

Control borroso de la fuerza del efector de un Brazo Robot

Bautista Blasco, Susana: subautis@fdi.ucm.es
Garmendia Salvador, Luis: lgarmend@fdi.ucm.es
Departamento de Ingeniería del Software e Inteligencia Artificial
Universidad Complutense de Madrid

RESUMEN

La lógica fuzzy en los últimos años ha surgido como una herramienta importante para el control de subsistemas y procesos industriales complejos y otros sistemas expertos. La lógica convencional, bivaluada, trabaja con predicados nítidos, verdaderos o falsos. El uso de lógica fuzzy permite trabajar con lógica multivaluada, permitiendo modelar incertidumbre mediante distintos grados de certeza en los predicados, utilizando para ello, variables lingüísticas y cuantificadores borrosos.

Son muchas las aplicaciones que han usado lógica fuzzy exitosamente, como por ejemplo, lavadoras, equipos de aire acondicionado, aspiradores, tostadores, televisiones, cajas de velocidades automáticas... la lista muestra distintas áreas de aplicación y podríamos seguir enumerando multitud de ejemplos.

En el siguiente trabajo se describe un método simple para aplicar lógica fuzzy en un sistema de control determinado. Es un brazo robot implementado en Matlab [1], en el que se determina a través de lógica fuzzy la fuerza del efector de la pinza que tiene que hacer para poder levantar o no un determinado objeto.

Palabras claves:

Lógica fuzzy; control; brazo robot; conjuntos difusos; Matlab;

1. INTRODUCCIÓN

Un controlador lógico difuso, contiene un algoritmo que es capaz de convertir una estrategia de control lingüística en una estrategia de control automático. Con la lógica difusa se pueden diseñar aplicaciones para que las máquinas respondan con mayor inteligencia a la imprecisión de la información y a las condiciones del mundo exterior, con lo que se busca imitar el comportamiento humano. La creación de una máquina con lógica difusa, es forjar un sistema experto, en donde el comportamiento de la máquina, va a estar basado totalmente en el conocimiento del experto o de la persona que aporta sus conocimientos empíricos para el funcionamiento de ésta. El conocimiento del experto es el conocimiento empírico de cómo controlar el fenómeno, sin conocer ningún modelo del sistema a controlar.

El introductor de la lógica difusa fue Lotfi Zadeh [8], un profesor matemático iraní residente en los Estados Unidos que en 1965 en la Universidad de California en Berkeley, Zadeh publicó un artículo llamado “Fuzzy Sets”, donde presentó formalmente la teoría de conjuntos difusos.

Lotfi Zadeh expande la idea de los estados clásicos de lógica Booleana a los que nombró estados difusos utilizando el concepto de grado de pertenencia. A diferencia de la lógica Booleana, la lógica difusa generaliza la lógica clásica y la lógica multivaluada, por lo que se considera un rango de grados de valor entre cierto y falso, en el intervalo $[0, 1]$. La lógica difusa define grados de relación y grados de verdad de algo, incluso algunas cosas pueden ser parcialmente verdaderas y parcialmente falsas.

En un sistema difuso, un pequeño cambio en la variable de entrada, da como resultado un pequeño cambio en el funcionamiento del sistema, por lo que el control difuso, produce comportamientos suaves.

Cuando la teoría de los conjuntos difusos se aplica para definir y resolver problemas de control se obtienen los denominados controladores difusos o lingüísticos. El control difuso fue la primera aplicación de la lógica difusa a la resolución de problemas reales (se aplicó al control de una planta de concreto en el año 1980). En el área del control la lógica difusa ha permitido obtener resultados espectaculares, resultados que han avalado la teoría. Si se hace un razonamiento con lógica difusa dentro de un sistema experto que modela la información con conjuntos difusos se

obtiene una expresión también difusa, siendo usual tener que asignarle posteriormente una interpretación lingüística que puede ser útil para la toma de decisiones. En algunos casos se requiere defuzzificar la salida si la decisión debe ser nítida, produciendo una pérdida de información. En cambio, cuando se realiza un razonamiento difuso para controlar un proceso, el resultado buscado es de una cualidad completamente diferente, ya que lo que ha de generarse es un conjunto de valores numéricos precisos que, una vez traducidos a las magnitudes físicas correspondientes: presión, voltajes, corrientes, etc., han de introducirse a los actuadores de control.

