

Heurísticas para el Ajuste de Algoritmos de Control de Plataformas Robóticas de Movimiento en Simuladores

Sergio Casas, Cristina Portalés*, José V. Riera, Marcos Fernández

Instituto de Robótica y Tecnologías de la Información y la Comunicación, Universitat de València, C/ Catedrático José Beltrán 2, 46980, Paterna, Valencia, España.

Resumen

Diversos tipos de plataformas robóticas son empleadas habitualmente para la generación de claves gravito-inerciales en simuladores. Además del control de los actuadores, dichas plataformas deben ejecutar complejos algoritmos de control conocidos como algoritmos de *washout*, que deben ser ajustados para que el movimiento generado sea similar al simulado. El ajuste de dichos algoritmos es complejo por el elevado número de parámetros que poseen. Además, dicho ajuste se ha venido realizando tradicionalmente de modo manual mediante evaluaciones subjetivas. En este trabajo, los autores proponen un método automático de ajuste basado en optimización heurística, métricas objetivas, y simulación de la plataforma robótica para conseguir realizar el ajuste de manera más rápida. Se valida la corrección de las soluciones, y se comparan diversas técnicas de optimización, para concluir que la técnica más apropiada es la de los algoritmos genéticos.

Palabras Clave:

Plataformas de movimiento, heurísticas, simuladores, algoritmos de control, ajuste de parámetros, robótica, optimización.

1. Introducción

El objetivo de todo simulador de vehículos es proporcionar al usuario una sensación de pertenencia a un entorno virtual alternativo. Para ello, se deben estimular, del modo más eficaz posible, las claves sensoriales que hacen posible que el usuario sienta como cierta esa pertenencia al mundo virtual. Aunque la mayoría de simuladores suelen centrarse en las claves visuales y sonoras, existen otras que también deben ser tenidas en cuenta. Entre las claves extra-audiovisuales más importantes se encuentran las claves gravito-inerciales, relacionadas con la percepción del movimiento. Este tipo de claves se estimulan habitualmente en simuladores de vehículos mediante la construcción de plataformas robóticas de movimiento, sobre las que se suele situar al usuario de la simulación. Estas plataformas están dotadas de actuadores que permiten desplazarlas y orientarlas, pero siempre dentro de unos límites (Figura 1). Para el control de los movimientos de la plataforma se diseñan unos algoritmos de control específicos conocidos como algoritmos de generación de claves gravito-inerciales (en inglés *Motion Cueing/Drive Algorithms – MCA/MDA*) (Schmidt and Conrad 1969), también conocidos como

algoritmos de *washout*. Estos algoritmos toman como entrada el estado físico del vehículo simulado y generan como salida la pose deseada para la plataforma robótica en forma de grados de libertad (GdL) traslacionales y rotacionales, que es transformada en consignas para el control de los actuadores (Garrett and Best 2010).

En un nivel de abstracción inferior se situarían los algoritmos de control de los propios actuadores, cuya solución no se analiza en este trabajo. La Figura 2 muestra el esquema completo de generación de claves gravito-inerciales con manipuladores robóticos. El sistema de simulación debe implementar un módulo físico, que puede simular diferentes tipos de vehículos (Slob 2008, Cossalter, Lot et al. 2011, Casas, Rueda et al. 2012) y que interactúa con el piloto a través de una reproducción de los mandos del vehículo. Éste módulo calcula un estado físico simulado, que es el que se desea generar sobre la plataforma robótica. Ésta, mediante un algoritmo de control (MCA) generará los comandos necesarios para que la plataforma se mueva provocando sobre el usuario un estado físico similar al simulado por el módulo físico. Para ello, genera la pose deseada para la plataforma de movimiento y, tras comprobar que no se exceden los límites físicos, se envían los comandos deseados para los actuadores, empleando las ecuaciones de la cinemática inversa del manipulador. Opcionalmente, aunque no es lo habitual, se puede calcular, mediante la cinemática directa, el estado físico real de la plataforma y comprobar que es el deseado o utilizarlo como entrada de algoritmo MCA.

Aunque este tipo de claves ha sido incluido en simuladores desde hace más de 50 años, se ha avanzado menos en este campo

* Autor en correspondencia.

Correos electrónicos: sergio.casas@uv.es (Sergio Casas),
cristina.portales@uv.es (Cristina Portalés),
jorieo@robotica.uv.es (José V. Riera), marcos.fernandez@uv.es
(Marcos Fernández),
URL: artec.uv.es (Sergio Casas)

que en otros aspectos de la simulación, como el de la generación de claves audiovisuales.



Figura 1: Plataforma robótica de movimiento. Fuente: Cobra Simulation®.

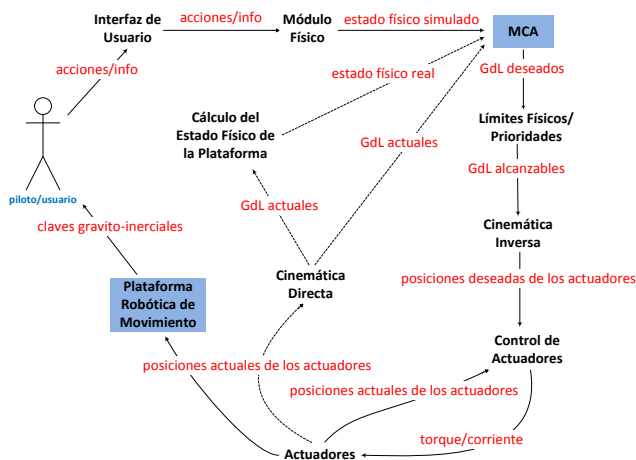


Figura 2: Esquema de generación de claves gravito-inerciales en simuladores.

