

# JRF: Una Herramienta Multiplataforma para el Diseño y Simulación de Estructuras Robóticas

J. L. Gómez, V. Moreno, B. Curto, A. M. Moreno, E. Sanz, I. Álvarez, F. J. Blanco  
Departamento de Informática y Automática.  
Facultad de Ciencias.  
Universidad de Salamanca  
Plaza de la Merced s/n 37008. Salamanca  
control@abedul.usal.es

## Resumen

*En este artículo se presenta la herramienta JRF (Java Robotics Factory) para ser utilizada en la docencia práctica de la materia de Robótica. Su objetivo primordial es que los alumnos puedan diseñar y simular el comportamiento de diferentes estructuras robóticas dentro de un espacio de trabajo especificado.*

*JRF es una herramienta altamente configurable en cuanto al tipo de robots que se pueden diseñar y en cuanto al entorno virtual en el que trabaja. Se consigue así que el alumno profundice en el conocimiento de las transformaciones cinemáticas y que comprenda las diferentes posibilidades de acceso a un punto dependiendo de la configuración seleccionada.*

*La ejecución de la aplicación se puede realizar en cualquier plataforma debido al lenguaje con el que se ha desarrollado. El realismo de las representaciones gráficas viene garantizado por la utilización de VRML y Java 3D.*

*De esta forma el estudiante podrá explotar al máximo las posibilidades de la aplicación, como paso previo al manejo de robots manipuladores reales. Con ello se alcanza un mayor nivel de seguridad, tanto para los alumnos como para los robots, y la disminución de los costes a la hora de montar un laboratorio de Robótica. Con esta experiencia, se ha mostrado cómo una herramienta de simulación puede ser de gran utilidad en la enseñanza práctica de la Robótica.*

**Palabras clave:** Simulación, Robótica, VRML, Java 3D

## 1. Introducción

La Robótica es una materia que se está impartiendo en diversas titulaciones de Ingeniería, por cuanto este tipo de dispositivos de actuación están presentes, cada vez en mayor grado, en la industria manufacturera. Desde el punto de vista docente, la enseñanza de esta materia consta de una serie de sesiones teóricas, donde se imparten los principales conceptos, y lo que es fundamental, de una serie de sesiones prácticas donde el alumno debe aplicar los

conocimientos adquiridos sobre manipuladores reales, donde puede comprobar las diferentes capacidades de movimiento y operación de cada estructura robótica.

Actualmente, los equipos necesarios para realizar prácticas de Robótica tienen un elevado coste que imposibilita, en muchas ocasiones, que el estudiante acceda y maneje diversos robots industriales. No obstante, una alternativa es recurrir a manipuladores enfocados a la docencia, como el PUMA de TecQuipment [5] o el MENTOR de Italtel/Aleco [6]. Su utilización permite realizar un amplio abanico de experiencias, sin embargo, siempre muestran las limitaciones asociadas a tener que trabajar con una única estructura cinemática sin poder estudiar las ventajas e inconvenientes de las diferentes configuraciones. No obstante, el coste de un laboratorio de Robótica para las Ingenierías sería muy elevado si únicamente se cuenta con robots educativos. Esto es debido al número de alumnos que cursan la asignatura, lo que supondría aumentar el número de puestos de trabajo o el número de horas de docencia práctica impartidas por el profesor.

Asimismo, un aspecto fundamental a tener en cuenta es la seguridad en el manejo de los robots, que va tomando mayor importancia conforme aumentan las prestaciones del manipulador. La seguridad se ha de tener en cuenta, por una parte, en lo que se refiere a la integridad del manipulador ante posibles colisiones con su entorno o con elementos de su propia cadena cinemática. Por otro lado, hay que tener en cuenta al operario, en este caso el alumno, para que no sea golpeado por el manipulador cuando, por descuido, se introduzca en la zona de trabajo del robot. Estos aspectos tienen mayor importancia por tratarse de unos usuarios, los alumnos, que están en la primera fase de aprendizaje.