La necesidad de utilizar lógica difusa para calcular la fuerza del efector de un brazo robot, viene dada debido a la cantidad de información con incertidumbre que manejamos a la hora de clasificar e identificar los objetos con los que trabajamos, y la mayor naturalidad que esto aporta. Un objeto es rugoso, o muy rugoso, no es una información que podamos verificar con un simple verdadero o falso, si no que hay que determinar con que grado de pertenencia el objeto es rugoso y/o muy rugoso.

1.1. Toolbox fuzzy de Matlab

En Matlab existen multitud de utilidades, toolboxes de control específicas. Entre ellas se encuentra la toolbox fuzzy, que permite definir un controlador difuso. Se accede al editor desde línea de comando escribiendo la palabra reservada *fuzzy* o bien desde el menú de inicio *Start* de Matlab.

A través de este editor, se pueden modificar los métodos de los operadores lógicos ‘and’ y ‘or’, los métodos de implicación, de agregación y de defuzzificación. Permite elegir el modelo a usar Sugeno o Mamdani, a través de las distintas opciones de la barra de menús. Se pueden agregar variables de entrada y de salida, así como definir sus conjuntos borrosos, eligiendo rango, forma y parámetros.

Las reglas de control del modelo, se editan a través de un editor específico, en el que se van seleccionando los distintos valores de las variables de entrada y de salida.

Para poder implementar el controlador es necesario guardarlo, para poder importar el archivo al workspace, para que luego Matlab lo pueda reconocer y pueda ser usado por cualquier aplicación que lo necesite, o por Simulink. Si se desea trabajar con un modelo ya guardado se debe importar desde el menú *fuzzy* primero y luego exportarlo al workspace.

2. BRAZO ROBOT

El sistema de control que se presenta, es un brazo robot, simulado en Matlab, en el que se aplica lógica fuzzy para calcular la fuerza del efector de la pinza y determinar si el brazo puede levantar el objeto generado, dependiendo de las características del mismo, el brazo lo levantará, o bien no lo levantará porque se le desliza de la pinza con la fuerza calculada o bien no lo levanta porque lo rompería.

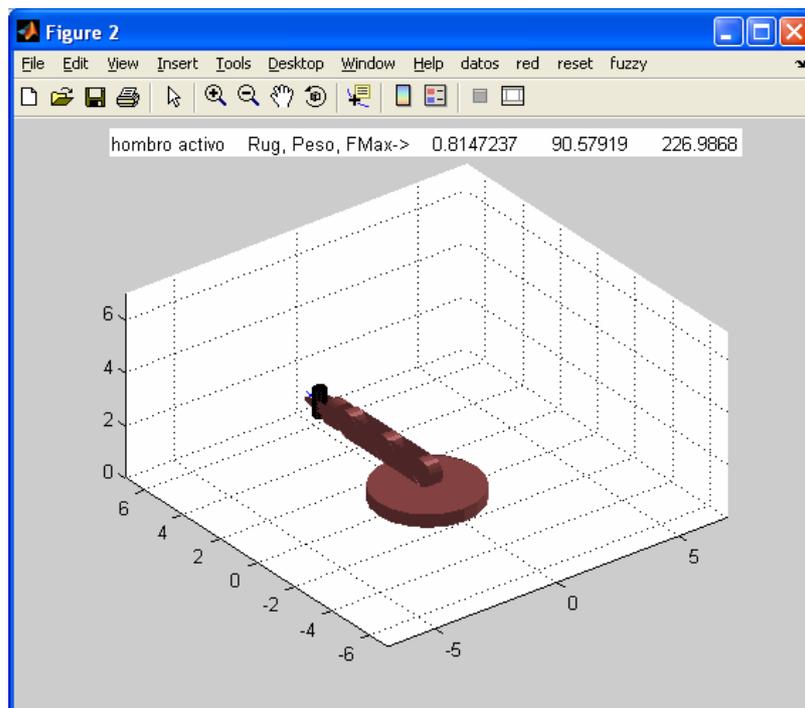


Figura 1: Brazo Robot

Para llevar a cabo la simulación, se implementa un controlador difuso, en el que cada variable del modelo se descompone en un conjunto de regiones difusas: *conjuntos difusos*. Las variables de entrada al controlador, son las que representan las características del objeto: *rugosidad* definida a través de cuatro conjuntos difusos (muy rugoso, rugoso, liso y muy liso), *peso* definido con cuatro conjuntos difusos (muy pesado, pesado, ligero y muy ligero) y *fragilidad* que es booleana. La salida del controlador es la *fuerza del efector* que esta definida con cinco conjuntos difusos.

