



MX0500325

*XVI Congreso Anual de la SNM y XXIII Reunión Anual de la SMSR*  
*XVI SNM Annual Meeting and XXIII SMSR Annual Meeting*  
Oaxaca, Oaxaca, México, Julio 10-13, 2005 / Oaxaca, Oaxaca, México, July 10-13, 2005

## **Metodología Basada en Lógica Difusa para Construir las Funciones Objetivo en Problemas de Optimización de Combustible Nuclear: Aplicación al Diseño Radial de Celdas**

***Alejandra Maribel Barragán Martínez***

*Facultad de Ingeniería, Universidad Nacional Autónoma de México*  
*División de Ingeniería Eléctrica*  
*Circuito exterior s/n, Ciudad Universitaria, México, D.F.*  
*ale\_bar\_m@yahoo.com.mx*

***Cecilia Martín del Campo Márquez, Miguel Ángel Palomera Pérez***

*Facultad de Ingeniería, Universidad Nacional Autónoma de México*  
*Laboratorio de Análisis en Ingeniería de Reactores Nucleares*  
*Paseo Cuauhnáhuac 8532, 62550 Jiutepec, Mor. México*  
*cmm@fi-b.unam.mx, mapp@uxmcc2.iimas.unam.mx*

### ***Resumen***

Se describe una metodología basada en Lógica Difusa para la construcción de la función objetivo de los problemas de optimización de combustible nuclear. Se creó un sistema de inferencia que responde, en cierta forma, como un experto humano cuando tiene la tarea de calificar diferentes diseños radiales de celdas de combustible. Específicamente se detalla cómo fue construido un sistema de inferencia basado en Lógica Difusa que tiene cinco variables de entrada y una variable de salida, la cual corresponde a la función objetivo para el diseño radial de una celda de combustible para un BWR. La utilización de Fuzzy con Matlab ofreció la capacidad de visualización de la variable de salida en función de una o dos variables de entrada a la vez. Esto permitió construir, de manera adecuada, la combinación de las reglas de inferencia y las funciones de membresía de los conjuntos difusos empleados para cada una de las variables de entrada. La función objetivo obtenida se utilizó en un proceso de optimización basado en Búsqueda Tabú. La nueva metodología fue probada para el diseño de una celda empleada en un ensamble combustible del reactor de Laguna Verde obteniéndose excelentes resultados.

### **1. INTRODUCCIÓN**

Todos los problemas de la Administración de Combustible dentro del Núcleo son problemas de tipo combinatorio complejo que no pueden ser resueltos por métodos analíticos. Actualmente se realizan muchos trabajos en los que se aplican una gran diversidad de técnicas o métodos de optimización metaheurísticos basados en computación evolutiva. Podemos citar sólo por

mencionar algunos de ellos, la aplicación de Algoritmos Genéticos al diseño axial de combustible [1], la utilización de Búsqueda Tabú al diseño axial de combustible [2, 3], el empleo de Búsqueda Tabú al diseño de patrones de recarga [4], la aplicación de variantes de los Algoritmos Genéticos al diseño de recargas [5, 6], la utilización de Algoritmos Genéticos al diseño de patrones de barras de control [7], y el diseño de patrones de recarga de combustible mediante Redes Neuronales [8].

Debe notarse que en los métodos de optimización evolutivos, cada solución examinada, debe ser evaluada para poderla calificar en cuanto a la satisfacción de los parámetros de diseño que se busquen. Por esta razón, adicionalmente, a la selección de los métodos de optimización a aplicarse en cada problema, también se debe investigar cómo representar matemáticamente la función objetivo del problema de optimización específico.

De manera muy resumida y general, se puede decir que tradicionalmente la *Función Objetivo* ( $FO$ ) se ha venido representando mediante un modelo matemático en el que se adicionan los  $k$  diferentes objetivos parciales  $P_k$ , afectados por un peso  $W_k$  el cual pondera la importancia que tiene el objetivo parcial  $P_k$  dentro de la función objetivo total, como se muestra en la ecuación (1).

$$FO(x) = \sum_1^K P_k(x) * W_k \quad (1)$$

Donde  $k$  es el número de parámetros de diseño que se consideran en la optimización para los cuales se define un objetivo parcial  $P_k$ , que puede ser la búsqueda de un parámetro a minimizar (o maximizar) o puede ser un parámetro deseado (target) o puede ser una restricción a satisfacer.

Generalmente los pesos  $W_k$  son valores constantes o que pueden tomar un cierto valor según se quiera variar la importancia del objetivo  $P_k$  durante la evolución del proceso de optimización de acuerdo a valores obtenidos para el parámetro de diseño involucrado en el objetivo parcial  $P_k$ . Al venir experimentando con la construcción de las funciones objetivo en los diferentes problemas de optimización (diseño radial y axial de combustible, diseño de recargas, y de patrones de barras de control) se ha visto la necesidad de utilizar más que valores constantes para los pesos  $W_k$ , funciones continuas que permitan variar el valor del peso  $W_k$  según se vayan cumpliendo los objetivos parciales.

El trabajo presente muestra los resultados de una tesis en la que la hipótesis es que las funciones objetivo pueden ser construidas mediante la técnica de Lógica Difusa, dibujando las gráficas “deseadas” para la Función Objetivo mediante la combinación de los parámetros de diseño, éstos calificados también dentro de una escala de variación con funciones de pertenencia de *calidad*.

Como ejemplo de aplicación se seleccionó el problema de diseño radial de celdas de combustible para Reactores de Agua en Ebullición. El método de optimización seleccionado es la técnica de Búsqueda Tabú la cual ya se presentó en un trabajo anterior [9].

## 2. DESCRIPCIÓN DEL PROBLEMA

Se busca el diseño radial óptimo de una celda representada por un arreglo de 10x10 barras de combustible como se esquematiza en la Figura 1. Las dimensiones, las zonas de agua y los

materiales estructurales se consideran previamente definidos. La tarea de diseñar la celda consiste en buscar la distribución óptima de 10 diferentes composiciones de combustible (dióxido de uranio con diferentes enriquecimientos en U-235 y diferentes concentraciones de gadolinia) en las 92 barras del arreglo. Las composiciones se consideran previamente definidas y son las que se muestran en la Tabla I.

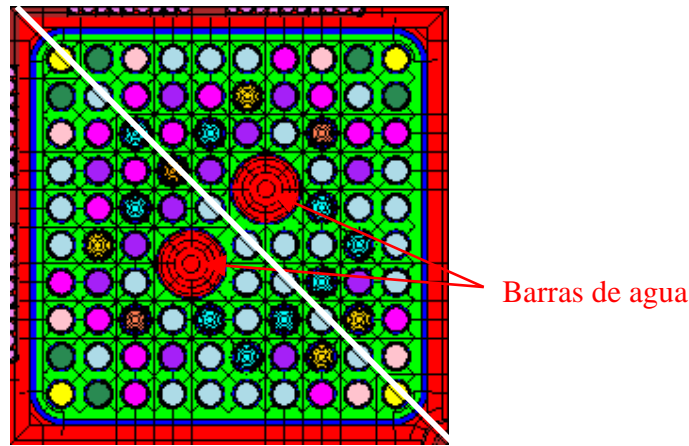


Figura 1. Sección radial de una celda de combustible 10x10 barras:

Tabla I. Enriquecimientos y concentraciones de gadolinia en las composiciones

Composición	U-1	U-2	U-3	U-4	U-5	U-6	UG-1	UG-2	UG-3	UG-4
U <sub>235</sub> %w	2.0	2.8	3.6	4.4	3.95	4.9	3.95	4.4	4.4	4.4
Gadolinia %w	0	0	0	0	0	0	5	5	4	2

