

Índice General

1	Sistemas de control para un microbot	1
1	Introducción	1
2	Tipos de sistemas de control	2
2.1	Arquitecturas deliberadas	3
2.2	Arquitecturas reactivas	4
2.3	Arquitecturas híbridas	5
3	Arquitecturas deliberadas aplicadas a microbots	6
4	Arquitecturas reactivas aplicadas a microbots	8
5	Arquitecturas híbridas aplicadas a microbots	10

Índice de Figuras

1.1	Estrategias de control	3
1.2	Descomposición del sistema de control de un robot en estructuras horizontales .	6
1.3	Ejemplo de control deliberado con descomposición horizontal: a) detección de obstáculo; b) modelado del entorno; c) planificación; d) ejecución	7
1.4	Descomposición del sistema en tareas con una estructura vertical	8
1.5	Ejemplo de arquitectura de subsunción de dos niveles: a) paso 1; b) paso 2; c) paso 3	9
1.6	Ejemplo de arquitectura híbrida	10

Índice de Tablas

Capítulo 1

Sistemas de control para un microbot

1 Introducción

A finales de los años 40, Norbert Wiener desarrolló el concepto de cibernética, que integraba teoría de control, ciencias de la información y biología en un intento de exponer los principios comunes de control y comunicaciones tanto en robots como en animales [?]. Ashby [2] y Wiener extendieron el concepto de organismo a una máquina que, haciendo uso de la matemática desarrollada para sistemas de control realimentados, fuese capaz de exhibir un comportamiento natural. La cibernética contempla la necesidad de una relación biunívoca entre un agente y su entorno, es decir, implica la necesidad de sensores y de condicionar el comportamiento de un programa a la respuesta de éstos.

Un punto lógico de partida a la hora de generar modelos de control inteligente para un robot es el estudio de sistemas biológicos. A lo largo de la historia se han llevado a cabo importantes esfuerzos para dotar a las máquinas de la más simple de las habilidades animales: percibir el entorno y actuar dentro de él de la forma más coherente posible de acuerdo a un propósito definido. Establecer el paralelismo deseado no es una tarea fácil por dos razones fundamentales: i) la configuración de una máquina es muy distinta de la de un animal, que además dispone de una cantidad muy superior de elementos sensores y de proceso, y ii) es extremadamente difícil determinar las bases psicológicas y fisiológicas del comportamiento animal. Debido a estos motivos, gran parte de los sistemas robóticos se apoyan en consideraciones puramente ingenieriles, pero existen varias aproximaciones que se apoyan en distintas disciplinas de la ciencia.

En particular, si bien la creación de robots que mimetizan sistemas biológicos resulta

excesivamente compleja, la psicología se ha aplicado en las áreas donde la biología no tiene cabida. A este respecto, la psicología que se aplica al campo de la robótica se centra en el estudio del comportamiento en función del binomio percepción/acción. El conductismo apareció como disciplina en 1910 y, descartando factores del tipo sensación, propósito, pensamiento o emoción, define el comportamiento sólo por observaciones [10]. La intención del conductismo es hacer la psicología más científica y menos filosófica, sociológica o teológica al basarse únicamente en datos empíricos. Posteriormente, la psicología fenomenológica (*gestalt*) [6] establece que el comportamiento está relacionado no sólo con el estímulo, sino también con la estructura del entorno en su interacción con el agente.

La psicología cognitiva surge en paralelo a las ciencias informáticas, y sus pilares son la adquisición, organización y uso del conocimiento [9]. En este punto, los modelos computacionales de la mente comenzaron a tener mayor peso y los experimentos conductistas fueron perdiendo prominencia en el estudio de la inteligencia humana. Las principales aportaciones de esta disciplina son:

- Una serie de subsistemas procesa la información sensorial de forma sistemática.
- La técnica de procesamiento de información en un individuo está fuertemente correlada con la de una computadora.
- Se definen dos tipos de procesamiento. El procesamiento tipo abajo/arriba (*bottom-up*) se dispara con un estímulo, mientras que el tipo arriba/abajo (*top-down*) responde a intenciones y expectativas.

