

**INVESTIGACIÓN DE NUEVOS PROTOTIPOS DE SENSORES DE VISCOSIDAD
Y SISTEMA DE CONTROL POR CLONACIÓN ARTIFICIAL, BASADOS EN
TÉCNICAS DE INTELIGENCIA ARTIFICIAL.**

PEDRO MIGUEL CAICEDO TORRES

**UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
FACULTAD DE INGENIRERIAS FISICOMECHANICAS
PROGRAMA DE INGENIERIA MECATRONICA
BUCARAMANGA
2007**

**DISEÑO, MODELACIÓN Y PRUEBAS DE UN PROTOTIPO SISTEMA DE
CONTROL DIGITAL DE LA VISCOSIDAD EN LA PLANTA DE
VISCORREDUCCIÓN DE BARRANCA BASADOS EN ALGORITMOS
GENÉTICOS Y CLONACIÓN ARTIFICIAL**

PEDRO MIGUEL CAICEDO TORRES

Informe final de Proyecto de Grado

**Director: ANTONIO FAUSTINO MUÑOZ MONER
Ph.D.Dr.Sc.Ing.**

**UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
FACULTAD DE INGENIRERIAS FISICOMECHANICAS
PROGRAMA DE INGENIERIA MECATRONICA
BUCARAMANGA
2006**

Nota de aceptación:

Firma del presidente del jurado

Firma del jurado

Firma del jurado

Bucaramanga, 6 de Julio de 2007

Texto de la dedicatoria

AGRADECIMIENTOS

CONTENIDO

	PAG.
INTRODUCCIÓN	14
1 OBJETIVOS	17
1.1 OBJETIVO GENERAL	17
1.2 OBJETIVOS ESPECÍFICOS	17
2 PLANTEAMIENTO DEL PROBLEMA	18
3 ANTECEDENTES	20
4 MARCO TEORICO	21
4.1 ALGORITMOS GENETICOS	21
4.1.1 Generación de la población inicial.	23
4.1.2 Codificación del problema.	23
4.1.3 Operadores genéticos.	25
4.1.3.1 Cruce.	25
4.1.3.2 Mutación.	26
4.1.3.3 Elitismo.	27
4.1.3.4 Inversión.	28
4.1.4 Evaluación de la población.	28
4.2 LÓGICA FUZZY	29
4.2.1 Agrupamiento difuso “Fuzzy C – Means”.	31
4.3 REDES NEURONALES	35

4.3.1 La neurona.	35
4.3.2 Arquitectura de las redes neuronales artificiales.	37
4.3.2.1 Capa de entrada.	37
4.3.2.2 Capa de salida.	38
4.3.2.3 Capas ocultas.	38
4.3.3 Formas de aprendizaje.	39
4.3.4 Modo de operación.	40
4.3.4.1 Redes estáticas.	40
4.3.4.2 Redes dinámicas.	41
4.3.5 Información adicional.	41
4.3.5.1 Normalización.	41
4.3.5.2 Competencia.	41
4.3.5.3 Operación sincronía.	41
4.3.5.4 Operación asincrónica.	42
4.4 FPGA (FIELD PROGRAMMABLE GATE ARRAYS)	42
4.5 EVENTOS ALEATORIOS Y PROBABILIDAD	44
4.5.1 Posibilidad matemática.	45
4.6 INSTRUMENTACIÓN Y SENSORICA	45
4.6.1 Sistemas de medida.	46
4.6.2 Elementos necesarios en instrumentación.	46
4.6.2.1 Transductor.	46
4.6.2.2 Sensor.	47

4.6.2.3	Acondicionador de señales.	47
4.6.3	Características estáticas.	48
5	METODOLOGIA	49
5.1	CLONACIÓN ARTIFICIAL EN INGENIERIA	49
5.2	METODOLÓGIA DE TIPO GENÉTICO	51
5.2.1	Creación de los cromosomas.	53
5.2.2	Codificación de cromosomas	54
5.2.2.1	Codificación binaria.	54
5.2.2.2	Codificación no binaria.	55
5.2.2.3	Codificación mixta.	55
5.2.3	Búsqueda de consecuentes.	55
5.2.4	Identificación de operadores genéticos correctos.	56
5.2.5	Algoritmos genéticos.	57
5.2.6	Inicialización del algoritmo genético.	57
5.2.7	Evaluar características de rendimiento.	58
5.2.8	Calificación de la población.	58
5.2.9	Elitismo.	60
5.2.10	Selección de pares.	60
5.2.11	Operadores genéticos.	61
5.2.12	Identificación de clusters de salida.	62
5.2.13	Dispositivo clonado.	63
5.3	METODOLÓGIA FUZZY – GENÉTICO	65

5.3.1 Codificación del problema.	66
5.3.2 Operadores genéticos Fuzzy.	70
5.3.2.1 Cruce fuzzy.	72
5.4 METODOLOGÍA NEURONAL	73
6 IMPLEMENTACIÓN DE LA METODOLOGÍA	74
6.1 SIMULACIÓN DEL SENSOR	74
6.2 HARDWARE EVOLUTIVO.	77
6.3 PROCESO DE CLONACIÓN DEL SENSOR.	79
7 CONCLUSIONES	85
8 RECOMENDACIONES	87
BIBLIOGRAFIA	88
ANEXOS	91

LISTA DE TABLAS

LISTA DE FIGURAS

LISTA DE ANEXOS

RESUMEN

INTRODUCCIÓN

En la actualidad los procesos de automatización industrial utilizan diversos dispositivos de alta precisión, los cuales, debido a sus características, son de un alto costo, valor que en la mayoría de veces se ve incrementado por la dificultad de adquirir dichos dispositivos en el mercado tecnológico nacional, esto sin duda alguna, inhibe el desarrollo de las empresas colombianas, marginándolas de producciones más rentables.