Los parámetros de evaluación se describen en la Tabla II. Se observan dos parámetros de composición: enriquecimiento promedio del uranio (*Enr*), y concentración promedio de gadolinia (*Gado*). El enriquecimiento promedio de la celda *x* se calcula sumando los enriquecimientos individuales de las barras y dividiendo entre el número total, de manera semejante se calcula la concentración promedio de gadolinia. Los parámetros neutrónicos de diseño que se evalúan son el factor pico de potencia máximo (*PPF*), el factor de multiplicación infinita (*K*) de la celda a 0 quemado, y el parámetro *S*, el cual se calcula como la suma de las diferencias cuadráticas de los valores de *k*-infinita de la solución *x* y *k*-infinita *target*, a *N* diferentes pasos de quemado. (ver ecuación (2)).

$$S(x) = \sum_{i=1}^N (K_i(x) - K_{i\_tar})^2 \tag{2}$$

Para obtener los parámetros *PPF* y *K* se simula la celda usando el simulador neutrónico HELIOS [9]. En la Figura 1 se muestra cómo se representa geoméricamente la celda con dicho simulador, los colores representan diferentes materiales (composiciones). En el ejemplo mostrado las barras que contienen óxido de uranio enriquecido mezclado con gadolinia tienen más nodos de cálculo que las barras sin gadolinia. Aprovechando la simetría diagonal de la celda y con el objeto de disminuir el tiempo de cálculo se aplicó la condición de reflexión en el eje de simetría.

**Tabla II. Parámetros que intervienen en la función objetivo**

$\mathbf{x}$	Vector de composiciones de combustible asociadas a las posiciones en la celda (solución $\mathbf{x}$ )
$Enr(\mathbf{x})$	Enriquecimiento promedio de la celda $\mathbf{x}$ , a 0 MWd/t
$Gado(\mathbf{x})$	Concentración promedio de gadolinia en la celda $\mathbf{x}$ , a 0 MWd/t
$PPF(\mathbf{x})$	Factor pico de potencia radial de la celda $\mathbf{x}$ , a 0 MWd/t
$K(\mathbf{x})$	k-infinita de la celda $\mathbf{x}$ , a 0 MWd/t,
$S(\mathbf{x})$	Suma de las diferencias cuadráticas de valores de k-infinita de la solución $\mathbf{x}$ y k-infinita <i>target</i> a diferentes pasos de quemado.

Como ya se mencionó, el método de optimización aplicado es la técnica de Búsqueda Tabú con algunas reglas heurísticas para acelerar la búsqueda y para evitar evaluar las soluciones que simplemente al calcular su *Enr* y su *Gado* no ameriten ser mandadas a evaluación neutrónica. También se evita la evaluación neutrónica completa (a diferentes pasos de quemado) de soluciones que tienen valores inadecuados de *PPF* y de *K* al paso 0 MWd/t. Para ver más detalles de cómo se aplicó el algoritmo de Búsqueda Tabú se recomienda leer el trabajo anterior [9]. En el presente trabajo nos avocaremos a describir esencialmente la metodología para construir la función objetivo, la cual es directamente leída por el sistema de optimización programado en lenguaje Phyton y ejecutado en una estación de trabajo con sistema operativo Unix.

### 3. CONSTRUCCIÓN DE LA FUNCIÓN OBJETIVO EMPLEANDO LÓGICA DIFUSA

#### 3.1. Fundamentos de Lógica Difusa

La lógica difusa es una rama de la inteligencia artificial que se funda en el concepto "Todo es cuestión de grado", lo cual permite manejar información vaga o de difícil especificación si quisiéramos hacer cambiar con esta información el funcionamiento o el estado de un sistema específico. Es entonces posible con la lógica difusa gobernar un sistema por medio de reglas de "sentido común" las cuales se refieren a cantidades indefinidas. En general la lógica difusa se aplica tanto a sistemas de control como para modelar cualquier sistema continuo de ingeniería, física, biología o economía. [10]

La lógica difusa es definida como un sistema matemático que modela funciones no lineales, que convierte unas entradas en salidas acordes con los planteamientos lógicos que usan el razonamiento aproximado. Se fundamenta en los denominados conjuntos difusos y un sistema de inferencia difuso basado en reglas de la forma " SI..... ENTONCES..... ", donde los valores lingüísticos de la premisa y el consecuente están definidos por conjuntos difusos, es así como las reglas siempre convierten un conjunto difuso en otro.

La Lógica Difusa, que hoy en día se encuentra en constante evolución, nació en los años 60 como la lógica del razonamiento aproximado, y en ese sentido podía considerarse una extensión de la Lógica Multivaluada. La Lógica Difusa actualmente está relacionada y fundamentada en la teoría

de los Conjuntos Difusos. Según esta teoría, el grado de pertenencia de un elemento a un conjunto va a venir determinado por una función de pertenencia, que puede tomar todos los valores reales comprendidos en el intervalo  $[0,1]$ .

La Lógica Difusa (llamada también Lógica Borrosa por otros autores) o Fuzzy Logic es básicamente una lógica con múltiples valores, que permite definir valores en las áreas oscuras entre las evaluaciones convencionales de la lógica precisa: Si / No, Cierto / Falso, Blanco / Negro, etc.

### 3.2. Construcción de la Función Objetivo con la Ayuda de Fuzzy - Matlab.

El proceso que se realizó para obtener la función objetivo utilizando el comando “fuzzy” de Matlab (versión 6.5) fue el siguiente:

#### Paso 1. Creación de variables:

Del menú “Edit” se selecciona “Add Variable” (“input or output”) para determinar si se añadirá una variable de entrada o de salida (ver figura 2).

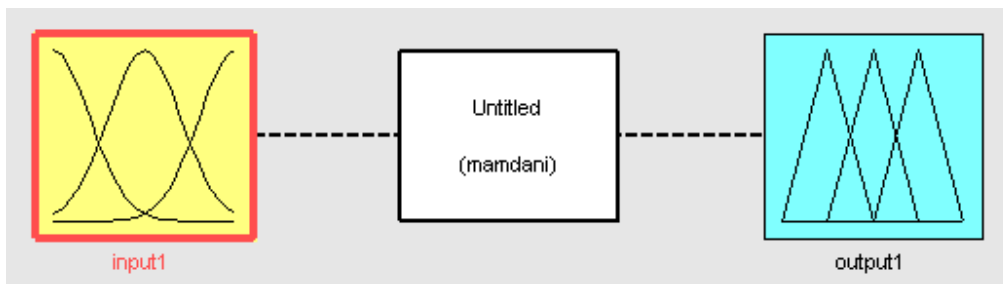
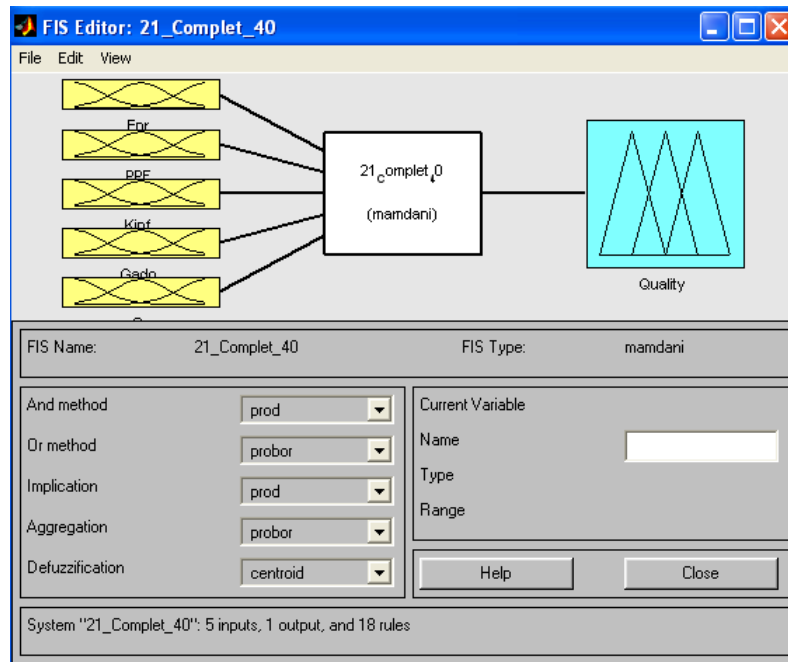


Figura 2. Creación de variables de entrada y salida

En este caso, se seleccionaron cinco variables de entrada y una de salida (ver Figura 3). Las variables de entrada representan a los parámetros involucrados en la función objetivo ( $Enr(x)$ ,  $PPF(x)$ ,  $K(x)$ ,  $Gago(x)$  y  $S(x)$ ), en la construcción cabe aclarar que en la construcción del sistema difuso a cada variable se le asignó una “variable lingüística” de “Enr”, “PPF”, “kinf”, “Gado” y “S” respectivamente; y la variable de salida representa la *función objetivo* (etiquetada con el nombre *Quality*).