2 Tipos de sistemas de control

Las estrategias clásicas de control de robots se dividen típicamente en dos grupos: deliberadas y las reactivas. Las estrategias reactivas se basan en asociar a cada entrada o conjunto de entradas sensoriales una acción de forma directa. Por ejemplo, si el robot dispone de un sensor delantero, un sistema simple reactivo sería detener los motores cuando el sensor se activa. Los sistemas deliberados se basan en procesar la información de los sensores de forma más sofisticada para conseguir respuestas más eficientes. En general, este tipo de estrategias requiere un modelo del entorno en que el robot está operando. Siguiendo con el ejemplo anterior, un sistema deliberado simple podría consistir en memorizar la posición del obstáculo detectado para futuras referencias. Obviamente, la principal desventaja de las arquitecturas deliberadas frente a las reactivas es que su tiempo de respuesta y su carga computacional son más elevados. A efectos de aprovechar



Figura 1.1: Estrategias de control

las ventajas de uno y otro grupo de estrategias, aparecen las estrategias híbridas, que combinan características de ambos. Así, los sistemas de control de robots se sitúan en el rango que se presenta en la Fig. 1.1.

2.1 Arquitecturas deliberadas

Las arquitecturas deliberadas suelen descomponerse en niveles o capas. Típicamente, una arquitectura de este tipo se divide en tres niveles [7]: i) el nivel de organización, que lleva a cabo la planificación de alto nivel y razonamiento, ii) el de coordinación, encargado de integrar las respuestas de los distintos módulos hardware, y iii) el de ejecución, que soporta el control a bajo nivel y acciones directas sobre el hardware. Las arquitecturas de este tipo presentan las siguientes características:

- Presentan una estructura jerárquica con una clara división de funcionalidades.
- La comunicación entre procesos y el control ocurren de forma predecible y determinista.
- Los niveles superiores de la jerarquía descomponen la tarea global en subtareas para los niveles inferiores

- Los niveles inferiores suelen trabajar de forma local, sin tener en cuenta el objetivo global del sistema
- Se basan fuertemente en representaciones simbólicas del entorno.

Un ejemplo típico de una arquitectura de este tipo podría consistir en buscar el camino óptimo entre dos puntos de un entorno explorado. El nivel de razonamiento determinaría, a partir del modelo del entorno adquirido, una trayectoria óptima mediante un algoritmo de planificación como Dykstra o el A* algoritmo. A continuación, ese camino se comunica al nivel de coordinación, que se encarga de transformar las coordenadas del camino establecido en órdenes a los motores y en determinar si hay algún obstáculo en el camino en función de los sensores del robot. Por último, los motores se encargan de cumplir las órdenes que les ha enviado el nivel de coordinación mediante, por ejemplo, contadores de vueltas.

El control jerárquico es claramente adecuado para entornos muy estructurados y fácilmente predecibles. No obstante, si el modelo del entorno no tiene una validez prolongada, es complicado de procesar o difícil de adquirir o actualizar, no resultan adecuados. Como solución a este problema, los sistemas reactivos asocian intrínsecamente percepción y acción. Así, son capaces de producir respuestas casi inmediatas incluso tratando con entornos fuertemente dinámicos.

2.2 Arquitecturas reactivas

Los sistemas reactivos definen un comportamiento individual como una pareja estímulo/respuesta para unas condiciones determinadas, viéndose éste afectado por la atención y la intención del agente. La atención depende del entorno en cada momento y se emplea para priorizar tareas y focalizar los sensores sobre zonas de interés [3]. Así, si el objetivo es seguir una pared a la derecha del robot, resulta más interesante prestar atención a los sensores a la derecha de éste que a un posible sensor frontal. Por otra parte, la intención depende del objetivo global del agente y determina qué conjunto de comportamientos deberían estar activos. Por ejemplo, si la intención del agente es desplazarse una determinada distancia y encuentra un obstáculo al frente, la respuesta debería ser girar a derecha o izquierda y seguir avanzando. Si la intención hubiese sido avanzar en línea recta una determinada distancia, la respuesta debería haber sido detener el robot y concluir la acción con fracaso. La combinación instantánea de un conjunto de comportamientos individuales se denomina comportamiento reflejo, mientras que la línea de acción global resultante se denomina comportamiento emergente. Por ejemplo, si un robot desea avanzar hacia el norte del entorno sin chocar con un obstáculo frontal, un comportamiento reflejo sería apartarse de los obstáculos en una dirección determinada. Si se combina este reflejo

