliu, eds.

World Scientific, 1990.

[Carbonell *et al.*, 1991]

PRODIGY: An Integrated Architecture for Planning and Learning

J. Carbonell, O. Etzioni, Y. Gil, R. Joseph, C. Knoblock, S. Minton y M. Veloso

SIGART Bulletin 2, pp. 51-55. 1991.

<http://www.cs.cmu.edu:8001/afs/cs.cmu.edu/project/prodigy/Web/prodigy-home.html>

[Canny y Reif, 1987]

New Lower Bound Techniques for Robot Motion Planning Problems

J. Canny y J. Reif

Proceedings of the 28th IEEE Symposium on Foundations of Computer Science, pp. 49-60. Los Angeles, 1987.

[Chang y Lee, 1973]

Symbolic Logic and Mechanical Theorem Proving

C.L. Chang y R.C.T. Lee

Academic Press, New York. 1973.

[Chazelle, 1987]

Aproximation and Decomposition of Shapes

B. Chazelle

Advances in Robotics 1: Algorithmic and Geometric Aspects of Robotics (J.T. Schwartz y C.K. Yap, editores), pp. 145-185, Lawrence Erlbaum Associates, Hillsdale, NJ. 1987.

[Chen y Brudick, 1993]

Finding Antipodal Point Grasps on Irregularly Shaped Objects

I.M. Chen y J. W. Burdick

IEEE Journal of Robotics and Automation, Vol.9, No.4, pp507-512. 1993.

[Cohen y Levesque, 1990]

Intention is Choice with Commitment

P.R. Cohen y H.J. Levesque

Artificial Intelligence, Vol. 42, pp. 213-261. 1990.

[Costa *et al.*, 1995]

Planificación y Control en Entornos Multi-Robot

R. Costa, L. Basáñez y R. Suárez

Actas del 4º Congreso de la Asociación Española de Robótica (AER'95), pp. 123-131, Zaragoza. 1995

[Crespo *et al.*, 1994]

A Temporal Blackboard for Real-time Process Control

Crespo, V. Botti, F. Barber, D. Gallardo y E. Onaindia
Journal of Engineering Applications of Artificial Intelligence, Vol. 7.
Nº 3. pp 255-266. 1994.

[Dean, 1989]

Using Temporal Hierarchies to Efficiently Maintain Large Temporal Databases

T. Dean
Journal of the Association for Computing Machinery, Vol. 36, Nº 4,
pp. 687-718. Octubre, 1989.

[de Lope, 1998]

Modelado de Entornos con Técnicas basadas en Redes de Petri Borrosas para la Exploración y Planificación de Robots Autónomos

J. de Lope
Tesis doctoral, Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid. 1998.
<http://www.sia.eui.upm.es/~jdelope/tesis.resumen.shtml>

[Doyle y Jones, 1994]

A Tangent Based Method for Robot Path Planning

A. Doyle y D. Jones
Proceedings of the 1994 IEEE International Conference on Robotics and Automation, Vol. 3, pp. 2230-2237, San Diego, California. 1994

[Englemore y Morgan, 1988]

Blackboard Systems

R. Englemore y T. Morgan (editores)
Addison-Wesley. Reading, Massachussets. 1988.

[Fikes y Nilsson, 1971]

STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving

R.E. Fikes y N.J. Nilsson
Artificial Intelligence, Vol. 2, pp. 189-208. 1971.

[Fiorini y Schiller, 1995]

Robot Motion Planning in Dynamic Environments

P. Fiorini y Z. Schiller
International Symposium of Robotic Research (G. Giraldo y G. Hirzinger, editores), pp. 237-248, Springer-Verlag, Munich, Alemania. 1995.

[Fisher, 1994]

Efficient Path Planning Strategies for Cooperating Manipulators in Environments with Obstacles

M. Fisher

Proceedings of the 1994 IEEE International Conference on Robotics and Automation, pp. 2418-2423, San Diego, California. 1994

[Floyd, 1962]

Algorithm 97: Shortest Path

R.W. Floyd

Communications of ACM, 5:6, p. 345. 1962.

[Fraichard y Laugier, 1993]

Dynamic Trajectory Planning, Path-Velocity Decomposition and Adjacent Paths

T. Fraichard y C. Laugier

Proceedings of the 1993 Int. Joint Conference on Artificial Intelligence, Vol. 2, pp. 1592-1597, Chambery, Francia. 1993.