Para la selección de las operaciones básicas de la Lógica Difusa, en esta misma pantalla de la Figura 3, se seleccionó “prod” (producto cartesiano) tanto para el operador AND como para el operador IMPLICATION. Esto significa que los valores de pertenencia de las funciones de entrada que son intersecadas (Con el operador AND) van a multiplicarse. Este producto se realiza de acuerdo a la función de pertenencia construida por Mandani en 1974. El operador OR no fue utilizado en las reglas por lo que su selección no tuvo ninguna importancia en este proceso. Para la agregación de las reglas “Aggregation” se utilizó el operador “probor”. Para realizar el proceso de defusificación se utilizó el método del centroide “centroid” el cual genera como salida el valor correspondiente al centro de gravedad de la función de pertenencia de la salida. Para conocer el

detalle de estas funciones y operadores se recomienda consultar el manual de la Herramienta de cálculo de Lógica Difusa de Matlab [11].



**Figura 3. Variables de entrada y salida y selección de métodos de Fuzzy**

## **Paso 2. Construcción de funciones de pertenencia para cada variable:**

Del menú Edit se selecciona Membership Function para visualizar la ventana donde se diseñan las funciones de pertenencia.

Para la variable de enriquecimiento promedio de la celda (*Enr*) se seleccionaron los tipos de curva zmf, psigmf y smf, para “regular”, “bien” y “mal” respectivamente (ver Figura 4), las cuales se ajustaron siguiendo el comportamiento buscado en las curvas de superficie para una variable que se describirá más adelante. Para seleccionar el rango que abarcarán las curvas de la variable, se determina un valor deseado (target) y se delimita el rango dependiendo de los valores que se pretendan analizar. Para el caso de la variable *Enr* se determinó el rango de [3 5] es decir enriquecimientos que varían de 3% a 5% en U-235. El proceso de construcción de todas las curvas de las variables involucradas es el mismo, se utilizan los mismos criterios que se describieron para la variable *Enr*.

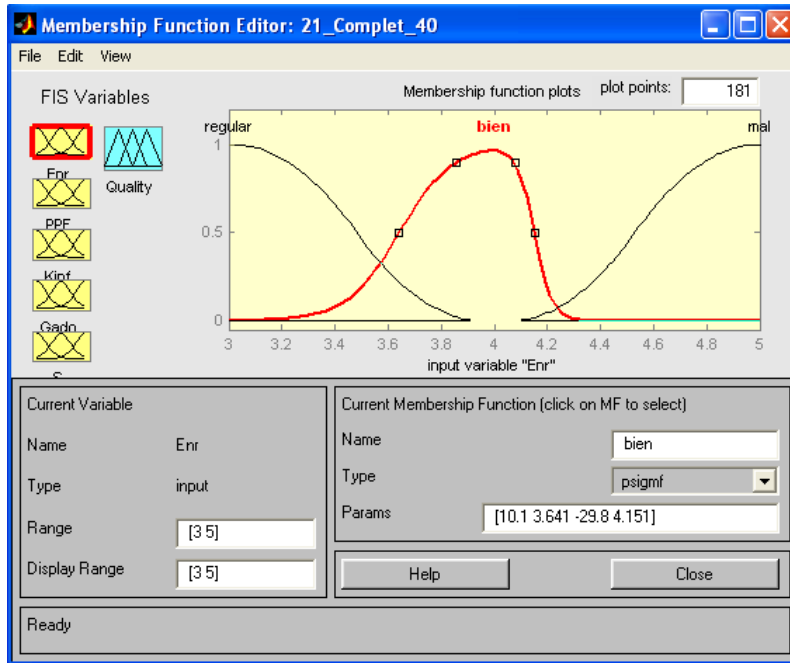


Figura 4. Función de pertenencia para la variable *Enr*

Para la variable *PPF* se utilizaron las curvas *zmf*, *gaussmf* y *smf* para “bien”, “regular” y “mal”, respectivamente (ver Figura 5).

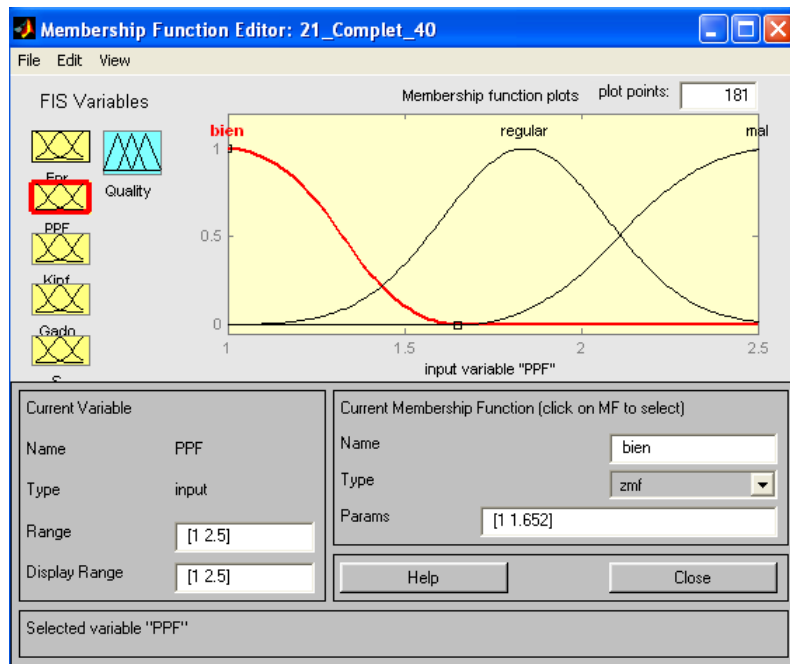
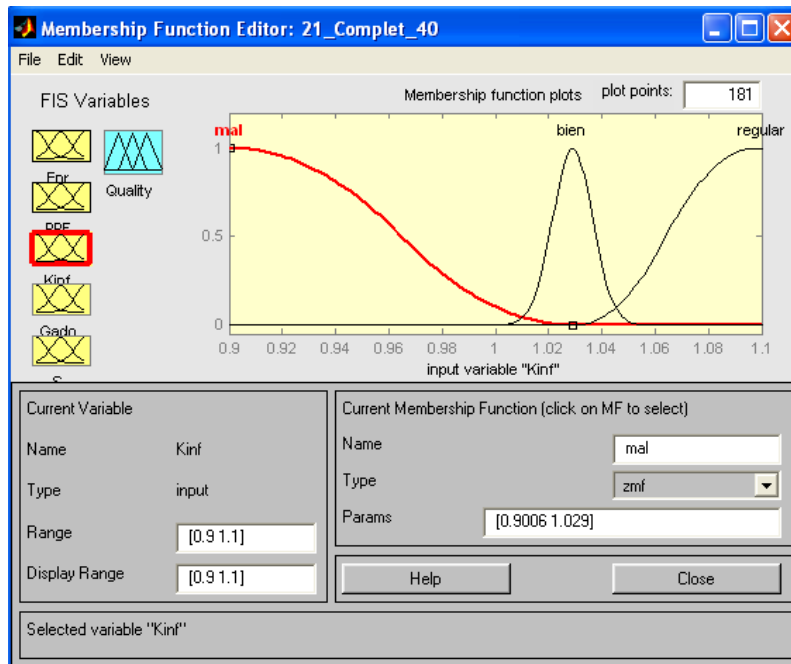