Algunos de estos dispositivos tecnológicos de gran demanda por parte de la industria nacional son los controladores y sensores para procesos industrializados, estos elementos son la columna vertebral para todos los procesos de automatización, por lo que una reducción en el costo final de los mismos, se ve reflejada en la viabilidad económica de dicha automatización.

Una de las vías para llegar a esta reducción de costos es la clonación artificial de los dispositivos mencionados, por métodos y procedimientos que logran replicar las funciones de los controladores y sensores, estos aplican operadores genéticos que involucran técnicas avanzadas de Inteligencia Artificial (**IA**), lo que evita la violación de derechos intelectuales de los sensores y controladores.

La respuesta al problema planteado, es desarrollar mediante la aplicación de redes neuronales, lógica fuzzy, la introducción de sistemas de control distribuido e inteligente con variantes de control adaptativo y predictivo y algoritmos genéticos sistemas mixtos de hardware y software que permitirán reproducir la estructura inteligente de los sensores y controladores, con el fin de mostrar y emular el flujo de información interna del sistema, tomando como referencia las características del dispositivo real.

Uno de los resultados más interesantes al aplicar esta técnica de clonación artificial es que se puede observar que en la investigación similar desarrollada con éxito en la Refinería de Petróleo de Barrancabermeja, dependencia de ECOPETROL, se logro replicar un sensor de viscosidad para la planta de Visco Reducción, y posteriormente se crearon 7 replicas de dicho transductor obteniéndose una disminución de los costos del 75% del valor total del sensor para cada replica.

Para el área de controladores los ejemplos no están totalmente desarrollados, se tienen referencia de los futuros trabajos a realizar en la planta **C.E.S. Termopaipa IV**, en donde se espera, por métodos de clonación artificial la reducción hasta de un 50% en los gastos de automatización.

Por otra parte la implementación de esa novedosa técnica le permitirá al empresario racionalizar sus recursos para enfocarlos a otras áreas de la producción, poseer un mayor control de esta (tiene más dispositivos que miden y controlan las variables del proceso), mejorar la calidad de sus sistemas de adquisición de datos al poder invertir en mejores dispositivos los cuales serian clonados posteriormente.

Como consecuencia de lo anterior se hace indispensable incursionar en el campo del control inteligente para encontrar soluciones óptimas a las necesidades de las empresas nacionales; definir los métodos, protocolos y técnicas que permitan la clonación tanto de sensores como de controladores.

Esta propuesta de investigación busca definir dichos métodos y crear las herramientas básicas que permitan la repetibilidad de estas experiencias con un menor costo de investigación por parte de las empresas nacionales, rigiéndose únicamente con el protocolo que se generará de esta investigación, aplicando su resultado en la Refinería de Petróleo de Barrancabermeja, dependencia de ECOPETROL.

Se implementarán las técnicas de clonación artificial por medio de hardware (el cual puede ser PLC, Microcontroladores, FPGA, DSP), en el software se utilizarán lenguajes de programación tipo VHDL y lenguaje ensamblador para los dispositivos de menor capacidad de procesamiento.

Esto contribuirá a la reducción de costos de los dispositivos y a la facilidad de implementación por cualquier medio experto en el tema.

1 OBJETIVOS

1.1 OBJETIVO GENERAL

Diseñar, modelar y evaluar sistema de control digital de la viscosidad en la planta de viscorreducción basados en algoritmos genéticos y clonación artificial

1.2 OBJETIVOS ESPECÍFICOS

- Diseñar sistema de control de la viscosidad para *La Planta de Viscorreducción de la Refinería de Petróleos de Barrancabermeja*
- Realizar pruebas al sistema de control de la viscosidad clonado, para la evaluación de la efectividad del sistema, con respecto a los datos suministrados por la Empresa Colombiana de Petróleos ECOPETROL.
- Generar informes de datos experimentales sobre la aplicación y validación del prototipo digital de clonación artificial para controladores.
- Desarrollar una aplicación de diseño de controladores aplicando FPGA a través de algoritmos evolutivos y verificar el que se puedan clonar los controladores industriales.

2. PLANTEAMIENTO DEL PROBLEMA

Los procesos de automatización industrial utilizan dispositivos de alta precisión, los cuales son de un altísimo costo por su instalación, mantenimiento, confiabilidad y disponibilidad, además de su dificultad para conseguirlos en el mercado tecnológico nacional, es por esto que parte de las empresas colombianas están sujetas en su desarrollo tecnológico a la dependencia de los países productores de alta tecnología; especialmente en el campo de los sensores para procesos industrializados, en los cuales el país tiene altos intereses y busca aumentar el margen de productividad de sus recursos, y una mejor prestación de servicios, como por ejemplo en el campo de la explotación petrolífera.

Como consecuencia de lo anterior se hace indispensable incursionar en el campo del control inteligente para encontrar soluciones óptimas a las necesidades de las empresas nacionales. Actualmente en la Empresa Nacional de Petróleos ECOPETROL refinería de Barrancabermeja Santander, se tiene la necesidad de contar con un sistema de control en tiempo real, que permita reemplazar un analizador (sensor) de viscosidad por un sensor inteligente clonado a partir del dispositivo real.

Para el control adecuado de los procesos de la Industria del Petróleo, se requiere medir muchas variables para la toma de decisiones adecuadas, en cuanto a la producción en cantidad y calidad.

Estas variables van desde simples mediciones de flujos, presiones, niveles y temperaturas, tomados con instrumentos estándar que funcionan por la relación de mediciones físicas en forma indirecta, hasta otras mucho más complejas que

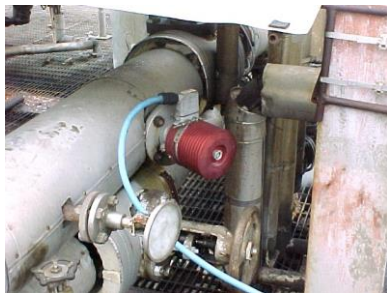
en ocasiones requieren análisis fuera de línea (en el laboratorio), pero que ocasionan retardos y muchas veces no son oportunos para tomar decisiones.