En la siguiente figura, Figura 2, se pueden ver los conjuntos difusos definidos para las variables de entrada, y donde se encuentra el objeto generado en cada uno de

esos conjuntos difusos, para poder aplicar una regla de control y calcular la fuerza del efector correspondiente a la variable de salida.

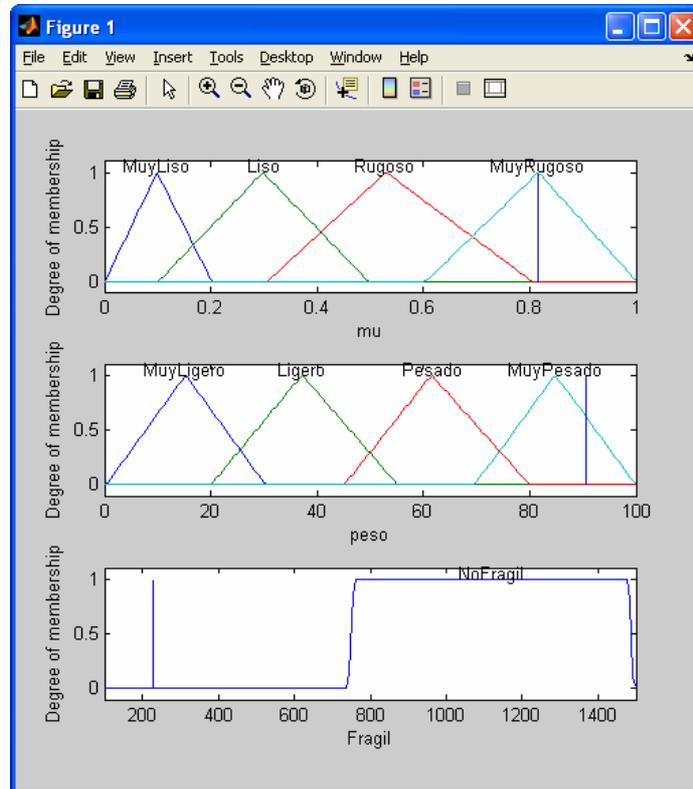


Figura 2 : Conjuntos difusos para las etiquetas lingüísticas de rugosidad, peso y fragilidad

La definición de los distintos conjuntos difusos, tiene un estudio físico, para determinar la rugosidad a partir del coeficiente de rozamiento, ya que cuando mayor es el coeficiente de rozamiento, más rugoso es el objeto, el peso es directamente proporcional y la fragilidad se ha considerado como una variable nítida, considerando los objetos frágiles o no, a partir de un límite de fuerza que aguanta sin romperse. Con estas características se calcula la fuerza para levantar o no el objeto generado.

El modelo está especificado con treinta y dos reglas de control (if-then), dependiendo del valor de las variables de entrada, para calcular la variable de salida.

La *toolbox* de Fuzzy Logic de Matlab, implementa *Mamdani* [7] con las siguientes características: conjunción (mínimo), disjunción (máximo), implicación (mínimo) y agregación (máximo). La salida de una regla difusa es un conjunto difuso, y para llevar a cabo la defuzzificación y obtener un valor concreto, utiliza el método del

Centroide [3]. Estas características se pueden ver en la Figura 3, que corresponde con el editor de Matlab.

