

CAPÍTULO 5

Resultados.

En este capítulo se presentan los resultados obtenidos con el algoritmo de optimización de este trabajo.

5.1 Resultados obtenidos.

El simulador realizado para este proyecto tiene la posibilidad de crear distintos ejemplos. El usuario del sistema decide el número de grados de libertad que debe tener el simulador, también decide que tan congestionado está el ambiente donde el robot simulador trata de alcanzar sus puntos. El número de puntos también es asignado por el usuario. Coloca los objetos en el espacio y decide su tamaño. De esta manera podemos obtener ejemplos de diversos tipos (Figura 5.1).

Al parecer, las soluciones obtenidas por el simulador fueron buenas, sobre todo si consideramos que la otra solución sería el actual proceso de programación manual del robot, expuesto en el Capítulo 1. Con el simulador, podríamos pensar en menor tiempo invertido para la programación del robot y en un recorrido más "óptimo" de los puntos. Sin embargo, pudimos observar lo siguiente:

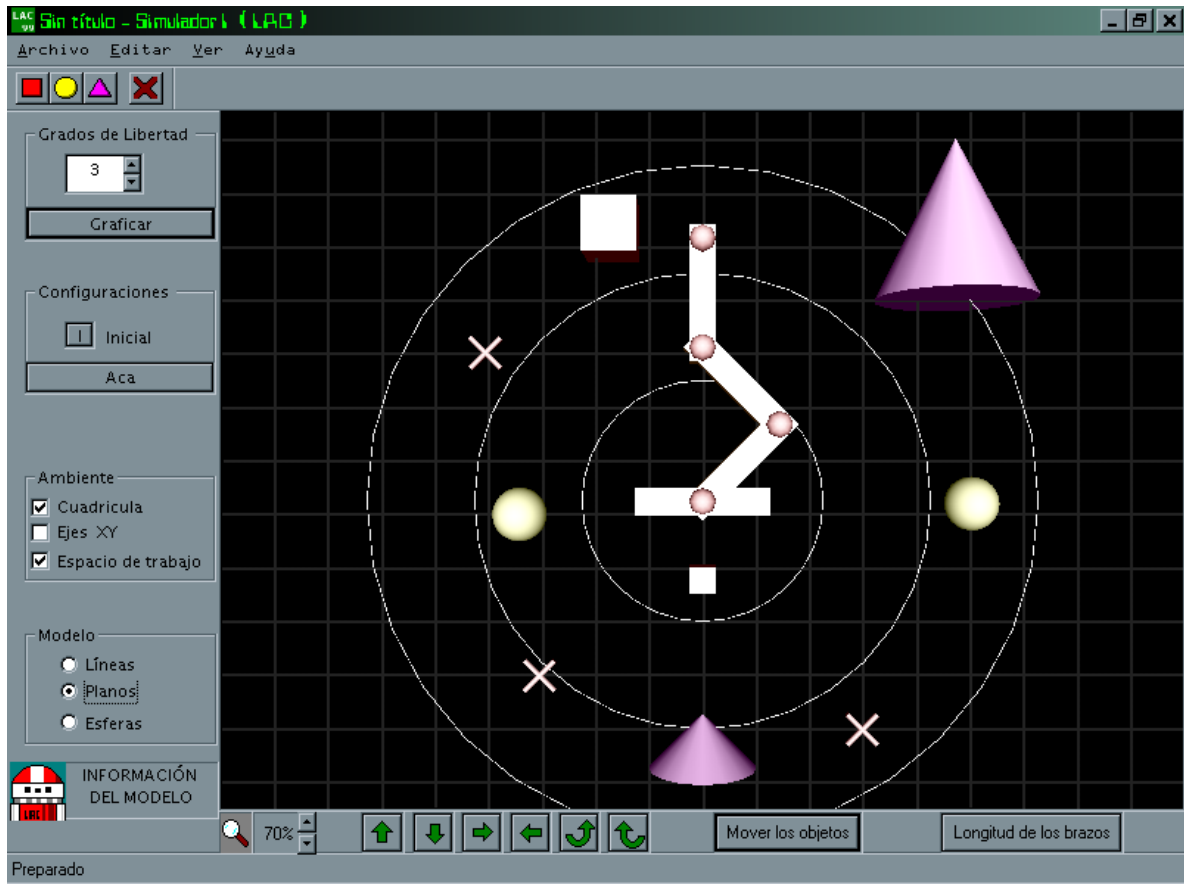


Figura 5.1 Ejemplo de problema.

En algunas ocasiones la solución era encontrada sin ningún contratiempo, pero hubo ejemplos en que se presentaron ciertos problemas. Estos fueron reflejados al momento de ejecución del algoritmo. El principal de todos ellos ocurre cuando en su recorrido, el simulador no logra tocar todos los puntos.

Este tipo de problema ocurre en casos en que los puntos están colocados en posiciones complicadas para ser alcanzados por el robot, ya sea porque existan demasiados obstáculos cerca de ellos o también cuando están en los límites externos de su espacio de trabajo

Nosotros pensamos que estos problemas no tienen que ver con el algoritmo de optimización, sino más bien con los algoritmos de robótica. En otras palabras, supongamos que el punto deseado está en una posición difícil de alcanzar para el simulador por lo que la cinemática inversa no logra resolver el problema de encontrar una configuración que le permita al robot alcanzar dicho punto. El algoritmo de optimización para ese punto tendría cero número de soluciones y el costo de la configuración sería también de cero. El algoritmo de optimización entonces, devolverá esta solución como la del mejor costo y entonces el recorrido siempre se hará a un mismo punto.

Es cierto que para disminuir la posibilidad de no encontrar una solución en la Cinemática Inversa, se puede aumentar el número de landmarks, pero esto también ocasiona que el problema sea demasiado complejo y lento para hallar una solución.

En conclusión podemos decir que el algoritmo no es erróneo y proporciona una posible solución al problema de optimización de trayectorias de puntos.

Además de que deja abierta la posibilidad de llevar a cabo un mejor desarrollo para un algoritmo con estas características.