

GENERACIÓN DE TRAYECTORIAS EN UN ROBOT DELTA POR MEDIO DE SU ADAPTABILIDAD A DIFERENTES ENTORNOS

Diego Edissón Flórez Vergara
deflorezv@correo.udistrital.edu.co
20161101008

Universidad Distrital "Francisco José de Caldas", Especialización en Informática y Automática Industrial, Bogotá, Colombia.

Resumen- Este trabajo de investigación pretende simular diferentes ambientes virtuales que interactúan con un robot Delta, con el propósito de que dicho robot pueda generar diferentes trayectorias dentro de su espacio de trabajo, a través de la exploración (Observación, reconocimiento y registro) e interacción con su entorno por medio de dispositivos de adquisición de datos, que le permitan implementar un control cinemático pudiendo actualizar su ubicación con el propósito de encontrar un punto dentro de su espacio de trabajo, el cual esté libre de obstrucciones en determinado rango de medidas.

Abstract- This research work pretends to simulate different virtual environments that interact with a Delta robot, with the purpose of which said robot can generate different trajectories within its work space, through the exploration (Observation, recognition and registration) and the interaction with its environment by means of data acquisition devices that allow the implementation of a kinematic control and can update its location for the purpose of finding a point within its workspace, which is free of obstructions in a range of measures.

Index Terms- Delta robot, workspace, forward kinematics, inverse kinematics, singularities, V-REP.

I. INTRODUCCIÓN

Los entornos de simulación están teniendo en la actualidad un rol principal no solo al reducir los tiempos de desarrollo y costos, por ejemplo, con técnicas como la sistematización del software y/o hardware-in-the-loop evaluando el desempeño de los robots, pero también en la exploración de nuevos tipos de robots y aplicaciones.[1]

Entre las razones principales por las cuales la simulación robótica está teniendo nuevamente una gran importancia, se pueden encontrar [2]:

- El poder computacional de los dispositivos electrónicos ha incrementado significativamente, lo cual hace posible ejecutar algoritmos con cálculos computacionales robustos en computadoras personales, en lugar de usar hardware de propósito especial.

- El esfuerzo en la industria de video juegos por crear entornos más reales de realidad virtual, donde, la creación de dichos entornos virtuales requiere una gran cantidad de poder de procesamiento para la reconstrucción gráfica y cálculos físicos. Lo que demuestra una afinidad en los objetivos a alcanzar por los video juegos y los simuladores robóticos (la creación de un homologo virtual realista de un uso en el mundo real).

El robot Delta (Figura 1) es un manipulador paralelo, el cual consta de tres cadenas cinemáticas cerradas de configuración PR(Ps), donde (Ps) representa el espacio paralelogramo de cuatro barras con cuatro articulaciones esféricas, y donde PR representa que cada una de las cadenas cinemáticas se compone de una articulación prismática pasiva y una articulación de rotación activa que conectan una base fija a una base móvil [3].

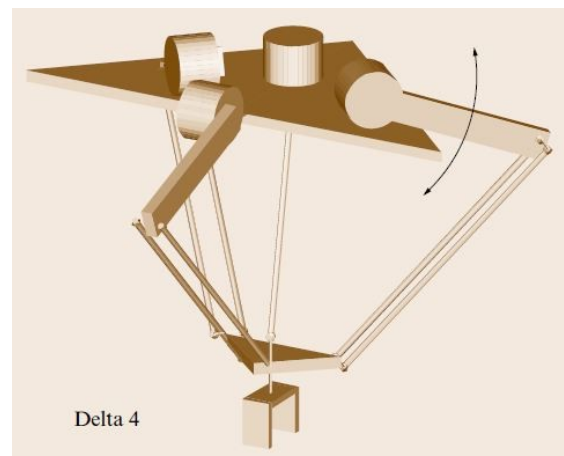


Figura 1. Representación de un Robot Delta [5].