Surge, por tanto, como alternativa la utilización de simuladores de robots que puedan apoyar tanto la enseñanza teórica como la práctica. Existen ya diversos trabajos que han enfocado este problema. Los trabajos iniciales, con unos entornos gráficos limitados, ofrecían la posibilidad de simulación de ciertas estructuras robóticas. Muchos de ellos, como [1][9], están limitados a uno de los manipuladores

más utilizados, el antropomórfico o tipo PUMA. Aunque este tipo está muy extendido en la industria de manufacturación, especialmente en el sector del automóvil, existen otras estructuras, como el SCARA o el Stanford, que también tienen una fuerte presencia en la realidad industrial. Sería muy conveniente disponer de una herramienta donde el alumno fácilmente pueda definir todas las estructuras estudiadas para analizar la accesibilidad a los distintos puntos del espacio de trabajo, y donde pueda experimentar otras que él diseñe.

En la mayoría de los trabajos previos únicamente se hace hincapié en la utilización de los métodos de aprendizaje y ensayo propios de los manipuladores industriales reales. En nuestro entorno docente es necesario además plantear al alumno que el robot industrial no es un mero actuador teleoperado si no un recurso programable.

Otra de las restricciones que presentan las herramientas de este tipo, es la capacidad limitada de representar los objetos que forman parte del entorno donde trabaja el robot. Aunque se han realizado importantes avances en este campo, en general, las soluciones definen un lenguaje propio para la representación de los objetos. Esto supone una dificultad añadida para el alumno, ya que debe aprender este lenguaje de definición antes de poder experimentar con la herramienta.

Asimismo cabe señalar que las aplicaciones existentes están asociadas a la presencia de una biblioteca gráfica y una plataforma hardware, que son propietarias. Por tanto, una característica tan deseable como la portabilidad queda muy limitada a la presencia de estos recursos. Serán estas limitaciones las que han servido como especificación para la herramienta objetivo de este trabajo.

En este artículo se van a presentar las principales características de la herramienta desarrollada. Así, en el apartado II se van a analizar las funcionalidades más destacadas que debe reunir un entorno de simulación de estructuras robóticas. A continuación, en el apartado III, se examinarán los requisitos fundamentales que han guiado el diseño de la herramienta presentada. Con ello, en el siguiente apartado, se presentan las herramientas utilizadas y que permiten alcanzar los requisitos planteados. En el apartado V se destacan las características más sobresalientes de la aplicación con su interfaz de usuario. Y, por último, en el apartado VI se finaliza con la presentación de las principales conclusiones.

## 2. Funcionalidades de un entorno de simulación de estructuras robóticas

En la Figura 1 se observan las funciones que se han de poder desarrollar en una aplicación que se utilice para la docencia en Robótica. Como se puede percibir, se plantea en primer lugar la dotación de una utilidad en la que el alumno pueda diseñar diferentes configuraciones de cadenas cinemáticas. Se deben incluir diferentes tipos de elementos con las dos posibilidades de articulación: prismática y de revolución. Para fijar los conceptos teóricos, se podrán parametrizar utilizando variables semejantes a las planteadas por Denavit-Hartenberg.

Una vez que el alumno ha construido su robot, es necesario incluirlo en su entorno de trabajo. Los avances en el campo de la realidad virtual son de gran utilidad, como se ha observado en diferentes experimentos docentes [9]. Así, disponer de representaciones dotadas de gran realismo en cuanto a iluminación, texturas, puntos de vista, etc. motiva al alumno para la utilización de la herramienta.

Además, se pretende que el alumno adquiera la capacidad de programar los robots. Para ello debe ser capaz de mover cada una de las articulaciones del robot y observar el efecto que este movimiento tiene en el efector final. El objetivo fundamental es definir un programa o secuencia de movimientos que se podrá ejecutar, posteriormente, en simulación. Se pretende que el alumno pueda, por una parte, comprender conceptos claves como la accesibilidad de cada cadena cinemática. Por otra parte, se plantea que la herramienta incorpore la capacidad de programación que disponen los sistemas robóticos comerciales.