Las plantas petroquímicas requieren del análisis de los productos resultantes a fin de controlar su calidad para cumplir con las exigencias del mercado sin regalar calidad.

Para ello se utiliza:

Figura 1.

1. ANALISIS EN LINEA



Ventajas:

Dato Continuo

Desventajas:

Altos Costos por Adquisición, Instalación, Mantenimiento y Calibración del Equipo

Figura 2.

2. ANALISIS EN EL LABORATORIO



Ventajas:

Precisión

Desventajas:

Costos de los Análisis
Baja Frecuencia de Muestreo
Tiempo de entrega de los Resultados
Carencia de equipos especializados para Laboratorio

3. ANTECEDENTES

La Universidad Autónoma de Bucaramanga en su Facultad de Ingeniería Mecatrónica, ha venido desarrollando investigaciones en tecnologías de control inteligente utilizando herramientas como lógica difusa, redes neuronales y algoritmos genéticos fortaleciendo la Línea de investigación en AUTOMATIZACIÓN Y CONTROL del Grupo de Tecnologías de la Información. Los trabajos realizados hasta el momento han consistido en el uso y aplicación de Redes Neuronales en la implementación del desarrollo de Proyectos de Grado.

Se tiene como referente también los trabajos realizados por el Laboratorio Computo Especializado de la UNAB (**LCE**), y Instituto de Investigación y Desarrollo de Tecnologías Aplicadas (**IIDTA**) de la Universidad de Pamplona e investigaciones de la Universidad Nacional de Colombia.

En el campo de la ingeniería la clonación artificial, se realiza con la utilización de técnicas de inteligencia artificial creando réplicas de medios y sistemas, cumplan con funciones equivalentes.

4. MARCO TEORICO

4.1 ALGORITMOS GENÉTICOS

El término de Algoritmos Genéticos (**AG**), se utiliza para referirse a una técnica de búsqueda que incorpora el concepto de la selección natural en sus pasos iterativos. Es un método de búsqueda y optimización de características basados en la reproducción natural, las cuales son sencillamente operaciones sobre el material genético de cada individuo, estas operaciones de reproducción se realizan rápidamente (a diferencia de la naturaleza), lo que permite una amplia exploración de diversas soluciones a los problemas a optimizar en un tiempo de computo relativamente bajo.

La idea de utilizar la naturaleza como fuente de inspiración para una técnica de búsqueda y optimización nació de la imaginación de John Holland quien describió por primera vez sobre los algoritmos adaptativos en 1962, para posteriormente desarrollar el teorema fundamental de los algoritmos genéticos en 1975.

El desarrollo de Holland permite definir la problemática del algoritmo genético en unos pocos pero sustanciales pasos, el primero de ellos consiste en hallar los parámetros de los cuales depende el problema, posterior a esto viene la tarea de codificarlos en un cromosoma, y finalmente aplicar métodos de evolución (selección, reproducción sexual, mutación entre otros) permitiendo un intercambio de información y alteraciones que generan diversidad.

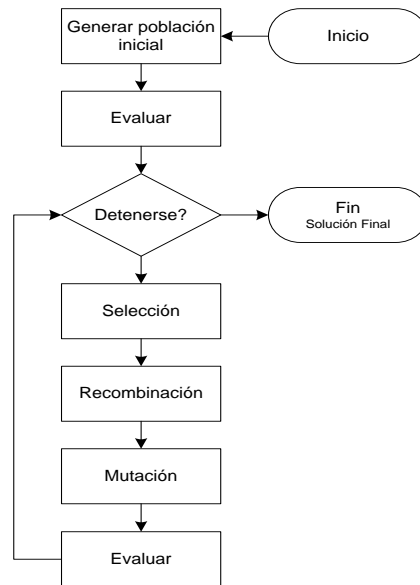
Esta metodología busca la combinación de las características óptimas de las soluciones parciales obtenidas en el proceso, cabe también mencionar, que en la combinación de características, es posible obtener soluciones de peor desempeño, esto permite la búsqueda de soluciones fuera del mini-universo de soluciones encontrados (es interesante observar como los AG ganan cada día mas terreno en la búsqueda de soluciones a problemas con un gran "universo" de respuestas).

Al mismo tiempo, la utilización de la mutación como operador genético, complementa ampliamente la necesidad buscara soluciones nuevas en espacios aun no explorados.

Si bien el AG es una representación (o intenta serlo en su estructura) de la evolución de los seres en la naturaleza, este proceso debe tener un fin, el cual le permite al usuario, obtener una solución óptima a su problema (el mejor individuo de la última población encontrada), este criterio, con el cual se da fin al proceso de evolución, varían de usuario a usuario, pero se resaltan los que implementan un máximo tiempo de iteración, un mínimo de "desempeño" por individuo y encontrar una población ampliamente semejante; todas estas características, permiten encontrar de una manera rápida y económica (hablando en términos computacionales) una solución, que en la mayoría de los casos debe estar ajustada a los deseos del programador.

El proceso que ejecuta el algoritmo genético, desde su inicio, hasta su finalización se aprecia en la figura 3.

Figura 3. Diagrama de flujo de un AG



4.1.1 Generación de la población inicial.

Para crear una población inicial, el primer paso que se debe realizar es la codificación de la solución al problema que hemos planteado, este es uno de los pasos más importantes, pues gracias a que tan bien se realice esta codificación se obtendrán resultados óptimos o por el contrario, luego de un amplio número de iteraciones, no se encontrará ningún tipo de solución.