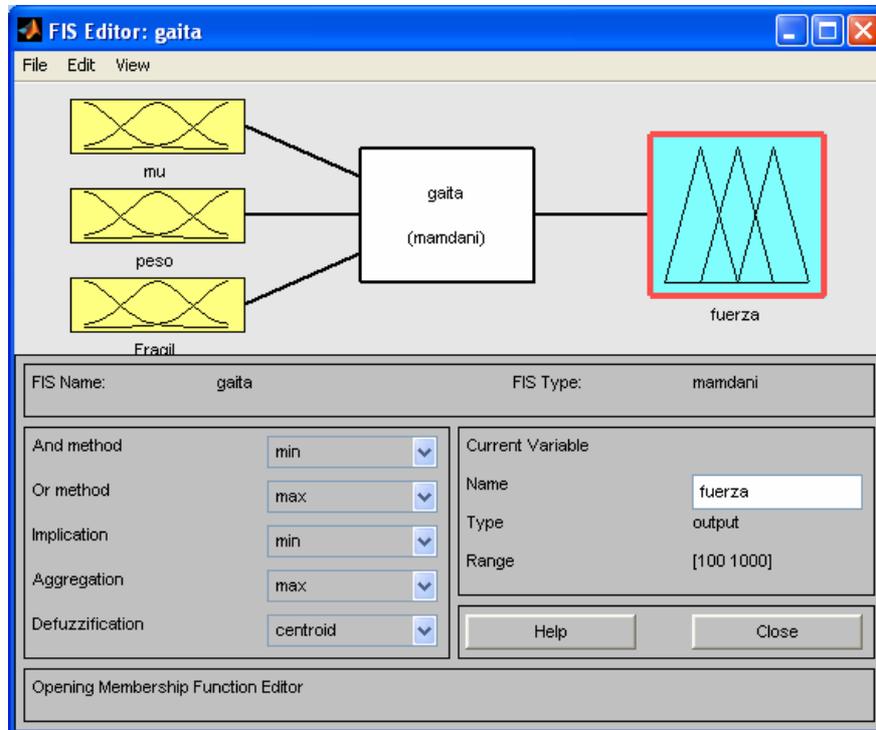


Figura 3: Editor del controlador fuzzy

Algunas de las 32 reglas de inferencia utilizadas para el aprendizaje de la fuerza de agarre a partir de premisas con incertidumbre, se muestran en la Figura 4.