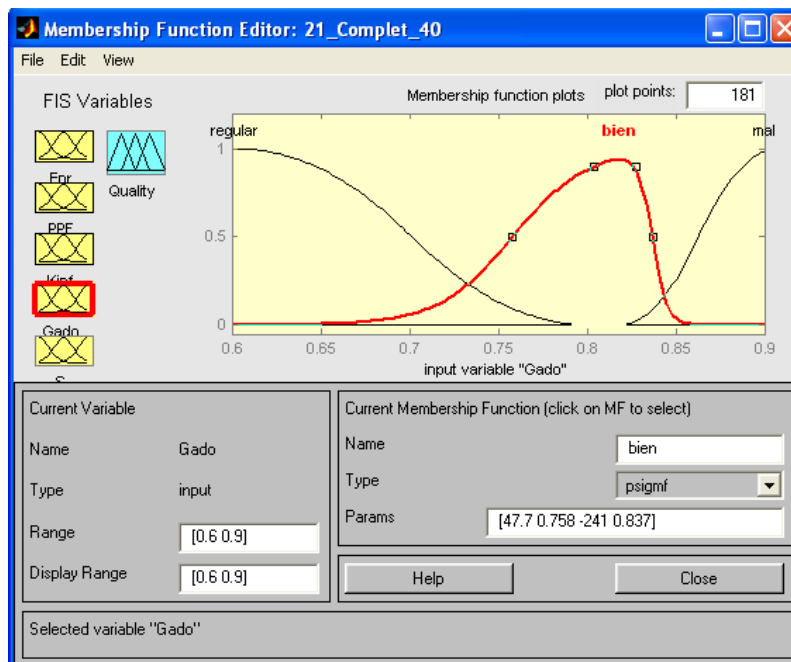
Figura 5. Función de pertenencia para la variable *PPF*

Para la variable  $K_{inf}$  se utilizaron las curvas *zmf*, *gaussmf* y *smf* para “mal”, “bien” y “regular” respectivamente (ver Figura 6).



**Figura 6. Función de pertenencia para la variable *Kinf***

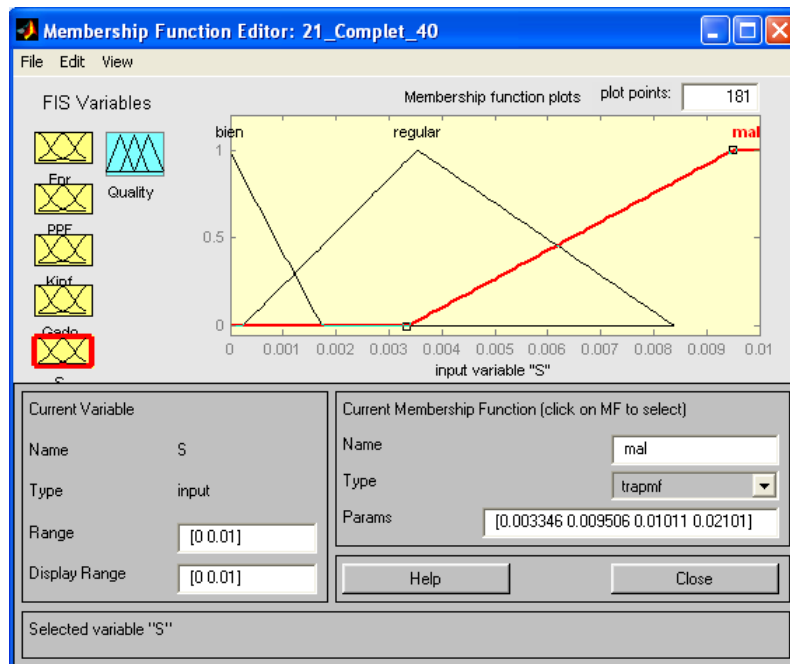
Para la variable *Gado* (gadolinia) se utilizaron las curvas zmf, psigmf y smf para “regular”, “bien” y “mal”, respectivamente (ver Figura 7).



**Figura 7. Función de pertenencia para la variable *Gado***

Finalmente, para la variable *S* se utilizaron las curvas trimf para “bien” y “regular”. Para “mal” se utilizó trapmf (ver Figura 8).





**Figura 8. Función de pertenencia para la variable S**

Cada una de las curvas fueron seleccionadas de acuerdo al comportamiento que siguen las variables cuando se simulan en HELIOS, por esta razón se eligieron en su mayoría curvas con pendientes suaves donde el rango buscado fuera mayor, y otras como en el caso de  $S$  donde las pendientes de las curvas son muy abruptas y el rango buscado se reduce a un valor target.

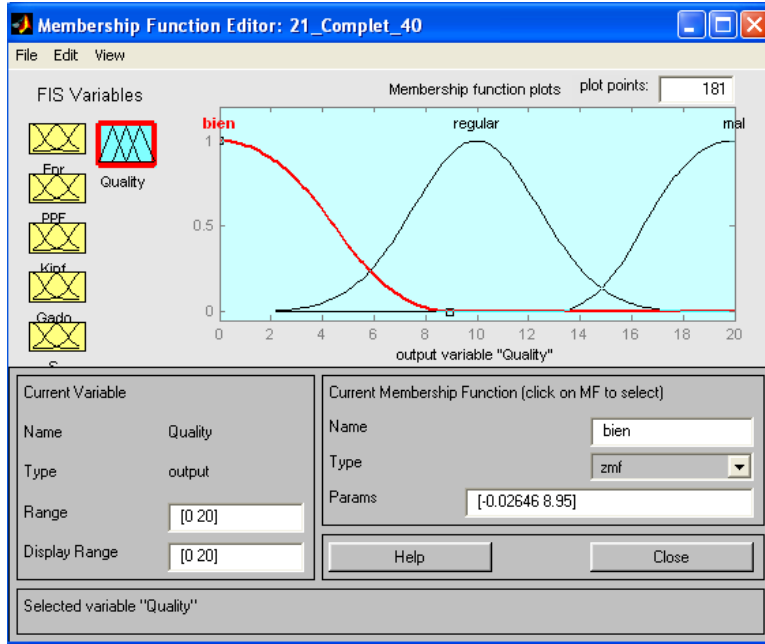
Para la variable de salida que representa la calificación (*Quality*), se seleccionaron las curvas zmf, gaussmf y smf para “bien”, “regular” y “mal”. Calificando como muy bien al rango de [0 4] (ver Figura 9).