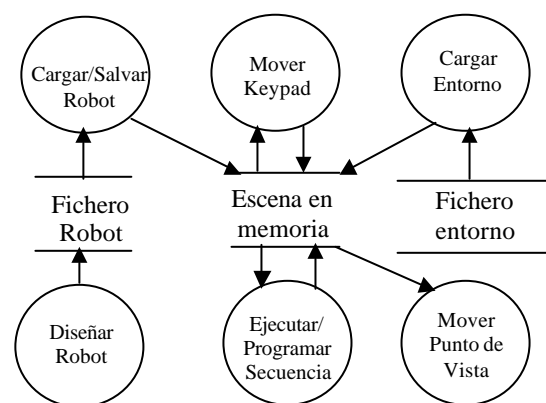


Figura 1.- Funcionalidades de JRF

La intención es que los estudiantes comprendan conceptos claves de la Robótica con la seguridad que aporta la simulación pero con el suficiente nivel

realismo para no perder la relación con los manipuladores reales.

### 3. Requisitos de diseño

Para definir los requisitos de esta aplicación se ha tenido en cuenta la experiencia de las personas que integran nuestro grupo de trabajo, tanto en el desarrollo de aplicaciones [1][3] como en el manejo de las disponibles en otros centros [2].

A continuación, se van a destacar las restricciones que han guiado el diseño y se señalará la solución aportada:

- **Alta disponibilidad.** Las aplicaciones existentes estaban, en general, restringidas a la presencia de recursos hardware y software concretos. Así, [3] dispone de unas grandes capacidades gráficas, pero que dependen de la presencia de las librerías específicas de *Silicon Graphics*. Respecto al software, por ejemplo, el simulador de *Robot Toolbox* [2] exige la presencia de una licencia de MatLab [11] que, aunque es muy común en el entorno de las enseñanzas de Ingeniería, siempre puede suponer una limitación a su utilización. Como solución, se plantea utilizar Java. La presencia de la máquina virtual se ha hecho común en los ordenadores debido al enorme éxito que tiene el carácter multiplataforma. No cabe duda que su carácter de software de libre acceso es responsable de este éxito.
- **Capacidad para configurar robots.** En muchos casos, las aplicaciones disponibles están limitadas a la utilización de un determinado tipo de estructura robótica. Así, [1][7] se restringen a robots tipo PUMA de Tecquipment [5]. Otras herramientas, como [2][8], son capaces de utilizar diferentes tipos de robots, pero presentan el problema de que la configuración cinemática y la definición de los elementos del robot se realiza a través de complejos ficheros de configuración en modo texto. Se pretende ahorrar al alumno la necesidad de aprender la sintaxis de estos ficheros y proporcionar un cómodo editor gráfico que, utilizando una estructura de árbol, permita al alumno concentrarse en la comprensión de la cadena cinemática del robot.
- **Representación del entorno.** Otro de los aspectos que se pretenden contemplar es la capacidad de incluir el entorno de trabajo

del manipulador. Con esto se puede conseguir que el alumno pueda observar las capacidades de accesibilidad del robot considerado. Además, se pretende dotar a la aplicación de un gran realismo. En este sentido, la utilización de Java 3D y VRML (*Virtual Reality Modeling Language*) nos van a permitir alcanzar este objetivo.

- **Capacidades de programación.** Una vez que el alumno ha experimentado las posibilidades y limitaciones de movimiento relacionadas con cada uno de los grados de libertad del robot, es necesario introducir su carácter de dispositivo programable. Así, se plantea la opción de disponer de un editor de programas del robot, así como la posibilidad de simular su ejecución. En este sentido, es importante introducir una interfaz para que el robot interactúe con su entorno a través de un conjunto de E/S.

### 4. Herramientas utilizadas

En este apartado, se va a realizar una descripción de las principales herramientas que se han utilizado en este proyecto: Java como lenguaje de programación y el estándar VRML para los modelos gráficos tanto de las piezas como de los entornos virtuales.