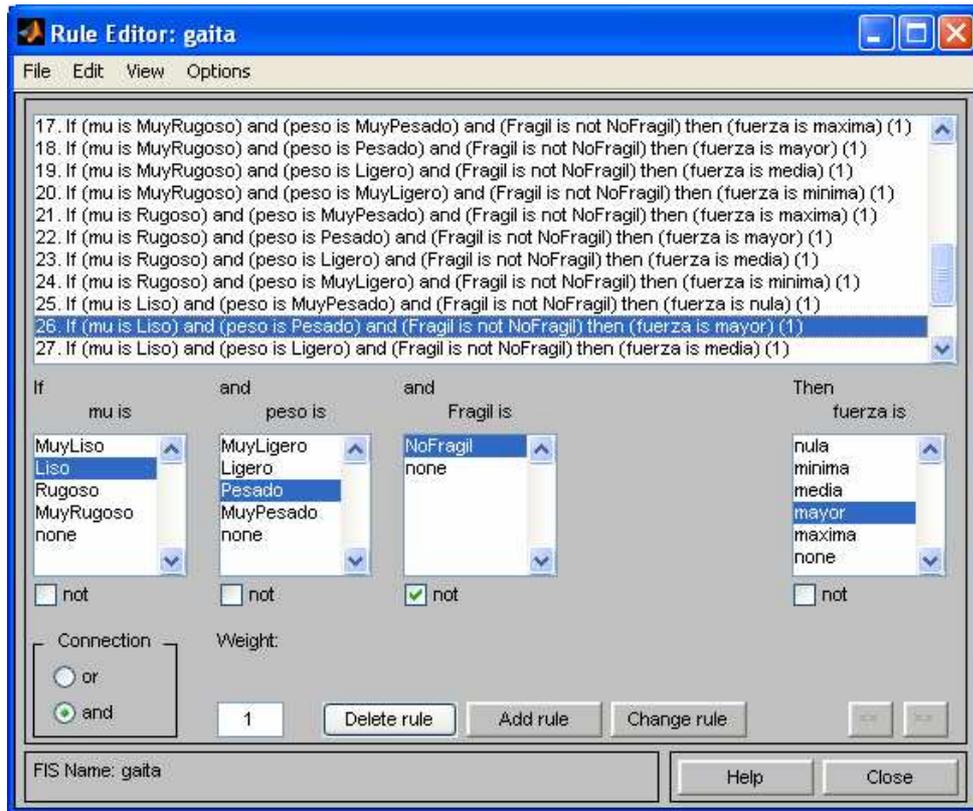


Figura 4: Reglas de inferencia

Ejemplo 2.1

En el siguiente ejemplo se muestra el funcionamiento del sistema para un objeto determinado generado con las siguientes características: rugosidad 0.81, peso 90.57 y no frágil. A partir de estos valores, se calculan los grados de pertenencia de los conjuntos borrosos de la Figura 2, que son las premisas de las reglas de inferencia de la Figura 4, y se obtienen los grados de pertenencia para los conjuntos borrosos de la salida del sistema que modelan la fuerza del efector (desde el conjunto borroso fuerza nula, mínima, media, mayor y hasta la fuerza máxima), que se observan en la Figura 5.

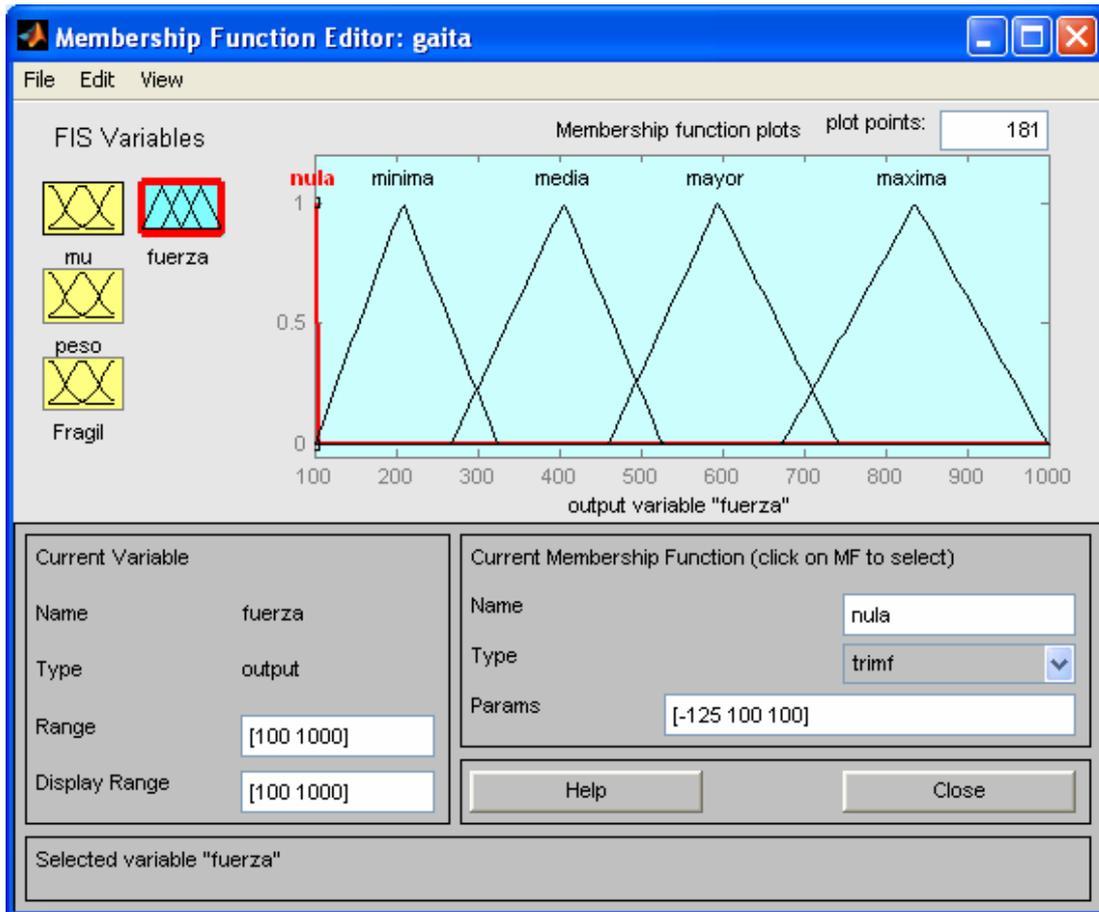


Figura 5: Conjuntos borrosos que modelan la fuerza del efector que aprende el sistema

Usando el método de defuzzificación del centroide, se aprende que la fuerza a aplicar es de 226.98 Newtons, por lo que el sistema decide no coger el objeto porque lo rompería, como se muestra en la Figura 6.