[Freksa, 1991]

Conceptual Neighborhood and its Role in Temporal and Spatial Reasoning

C. Freksa

Proc. of the IMACS Workshop on Decision Support Systems and Qualitative Reasoning. Eds.: Singh, M.; Trav-Massuys, L., Elsevier Science Publishers, Amsterdam. 1991.

[Freund y Hoyer, 1988]

Real-Time Pathfinding in Multirobot Systems Including Obstacle Avoidance

E. Freund y H. Hoyer

The International Journal of Robotics Research, 7 (1), pp. 42-70. 1988.

[Fu, 1977]

Syntactic Pattern Recognition Applications

K.S. Fu

Springer-Verlag, 1977.

[Fu *et al.*, 1988]

ROBÓTICA: Control, Detección, Visión e Inteligencia

K.S. Fu, R.C. González y C.S.G. Lee

McGraw-Hill, Madrid. 1988.

[Gachet *et al.*, 1992]

A Software Architecture for Behavioural Control Strategies of Autonomous Systems

D. Gachet, M.A. Salichs, L. Moreno, J.R. Pimentel y A. de la Escalera
Proc. IECON'92. San Diego, CA. 1992.

[Garbajosa *et al.*, 1997]

An Intelligent Control System for an Autonomous Guided Vehicle to Support Material Transport

J. Garbajosa, C. Lomba, M. Mellado, E. Vendrell, A. Crespo, K. Schilling y R. Mayerhofer
6th Interfaces'97. Man machine Interaction. Intelligent Systems in Bussines. Montpellier, Francia. 1997.

[Gerevini y Serina, 1999]

Fast Planning through Greedy Action Graphs

A. Gerevini e I. Serina

Technical Report. Computer Science Department, University of Rochester. Febrero, 1999.

ftp://ftp.cs.rochester.edu/pub/papers/ai/99.tr710.Fast_planning_through_greedy_action_graphs.ps.gz

[Ghezzi *et al.*, 1991]

A Unified High-Level Petri Net Formalism for Time-Critical Systems

C. Ghezzi, D. Mandrioli, S. Morasca y M. Pezzè

IEEE Transactions On Software Engineering, Vol. 17 N^o. 2. Febrero, 1991.

[Goldman y Lang, 1993]

Intentions in Time

R.P. Goldman y R.R. Lang

Technical Report TUTR 93-101, Computer Scienctie Dep. and Center for Automation and Autonomous Complex Systems, Tulane University, New Orleans. Enero, 1993

[Graves, 1995]

A Generalized Teleautonomous Architecture Using Situation-Based Action Selection

B. S. Graves

Tesis Doctoral. Texas A&M University, College Station. 1995.

http://www.cs.tamu.edu/research/robotics/Sean/Papers/sean_diss.ps.gz

[Green, 1969]

An Application of Theorem Proving to Problem Solving

C. Green

Proceedings of the First International Joint Conference on Artificial Intelligence (IJCIA-69), pp 741-747, Washington, DC. Mayo, 1969.

[Gupta, 1987]

Parallelism in Production Systems

A. Gupta

Pitman/Morgan Kaufmann Publishers, Inc., Los Altos, CA. 1987.

[Hague y Tillet, 1996]

Navigation and Control of an Autonomous Horticultural Robot

T. Hague y N.D. Tillet

Mechatronics, 6 (2), pp. 165-180. 1996.

[Harary, 1969]

Graph Theory

F. Harary

Addison-Wesley, 1969.

[Heiserman, 1981]

Robot Intelligence with Experiments

D.L. Heiserman

Blue Ridge Summit TAB Books. 1981.

[Hendler *et al.*, 1990]

AI Planning: Systems and Techniques

J. Hendler, A. Tate y M. Drummond

AI Magazine, vol. 11, n^o. 2, pp. 61-77. 1990.

[Homem de Mello y Sanderson, 1986]

AND/OR Graph Representation of Assembly Plans

L. S. Homem de Mello y A. C. Sanderson

Robotics Institute Technical Report, Carnegie-Mellon University, CMU-RI-TR-86-8. Proceedings of the 5th National Conference on Artificial Intelligence (AAAI-86), pp 1113-1119, 1986.