jo con la acción de avanzar hacia el norte, el vector de movimiento resultante se denominaría comportamiento emergente.

Los robots basados en comportamientos puramente reactivos también presentan desventajas, en tanto que no trabajan con representaciones del entorno y por tanto no admiten ningún tipo de previsión. Debido a ello, pierden eficacia y versatilidad y pueden caer en trampas derivadas de mínimos locales.

2.3 Arquitecturas híbridas

Las arquitecturas híbridas han aparecido recientemente, combinando aspectos tradicionales de la inteligencia artificial, como es el uso de representaciones simbólicas, con el objetivo de ofrecer la robustez, rapidez y flexibilidad de los sistemas reactivos. Estas arquitecturas reconfiguran los módulos de control reactivo de acuerdo a la información disponible del entorno y a estrategias basadas en razonamiento global. En general, varias tareas necesitan control deliberado, como la planificación o toma de decisiones, el manejo de problemas inesperados, las situaciones novedosas o aprendidas de forma errónea, las acciones peligrosas y la superación de hábitos. Por su parte, otras acciones más inmediatas como comportamientos reflejos o adquiridos no requieren supervisión global. Una de las opciones más comunes consiste en controlar los comportamientos reactivos mediante umbrales, que fijan los módulos de planificación: si bien la percepción dispara la acción, los procesos atencionales la modulan. Así, el módulo de control de alto nivel permite una integración coherente de múltiples comportamientos concurrentes.

Las estrategias más comunes en este tipo de arquitecturas son las siguientes:

- Selección [1]: La planificación se entiende como una configuración que puede alterarse de acuerdo al porcentaje de éxitos del sistema. El planificador determina qué comportamientos actúan y con qué parámetros lo hacen durante la operación del agente.
- Recomendación [4]: En este caso la planificación se toma como un consejo que puede o no ser aceptado por los comportamientos.
- Adaptación [8]: El planificador cambia continuamente los distintos comportamientos en función de las condiciones del entorno y las tareas requeridas.
- Posposición [5]: El planificador evita tomar decisiones o hacer cambios durante el mayor tiempo posible. Así, el plan de acción está adaptado a los datos adquiridos de forma más reciente en lugar de a las condiciones iniciales.

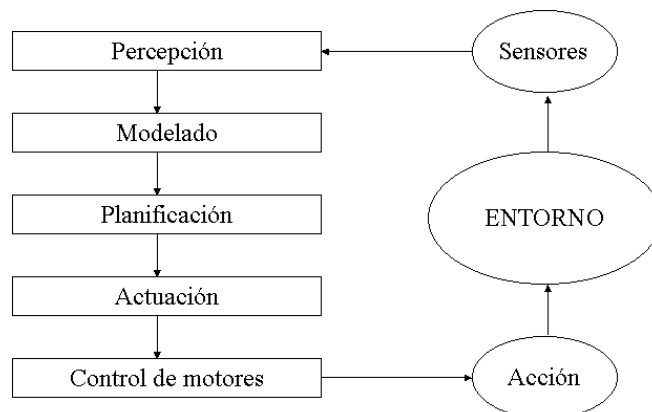


Figura 1.2: Descomposición del sistema de control de un robot en estructuras horizontales

3 Arquitecturas deliberadas aplicadas a microbots

Las arquitecturas deliberadas tienden a estructurarse de acuerdo al modelo *top-down* comentado anteriormente y describen y procesan el entorno mediante un conjunto discreto de acciones y eventos. Habitualmente, para implementarlas se recurre a una descomposición horizontal como la que se muestra en la Fig. 1.2. Los bloques que se incluyen son los siguientes:

- **Percepción:** Este módulo se encarga de recoger información del entorno. Básicamente, lee cada sensor y le da un sentido físico. Por ejemplo, un sensor infrarrojo sólo indica la presencia o ausencia de obstáculos. El módulo de percepción añadiría la dirección en que se ha detectado un obstáculo en función de la ubicación del sensor respecto al robot (Fig. 1.3.a).
- **Modelado:** Este módulo se encarga de fundir la información de los sensores en el espacio y en el tiempo para construir un mapa del entorno. Por ejemplo, si se dispone de un robot con un sensor infrarrojo situado a la derecha y dicho sensor se activa dos veces consecutivas y se desactiva en una tercera lectura, el módulo de modelado generaría un modelo del entorno como el que se marca en rojo en la Fig. 1.3.b).
- **Planificación:** Este módulo se encarga de construir un plan de acción para el robot. Para ello, se basa en el modelo del entorno disponible. Por ejemplo, si se dispone del modelo de entorno de la Fig. 1.3.b, un planificador podría diseñar una trayectoria suave para doblar la esquina norte (Fig. 1.3.c).

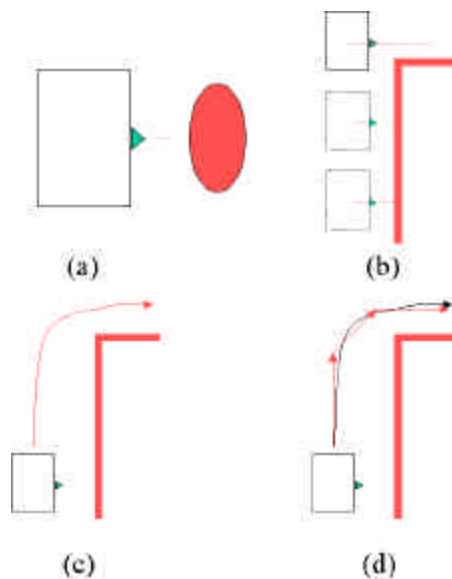


Figura 1.3: Ejemplo de control deliberado con descomposición horizontal: a) detección de obstáculo; b) modelado del entorno; c) planificación; d) ejecución

- **Actuación:** Este módulo se encarga de mover el robot de acuerdo al plan del planificador. Siguiendo con el ejemplo anterior, un actuador podría encargarse de descomponer la trayectoria deseada en órdenes que comunicar a los motores, como se muestra en la Fig. Fig. 1.3.d.
- **Control de motores:** Este último módulo actúa a bajo nivel directamente sobre los motores. En el ejemplo comentado, se encargaría de avanzar y girar cuanto sea necesario para cumplir las órdenes del actuador de acuerdo a los contadores de vueltas de las ruedas.

En este tipo de sistemas es importante establecer una comunicación eficaz entre capas. El modelo más habitual es el modelo pizarra (*Blackboard*), consistente en establecer una zona de memoria común a la que pueden acudir todos los procesos a leer o escribir información. Es importante tener cuidado de que no aparezcan cuellos de botella ni errores de temporización cuando se trabaja con este esquema. Para ello, hay que evitar colas en el acceso a la información y cada proceso debe saber cómo de actualizada está la información disponible.

El conjunto de todos los módulos se conoce como el ciclo sensor-modelo-plan-acción (*sense-model-plan-act cycle*). Siguiendo este esquema, se consigue un módulo de control que opera siempre de forma determinista. Es importante notar que el sistema no funciona hasta que todos los módulos se han implementado. No obstante, la mayor desventaja de estas arquitecturas es su incapacidad de generar respuestas rápidas, ya que la entrada de los sensores ha de atravesar todos los módulos del sistema para que se genere una respuesta motora. El tiempo que media



Figura 1.4: Descomposición del sistema en tareas con una estructura vertical

entre que un sensor se lee y se genera una respuesta se conoce como periodo de latencia.