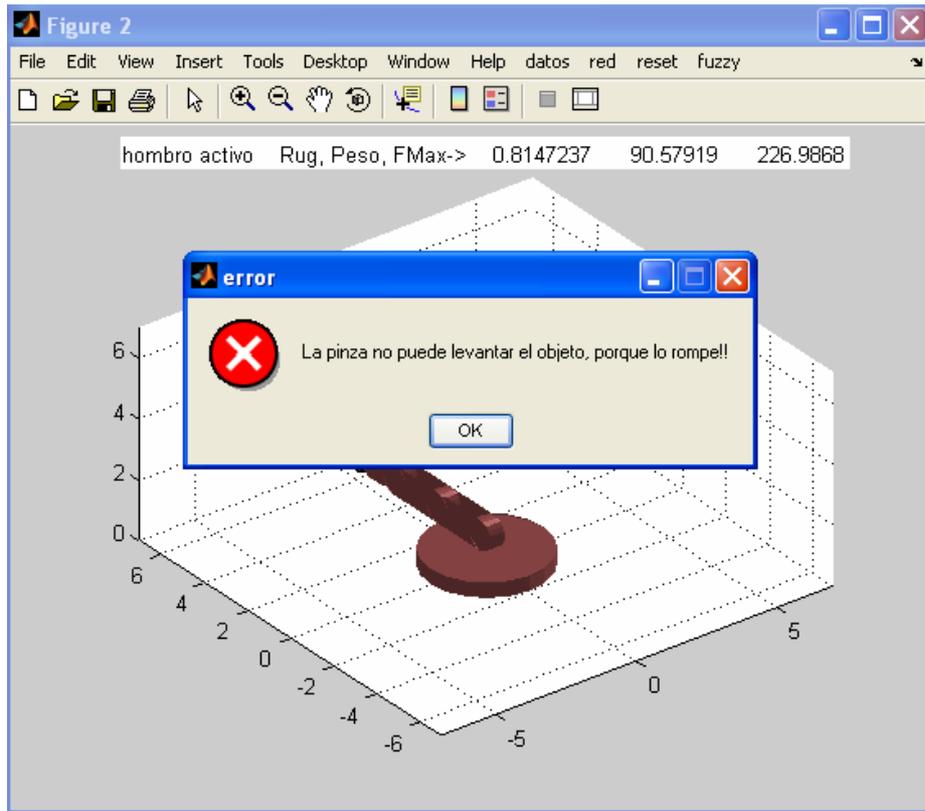


Figura 6: Ejemplo en que el brazo rompería el objeto

Ejemplo 2.2

En este otro ejemplo se muestra el funcionamiento del sistema para un objeto generado con las siguientes características: rugosidad 0.035, peso 84.91 y frágil. A partir de estos valores, igual que en el ejemplo anterior, se calculan los grados de pertenencia de los conjuntos borrosos de la Figura 2, que son las premisas de las reglas de inferencia de la Figura 4, y se obtienen los grados de pertenencia para los conjuntos borrosos de la salida del sistema que modelan la fuerza del efector (desde el conjunto borroso fuerza nula, mínima, media, mayor y hasta la fuerza máxima), que se observan en la Figura 5.

Usando el método de defuzzificación del centroide, se aprende que la fuerza a aplicar es de 1033.99 Newtons, por lo que el sistema decide no coger el objeto porque se le deslizaría, el brazo se levanta, pero no levanta el objeto, como se muestra en la Figura 7.