[Homem de Mello y Sanderson, 1990]

AND/OR Graph Representation of All Assembly Sequences

L. S. Homem de Mello y A. C. Sanderson

IEEE Transactions of Robotics and Automation, Vol. 6 (2) pp. 188-199. Abril, 1990.

[Hsieh y Archibald, 1996]

Flexibility in a Blackboard System for Solving Dynamic Resource-Constrained Scheduling Problems

Ch.L. Hsieh y J. Archibald

Occasional Paper N° CIS-15-96. School of Computing & Information Systems. University of Sunderland. Agosto, 1996.

http://osiris.sund.ac.uk/~esomhs/publishp/cis15_96.ps

[Hwang y Ahuja, 1992]

Gross Motion Planning -- A Survey

Y.K. Hwang y N. Ahuja

ACM Computing Surveys, 24. pp. 220-291. 1992.

[Ishida, 1994]

An Optimization Algorithm for Production Systems

T. Ishida

IEEE Transactions on Knowledge and Data Engineering, 6 (4), pp. 549-558. 1994.

[Jensen, 1992]

Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. Volume 1 : Basic Concepts

K. Jensen

EATCS Monographs on Theoretical Computer Science, Springer-Verlag. 1992.

[Jensen y Rozenberg, 1991]

High-level Petri Nets: Theory and Application

K. Jensen y G. Rozenberg (editores)

Springer-Verlag, New York. 1991.

[Kant y Zucker, 1986]

Toward Efficiency Trajectory Planning: The Path-Velocity Decomposition

K. Kant y S.W. Zucker

The International Journal of Robotics Research, 5 (3), pp. 72-89. 1986.

[Kavraki y Latombe, 1994]

Randomized Preprocessing of Configuration Space for Fast Path Planning

L. Kavraki y J.C. Latombe

Proceedings of the 1994 IEEE International Conference on Robotics and Automation, pp. 2138-2145, San Diego, California. 1994.

[Khatib, 1986]

Real-Time Obstacle Avoidance for Manipulators and Mobile Robots

O. Khatib

International Journal of Robotics Research, 5 (1), pp. 90-98. 1986.

[Khatib, 1994]

Sensor-Based Robotic Manipulation

O. Khatib

Proceedings of the 2nd Japan-France Congress on Mechatronics, Takamatsu, Japón, pp. 193-198. 1994.

[Khosla y Volpe, 1988]

Superquadratic Artificial Potentials for Obstacle Avoidance and Approach

P. Khosla y R. Volpe

Proceedings of the 1988 IEEE International Conference on Robotics and Automation, pp. 1778-1784, Philadelphia. 1988

[Knoblock, 1994]

Automatically Generating Abstractions for Planning

C. A. Knoblock

Artificial Intelligence, 68, pp. 243-302. 1994.

[Koga y Latombe, 1994]

On Multi-Arm Manipulation Planning

Y. Koga y J.C. Latombe

Proceedings of the 1994 IEEE International Conference on Robotics and Automation, pp. 945-952, San Diego, California. 1994.

[Korf, 1985]

Planning as Search: A Quantitative Approach

R. Korf

Artificial Intelligence, 33, pp. 65-88. 1985.

[Kuhn, 1972]

Mechanism of Change in the Development of Cognitive Structures

D. Kuhn

Child Development, 43, pp. 833-844. 1972.

[Kuo y Moldovan, 1992]

The State of the Art in Parallel Production Systems

S. Kuo y D. Moldovan

Journal of Parallel and Distributed Computing, 15, pp.1-26. 1992.

[Kuokka, 1991]

MAX: A meta-reasoning architecture for 'X'

D.R. Kuokka

SIGART Bulletin 2, pp. 93-97. 1991.

<http://ai.eecs.umich.edu/cogarcho/max/index.html>

[Laffey *et al.*, 1988]

Real-Time Knowledge-Based Systems

T.J. Laffey, P.A. Cox, J.L. Schmidt, S.M. Kao y J.Y. Read

AI Magazine, vol. 9, n^o 1, pp. 27-45. 1988.