Son sobradamente conocidas las ventajas que aporta Java como lenguaje de programación. Las aplicaciones construidas adquieren un carácter multiplataforma que podrán ser ejecutadas sobre cualquier ordenador que disponga una máquina virtual. Además, dispone de bibliotecas de clases que permiten desarrollar interfaces gráficas de usuario potentes y amigables. En este sentido, otra de las razones para elegir Java ha sido que posee un API (*Application Program Interface*) de gran potencia, como es Java 3D, para el manejo de gráficos tridimensionales: proporciona las funciones para crear imágenes, visualizaciones, animaciones y aplicaciones interactivas con gráficos 3D. Esta elección ha sido fundamental para poder visualizar e interactuar con los robots creados. Además, con Java 3D se consigue la simulación de operaciones de movimiento de gran calidad.

VRML [10] es un lenguaje de modelado con el que se pueden desarrollar mundos interactivos en tres dimensiones (3-D). Estos mundos constituyen lo que se denomina la "realidad virtual", porque los usuarios pueden interactuar con los objetos de una forma similar a como lo hacen en la realidad "normal". Las posibilidades de la realidad virtual son innumerables: simulaciones educativas, nuevos

métodos de organizar la información, nuevas formas de entretenimiento, etc. La manera de acceder a estos mundos virtuales es por medio de un navegador de VRML que interprete los comandos del lenguaje, y permita adentrarse e interactuar con el mundo virtual. Estos navegadores de VRML funcionan normalmente como plug-ins de los navegadores tradicionales del Web (Explorer, Netscape, etc). Esta posibilidad abre las puertas del uso de la aplicación en un entorno abierto como es Internet. VMRL también está dirigido a ser un formato universal de intercambio para integrar gráficos 3D y multimedia. Los navegadores VRML, así como las herramientas de autor para la creación de ficheros VRML, están disponibles ampliamente para la mayoría de las diferentes plataformas.

## 5. Interfaz de usuario de JRF

En esta sección se van a presentar las principales utilidades de la herramienta JRF. La aplicación consta de una serie de ventanas, donde destaca la ventana principal en la que van a aparecer los dos componentes principales: el robot y su entorno. Hay que resaltar que en todo momento el alumno puede modificar de forma sencilla el punto de vista con el que observar el espacio de trabajo. El aspecto general de la aplicación se presenta en la Figura 2.

Para guiar al estudiante en su proceso de aprendizaje, se puede disponer de robots y entornos ya diseñados. Así, de forma inmediata, puede proceder a realizar diferentes experiencias de simulación de movimientos.

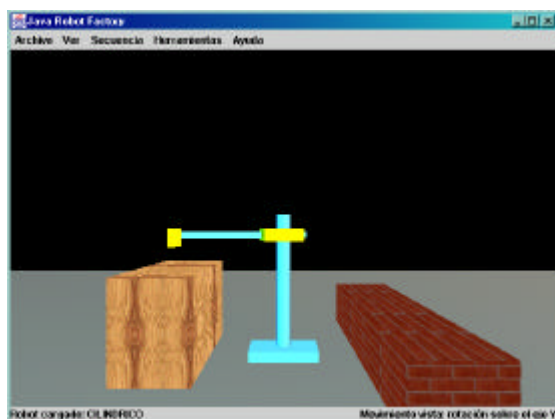


Figura 2.- Aspecto general de la aplicación

Sin embargo, una de las principales ventajas de JRF es que está enfocada al diseño de manipuladores, sin la necesidad de complejos ficheros de configuración. Para ello, se utilizan ficheros donde se definen los diferentes elementos que constituyen un robot.

Fácilmente, permite diseñar cualquier estructura robótica clásica y cualquier cadena redundante, sin limitaciones en cuanto al número de elementos considerados. Esto es debido a que se ha utilizado una estructura en árbol (parte izquierda de la Figura 3) en la que el nodo padre es el robot y los sucesores son los sucesivos elementos de la cadena cinemática. Se abre, así, la posibilidad de introducir sistemas multirobot, lo que constituye una de las posibilidades de ampliación de nuestra aplicación.

