

Proyecto de grado

Control y Comportamiento de Robots Omnidireccionales

Alcance del Sistema

Sistema EasyRobots

Santiago Martínez, Rafael Sisto

pgomni@fing.edu.uy

<http://www.fing.edu.uy/~pgomni>

Tutor

Gonzalo Tejera

Cotutores

Facundo Benavides, Santiago Margni

Versión 1.3

Instituto de Computación
Facultad de Ingeniería - Universidad de la República
Montevideo - Uruguay

31 de mayo de 2009

Índice

1. Introducción	7
1.1. Proposito	7
1.2. Alcance	7
2. Casos de Uso del Sistema	9
2.1. Inicialización del sistema	9
2.2. Activación de Estrategia	9
2.3. Desactivación de Estrategia	9
2.4. Recibimiento y envío de datos de entidades	9
3. Requerimientos del Sistema	11
3.1. Requerimientos dentro del Alcance	11
3.2. Requerimientos fuera del Alcance	11

Índice de figuras

1.	Diagrama general del sistema a implementar	8
----	--	---

1. Introducción

1.1. Proposito

Este documento presenta los requerimientos del sistema y los casos de uso que el sistema debe soportar.

1.2. Alcance

El control y comportamiento de robots omnidireccionales requiere de un sistema computacional. El sistema computacional es el medio que recibe datos del estado de la simulación de los robots, los procesa y determina el comportamiento próximo a ejecutarse por los robots. Por esta razón, en el proyecto se debe desarrollar un sistema computacional que permita controlar, determinar e indicar la acción a realizar a los robots que se construyan. El objetivo del sistema es que posea la abstracción necesaria para ejecutar un amplio espectro de aplicaciones de control y comportamiento de robots, y que imponga la menor cantidad de limitantes posibles sobre las estrategias de comportamiento, dispositivos de visión del sistema físico y mecanismos de control a fin de ser utilizado en diferentes robots omnidireccionales con diferentes estrategias de comportamiento y control.

Por lo anterior, se desarrolla un sistema que brinde las interfaces necesarias para desarrollar diferentes módulos para el manejo de robots y permita integrar entre sí las diferentes funcionalidades de forma simple y sin tener la necesidad de conocer aspectos específicos como el manejo de eventos, la comunicación entre el medio físico y virtual y el modelado del mismo en el mundo físico. Las abstracciones buscadas pueden ser a grandes razgos:

Abstracción del control de los robots y comunicación con los mismos: el usuario no debe considerar como se realiza el control del robot y como se envían los datos al mismo. Como ejemplo, para indicarle al robot que se desplace, al usuario le basta con indicar la posición y ángulo final, y la velocidad requerida o el tiempo en el cual se quiere que se ejecute el desplazamiento completo. De esta forma no es necesario conocer la velocidad que tendrá cada rueda ni como se envía dicha indicación al robot.

Abstracción de las implementaciones de estrategias: el usuario define solamente los aspectos más relevantes de la estrategia. Como ejemplo, utilizando la estrategia de Campos Potenciales, el usuario define atractores y repulsores mientras que la evaluación y demás análisis de los campos son internos al sistema, transparentes al usuario. Similar para otras estrategias como por ejemplo Diagramas de Voronoi.

Abstracción de la integración de los datos recibidos de los sensores: el usuario define sensores, la relevancia y confianza de cada uno, y el sistema los procesa de acuerdo a diferentes estrategias de fusión. Nuevamente, la recepción de los datos, la fusión y procesamiento de los mismos es transparente al usuario.

Abstracción de la construcción del robot que es utilizado: el usuario especifica la forma en como está construido el robot y luego el sistema envía los datos para realizar una acción considerando la construcción indicada. Por ejemplo, si se requiere que un robot de tres ruedas se desplace sobre la superficie, se enviará un mensaje a éste que será distinto al mensaje que se enviaría por ejemplo en el caso de contar con un robot de cuatro ruedas.

El desarrollo se realiza considerando las diferentes aplicaciones como pueden ser futuros cursos de robótica, competencias entre universidades o a nivel interno de la Facultad de Ingeniería de la Universidad de la República [2], en particular el grupo MINA [3] tales como el Sumo de Robots [1] o Fútbol de Robots [1].

Como ejemplo específico de aplicación de este sistema se desarrollará entre otros un robot omni-direccional de tres ruedas luchador de sumo y se verificará el correcto funcionamiento.

De forma general, la figura 1 representa la funcionalidad del sistema: se reciben datos de lectura de diferentes sensores los cuales pueden variar en tipo y precisión de las mediciones, se fusionan y luego se determina la acción a seguir. Para ello se utilizan los datos fusionados, la estrategia de planificación deseada y la representación lógica del robot que se busca controlar y se determina la acción que debe ejecutar el robot. Esta acción es enviada al servidor que controla el robot en un mensaje que puede ser interpretado por el servidor. Este mensaje debe contener los datos a enviar al robot asignándole la velocidad de cada rueda.