[Langely *et al.*, 1991]

A design for the ICARUS Architecture

P. Langely, K.B. McKusick, J.A. Allen, W.F. Iba y K. Thompson

SIGART Bulletin 2, pp. 104-109. 1991

<http://ai.eecs.umich.edu/cogarcho/icarus/index.html>

[Latombe, 1991]

Robot Motion Planning

J.C. Latombe

Kluwer Academic Publishers. 1991.

[Lin y Tsai, 1991]

Motion Planning for Multiple Robots with Multi-mode Operations via Disjunctive Graphs

C.F. Lin y W.H. Tsai

Robotica, 9, pp. 393-408. 1991.

[López, 2000]

Supervisión, Detección, Diagnóstico y Recuperación de Fallos en Robots Móviles Autónomos

J. López

Tesis Doctoral. Departamento de Ingeniería de Sistemas y Automática. Universidad de Vigo. 2000.

[Lozano-Pérez, 1983]

Spatial Planning: A Configuration Space Approach

T. Lozano-Pérez

IEEE Transactions on Computers, 32 (2), pp. 108-120. 1983.

[Lozano-Pérez, 1987]

A Simple Motion-Planning Algorithm for General Robot Manipulators

T. Lozano-Pérez

IEEE Journal of Robotics and Automation, 3 (3), pp. 224-238. 1987.

[Lozano-Pérez *et al.*, 1984]

Automatic Synthesis of Fine-Motion Strategies

T. Lozano-Pérez, M.T. Mason y R.H. Taylor

The International Journal of Robotics Research, Vol. 3, n° 1, pp. 3-24. 1984.

[Marek *et al.*, 1990]

A Theory of Nonmonotonic Rule Systems

W. Marek, A. Nerode y J. Remmel

Proceedings of 5th IEEE Symposium on Logic in Computer Science, pp. 79-94. 1990.

[Marek *et al.*, 1992]

A Theory of Nonmonotonic Rule Systems II

W. Marek, A. Nerode y J. Remmel

Annals of Mathematics and Artificial Intelligence 5, pp. 229-263. 1992.

[Mason y Salisbury, 1985]

Robot Hands and the Mechanics of Manipulation

M. Mason y J.K. Salisbury

MIT Press, Cambridge, MA, 1985.

[McCarthy y Hayes, 1969]

Some Philosophical Problems from the Standpoint of Artificial Intelligence

J. McCarthy y P.J. Hayes

Machine Intelligence, vol 4, pp. 463-502. B. Meltzer y D. Michie eds. American Elsevier Publishing Co. Inc. 1969
<http://www-formal.stanford.edu/jmc/mechay69.html>

[McCarthy, 1997]

Situation Calculus with Concurrent Events and Narrative

J. McCarthy

Artículo no publicado. Computer Science Department, Stanford University. 1997.

<http://www-formal.stanford.edu/jmc/narrative/narrative.html>

[McCarthy, 1999]

Making Robot Conscious of Their Mental States

J. McCarthy

Computer Science Department, Stanford University. 1999
Actualización no publicada del artículo incluido en Machine Intelligence 15 workshop, Oxford University. 1995
<http://www-formal.stanford.edu/jmc/consciousness/consciousness.html>

[Mellado et al., 1992]

Sistema CAD Para Programar Robots

M. Mellado, J. Tornero y E. Vendrell

Revista Proyecto 2000, Ingeniería del Producto, N° 82. Ed. Pulsar, ISSN 0210-0118. Diciembre, 1992.

[Mellado y Tornero, 1993]

On the Spherical Splines for Robot Modeling

M. Mellado y J. Tornero

Working Conference on Geometric Modeling in Computer Graphics, Génova. Julio, 1993

[Mellado y Vendrell, 1994]

Software Tools for Teaching in Robotics

M. Mellado y E. Vendrell

TEMPUS - IMPACT Workshop "Teaching Control Engineering and Process Control for Engineers in Mining, Mechanical Engineering, Metallurgy". Miskolc, Hungría. 1994

[Mellado y Vendrell, 1995]

Software Específico Para La Docencia Práctica En Robótica

M. Mellado y E. Vendrell

XVI Jornadas De Automática, San Sebastián, España. Septiembre, 1995.

[Mellado, 1996]

Modelado, Detección de Colisiones y Planificación de Movimientos en Sistemas Robotizados mediante Volúmenes Esféricos

M. Mellado

Tesis Doctoral. Departamento de Ingeniería de Sistemas, Computadores y Automática. Universidad Politécnica de Valencia. 1996.