De forma sencilla, como puede observar en la Figura 3, se puede definir cada uno de los elementos de la cadena cinemática. Por una parte, la herramienta proporciona la capacidad de utilizar diferentes tipos de elementos (parte superior), incluyendo un fichero VRML donde se encuentra definido cada elemento o pieza. Por otra parte, JRF permite parametrizar la naturaleza cinemática del robot de forma semejante, aunque más intuitiva, que la propuesta por Denavit-Hartenberg. Así, se pueden especificar articulaciones de rotación y traslación donde, mediante menús desplegables, se definen la posición relativa, así como los rangos de las variables de configuración asociadas a cada uno de los elementos.

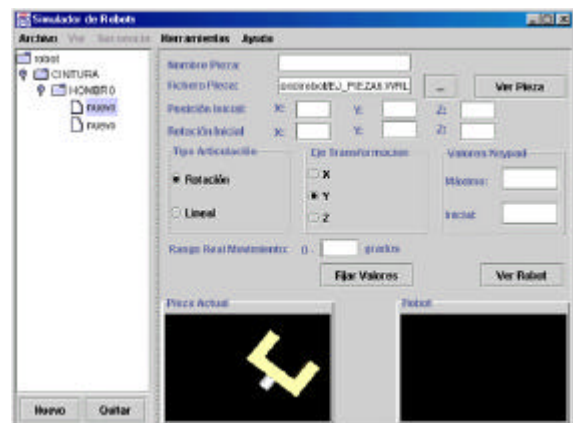


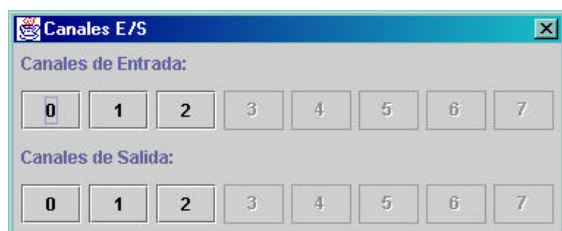
Figura 3.- Pantalla de diseño de robots

Una vez que el alumno ha definido la estructura del manipulador, se proporciona la posibilidad de simular el movimiento de cada uno de los elementos del robot. Así, de acuerdo con los rangos establecidos en la pantalla de diseño, se puede modificar el valor de la variable de configuración correspondiente (Figura 4) y visualizar el movimiento en la pantalla principal de la aplicación. En este sentido, el estudiante puede familiarizarse con el manejo del *keypad*, para llevar a cabo la etapa de entrenamiento del robot..



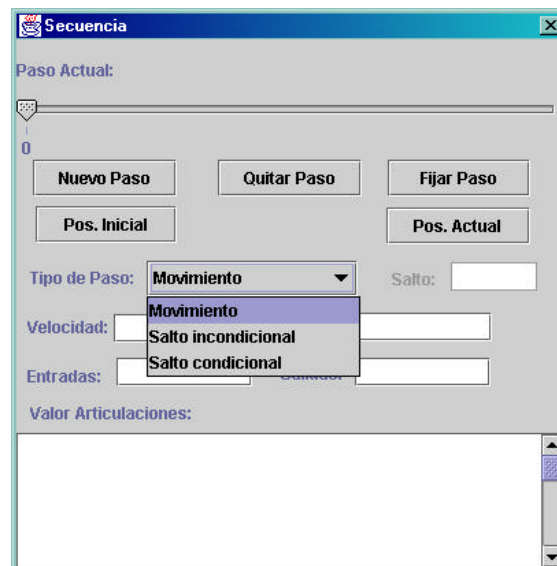
**Figura 4.- Keypad virtual**

El alumno puede comprobar la accesibilidad del efector final del robot para la operación que se pretende desarrollar. Incluyendo un fichero VMRL se puede disponer del entorno donde el robot trabaja. La utilización del formato de ficheros VRML permite utilizar un gran realismo en la simulación desarrollada.