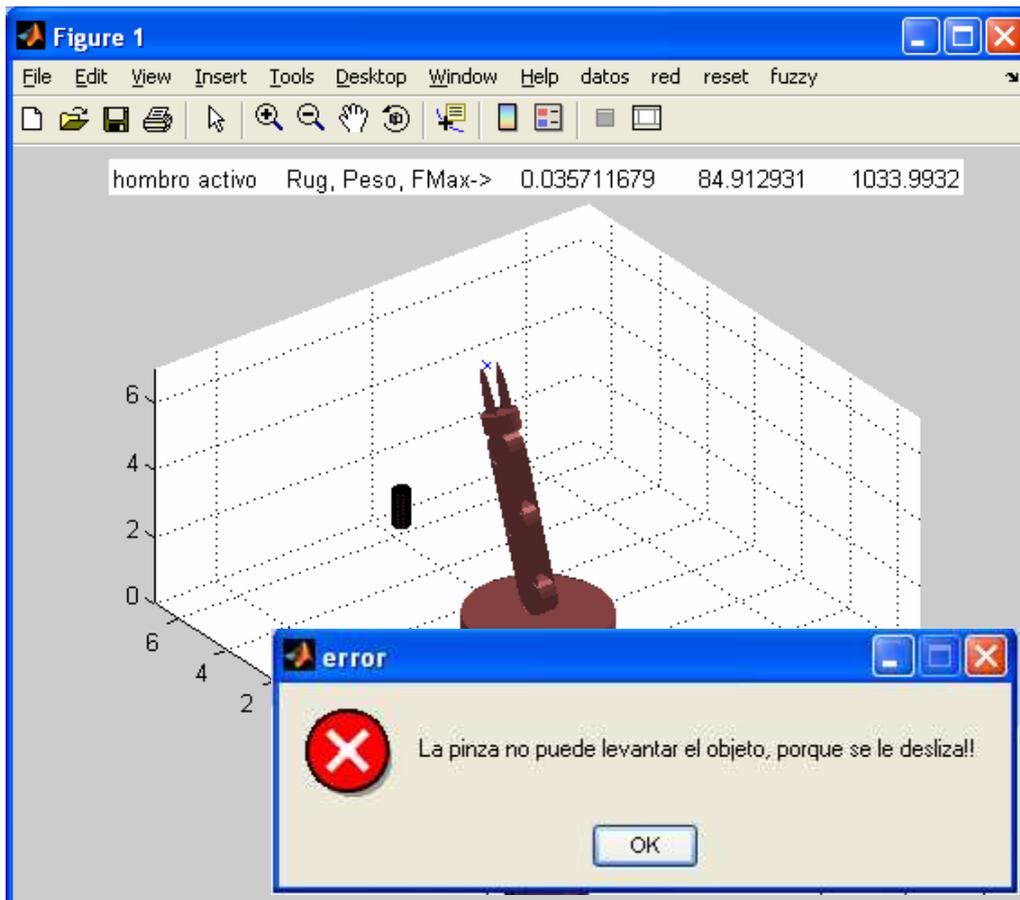


Figura 7: Ejemplo en el que el objeto se le desliza al brazo

Ejemplo 2.3

En este último ejemplo se muestra el funcionamiento del sistema para un objeto generado con las siguientes características: rugosidad 0.95, peso 48.53 y no frágil. A partir de estos valores, igual que en los ejemplos anteriores, se calculan los grados de pertenencia de los conjuntos borrosos de la Figura 2, que son las premisas de las reglas de inferencia de la Figura 4, y se obtienen los grados de pertenencia para los conjuntos borrosos de la salida del sistema que modelan la fuerza del efector (desde el conjunto borroso fuerza nula, mínima, media, mayor y hasta la fuerza máxima), que se observan en la Figura 5.

Usando el método de defuzzificación del centroide, se aprende que la fuerza a aplicar es de 900.28 Newtons, por lo que el sistema decide coger el objeto aplicando la fuerza calculada en la pinza del brazo para sujetarlo y poder levantarlo y moverlo, como se muestra en la Figura 8.