### Paso 3. Construcción de reglas:

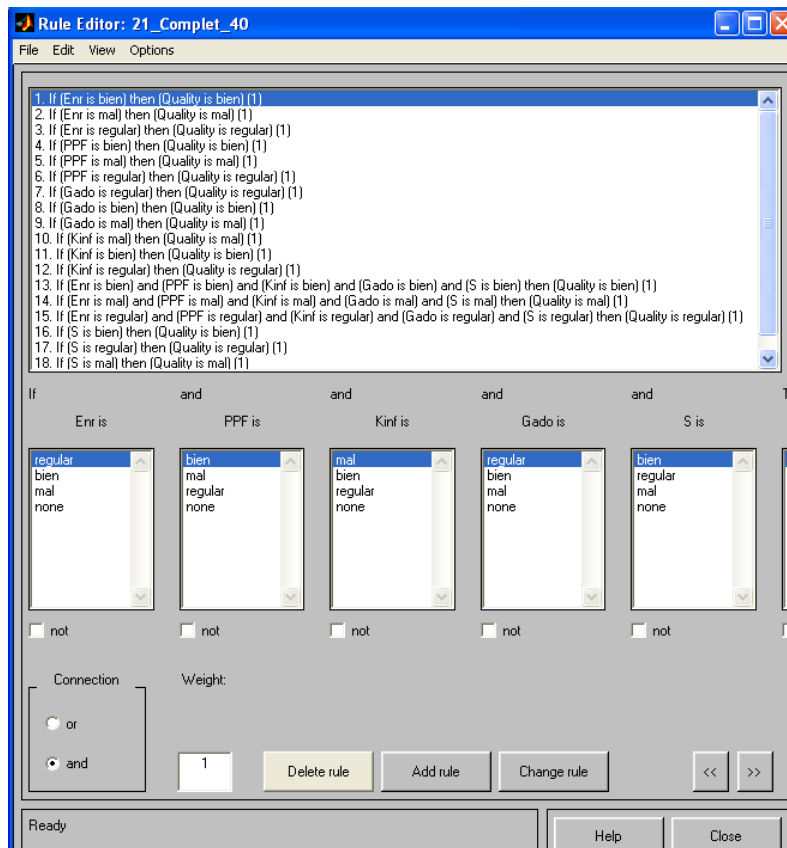
Del menú Edit en el apartado Rules se pueden seleccionar las reglas que seguirán las curvas construidas para cada variable. Estas reglas son las que determinaran el comportamiento final de las variables en el proceso de optimización cuando el archivo creado en Matlab (archivo.fis) es utilizado en el programa que simulara dicho proceso.

Las reglas que se utilizaron consideran el comportamiento que debe seguir cada variable individualmente y en conjunto. En este caso sólo se utilizó el conector And y se crearon 18 reglas (ver Figura 10).

Para observar cómo actúan las reglas sobre las variables se selecciona del menú View, el apartado Rules, es aquí donde se involucran los valores Target de cada variable (los cuales se pueden variar al mover la línea vertical de color rojo) y de acuerdo a ellos obtener la calificación de las combinaciones que se realicen (ver Figura 11).



**Figura 9. Función de pertenencia para la variable *Quality* (Función Objetivo)**



**Figura 10. Edición de las reglas**

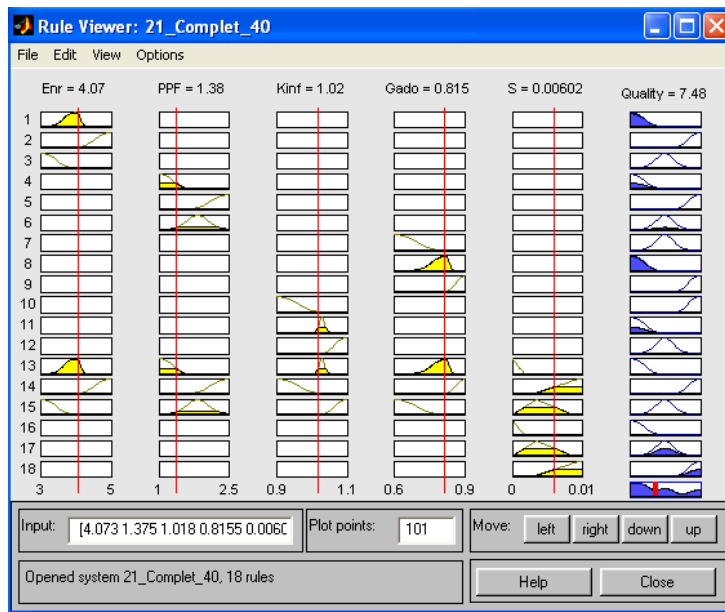


Figura 11. Vista de las reglas

Mediante el apartado de Surface en el menú View es posible darle a cada variable las curvas y los rangos más adecuados para representar su comportamiento, esto se observa al analizar cada variable por separado (su curva de superficie). Por ejemplo, para la variable *Enr* observamos que la mejor calificación la obtiene para un enriquecimiento cercano a 4 y que, conforme se incrementa ese valor su calificación empeora drásticamente (ver Figura 12). La pendiente que se presenta en la curva antes del 4 es más suave, por lo tanto, es más aceptable un valor menor que uno mayor.

En la variable *PPF*, se pretende obtener una curva lo más cercana a una línea recta, siendo su calificación ideal aquella presente cuando el PPF vale 1 (Ver Figura 13).

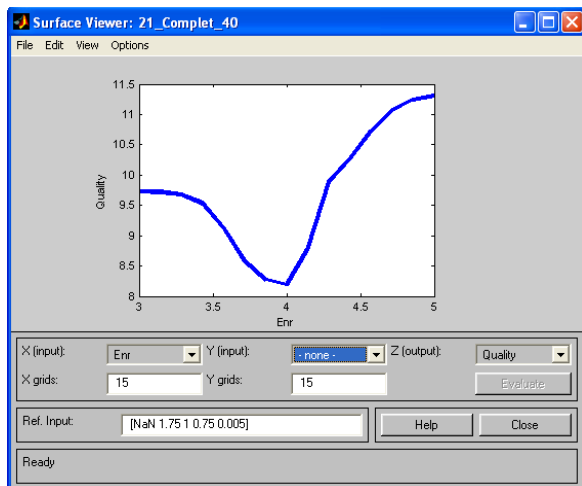


Figura 12. Calificación en función de *Enr*

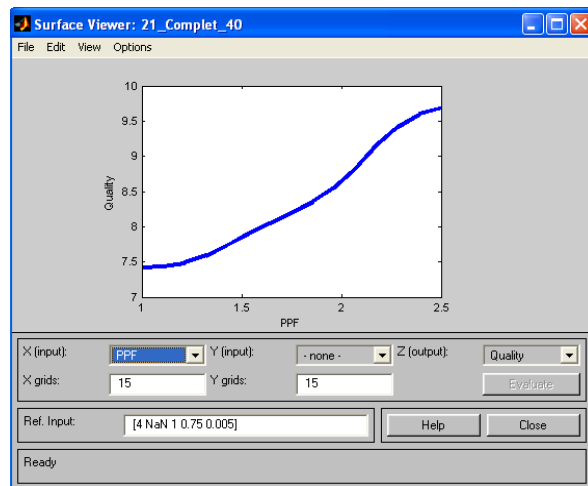
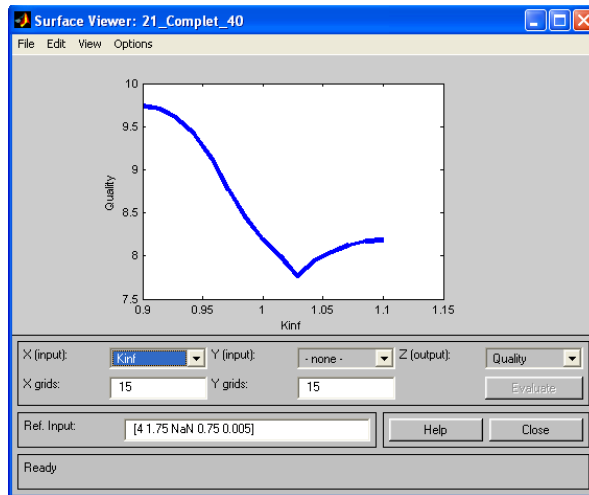


Figura 13. Calificación en función de *PPF*

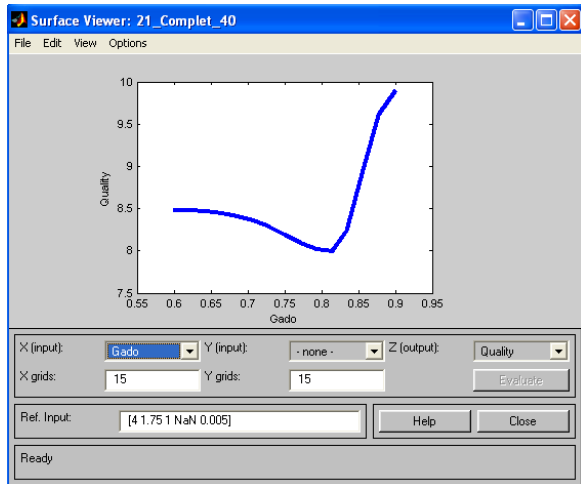
Para la variable  $K_{inf}$ , la mejor calificación se encuentra en 1.03 aproximadamente. Las pendientes presentes en la curva antes y después del valor 1.03 son muy abruptas, por lo tanto, se pretende que el valor esperado de esta variable tenga un rango muy reducido de variación (ver figura 14).