4.1.2 Codificaciones del problema.

Existe un número variado de tipos de problemas, al igual que una cantidad de formas de codificación de los mismos, pero debido a que la implementación de los AG es netamente computacional (realizar este tipo de búsqueda utilizando otros

medios sería sencillamente ineficiente) la forma de codificación más implementada es la codificación binaria, esta incluye otras ventajas como su simplicidad a la hora de la manipulación con operadores genéticos, su facilidad para convertir esta información de binaria a entero o real.

Es necesario mencionar, que existen problemas en la que una codificación binaria no es la adecuada, esto se debe a que la codificación del dominio se define mejor con algún tipo de otro código, en algunos casos se utiliza la codificación por números, esta es útil para determinar la secuencia lógica de un proceso, o en el caso de esta investigación se implementa la codificación de estados *fuzzy*, genéticos e inclusive Redes neuronales, donde cada cromosoma, representa dichas características. Estos tipos de representación se denominan representación no binaria.

Finalmente, es necesario mencionar que existe un tercer tipo de codificación que es la llamada codificación mixta, en la cual se pretende utilizar los dos tipos de representación expuestos anteriormente. Los tipos de representación los podemos apreciar en la **Figura 4**.

Figura 4. Codificaciones.

Binaria	1	0	0	1	0	0
No binaria	A	B	C			
Mixta	A	B	10	1	0	

Posterior a la identificación de las necesidades del problema y a la selección de la codificación del mismo, es necesario generar la primera población, esta generalmente se crea de forma aleatoria, permitiendo una generación de semillas en un amplio dominio del problema, también se encuentran poblaciones iniciales

en las que "la mano del creador" tiene algo que ver, al utilizar soluciones (individuos) que desde el punto de vista del desarrollador pueden aportar información inicial relevante.

Para la presente investigación se utilizarán los algoritmos genéticos para la aplicación e interpretación del mapeo genético, que contiene los códigos de la estructura funcional del controlador o sensor. El mapeo es un conjunto de barras de códigos que describe las unidades operativas funcionales del controlador o sensor. Cada unidad operativa esta formada por elementos unitarios que representan una parte de la operación del dispositivo, lo que permitirá estudiar y optimizar la repercusión de cada variable en la salida del controlador o sensor.

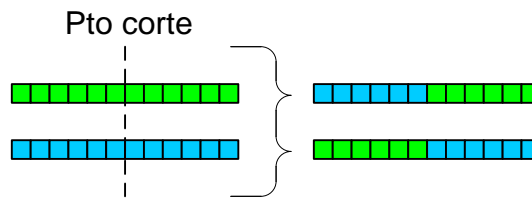
4.1.3 Operadores Genéticos.

Los operadores genéticos, son sin lugar a dudas los elementos más importantes del proceso de clonación, esto se debe a su importancia en la conversión de los cromosomas de entrada en cromosomas de salida, algunos de los operadores más conocidos son: Cruce, Mutación, Elitismo, de la misma manera introducimos algunos otros operadores utilizados en el proceso de clonación (inversión).

4.1.3.1 Cruce: Este es un operador muy importante, pues es el encargado de realizar la optimización de los individuos mezclando las características (buenas y malas) de respectivos padres, lo cual busca preservar las buenas características al aplicar este operador a excelentes padres y crear mejores individuos con el transcurso de generaciones.

Una de las metodologías aplicadas en el procedimiento de clonación artificial, es el cruce con un punto de corte, este punto de corte se ubica aleatoriamente a lo largo del tamaño del cromosoma, y se realiza una combinación de código basándose en el punto de corte, este efecto se aprecia en la **Figura 5**.

Figura 5. Cruce



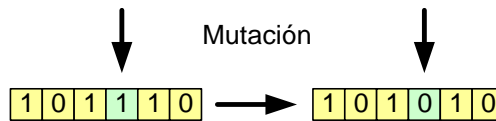
Este tipo de cruce se utiliza por la facilidad de computo (existe otras modalidades que utilizan varios puntos de cruce) y facilidad de combinar diferentes características (al elegir el punto de corte de forma aleatoria).

4.1.3.2 Mutación: La mutación es un operador genético el cual, le permite al algoritmo realizar búsquedas en espacios de la solución que no se han tenido en cuenta, es sencillo encontrar la importancia de la mutación al recordar que algunas características que hacen mejorar a los individuos no se encuentran precisamente en las poblaciones actuales, hace falta la aplicación de mutaciones para encontrar estas características.

Cabe recordar que en la mayoría de las veces, las mutaciones generan individuos que no se desempeñaran de la mejor manera, pero algunas veces, se encuentran características que no habían sido tenidas en cuenta.

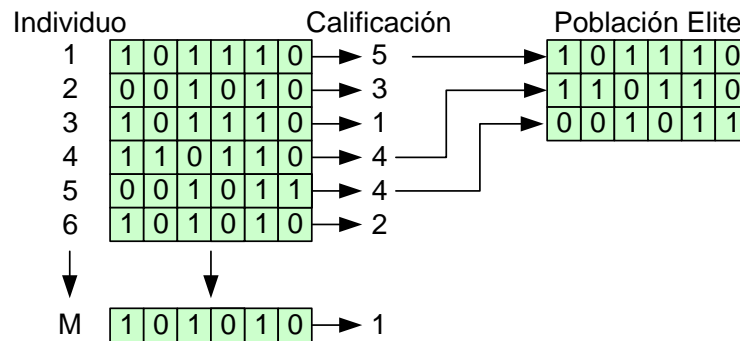
Un proceso de mutación clásico se aprecia en los cromosomas tipo binario, esto se observe en la **Figura 6**.