[Mellado et al., 1997]

Application of a Real Time Expert System Platform for Flexible Autonomous Transport in Industrial Production

M. Mellado, E. Vendrell, A. Crespo, J. Garbajosa, C. Lomba, K. Schilling y R. Mayerhofer

Advanced Summer Institute (ASI'97). Life Cycle Approaches to Production Systems: Management, Control, Supervision. Budapest, Hungría. 1997.

[Mellado *et al.*, 1998]

Architecture Definition for a Flexible Transport System in Industrial Production

M. Mellado, E. Vendrell, A. Crespo, E. Ballester, J.M. Aparicio, P. López y C. Lomba

3rd IFAC Symposium on Intelligent Autonomous Vehicles (IAV'98). Madrid, España. 1998.

[Mellado *et al.*, 1999]

Application of a Real Time Expert System Platform for Flexible Autonomous Transport in Industrial Production

M. Mellado, E. Vendrell, A. Crespo, P. López, J. Garbajosa, C.

Lomba, K. Schilling, H. Stützle & R. Mayerhofer

Special Issue on Computers in Industry, Elsevier Science, vol. 38. n^o.2, ISSN 0166-3615, pp. 187-200, March 1999.

[Mensch *et al.*, 1993]

REAKT Architecture

Mensch, D. Kersual, A. Crespo y F. Charpillet

Workshop on Integration in Real-Time Intelligent Control Systems IRTICS'93. Madrid, Spain. 1993.

[Mohri *et al.*, 1993]

Collision-free Trajectory Planning for Two manipulators Using Virtual Coordination Space

A. Mohri, M. Yamamoto y S. Marushima

Proceedings of the 1993 IEEE international Conference on Robotics and Automation, Vol. 1, pp. 674-679, Atlanta. 1993

[Möller *et al.*, 1998]

Modeling Ant Navigation with an Autonomous Agent

R. Möller, D. Lambrinos, R. Pfeifer, T. Labhart y R. Wehner

Proceedings of the 5th International Conference of the Society for Adaptive Behaviour, pp. 185-194. MIT Press. 1998.

[Moorehead *et al.*, 1999]

Autonomous Navigation Field Results of a Planetary Analog Robot in Anctartica

S. Moorehead, R. Simmons, D. Apostolopoulos y W. Whittaker

International Symposium on Artificial Intelligence, Robotics and Automation in Space. Junio, 1999.

http://www.ri.cmu.edu/pubs/pub_2122.html

[Muñoz y Ollero, 1996]

Smooth Trajectory Planning Meted for Mobile Robots

V.F. Muñoz y A. Ollero

Computational Engineering in System Applications (CESA'96), Lille, Francia. 1996.

[Muñoz *et al.*, 1995]

Analisy of Reactive Behaviour for the Implementation of a Reactive Architecture

J.L. Muñoz, J. Tornero, L.M. Tomás y F.J. García

The 4th International Symposium on Measurement and Control in Robotics. Smolenice (Slovakia). 1995.

[Muñoz, 1998]

Control en Robótica Móvil. Arquitectura y Metodología

J.L. Muñoz

Tesis Doctoral. Departamento de Automática, Electricidad y Electrónica Industrial. Universidad de Murcia. 1998.

[Murata, 1989]

Petri nets: properties, analysis and applications

T. Murata

Proceedings of IEEE, vol.77, n^o 4, pp. 541-580. 1989.

[Naphade *et al.*, 1999a]

Graph-Theoretic Generation of Assembly Plans, Part I: Correct Generation of Precedence Graphs

K.S. Naphade, R.H. Storer y S.D. Wu

Bell Labs, Lucent Technologies. Lehigh University. 1999.
(Artículo pendiente de revisión)

[Naphade *et al.*, 1999b]

Graph-Theoretic Generation of Assembly Plans, Part II: Problem Decomposition and Optimization Algorithms

K.S. Naphade, S.D. Wu y R.H. Storer

Bell Labs, Lucent Technologies. Lehigh University. 1999.
(Artículo pendiente de revisión)

[Navarro *et al.*, 1996]

Management and Monitoring of a Robotic System in a Windows Based Environment

S.F. Navarro, J.R. Díaz, M. Mellado y E. Vendrell

Advanced Manufacturing Processes, Systems and Technologies (AMPST'96), Bradford, Reino Unido. 1996.