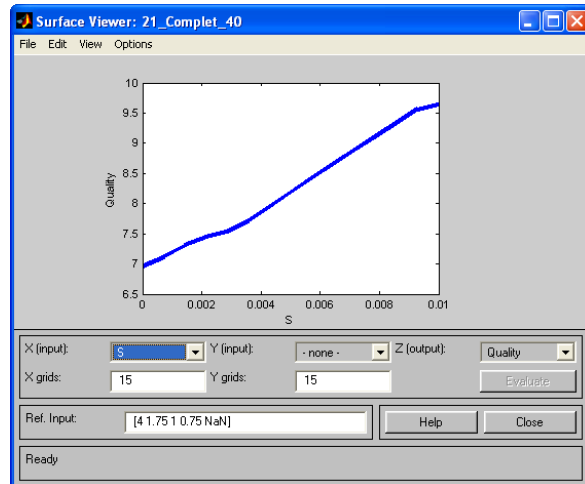
**Figura 14. Calificación en función de  $K_{inf}$**

Para la variable  $Gado$ , se pretende obtener un rango reducido y de preferencia un valor target de 0.815. En la curva de superficie de esta variable, antes del valor deseado la pendiente es más suave que después de dicho valor donde la calificación rápidamente se convierte en mala. Se prefiere un valor menor al deseado que uno mayor (ver Figura 15).

Al igual que para  $PPF$ , se busca que la curva de la variable  $S$  sea lo mas cercana a una línea recta de pendiente positiva, siendo la mejor calificación cuando  $S = 0$  (ver Figura 16).



**Figura 15. Calificación en función de  $Gado$**



**Figura 16. Calificación en función de  $S$**

Utilizando este mismo apartado (surface) se puede observar el comportamiento de dos variables simultáneamente y la calificación de ambas (ver Figuras 17, 18 y 19).