Figura 6. Mutación



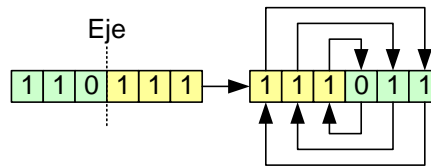
4.1.3.3 Elitismo: El elitismo es una de las características más importantes para nuestro proceso de clonación, esta metodología o operador genético, garantiza que el AG siempre mejorara o mantendrá su “desempeño”, esto se logra gracias a la selección y preservación de los N mejores individuos, donde N , es menor al tamaño de la población M , este operador se aprecia de mejor manera en la Figura 7.

Figura 7. Elitismo



4.1.3.4 Inversión: Este operador genético, se encarga de intercambiar información con la parte posterior del cromosoma (consecuentes) y la parte inicial del mismo (antecedentes), de esta forma se busca emular el comportamiento de los consecuentes basándose en la inclusión de estas características en los antecedentes, gráficamente podemos imaginar el uso de un espejo en todo el centro del cromosoma, esto se observa en la Figura 8.

Figura 8. Inversión



4.1.4 Evaluación De La Población

Esta evaluación de la población, es la característica principal y primordial de los algoritmos genéticos, generalmente es denominada función de evaluación y o "fitness functions", donde esta función representa una medida de cómo cada individuo se desempeña en el dominio del problema o un tipo de calificación similar a la de un juego, en muchos casos la correcta implementación de esta función, permite encontrar una solución al problema planteado (a la búsqueda planteada) y esta función es la que en instancias finales, repercutirá en la "supervivencia" de cada individuo, algunos autores hacen referencia a esta característica denominando a la función de evaluación como el "medio ambiente" del problema.

En la evaluación también se pueden utilizar técnicas de crianza, es decir, se premian ciertas características que sin afectar el desempeño de la solución, son individuos mejores adaptados, es el ejemplo de individuos que requieran menor tiempo de procesamiento, soluciones menos complejas (en el caso de la codificación variable) entre otros casos, todo con el fin de obtener la mejor solución a los problemas planteados.

4.2 LÓGICA FUZZY

Uno de los primeros conceptos que se debe conocer para entender la lógica difusa es el de conjunto difuso. Básicamente, un conjunto difuso es un conjunto que no tiene límites claramente definidos o precisos.

A diferencia de los conjuntos clásicos, en los conjuntos difusos la transición de la pertenencia o no-pertenencia de un elemento a un cierto conjunto, es gradual, y esta transición está caracterizada por funciones de membresía, las cuales les dan a los conjuntos difusos flexibilidad para modelar expresiones lingüísticas empleadas cotidianamente. Sea X una colección de objetos denotados genéricamente por x ; entonces, un conjunto difuso A en X se define como el siguiente conjunto de pares ordenados:

$$A = \{(x, \nu_A(x) | x \in X)\}$$

Donde $\nu_A(x)$ se conoce como la función de membresía (MF por sus siglas en ingles) del conjunto difuso A . La MF transforma cada elemento de X hacia un grado de membresía (o valor de membresía) entre 0 y 1.

En otras palabras identifica que tanto pertenece este valor x al conjunto difuso A , si tiene un total grado de pertenencia (se denota con el valor de 1) o si por el contrario no pertenece a este conjunto (asignándole un valor de 0).

Generalmente, X es llamado el universo de discurso o, simplemente, el universo, el cual puede consistir de objetos discretos (ordenados o no ordenados) o ser un espacio continuo, este universo debe estar cubierto por conjuntos difusos, los cuales representan la lógica difusa.

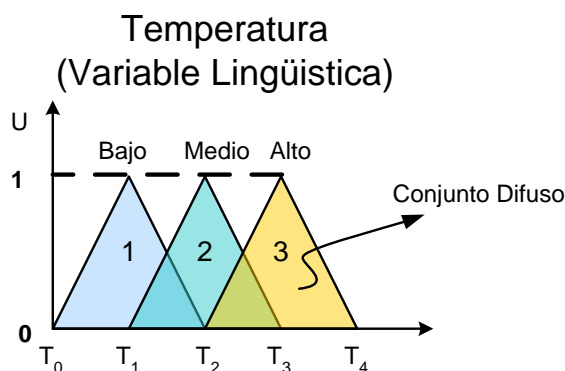
Por ejemplo, el universo ordenado discreto definido por el número de hijos que una pareja puede desear tener, se expresa como: $X = \{0, 1, 2, 3, 4, 5, 6\}$. De esta forma, el conjunto $A =$ "número ideal de hijos en una familia" se puede describir como:

$$A = \{(0, 0.1), (1, 0.3), (2, 0.7), (3, 1), (4, 0.6), (5, 0.2), (6, 0.1)\}$$

Donde los grados de membresía asignados, son medidas subjetivas.[Are00].

Los elementos de la lógica difusa, se aprecian en la **Figura 9**, en la que se introduce una variable lingüística denominada "Temperatura", con tres conjuntos difusos (Bajo, Medio y Alto).

Figura 9. Lógica fuzzy



4.2.1 Agrupamiento Difuso "Fuzzy C-Means".

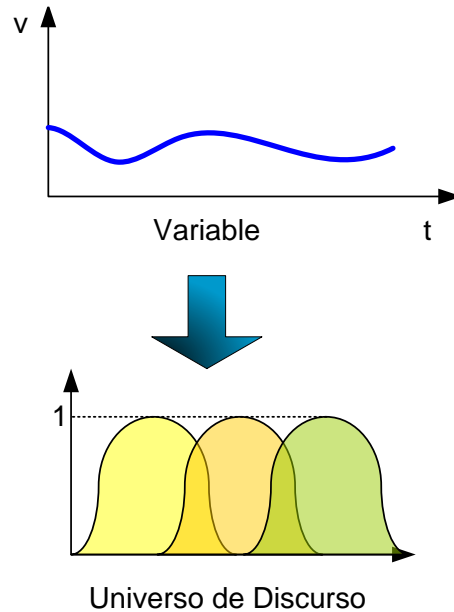
Son las relaciones funcionales transformadas en cluster fuzzy, con el objetivo de estructurar el comportamiento FUNCIONAL tanto de la entrada como de la salida.