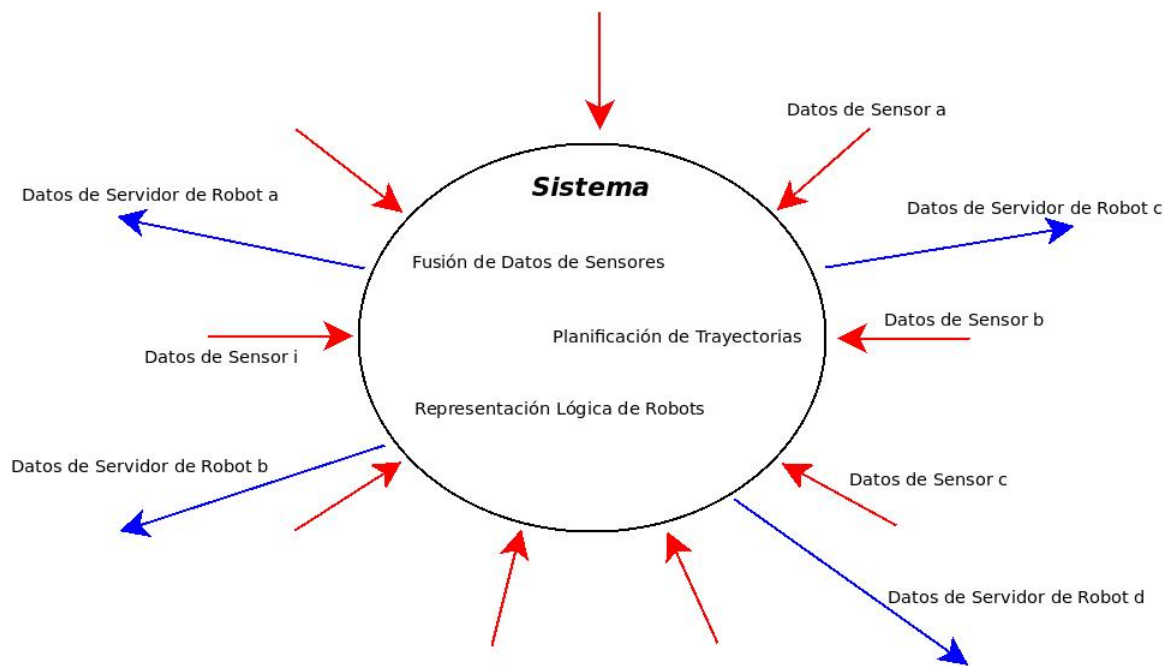


Figura 1: Diagrama general del sistema a implementar

2. Casos de Uso del Sistema

2.1. Inicialización del sistema

El usuario inicializa el sistema y éste crea las instancias que reflejen el estado inicial solicitado. Para ello el usuario debe brindar un archivo que describa una aplicación robótica conteniendo una o varias estrategias que se podrán utilizar, los Sistemas de Sensado para la obtención de datos, la especificación de la estructura del robot como es la posición de cada rueda respecto del centro del robot, la orientación de la misma respecto del eje así como otros atributos necesarios para definir completamente la aplicación y el Sistema de Actuación con dicho robot.

2.2. Activación de Estrategia

El usuario activa una de las estrategias definidas que controlará a los robots identificados en ella. De esta forma, se creará un camino desde las mediciones que son obtenidas por los sensores, luego procesadas y luego enviadas mediante acciones a seguir a los robots controlados.

2.3. Desactivación de Estrategia

El usuario desactiva una de las estrategias que se encuentren activas originando la detención de los robots controlados por dicha estrategia. Se debe estar seguro de enviar una orden de detención a los robots de forma previa a la desactivación para asegurar que los robots no continúen desplazandose en el caso de aquellos que hasta no recibir una nueva orden ejecutan la última recibida.

2.4. Recibimiento y envío de datos de entidades

El sistema debe obtener posiciones de los robots y otras entidades que controle. Las posiciones las brindan diferentes tipos y cantidades de sensores por lo que deben ser traducidas a un mismo sistema de referencia para luego ser fusionadas y procesadas. El procesamiento se realiza a partir de la estrategia de planificación que se utilice y la representación del robot como puede ser el posicionamiento de las ruedas, el radio de éste entre otros. A partir de esto, se determina la velocidad a asignar a cada rueda. Las velocidades son ingresadas en un mensaje que será enviado al servidor que maneje el robot destino.

3. Requerimientos del Sistema

A fin de que el caso de uso descrito en 2.4 se cumpla, se impusieron ciertos requerimientos funcionales en el alcance del sistema. A continuación se describirán dichos requerimientos así como aquellos que se eligieron no incluir en el alcance, pudiendo éstos ser implementados en trabajos futuros.

3.1. Requerimientos dentro del Alcance

Los requerimientos considerados dentro del alcance son todos aquellos definidos en el documento Especificación de Requerimientos de Software [4].

3.2. Requerimientos fuera del Alcance

Como requerimientos no considerado en el alcance, se encuentra los siguientes:

- Interoperabilidad: No se implementará la interfaz que permita solicitar y modificar velocidades o posiciones de los robots controlados por la aplicación desde otras aplicaciones. Se tomó la decisión de no incluir este requerimiento ya que se consideró que no es objetivo de la aplicación en el proyecto, aunque puede ser implementada en futuros trabajos sobre este desarrollo.

Referencias

- [1] CAETI Centro de Altos Estudios en Tecnologia Informatica. Campeonato argentino de fútbol de robots. Retrieved 19 May 2009 <http://www.vaneduc.edu.ar/cafr/>.
- [2] Facultad de Ingeniería. Retrieved 18 May 2009 <http://www.fing.edu.uy>.
- [3] Grupo MINA. Network management - artificial intelligence. Retrieved 18 May 2009 <http://http://www.fing.edu.uy/inco/grupos/mina/index.html>.
- [4] Rafael Sisto and Santiago Martínez. Especificación de requerimientos de software. Technical report, Instituto de Computación, Facultad de Ingeniería, Universidad de la República, Uruguay, 2009.