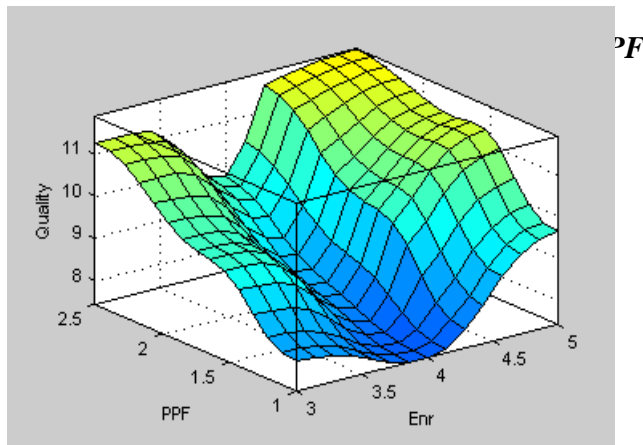


Figura 17. Calificación en vs  $Enr$  y  $PPF$

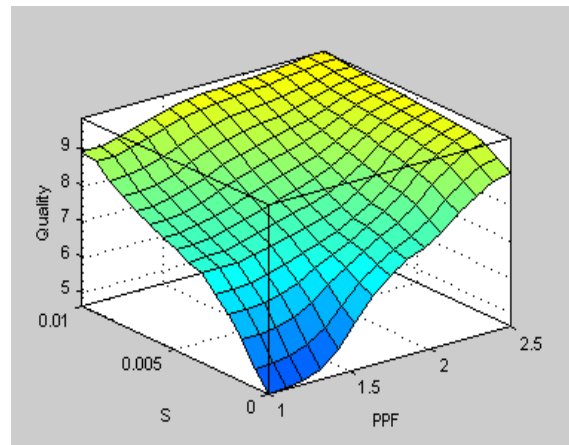


Figura 18. Calificación vs de  $PPF$  y  $S$

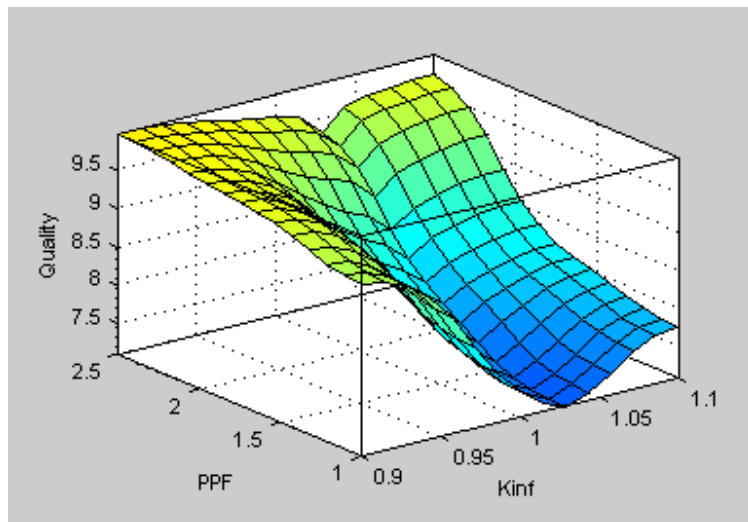


Figura 19. Calificación vs de  $K_{inf}$  y  $PPF$

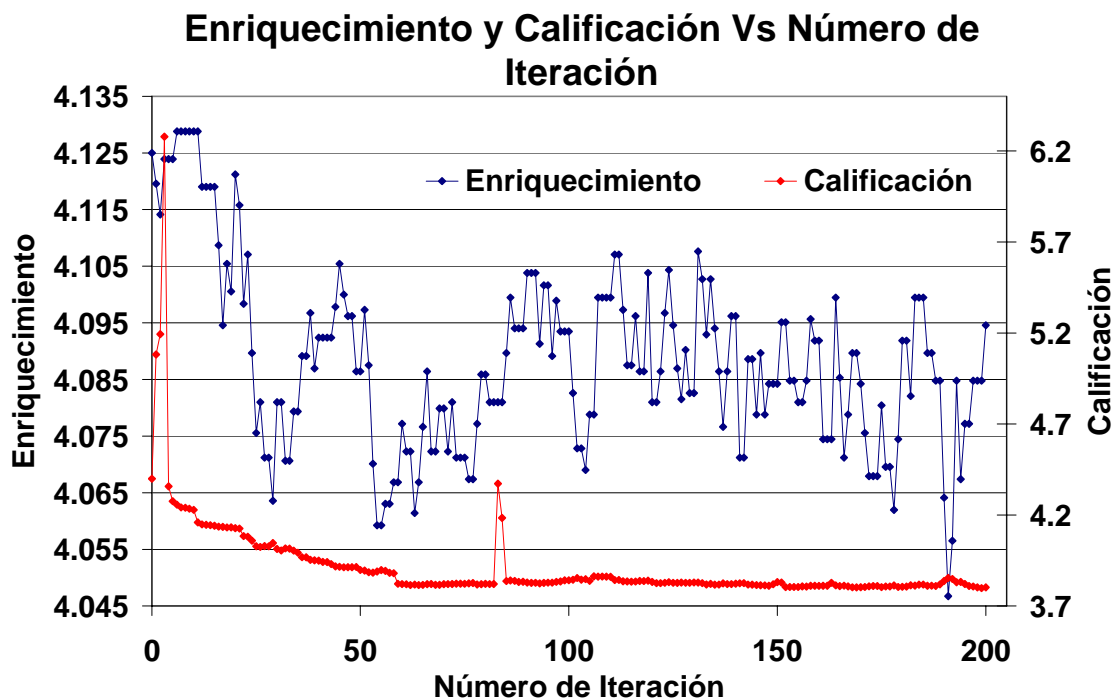
### 3. RESULTADOS DE LA OPTIMIZACIÓN

En las Figuras 20 a 24 se muestran las evoluciones de  $Enr(x)$ , del  $PPF(x)$ , de  $K_{inf}(x)$  y de  $S(x)$  en función del número de iteración del proceso de optimización. En la Tabla III se muestran los resultados para la “mejor” solución, la cual tiene un valor de  $PPF$  inferior que en los casos de la celda “mejor” del trabajo anterior (en el cual la función objetivo se calcula de manera tradicional), y de la celda de referencia. En la última columna se muestra el número de iteración y el número de la solución asociados a la “mejor” solución. Se observa que la “mejor” solución se encontró en la iteración 199 y corresponde a la solución número 607 de las evaluaciones completas. Se trata de un diseño con un  $PPF$  6.5 inferior al de la celda de referencia, que tiene reactividad muy semejante y además es de menor enriquecimiento. Con respecto a la celda

“mejor” del caso con función objetivo tradicional se puede apreciar que ésta es mejor en enriquecimiento pero peor en PPF.

**Tabla III. Resultados para la “mejor” solución investigada**

		$K_{inf}$	PPF	S pcm	Gado (w%)	Enr (w%)	Calificación	Iteracion / Solucion
TS+FO <sub>fuzzy</sub>	“mejor”	1.0305	1.351	3.48	0.815	4.084	3.80	199 / 607
TS+FO <sub>trad</sub>	“mejor”[9]	1.0298	1.438	7.0	0.815	4.026	4.41	
Referencia		1.0293	1.438	0	0.815	4.107	4.43	



**Figura 20. Evolución de la Calificación y del Enriquecimiento en función del Número de Iteración**

#### 4. CONCLUSIONES

La metodología de construcción de funciones objetivo utilizando Lógica Difusa dio resultados muy satisfactorios y se estima que puede ser mejorada ajustando algunos de los parámetros y operadores de Fuzzy. Esto se está realizando en un trabajo de tesis en proceso. La lógica difusa permitió crear un sistema de inferencia que responde en cierta forma como un experto humano cuando tiene la tarea de calificar diferentes diseños radiales de celdas de combustible. La utilización de Fuzzy con Matlab resulta muy adecuada para construir el sistema de inferencia difuso. Esto dado a que es una herramienta que ofrece la capacidad de visualización de la variable de salida en función de una o dos variables de entrada a la vez. Esto permite construir de manera

adecuada la combinación de las reglas de inferencia y las funciones de membresía de los conjuntos difusos empleados para cada una de las variables de entrada.

### PPF y Calificación Vs Número de Iteración

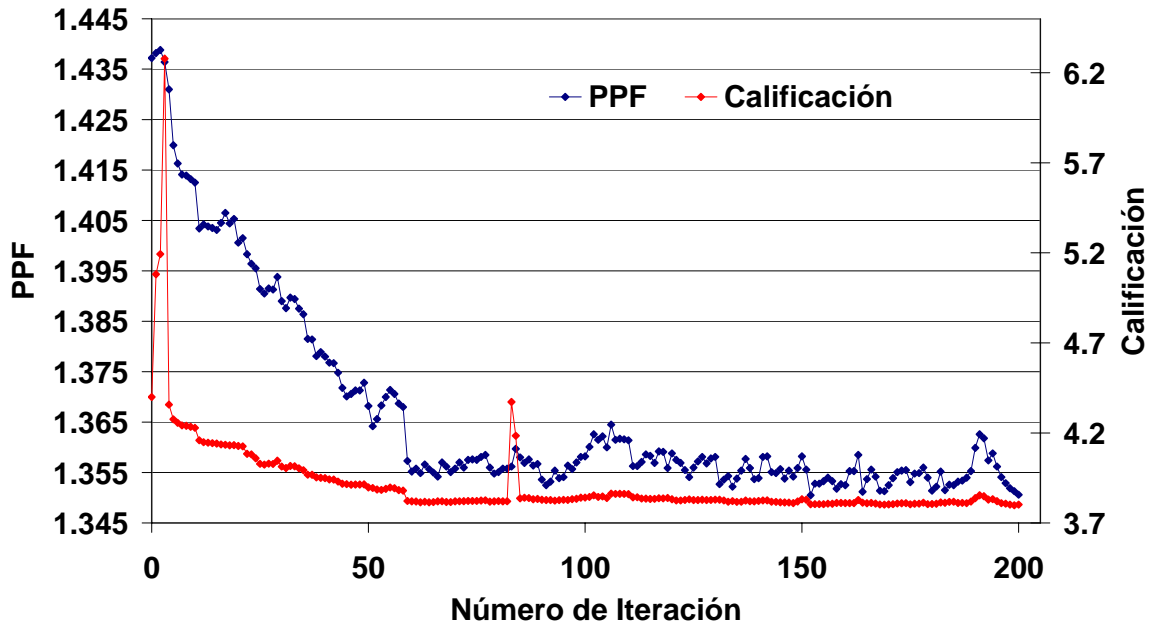


Figura 21. Evolución de la Calificación y del PPF en función del Número de Iteración

### kinf y Calificación Vs Número de Iteración

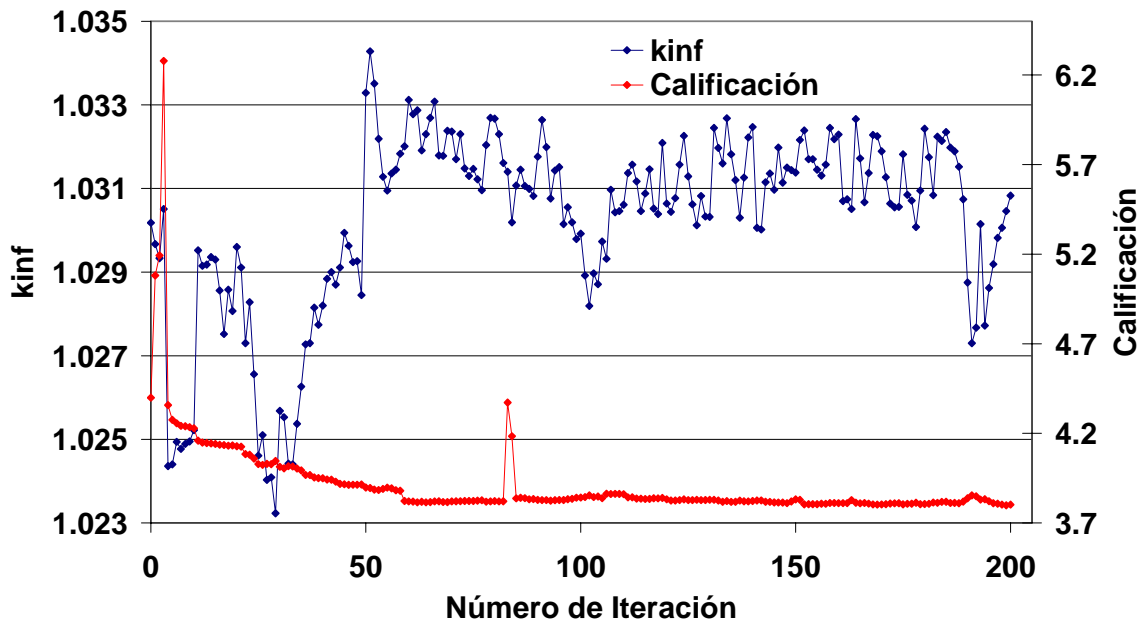


Figura 22. Evolución de la Calificación y de Kinf en función del Número de Iteración