Tal vez la estructura robótica paralela que tiene mayor aceptación, uso y desarrollo en el ámbito industrial sea el robot Delta, ya que, al contar con una relación velocidad-precisión tan elevada es ampliamente utilizado en diferentes industrias y en los diferentes procesos que son desarrollados en dichas industrias. Con los constantes desarrollos en áreas como la electrónica y/o la ciencia de los materiales se obtiene cada vez más provecho de las propiedades del robot Delta y se amplían sus campos de aplicación.

II. SIMULADORES ROBOTICOS

En la actualidad industrial y académica se pueden encontrar diversos paquetes de software para el diseño, análisis, simulación, visualización, comunicación, programación y demás actividades relacionadas con el modelamiento de un dispositivo específico o de un conjunto de dispositivos que componen un sistema físico.

Dentro del campo práctico de simulación robótica, se pueden encontrar típicamente tres tipos de paquetes de software con los que el usuario puede interactuar, dichos paquetes de software se pueden clasificar de la siguiente forma [11].

- A.** Paquetes de diseño CAD (Computer-Aided Design), estos paquetes de software tienen en común la habilidad para construir modelos de sólidos en 3D, construir ensamblajes 3D de dichos modelos, realizar sus respectivos dibujos de Ingeniería y simular movimientos.

PTC® Creo®: Paquete de software ofrecido por PTC Inc., el cual brinda un conjunto de aplicaciones para el diseño de cualquier producto que se desee por medio de diseño 2D y 3D, diseño industrial, simulación, visualización, CAM (Computer-Aided Manufacturing), diseños integrando datos de múltiples sistemas CAD, entre otras herramientas.

SolidWorks®: Paquete de software ofrecido por Dassault Systèmes SolidWorks Corp., el cual brinda herramientas de diseño sostenible 2D y 3D para crear, simular, validar, publicar, administrar y gestionar los datos, con el propósito de aumentar la productividad y agilizar la introducción de los productos en el mercado.

AutoCAD®: Paquete de software ofrecido por Autodesk Inc., el cual brinda herramientas de diseño mecánico 2D y 3D, simulación de productos, creación de mecanizado, comunicación entre diseños y creación de prototipos digitales.

- B.** Paquetes con un enfoque principalmente orientado hacia la animación para la visualización del robot, donde se desee o necesite programar el robot directamente y/o su comportamiento con el entorno con el cual va a interactuar.

GAZEBO®: Paquete de software libre de simulación robótica que ofrece un entorno de simulación preciso y eficiente por medio de un motor robusto de física, gráficos de alta calidad e interfaces gráficas para una gran variedad de robots en entornos virtuales complejos.

RoboDK®: Paquete de software para la simulación de robots industriales que brinda una amplia librería de robots y dispositivos industriales de distintas marcas comerciales, proporciona herramientas para simulación, visualización, programación offline en diversos lenguajes de programación de alto nivel.

V-REP®: Paquete de software de simulación robótica que brinda gran versatilidad e ideal en aplicaciones multi-robot, con herramientas para simulación, alto prototipado y verificación, monitoreo remoto, entre otras, con diferentes enfoques de programación en diferentes lenguajes de programación.

- C.** Paquetes con herramientas sofisticadas de simulación numérica para implementar estrategias de control a diferente nivel, que permitan controlar el comportamiento del robot por medio de la formulación de ecuaciones dinámicas de movimiento.

MATLAB®: Paquete de software ofrecido por MathWorks Inc., el cual brinda herramientas matemáticas computacionales para resolver problemas científicos y de ingeniería en el procesamiento de señales, procesamiento de imágenes, visión por computadora, control de sistemas, robótica y muchos más.

Simulink®: Paquete de software ofrecido por MathWorks Inc., basado en un entorno de diagrama de bloques, el cual proporciona un editor gráfico, librerías de bloques personalizables, solucionadores para el modelado y simulación de sistemas dinámicos, y que se encuentra integrado con MATLAB® permitiendo incorporar sus algoritmos y exportar simulaciones para un análisis de mayor profundidad.