El agrupamiento difuso o también llamado "Clustering", es una metodología, que permite la partición del universo de discurso en diferentes grupos; el propósito de este agrupamiento es identificar los "grupos naturales" de datos a lo largo de un "data set", donde este grupo representa fielmente el comportamiento del sistema. Para realizar esta operación, se toma un universo de discurso Z , en el cual se define una *c-partición* como una recopilación de c conjuntos difusos definidos sobre Z , y con funciones de pertenencia $u_1(x)$, $u_2(x)$, ..., $u_c(x)$ respectivamente, esto será de gran utilidad siempre y cuando los conjuntos difusos cumplan con las siguientes características:

- Para todos los elementos de Z , la suma de sus funciones de pertenencia a los c conjuntos debe ser 1, esto quiere decir que para todo valor del universo de discurso, se encuentra un grado correspondiente a cada conjunto c .
- Para todos los conjuntos debe existir al menos un elemento de Z tal que su función de pertenencia al conjunto sea distinta de cero, por lo que cada conjunto tiene elementos "activos" en el universo de discurso.

El resultado esperado del proceso de agrupamiento se aprecia en la **Figura 10**

Figura 10. Representación de una magnitud



Como es evidente, el procedimiento, busca poder clasificar de la mejor manera los datos de entrada en respectivos clusters, estos clusters o conjuntos de agrupamiento, serán posteriormente reflejados en conjuntos difusos, este último paso, revela la gran importancia de la teoría de agrupamiento.

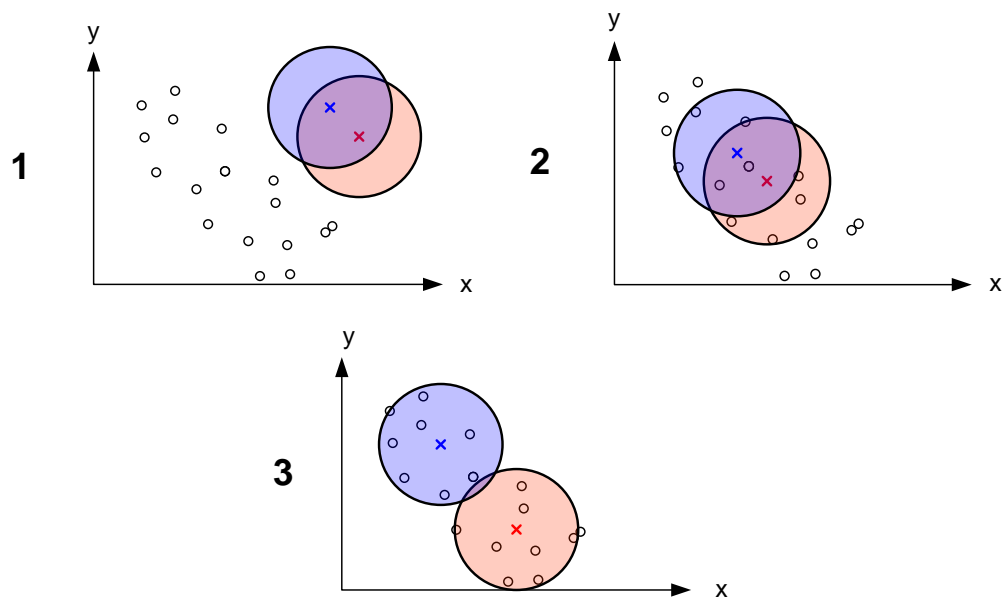
Para realizar este tipo de “*Clustering*” se utilizan diversas técnicas, una de las más populares es la denominada “*Fuzzy c-means*” esta es una técnica de agrupamiento de datos, en la cual a cada dato le corresponde un grado de pertenencia. Esta técnica fue originalmente introducida por Jim Bezded en 1981, como un mejoramiento a los métodos iniciales de “*Clustering*”, el gran aporte de esta metodología radica en su facilidad para agrupar datos de espacios multidimensionales dentro de un número de diferentes clusters.

Para implementar el “*Fuzzy c-mean*”, se deben realizar los siguientes pasos:

1. Seleccionar el número de clusters
2. Poner en el espacio los centros de los clusters aleatoriamente.
3. Asignar un grado de membresía a cada punto dependiendo de su distancia a cada centro.
4. Calcular nuevamente los centros.
5. Repetir los pasos hasta que los centros no cambien significativamente.

Este procedimiento se ilustra ampliamente en la **Figura 11**, para este caso se estudio, en el primer paso, se eligieron centros aleatorios, posterior a esto, se realiza la primera iteración, se calculan los grados de pertenencia y se vuelven a recalculara los respectivos centros de clusters (para nuestro ejemplo, fueron seleccionados 2 clusters), esto representa el 2 paso.

Figura 11. Proceso de Fuzzy c-mean



Finalmente, luego de varias iteraciones, la posición de los centros no varía significativamente, por lo que el proceso de cálculo de los centros se da por terminado, esto se aprecia en la figura en el paso 3, en este punto, todos los datos, tienen un grado de pertenencia respecto a cada cluster, donde el centro del cluster, define un grado de pertenencia igual a 1, grado de pertenencia que disminuye proporcionalmente a medida que se aleja del centro del cluster.

$$m_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{q-1}}} \quad \text{donde } d_{ik} = \|u_k - c_i\|$$

Donde:

m_{ik} = Grado de membresía para el punto k en el cluster i.

d_{ik} = Distancia entre el punto k y el centro del cluster i.

d_{jk} = Distancia entre el punto k y el centro del cluster j.