4 Arquitecturas reactivas aplicadas a microbots

Las arquitecturas reactivas se basan, tal como se ha comentado, en establecer una correspondencia directa entre sensores y actuadores. Este tipo de arquitecturas pretenden cubrir varias deficiencias de los planificadores clásicos, como son: i) la incapacidad de trabajar con múltiples objetivos; ii) la dificultad de manejar múltiples sensores; y iii) la falta de robustez al ruido y a eventos inesperados.

Una de las arquitecturas reactivas más habituales es la arquitectura de subsunción, que consiste en definir comportamientos deseables de bajo a alto nivel, de tal forma que cada comportamiento engloba al inmediatamente anterior. Un comportamiento cualquiera se genera de acuerdo a la respuesta de uno o varios módulos. En este caso, los niveles inferiores siempre tienen prioridad sobre los superiores. La arquitectura ofrece una división básica en los siguientes niveles:

- Evitar colisiones.
- Desplazarse por el entorno.
- Explorar el mundo.
- Construir modelos del mundo.
- Actualizar modelos del mundo.
- Razonar acerca de los modelos.

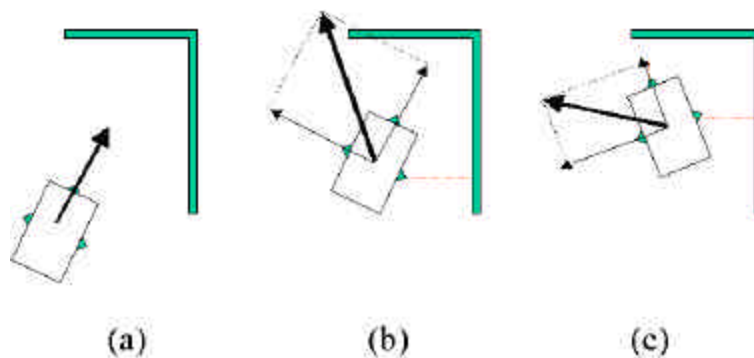


Figura 1.5: Ejemplo de arquitectura de subsunción de dos niveles: a) paso 1; b) paso 2; c) paso 3

- Tomar decisiones y llevarlas a cabo.
- Razonar sobre el comportamiento de otros objetos y actuar en consecuencia.

La acción resultante de esta arquitectura, dado un conjunto de estímulos, es igual a la combinación de las respuestas de todos los módulos del sistema (Fig. 1.4).

Un ejemplo de este tipo de arquitecturas con dos únicos niveles podría ser el siguiente. Dado un robot dotado de un sensor frontal y dos laterales, uno a cada lado, un primer nivel que evita colisiones consistiría en doblar a la derecha si se activa el sensor izquierdo y viceversa y frenar si se activa el sensor frontal. Un segundo nivel, que trata de que el robot se desplace, consistiría simplemente en avanzar en la dirección en que el robot se encuentra. La Fig. 1.5 muestra un ejemplo del comportamiento resultante del robot en un caso cualquiera. En la Fig. 1.5.a, el robot avanza hacia un obstáculo. Sus sensores aún no se han activado, por tanto el camino está libre -se cumple el primer nivel- y se puede avanzar en línea recta -se cumple el segundo nivel-. De repente (Fig. 1.5.b) se activa el sensor de la derecha y el comportamiento asociado a este evento es girar 90 grados a la izquierda. No obstante, sigue activo el comportamiento de avance, ya que no se ha detectado nada con el sensor frontal. Así pues, la resultante de ambos comportamientos es girar a la izquierda, pero no con un ángulo de 90 grados, sino de forma mucho más suave. Cuando el agente se acerca a la pared, el sensor frontal se activa, con lo que se produce un decremento en el vector de avance. Dado que el sensor derecho sigue activo, el comportamiento generaría un giro más pronunciado a la izquierda para evitar el obstáculo y seguir avanzando en la medida de lo posible. La mayor ventaja de esta arquitectura es que los módulos se pueden reemplazar o añadir fácilmente sin perjuicio para la estructura global. La mayor desventaja es que el comportamiento resultante es poco predecible y, por tanto, es difícil depurar el sistema completo.