Maplesoft®: Paquete de software ofrecido por Maple Inc., el cual brinda un amplio conjunto de herramientas matemáticas por medio de una interfaz para analizar, explorar, visualizar y resolver problemas matemáticos e ingenieriles con técnicas como modelado de sistemas, simulación, entre otras, proporcionando así una reducción en los riesgos y una rápida y alta calidad en los productos.

III. TRAYECTORIAS

Es el proceso por el cual se selecciona un movimiento y el conjunto de variables de entrada relacionados a este, partiendo de todos los posibles movimientos y variables de entrada que satisfagan las correspondientes restricciones [7].

Dependiendo del movimiento a seguir por el manipulador, en ocasiones se hace necesario especificar detalles adicionales que solamente las configuraciones iniciales y finales, por ende, una manera de incluir más detalle en una descripción de movimiento es por medio de una secuencia de puntos intermedios comúnmente conocidos como *puntos vía* [6].

La principal distinción entre las distintas técnicas para el análisis y generación de trayectorias se puede clasificar como se muestra a continuación (Figura 2) [8]:

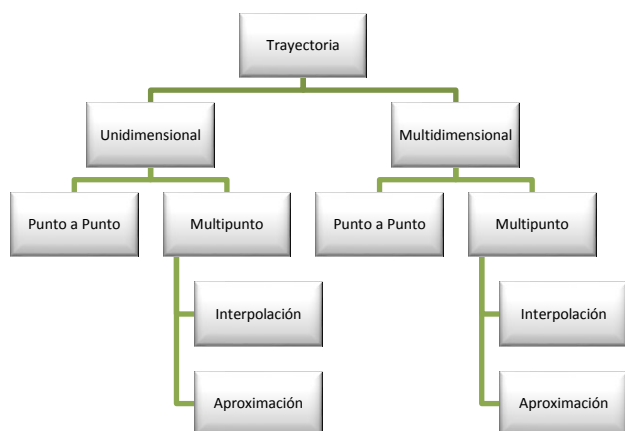


Figura 2. Clasificación de técnicas para generar trayectorias.

Trayectorias Punto a Punto

Esta metodología consiste en determinar una serie de puntos en el espacio de trabajo del manipulador, que el efector final debe atravesar independientemente de la trayectoria seguida desde su configuración inicial [9].

Trayectorias Multipunto

Esta metodología consiste en una trayectoria continua o trayectoria de trabajo, que tiene como objetivo hacer que el efector final la siga o complete lo más estrechamente posible [9].

✓ Interpolación

En este caso, la trayectoria atraviesa los distintos puntos ubicados en el espacio para ciertos valores en el tiempo (Figura 3) [8].

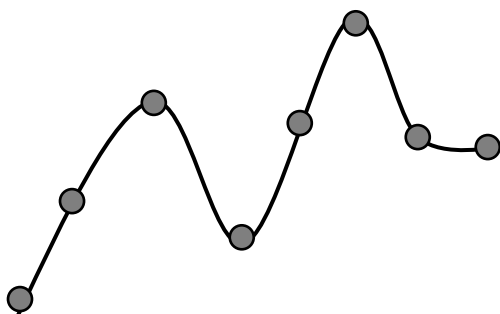


Figura 3. Interpolación de una trayectoria a través de un conjunto de puntos.

✓ Aproximación

En este caso, la trayectoria no pasa exactamente por los puntos ubicados en el espacio, pero se tiene la capacidad de asignar un error especificando cierta tolerancia (Figura 4) [8].

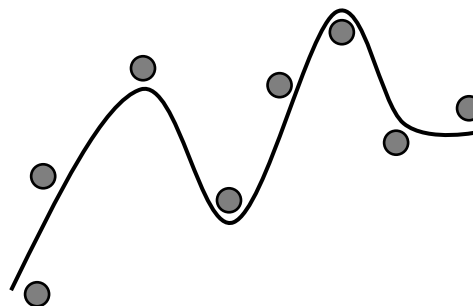


Figura 4. Aproximación de una trayectoria a través de un conjunto de puntos.