Los autores identifican tres razones fundamentales para ello. La primera es la naturaleza del problema. Es un problema complejo, cuya solución depende de múltiples factores, entre ellos factores humanos difíciles de medir (y todavía parcialmente desconocidos) relacionados con la percepción del movimiento. La segunda es la falta de un criterio de evaluación para poder comparar diferentes soluciones. Y la tercera, la multitud de parámetros que se deben ajustar en los distintos algoritmos de *washout* para conseguir que se comporten de la forma esperada (Grant and Reid 1997).

La evaluación de este tipo de algoritmos está íntimamente relacionada con el ajuste de parámetros, ya que éstos se deben ajustar en base a un criterio de evaluación, y a su vez, el resultado de la evaluación depende en gran medida del ajuste realizado.

1.1. Algoritmos de Washout/MCA

El objetivo de estos algoritmos es generar las claves gravito-inerciales necesarias para que el piloto sienta mayor inmersión en el manejo del vehículo simulado. Para ello, los algoritmos MCA deben proporcionar siempre la pose deseada para la plataforma de

movimiento. Esta pose se especifica siempre en forma de 6 GdL, aunque si la plataforma tiene menos de 6 GdL, algunos no se emplearán. Las entradas del simulador, sin embargo, pueden variar de un algoritmo a otro, aunque casi siempre se corresponden con la aceleración lineal (o fuerza específica) y la velocidad angular del vehículo simulado. Estas magnitudes son las que es capaz de sensorizar el sistema vestibular humano, por lo que es lógico emplearlas como entradas. Opcionalmente se pueden utilizar otras, como la posición o la orientación del vehículo simulado, aunque no es habitual. El objetivo fundamental de cualquier algoritmo de *washout* es reproducir el movimiento que está experimentando el vehículo simulado, intentando al mismo tiempo que la plataforma robótica no alcance sus límites físicos (porque si lo hace, el control deberá detenerla). Lógicamente, ambos objetivos son contrapuestos, por lo que rara vez es posible reproducir el movimiento original.

Existen numerosos tipos de algoritmos de *washout* que se aproximan al problema de manera diferente (Stahl, Abdulsamad et al. 2014) aunque todos ellos hacen uso de una serie de principios básicos comunes. La primera idea es reproducir los mismos movimientos que el vehículo virtual, pero a menor escala, reduciendo su intensidad en un factor prefijado. La segunda idea, más sofisticada, es la aplicación de filtros pasa-alta a las entradas del algoritmo (normalmente aceleraciones y velocidades angulares) que eliminen los movimientos de baja frecuencia que son los responsables de los desplazamientos largos y sostenidos (Reymond and Kemeny 2000). Al eliminar las aceleraciones o velocidades de baja frecuencia, perdemos fidelidad en la reproducción del movimiento, pero nos aseguramos que la plataforma de movimiento vuelva eventualmente a su posición neutral. La tercera idea consiste en aprovechar la ilusión somatográfica (MacNeilage, Banks et al. 2007), para hacer percibir al cerebro las aceleraciones sostenidas (eliminadas por la aplicación de filtros pasa-alta) mediante pequeñas inclinaciones que modifican la dirección relativa del vector gravedad con respecto al sistema de referencia del aparato vestibular. Esta idea se implementa aplicando filtros pasa-baja a las aceleraciones lineales para inclinar ligera y lentamente la plataforma en los ejes lateral y longitudinal, en función del valor de estas aceleraciones filtradas. A esta técnica se le conoce como *tilt coordination* (coordinación angular). Para una explicación más detallada de estas técnicas y de algunos de los tipos de algoritmos MCA más utilizados, el lector puede consultar (Nahon and Reid 1990).

1.2. Evaluación los Algoritmos de Washout/MCA

Dado que por las restricciones físicas de los actuadores (tanto de espacio como de fuerza), no es posible generar un movimiento idéntico al original, se define la fidelidad (Sinacori 1977) gravito-inercial de un simulador como la capacidad de éste para generar las sensaciones de movimiento que se percibirían en una situación real con un vehículo. La evaluación de un algoritmo de *washout* es, por tanto, un método para obtener un valor de fidelidad para el simulador. No existen demasiadas referencias en la literatura sobre la evaluación de este tipo de algoritmos, y lo habitual es que el criterio de evaluación sea simplemente la opinión de uno o varios pilotos. Desde un punto de vista taxonómico, la evaluación se puede hacer básicamente de dos maneras: objetiva (cuantitativa) o subjetivamente (cualitativamente). La evaluación objetiva puede, además, ser directa o indirecta.

En la evaluación objetiva directa lo que se evalúa son ciertas magnitudes físicas que se corresponden con la ejecución del

Download English Version:

<https://daneshyari.com/en/article/8050537>

Download Persian Version:

<https://daneshyari.com/article/8050537>

[Daneshyari.com](https://daneshyari.com)