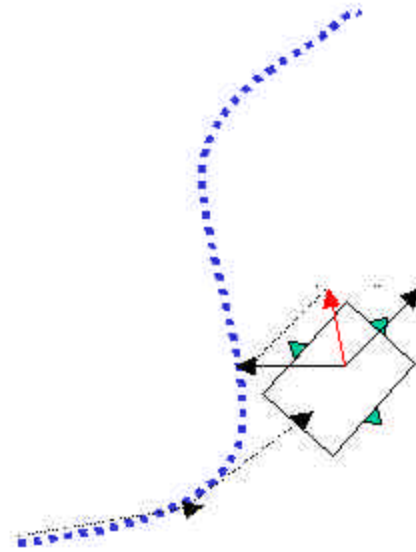


Figura 1.6: Ejemplo de arquitectura híbrida

5 Arquitecturas híbridas aplicadas a microbots

En general, y dado que tanto los sistemas reactivos como los deliberados presentan ventajas e inconvenientes, lo habitual es combinar ambas estrategias en mayor o menor medida para obtener una arquitectura híbrida. Usualmente, se usan técnicas deliberadas en los módulos de alto nivel, que funcionan de forma discreta de acuerdo a la carga computacional del sistema (control estratégico). A bajo nivel, sin embargo, se emplean estrategias reactivas de manera continua (control táctico). Así, por ejemplo, un planificador global podría devolver una trayectoria eficiente al robot, mientras que un módulo reactivo se encarga de seguir dicha trayectoria adecuándose a cualquier imprevisto. Por ejemplo, la Fig. 1.6 muestra una arquitectura híbrida donde un planificador global devuelve la trayectoria azul y uno local devuelve el comando en rojo. Puede observarse que el agente, debido a cuestiones mecánicas, no ha efectuado correctamente el giro y ha abandonado la trayectoria prevista. A nivel reactivo, el agente tiende a avanzar en línea recta por comodidad. Sin embargo, también debe tender a seguir el camino marcado por el nivel superior. Así, la composición de ambas fuerzas devuelve un comando de movimiento que mantiene la continuidad del movimiento pero que, poco a poco, también regresa a la trayectoria prevista.

Bibliografía

- [1] Arkin, R., "Path planning for a vision-based autonomous robot", *Proc. of the SPIE Conference on Mobile Robots*, pp. 240-249, Cambridge-MA, 1986.
- [2] Ashby, W.R., *Design for a brain: the origin of adaptive behaviour*, J. Wiley: New York, 1952.
- [3] Brooks, R.A., *A robust layered control system for a mobile robot*, en P.H. Winston and S.A. Shellard (eds.), *Artificial Intelligence at MIT Expanding Frontiers*, **2**, pp. 3-27, MIT Press Boston, 1990.
- [4] Gat, E., "ALFA: a language for programming reactive robotic control systems", *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'91)*, **2**, pp. 1116-1120, Sacramento-EEUU, 1991.
- [5] Georgeff, M. y Lansky, A., "Reactive reasoning and planning", *Proc. of the 6th national conf. on Artificial Intelligence of the American Assoc. for Artificial Intelligence (AAAI-87)*, **677**, pp. 677-682, Seattle-EEUU, 1987.
- [6] Kohler, W., *Gestalt psychology: an introduction to new concepts in modern psychology*, Liveright Publishing Co.: New York, 1947.
- [7] Lefebvre, D. y Saridis, G., "A computer architecture for intelligent machines", *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'92)*, **3**, pp. 2745-2750, Nice-Francia, 1992.
- [8] Lyons, D. y Hendricks, A., "Planning for reactive robot behavior", *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA'92)*, **3**, pp. 2675-2680, Nice-Francia, 1992.
- [9] Neisser, U., *Cognition and reality: principles and implications of cognitive psychology*, W.H. Freeman: San Francisco, 1976.
- [10] Watson, J., *Behaviourism*, People's Institute Publishing Co.: New York, 1925.