### S y Calificación Vs Número de Iteración

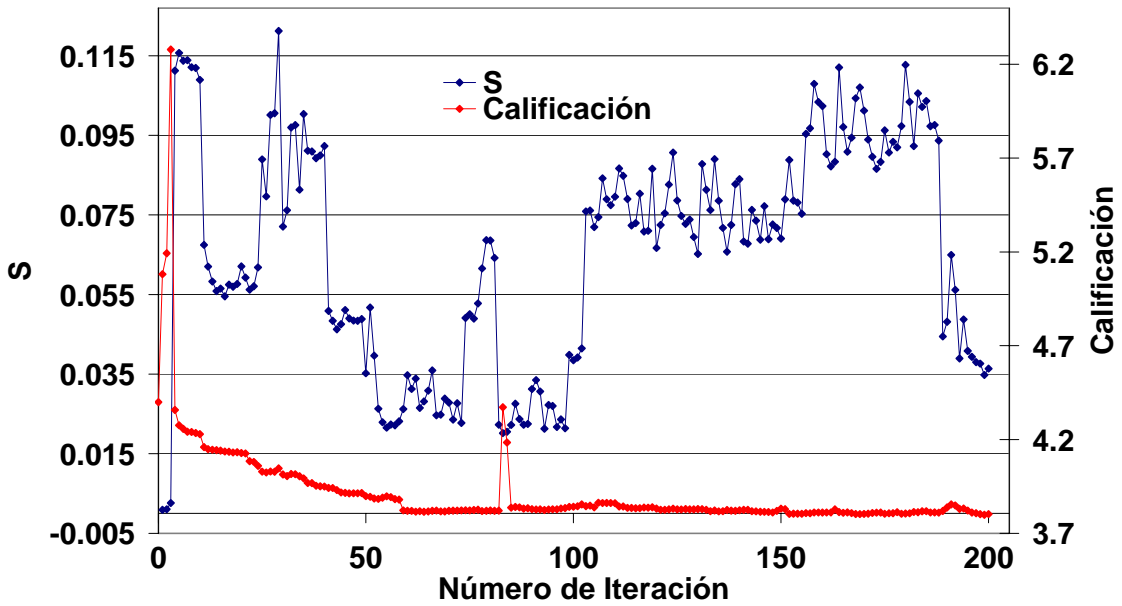


Figura 23. Evolución de la Calificación y de S en función del Número de Iteración

### Gadolina y Calificación Vs Número de Iteración

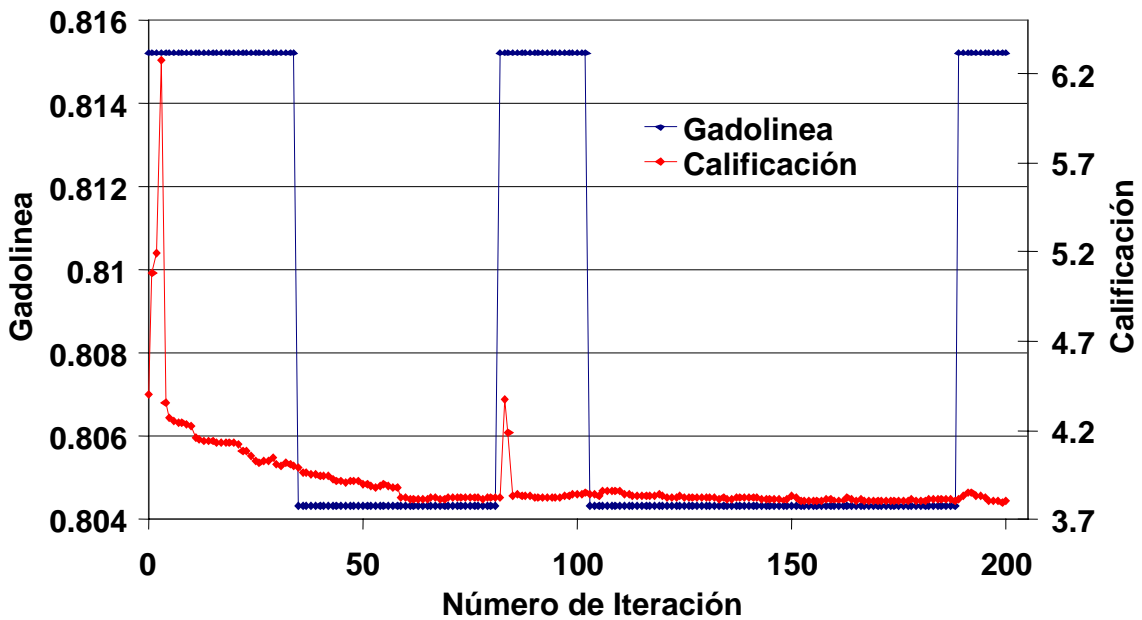


Figura 24. Evolución de la Calificación y de la Gadolina en función del Número de Iteración



## AGRADECIMIENTOS

Este trabajo ha sido realizado en la Facultad de Ingeniería de la Universidad Nacional Autónoma de México (UNAM) y ha recibido apoyo del CONACyT bajo el proyecto 41592-Y y de la UNAM a través del proyecto PAPIIT IN106803-2.

## REFERENCIAS

1. C. Martín del Campo, J. L. François, H. A. López, "AXIAL: a system for boiling water reactor fuel assembly axial optimization using genetic algorithms", *Annals of Nuclear Energy* Vol. **28(16)**, p. 1667-1682 (2001)
2. C. Martín-del-Campo, J. L. François, Luis B. Morales, "Boiling Water Reactor Fuel Assembly Axial Design Optimization Using Tabu Search", *Nuclear Sciences and Engineering*, Vol. **142 (1)**, p. 107-115 (2002)
3. J. L. François, C. Martín-del-Campo, R. François, L. Morales, "A Practical Optimization Procedure for Radial BWR Fuel Lattice Design Using Tabu Search with a Multiobjective Function", *Annals of Nuclear Energy*, Vol. **30(12)** p. 1213-1229 (2003)
4. A. Castillo, G. Alonso, L. Morales, C. Martín del Campo, J. L. Francois, E. del Valle, "BWR fuel reloads design using a Tabu search technique", *Annals of Nuclear Energy*, Vol. **31(2)**, p.151-161 (2004)
5. C. Martín del Campo, J. L. François, L. Avendaño, M. González. "Development of a BWR Loading Pattern Design System Based on Modified Genetic Algorithms and Knowledge". *Annals of Nuclear Energy*. Vol. **31(16)**, p. 1901-1911 (2004)
6. J. J. Ortiz, I. Requena, "An Order Coding Genetic Algorithms to Optimize Fuel Reload in a Nuclear Boiling Water Reactor", *Nuclear Science & Engineering*. Vol. **146**. p. 88-98 (2004)
7. J. L. Montes, J. J. Ortiz, I. Requena, R. Perusquía, "Searching for Full Power Control Rod Patterns in a BWR using Genetic Algorithms", *Annals of Nuclear Energy*. Vol. **31(16)** p. 1939-1954 (2004)
8. J. J. Ortiz, I. Requena, "Using a Multi-State Recurrent Neural Network to Optimize Fuel Reloads in BWRs", *Annals of Nuclear Energy*, Vol. **31**, p. 789-803 (2004)
9. C. Martín del Campo M., J. L. François L., M. A. Palomera, "Optimización de Celdas de Combustible para BWR Basada en Búsqueda Tabú Modificada", *Memorias del Congreso Internacional Conjunto Cancún 2004 LAS/ANS-SNM-SMSR*. Cancún, Q. R., México, 11-14 de Julio (2004)
10. "Lógica Difusa", <http://personales.ya.com/casanchi/mat/difusa01.htm>
11. Fuzzy Logic Toolbox User`s Guide for use with Matlab. Version 2. June 2004 The MathWorks, Inc.