[Navarro *et al.*, 1997]

Operating Over A Robotized System Through An Ethernet Network

S.F. Navarro, E. Ballester, J.R. Díaz, E. Vendrell y M. Mellado
Proceedings of the 2nd IEEE International Workshop on Factory
Communication Systems (WFCS'97), IEEE Press, The Institute of
Electrical and Electronics Engineers, Inc, ISBN 0-7803-4182-1,
Barcelona. Octubre, 1997.

[Newell y Simon, 1963]

GPS: A program that simulates human thought

A. Newell y H. Simon
Computers and Thought, eds. Feigenbaum and Feldman, McGraw-
Hill, New York. 1963.

[Newell y Simon, 1972]

Human Problem Solving

A. Newell y H. Simon
Englewood Cliffs, New Jersey, Prentice-Hall. 1972.

[Newell, 1973]

Production Systems: Models of Control Structures

A. Newell
Visual Information Processing (W. Chase, editor), cap. 10, pp. 463-
526, Academic Press, New York. 1973.

[O'Donnell y Lozano-Pérez, 1989]

**Deadlock-free and Collision-free Coordination of Two
Robot Manipulators**

P.A. O'Donnell y T. Lozano-Pérez
Proceedings of the 1989 IEEE International Conference on Robotics
and Automation, Vol. 1, pp. 484-489, Scottsdale, Arizona. 1989.

[Onaindía, 1997]

**Modelo de Representación y Razonamiento Temporal para
Sistemas Basados en el Conocimiento de Tiempo Real**

E. Onaindía
Tesis Doctoral. Departamento de Sistemas Informáticos y
Computación. Universidad Politécnica de Valencia. 1997.

[Peterson, 1981]

Petri Net Theory and the Modeling of Systems

J. L. Peterson
Englewood Cliffs, New Jersey, Prentice-Hall, 1981.

[Perkins y Austin, 1990]

Adding Temporal Reasoning to Expert System Building Environments

W. A. Perkins y A. Austin

IEEE Expert, 5 (1), pp. 23-30. Febrero, 1990.

[Piaget, 1926]

The Language and Thought of the Child

J. Piaget

New York, Harcourt, Brace. 1926.

[PSADD RETRARO, 1997]

Preliminary System Architectural Design Document

Project Report. ESPRIT IV Project N° 20.788 "Use of the Expert System Platform REAKT for Transport Robot Guidance (RETRARO)". Marzo, 1997.

[Reif y Sharir, 1994]

Motion Planning in the Presence of Moving Obstacles

J. Reif y M. Sharir

Journals of ACM 41, 4, pp. 764-790. 1994.

[Rich, 1983]

Artificial Intelligence

E. Rich

McGraw-Hill. New York. 1983.

[Rich y Knight, 1991]

Artificial Intelligence

E. Rich y K. Knight

2nd edition. McGraw-Hill. New York. 1991.

[Rietman, 1994]

Genesis Redux

E. Rietman

Windcrest/McGraw-Hill. 1994.

[Rosell *et al.*, 1999]

Compliant-Motion Planning and Execution for Robotic Assembly

J. Rosell, L. Basáñez y R. Suárez

Proceedings of the 1999 IEEE International Conference on Robotics and Automation, pp. 2774-2779, Detroit. Mayo, 1999.

[Sacerdoti, 1974]

Planning in a Hierarchy of Abstraction Spaces

E. Sacerdoti

Artificial Intelligence, Vol. 5, nº 2, pp. 115-135. Junio, 1974.

[Sánchez *et al.*, 1996]

Integrated System for Computer Aided Robot Programming

A.J. Sánchez, E. Vendrell y M. Mellado

Advanced Manufacturing Processes, Systems and Technologies (AMPST'96), Bradford, Reino Unido. 1996.

[Sánchez y Marchant, 1997]

Fast and Robust Method for Tracking Crop Rows Using a Two Point Hough Transform

A.J. Sánchez y J. Marchant

Proceedings of the International Workshop on Robotics and Automated Machinery for Bio-Productions (Bio-Robotics'97), pp. 33-38. Gandía, España. 1997.