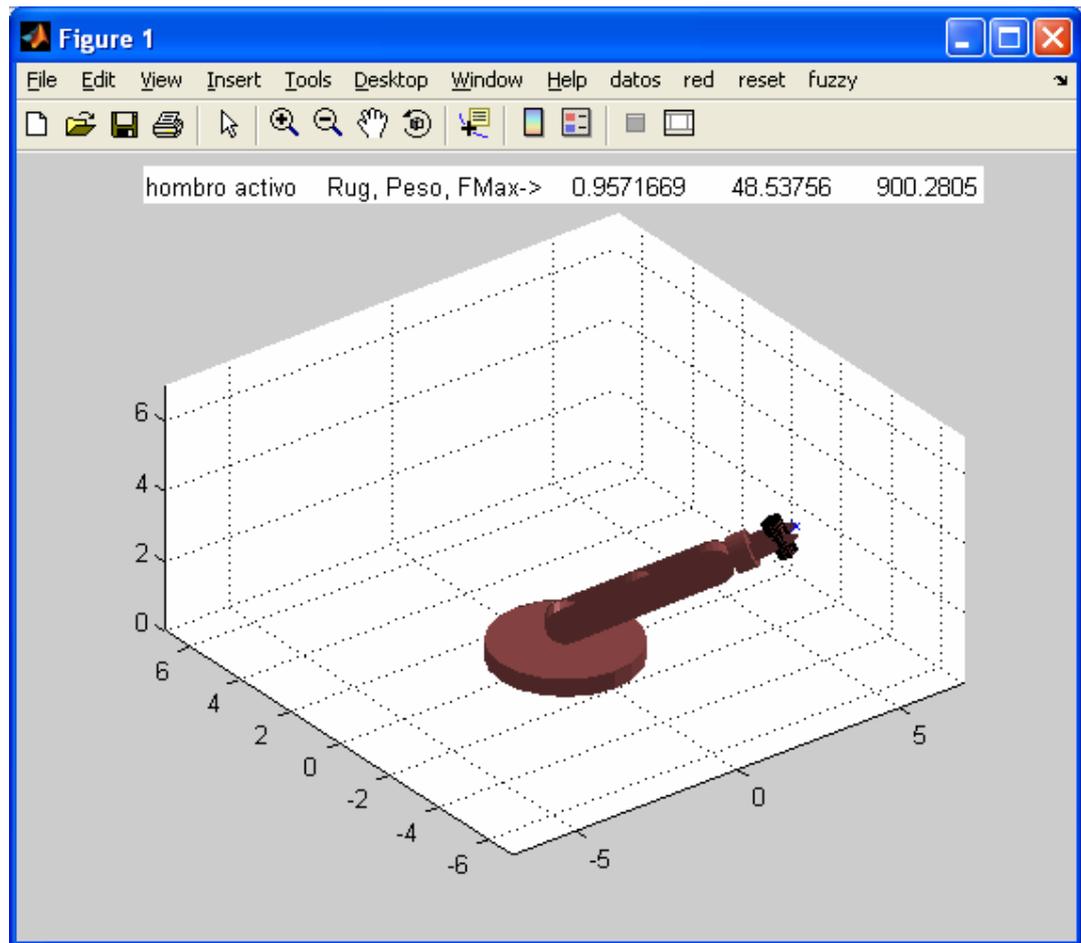


Figura 8: Ejemplo en el que el brazo levanta y mueve el objeto

3. CONCLUSIONES

La toolbox fuzzy de Matlab [1] es una herramienta útil para el aprendizaje y la implementación de sistemas de control inteligente borrosos.

La fuerza de un brazo robot puede ser decidida con inferencia borrosa y razonamiento aproximado con éxito en base a características poco nítidas, como la rugosidad, tipo de peso y fragilidad del objeto a agarrar.

La lógica de Zadeh [8], con el operador de implicación de Mandami [7] y el método de defuzzyficación del centroide [3] ha dado resultados satisfactorios para controlar dicha fuerza de agarre del brazo robot.

Se han añadido ejemplos que muestran el funcionamiento y las diferentes decisiones tomadas por el sistema de control difuso.

4. REFERENCIAS BIBLIOGRÁFICAS

- [1] Home Page: <http://www.mathworks.com>
- [2] Home Page: <http://www.cs.berkeley.edu/~zadeh/>
- [3] KLIR, J., YUAN, B. Fuzzy Sets and Fuzzy Logic. Theory and Applications, Prentice Hall, New Jersey, 1995.
- [4] PAJARES, G. SANTOS, M. Inteligencia Artificial e Ingeniería del Conocimiento; RA-MA, 2005;
- [5] SKLAR A. SCHWEIZER, B. Probabilistic Metric Spaces. Dover Publications, 1984.
- [6] TRILLAS, E, ALSINA, C., TERRICABRAS, J.M. Introducción a la Lógica Borrosa. Editorial Ariel. 1995.
- [7] TRILLAS E. AND L. VALVERDE (1985). On mode and implication in approximate reasoning. Approximate reasoning in expert systems. Eds. M. M. Gupta. North-Holland. pp. 157-166.
- [8] L. A. ZADEH. Fuzzy sets. Inform. and Control 8, 338–353, 1965.