IV. SIMULACION TRAYECTORIAS

Esta sección está dedicada al análisis y ejecución de las trayectorias que se implementaran en el robot Delta a través del simulador V-rep, con el objetivo de probar el comportamiento del robot frente al algoritmo de cinemática inversa en una trayectoria punto a punto.

El objetivo de la planificación de trayectorias es el poder generar unas referencias de entrada al sistema de control de movimiento para que este garantice que el manipulador ejecutara la trayectoria planeada, generando así, una secuencia en el tiempo de los valores alcanzados, dichas referencias de entrada se obtendrán a través de unos sensores implantados en la base móvil del manipulador y cuya función es la adquisición de datos de los diferentes entornos con los que el manipulador tendrá que interactuar. Dicho sensores se clasifican en:

- ✓ **Sensor de visión:** Este sensor (Figura 5) se ubicara en la parte inferior de una forma primitiva (cylinder) la cual se vinculara a la base móvil del robot Delta, con el fin de ir visualizando la trayectoria generada a través de los obstáculos. Entre las propiedades básicas que se pueden encontrar en este tipo de sensores suministrados por el simulador robótico están los diferentes tipos de proyección, ya sea, de tipo ortográfica o de tipo perspectiva [12].

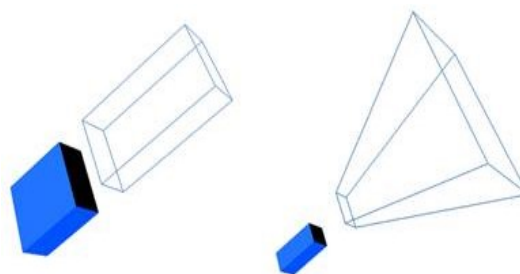


Figura 5. Tipos de operación del sensor de visión [12].

- ✓ **Sensor de proximidad:** Estos sensores (Figura 6) se incluirán dentro del entorno virtual alrededor la forma primitiva (cylinder) que estará vinculada a la base móvil del robot Delta, y tendrán como función adquirir los valores que serán enviados al controlador para que este genere la respectiva señal de control que permita ir ajustando la trayectoria del robot, corroborando a su vez, el correcto funcionamiento del control cinemático del robot Delta y sus diferentes componentes, donde los obstáculos presentes se combinarán entre sí para poder diseñar obstáculos de mayor complejidad, tipo laberinto. Estos sensores son obtenidos a partir de las librerías de componentes con las que cuenta el simulador V-REP [12].

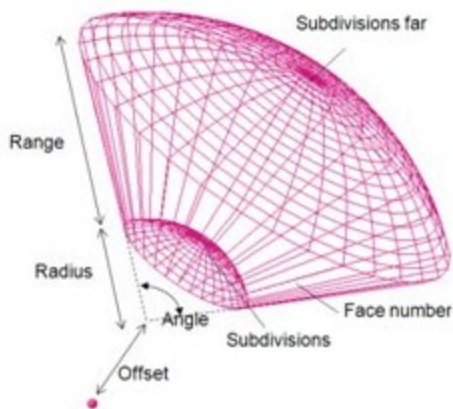


Figura 6. Sensor de proximidad de tipo volumétrico [12].

Para el diseño de los ambientes de simulación con los que el robot Delta debe interactuar se debieron tener en cuenta diferentes propiedades del mecanismo en sí, como fueron su espacio de trabajo, sus singularidades, el alcance y resolución de los sensores añadidos al mecanismo y la configuración de las propiedades de las articulaciones que componen el mecanismo, el cual se escogió en modo de cinemática inversa.

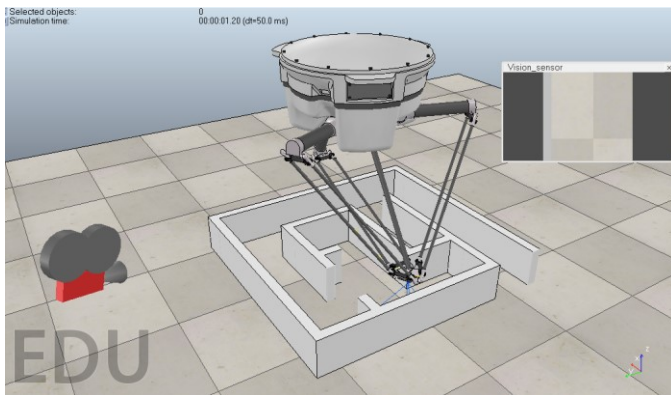


Figura 7. Ejecución de trayectoria en ambiente de simulación1.