Para realizar este tipo de agrupamientos, Bezdek y sus colegas [Bezdek01] desarrollaron un algoritmo para obtener una c-partición para un Universo de Discurso finito; dicho algoritmo se conoce como el *agrupamiento "fuzzy c-means"*, y es uno de los más populares, pese a que en la literatura se reportan muchos otros algoritmos distintos.

4.3 REDES NEURONALES.

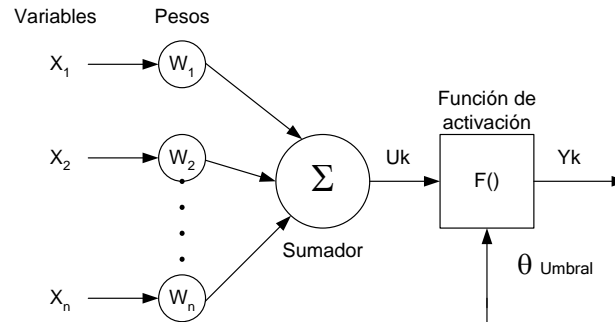
Las redes neuronales artificiales consisten en la implementación de una técnica que permite imitar el cerebro, específicamente la disposición física de las neuronas y su interconexión, pues es debida a esta intrincada conexión como podemos realizar tareas cognoscitivas, como hablar, reconocer un rostro entre otras, he aquí la importancia de esta técnica, la emulación del cerebro humano.

Al imitar esta disposición de elementos de computo (neurona) las redes neuronales artificiales poseen una característica que las convierte en piezas indiscutibles a la hora de realizar dispositivos con algún nivel de IA, su capacidad de aprendizaje. Para la presente investigación se utilizara la gran habilidad de las redes neuronales artificiales para encontrar las respuestas correctas que debe proporcionar el controlador o sensor clonado, basándose en las entradas y las salidas del dispositivo ha clonar, requiriendo según sea el caso de otro tipo de entradas que ayudaran a inferir el valor deseado, lo que contribuirá a la disminución de costos del dispositivo clonado.

4.3.1 La Neurona.

Una neurona es una unidad de procesamiento de la información, la cual es el eje central de una red neuronal. La composición típica de una neurona se puede observar en la Figura 12, el cual es un modelo de neurona con sus tres elementos fundamentales.

Figura 12. Representación de una neurona



Un conjunto de conexiones, cada una de ellas caracterizada por su fuerza o peso. Lo que le permite a una señal de entrada X_n tras pasar la conexión W_k , convertirse en una señal $X_n W_k$, por lo que si el peso W_k es positivo, tiene un efecto estimulador y si su peso es negativo o muy pequeño, tiende a "pasar" por alto esta entrada.

El segundo elemento es un sumador, que produce la suma ponderada de las entradas de acuerdo a los correspondientes pesos de las conexiones dando a su salida un único valor. Este valor es entregado al tercer elemento, una función de activación o también denominada de transferencia, que tiene como misión limitar la amplitud de la salida generada por la neurona.

Es habitual la inclusión en el modelo un umbral o polarización representado por cuya misión es controlar el nivel a partir del cual la neurona produce su salida. Este valor es generalmente añadido a la suma ponderada, que posteriormente es transformada por la función de activación.

De acuerdo con este modelo, se puede describir el comportamiento de la neurona mediante las siguientes expresiones:

$$U_k = \sum_{j=1}^n X_j \cdot W_j$$

$$V = U_k - \theta$$

$$Y_k = F(V)$$

4.3.2 Arquitecturas de las Redes Neuronales Artificiales-

Muchas veces se habla de arquitectura de una red neuronal. Este concepto se refiere a la manera en que se interconectan los distintos elementos llamados neuronas, que forman la red.

Normalmente los elementos de proceso se organizan como una secuencia de capas con un determinado patrón de interconexión entre los diferentes elementos de proceso que las forman, y con un modelo de conexión entre los elementos de proceso de las distintas capas. Uno de los rasgos que puede ayudar a definir una capa es el hecho de que todos los elementos de proceso que la forman usan la misma función de transferencia.

En la gran mayoría de las arquitecturas de las redes neuronales, se pueden hacer las siguientes distinciones entre capas:

4.3.2.1 Capa de entrada. Es la capa que recibe los estímulos del entorno o las variables a estudiar. No suele tener asociado un mecanismo de aprendizaje como tal, y su función es la de participar como receptora y acondicionadora de las entradas, su misión simplemente es la de distribuir cada entrada al resto de los elementos de proceso que constituyen la red.

4.3.2.2 Capa de salida. Es la capa sobre la que se forman las salidas de la red, en ella se encuentra el valor correspondiente a un estímulo (dado por las entradas) esta capa es conformada generalmente por una neurona, pero la arquitectura es flexible para la presentación de diversas neuronas a la salida.

4.3.2.3 Capas ocultas. Son las demás capas que no son ni de entrada ni de salida, en ellas está condensado todo el aprendizaje y la "maquinaria" de funcionamiento de la red neuronal, es en esta capa donde radica la capacidad de adaptarse al entorno, aprendiendo a proporcionar la respuesta adecuada ante los estímulos que recibe de sus entradas. Este aprendizaje se plasma en la modificación de los pesos de las conexiones entre los distintos elementos que forman la red, agregando valor a ciertos tipos de conexiones y debilitando otro tanto.

Existen muchos tipos de aprendizaje dependiendo del modo en que es realizado el ajuste de los pesos que como se dijo anteriormente es el encargado del proceso de adaptación. En un principio, los pesos pueden ser considerados parámetros libres, inclusive aleatorios, aunque es posible, fijar restricciones a los valores iniciales de los pesos, o a los valores que puedan tomar a lo largo del proceso de aprendizaje.