**Figura 5. - Pantalla de canales E/S del robot**

Asimismo, se plantea el robot como un elemento más de una célula de fabricación flexible y, que su interacción con el entorno se realiza a través de una serie de líneas de comunicación. Para que el estudiante asimile esta situación, en JRF se ha dispuesto de una opción de menú donde se puede encontrar y modificar, en su caso, un conjunto de canales de E/S, como aparece en la Figura 5. De esta forma se puede incluir en la programación del robot sentencias de bifurcación basándose en el estado simulado de los canales de entrada o sentencias de activación/desactivación de los canales de salida. Esta es una de las características más importantes de los manipuladores: su capacidad de realizar, mediante programación, diferentes secuencias de movimientos de acuerdo con el estado de la célula, a través de sus canales de E/S, en la que el robot se encuentra inmerso. Se ha partido de un lenguaje de programación como el de [5] en la que se incluyen la capacidad de incorporar sentencias de control de flujo. Como se puede observar en la Figura 6, se pueden incorporar diferentes tipos de instrucción.



**Figura 6.- Edición de secuencia de movimientos**

Para finalizar, resaltar que la aplicación permite simular la ejecución de las secuencias programadas, donde un aspecto de gran utilidad es la posibilidad de incorporar diferentes entornos de trabajo. Así, en un futuro, se plantea la incorporación de algoritmos de detección de colisiones como los que se proponen en [4]. Además, la utilización de Java de forma remota, a través de Internet permitiría, como en [7], explotar en cualquier punto las ventajas de la herramienta.

## 6. Conclusiones

En este trabajo se ha presentado la herramienta JRF para complementar tanto la docencia teórica como práctica de Robótica en las diferentes enseñanzas de ingeniería. Como se ha mostrado, la utilización de herramientas para generar software multiplataforma permite disponer de una aplicación para cualquier plataforma, con las ventajas que conlleva en un entorno docente. Asimismo, la aplicación permite diseñar cualquier cadena cinemática, con lo cual el alumno puede experimentar una simulación realista del comportamiento de los robots que en un futuro pueda encontrar en el entorno industrial.

La aplicación está siendo utilizada en las prácticas de la asignatura de Robótica que se imparte con éxito dentro de diversas titulaciones de la Universidad de Salamanca. Con ello se valida la aplicación desarrollada.

## Agradecimientos

Los autores desean agradecer el apoyo parcial de la Junta de Castilla y León a través de los proyectos JOPI y JOCL.

## Bibliografía

- [1] Alvarez I., Zazo A. F. (1994) “*Simurob: A graphic Simulator for robotic sequence programming*”, *IASTED Conference on Modelling and Simulation*. Lugano, Suiza. Pp. 215-221
- [2] Corke P.I. (1996), "A Robotics Toolbox for MATLAB", *IEEE Robotics and Automation Magazine*, Volume 3(1), March 1996, pp. 24-32
- [3] Gómez J. L., Morales (1998), A. Herramienta integrada para la generación de entornos y simulación de movimientos en un robot PUMA, Proyecto Fin de Carrera. Universidad de Salamanca.
- [4] Gottschalk, S., Lin M., Manocha D. (1996). “*OBB-Tree: A Hierarchical Structure for Rapid Interference Detection*”. SIGGRAPH 96 Conference Proceedings. Pp. 171-180
- [5] Robot educativo *MA2000*. Tecquipment Co. <http://www.tecquip.co.uk>
- [6] Robot educativo *Mentor*. Italtel Co. <http://www.italtec.it>
- [7] Safaric R., Calkin D.W., Parkin R.M. and Czarnecki C.A. (1998), "Control of a Robot System via Internet", Proc IEE/IMEchE Mechatronics 98, Skovde, Sweden, September 9-11, pp 829-834.
- [8] Software de Simulación *MA9080*. Tecquipment Co. <http://www.tecquip.co.uk>
- [9] Torres, et al (1999). “ASTRO: Aprendizaje mediante simulación y teleoperación de robots”. XX Jornadas de Automatica. Salamanca. pp. 209-215
- [10] Virtual Reality Modeling Language. International Standard ISO/IEC 14772-1:1997. <http://www.vrml.org/technicalinfo/specifications/vrml97/index.htm>
- [11] <http://www.mathworks.com>