[Sánchez *et al.*, 2000]

Línea de Investigación en Tele-Robótica

A.J. Sánchez, M. Mellado y E. Vendrell

Foro Iberoamericano de Reconocimiento de Formas y Análisis de Imágenes, Barcelona. 2000

[Sanz *et al.*, 1998]

Vision-Guided Grasping of Unknown Objects for Service Robots

P.J. Sanz, A.P. del Pobil, J.M. Iñesta y G. Recatalá

1998 International Conference on Robotics and Automation (ICRA'98), pp.3018-3025, IEEE Computer Society Press. ISBN: 0-7803-4758-7. 1998.

[Schoner *et al.*, 1995]

Dynamics of behaviour: Theory and applications for autonomous robot architectures

G. Schoner, M. Dose y C. Engels

Robotics and Autonomous Systems, 16 (2), pp. 213-246. 1995.

[Schwartz y Sharir, 1983]

On the “Piano Movers” Problem I: The Case of a Two-Dimensional Rigid Polygonal Body Moving amidst Polygonal Barriers

J. T. Schwartz y M. Sharir

Communications on Pure and Applied Mathematics, 36, pp. 345-398. 1983.

[Sergeyev y Chernov, 1997]

Image Recognition Based on the Principles of Pattern Recognition Theory

V.V. Sergeyev y A.V. Chernov

Pattern Recognition and Image Analysis, Vol.7, n^o.4, pp. 474-479, 1997.

[Siegler, 1994]

Cognitive variability: A key to understanding cognitive development

R.S. Siegler

Current Directions in Psychological Science, 3, pp. 1-5. 1994.

[Silva, 1985]

Las Redes de Petri: en la Automática y la Informática

M. Silva

Editorial AC. 1985.

[Simó, 1997]

Una Arquitectura Basada en Motivaciones para el Control de Robots Móviles

J. Simó

Tesis doctoral, Departamento de Ingeniería de Sistemas y Automática, Universidad Politécnica de Valencia. 1997.

[Sleumer y Tschichold-Gürman, 1999]

Exact Cell Decomposition of Arrangements used for Path Planning in Robotics

Nora H. Sleumer y N. Tschichold-Gürman

Institute of Theoretical Computer Science Swiss Federal Institute of Technology Zurich, Switzerland. 1999.

<ftp://ftp.inf.ethz.ch/pub/publications/tech-reports/3xx/329.ps.gz>

[Stonebreaker, 1992]

The Integration of Rule Systems and Database Systems

M. Stonebreaker

IEEE Transactions on Knowledge and Data Engineering, 4 (5), pp. 415-423. Invited Paper. 1992.

[Suárez *et al.*, 1995]

Using Configuration and Force Sensing in Assembly Task Planning and Execution

R. Suárez, L. Basáñez y J. Rosell

Proceedings of the 1995 IEEE International Symposium on Assembly Task Planning, pp. 273-279. 1995.

[Sutner y Maass, 1988]

Motion Planning Among Time Dependent Obstacles

K. Sutner y W. Maass

Acta Informatica, 26, pp. 93-133. 1988.

[Thrun, 1998]

Learning Metric-Topological Maps for Indoor Mobile Robot Navigation

S. Thrun

Artificial Intelligence Journal, 99 (1), pp. 21-71. 1998.

[Traven, 1993]

On Pattern Recognition Applications of Artificial Neural Networks

H.G.C. Traven

PhD thesis, Dept. of Numerical Analysis and Computing Science, Royal Institute of Technology, Stockholm, Sweden. 1993.

[Unruh y Rosenbloom, 1989]

Abstraction in Problem Solving and Learning

A. Unruh y P.S. Rosenbloom

Proceedings of the 11th International Joint Conference on Artificial Intelligence (IJCAI-89), pp. 681-687, Detroit. 1989.

[Vandorpe, 1997]

Navigation Techniques for the Mobile Robot LiAS

J. Vandorpe

Tesis doctoral, Division of Production Engineering, Machine Design and Automation, Department of Mechanical Engineering, Katholieke Universiteit Leuven, Bélgica. 1997.

http://www.mech.kuleuven.ac.be/pma/research/rim/rim_pl.html

[Vendrell *et al.*, 1991]

Programación de Robots mediante una Herramienta CAD Comercial

E. Vendrell, M. Mellado y J. Tornero

Proceedings del 2º Congreso de la Asociación Española de Robótica (AER), Zaragoza. 1991.