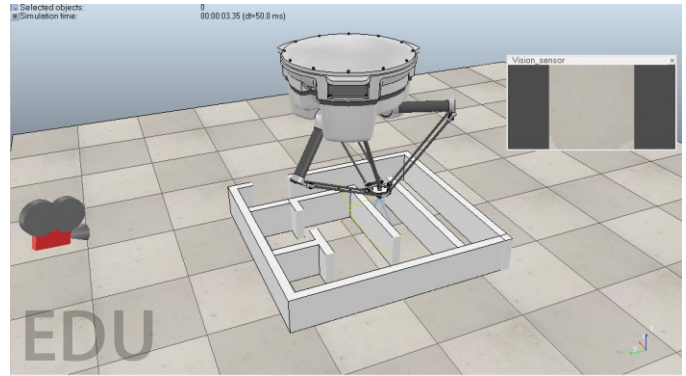


Figura 8. Ejecución de trayectoria en ambiente de simulación2.

V. CONCLUSIONES

Se corroboró el adecuado comportamiento del robot Delta al ejecutar diferentes trayectorias especificadas en el ambiente virtual V-REP, por medio de la configuración de la propiedad *inverse kinematics mode* que especifica el modo de simulación de las articulaciones que componen el mecanismo en general.

Para la ejecución de trayectorias se trabajó con un criterio de velocidad constante y un buffer de tiempo de 5 minutos para cada ambiente de simulación, esto debido a que la adquisición de datos de los sensores y el seguimiento de la trayectoria por parte del robot Delta tuvieron un lapso de tiempo mayor para la reacción del mismo.

Durante ciertos intervalos en la ejecución de las trayectorias por parte del robot Delta, el comportamiento del robot pasaba de simular un proceso de interpolación a simular un proceso de aproximación, generando un error de posición.

REFERENCIAS

- [1] Ando, Noriaki., Balakirsky, Stephen., Hemker, Thomas., Reggiani, Monica., Von Stryk, Oskar., 2010. Simulation, Modeling and Programming for Autonomous Robots. Springer.
- [2] Reckhaus, Michael., Hochgeschwender, Nico., Paulus, Jan., Shakhimardanov, Azamat., Kraetzschmar, Gerhard. An Overview about Simulation and Emulation in Robotics. Bonn-Rhein-Sieg University of Applied Sciences. Computer Science Department. 2010
- [3] Liu, Xin-Jun., Wang, Jinsong., 2014. Parallel Kinematics: Type, Kinematics, and Optimal Design. Springer.
- [4] Merlet, J.-P., 2006. Parallel Robots, 2nded. Springer.
- [5] Siciliano, Bruno., Khatib, Oussama., 2008. Springer handbook of robotics. Springer.
- [6] Craig, John J., 2006. Robótica, 3rd ed. Pearson.

- [7] Nof, Shimon Y., 1999. Handbook of Industrial Robotics, 2nded. Wiley.
- [8] Biagiotti, Luigi., Melchiorri, Claudio., 2008. Trajectory Planning for Automatic Machines and Robots, Springer
- [9] Kelly, R., Santibáñez, V., Loría, A., 2005. Control of Robot Manipulators in Joint Space. Springer.
- [10] Siciliano, Bruno., Sciavicco, Lorenzo., Villani, Luigi., Oriolo, Giuseppe., 2009. Robotics: Modelling, Planning and Control. Springer.
- [11] Kurfess R. Thomas., 2005. Robotics and Automation Handbook, 1sted. CRC Press.
- [12] [Online] <http://www.coppeliarobotics.com/helpFiles/>