[Vendrell *et al.*, 1992a]

Robot Programming based on Commercial CAD Systems

E. Vendrell, M. Mellado y J. Tornero

Proceedings of the IFAC Symposium on Low Cost Automation (LCA'92), Viena. 1992.

[Vendrell *et al.*, 1992b]

Programación De Robots. Desarrollo De Un Entorno Basado En Un Sistema CAD

E. Vendrell, M. Mellado y J. Tornero

Revista Automatización Integrada & Robótica, N 72. Ed. Pulsar, ISSN 0210-0118, Noviembre, 1992.

[Vendrell *et al.*, 1993]

Sistema Educativo de Programación de Robots basado en PC

E. Vendrell, M. Mellado y E. Vidal

Proceedings del 3º Congreso de la Asociación Española de Robótica (AER), Zaragoza. 1993.

[Vendrell y Mellado, 1998]

Human Intuition Based Planning for Mobile Robots Considering Dynamic Environments

E. Vendrell y M. Mellado

Proceedings of the IFAC Symposium on Intelligent Manufacturing Systems 1998, Elsevier Science, Brasil. Noviembre, 1998.

[Vendrell *et al.*, 1999]

Management of a Robotised System on an Ethernet Network

E. Vendrell, M. Mellado, E. Ballester, S.F. Navarro y J.R. Díaz

Transactions of the Institute of Measurement & Control, Vol 21, N° 1, ISSN 0142-3312, pp 8-13. November, 1999.

[Vere y Bickmore, 1990]

A Basic Agent

S. Vere y T. Bickmore

Computational Intelligence 6, pp. 41-60. 1990.

<http://ai.eecs.umich.edu/cogarcho/homer/index.html>

[Warren, 1990]

Multiple Robot Path Coordination using Artificial Potential Fields

C. Warren

Proceedings of the 1990 IEEE International Conference on Robotics and Automation, Vol. 1, pp. 500-505, Cincinnati, Ohio. 1990

[Washington y Hayes-Roth, 1992]

Practical Real-Time Planning

R. Washington y B. Hayes-Roth

Research Report KSL-92-80. Knowledge Systems Lab. Stanford University. 1992.

[Washington, 1994]

Abstraction Planning in Real-Time

R. Washington

Tesis Doctoral. Department of Computer Science. Stanford University. 1994.

[Watanabe *et al.*, 1992]

Architecture of Behaviour-Based Mobile Robot in Dynamic Environment

M. Watanabe, K. Onoguchi, I. Kweon e Y. Kuno

Proceedings of the 1992 IEEE International Conference on Robotics and Automation, Vol. 3, pp. 2711-2718. 1992.

[Weld, 1998]

Recent Advances in AI Planning

D.S. Weld

Internal Report. Dep. of Computer Science Engineering, University of Washington. 1998

(Artículo pendiente de publicación en AI Magazine)

[Wirth, 1987]

Algoritmos y Estructuras de Datos

N. Wirth

Prentice Hall. 1987

[Xiao y Volz, 1989]

On Replanning for Assembly Tasks Using Robots in Presence of Uncertainties

J. Xiao y R. Volz

Proceedings of the 1989 IEEE International Conference on Robotics and Automation, pp. 638-645. 1989.

[Xu y Van Brussel, 1997]

A Behaviour-Based Blackboard Architecture for Reactive and Efficient Task Execution of an Autonomous Robot

H. Xu y H. Van Brussel

Robotics And Autonomous Systems, Vol 22, Issue 2, pp. 115-132. November, 1997.

[Yang *et al.*, 1996]

On the Implementation and Evaluation of AVTWEAK

Q. Yang, J. Tenenbergy S. Woods

Computational Intelligence, 12, pp. 307-330. 1996.

[Yang, 1997]

Intelligent Planning. A Decomposition and Abstraction Approach

Q. Yang

Springer Verlag. Berlin Heidelberg. 1997.

✱

*esta tesis doctoral se acabó de
realizar un relativamente frío
día de Noviembre del año 2000*