Un conjunto de reglas bien definidas que describen el método de adaptación o modificación de los pesos de acuerdo con el entorno en el que se encuentra sumergida la red recibe el nombre de regla de aprendizaje, y su transcripción en forma de procedimiento se denomina algoritmo de aprendizaje. Existe una relación muy fuerte entre la arquitectura de una red neuronal artificial y el o los algoritmos de aprendizaje que puede usar, de tal modo que diferentes arquitecturas de redes neuronales requieren diferentes algoritmos de aprendizaje.

En el sistema que se ha definido anteriormente como red neuronal artificial se puede encontrar una serie de rasgos funcionales presentes en muchas de las implantaciones actuales. Dichas características operacionales se describen en los siguientes puntos.

4.3.3 Formas de aprendizaje.

Cada red neuronal tiene un modelo o paradigma con el cual aprende y por ende trabaja; Una de las principales ideas sobre las que se basan las redes neuronales artificiales es la de responder a los estímulos de las entradas mediante un proceso de aprendizaje por el cual va adaptando los pesos de las conexiones de sus elementos de proceso, de tal forma que la red aprende o mejor "memoriza" los ejemplos de entrenamiento que se le presentan. Estas formas de aprendizaje pueden ser de dos tipos supervisadas y no supervisadas.

El entrenamiento o aprendizaje supervisado consiste en presentar los conocimientos en forma de pares de [entrada, salida deseada]; si la entrada es distinta de la salida, se tendrá una red heteroasociativa; si la entrada es igual que la salida, se tratará de una red autoasociativa. Con este modelo, se busca la memorización de los ejemplos, para que posteriormente la red responda igual a como lo haría el modelo; es evidente que la red neuronal responderá en mejor medida siempre y cuando los ejemplos hallan sido significativos y lo suficientemente variados. Sigue el aprendizaje no supervisado como el segundo paradigma, durante este proceso de aprendizaje a la red no se la presenta la salida deseada.

El instructor o el algoritmo exterior sólo señalan cuando la salida es correcta o no, pero no indica en cuanto se diferencia de la salida buscada. Si se compara este paradigma con el supervisado, se observa que si bien el supervisado proporciona

una información relativa a la dirección en la que se deben realizar los cambios en el sistema (ajuste de los pesos), en el caso de un aprendizaje por refuerzo no se tiene información acerca de la “dirección” del cambio, lo cual hace que su ámbito de aplicación sea mucho más reducido comparado con el modo supervisado, aunque presenta interés en la comunidad científica dedicada al estudio de las máquinas capaces de aprender.

4.3.4 Modo de operación.

Al hablar de modo de operación se está haciendo referencia a la manera en que la red neuronal procesa los estímulos externos y crea la respuesta de salida. Puede considerarse a una red neuronal como perteneciente a una de dos grandes categorías:

4.3.4.1 *Redes estáticas.* En este tipo de red una vez establecido el valor de las entradas las salidas alcanzan un valor estacionario independientemente de las entradas en el instante anterior, y en un tiempo siempre por debajo de una determinada cota. Estas redes se pueden caracterizar estructuralmente por la inexistencia de bucles de realimentación y de elementos de retardo entre los distintos elementos de proceso que las forman.

Debido a su modo de funcionamiento, estas redes tienen una capacidad limitada para sintetizar funciones dependientes del tiempo en comparación con lo que se expone en el siguiente punto.

4.3.4.2 Redes dinámicas. Este tipo responde de manera diferente ante diferentes secuencias de entradas, haciendo uso de manera implícita o explícita de la variable *tiempo*. Este aspecto las hace en principio más idóneas que las redes estáticas para la síntesis de funciones en las que aparezca de alguna manera el parámetro tiempo.

4.3.5 Información adicional.

Existen las denominadas operaciones de capa, estas consisten en operaciones que afectan a la capa como un todo. En principio se considerarán las siguientes:

4.3.5.1 Normalización. Cada elemento de proceso de la capa considerada ajusta su salida para dar un nivel constante de actividad (considerando las salidas de todos los elementos de proceso que forman la capa).

4.3.5.2 Competencia. Sólo uno o unos pocos elementos de proceso de una capa ganan y producen salida, inhibiendo la generación de las salidas en el resto.

También es posible caracterizar a las redes neuronales teniendo en cuenta la forma de operar a la hora de generar la salida o al actualizar los pesos. Así se tiene:

4.3.5.3 Operación síncrona. Se dice que una red neuronal opera de forma síncrona cuando todos los elementos de proceso del sistema generan la salida a la vez.

4.3.5.4 Operación asíncrona. Se dice que una red opera de forma asíncrona cuando los elementos de proceso que la constituyen generan la salida aleatoriamente e independientemente unos de otros.

En estos casos puede añadirse a los elementos de proceso entradas de control que indiquen cuando han de ser actualizados los pesos de sus conexiones con los otros elementos de proceso.

Téngase en cuenta que en muchos modelos de redes la actualización de los pesos requiere la ejecución de la red a fin de calcular el correspondiente error, luego el cambio en el valor de los pesos también queda influido por el modo de operación síncrona o asíncrona.

4.4 FPGA (FIELD PROGRAMMABLE GATE ARRAYS).

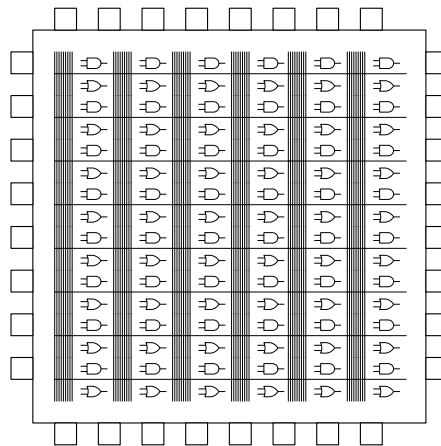
En el año de 1985 una compañía llamada Xilinx introdujo un nuevo concepto en lo que a dispositivos lógicos se refería, el concepto consistía en combinar el control que hacía el usuario (desarrollo de la aplicación) con un menor tiempo empleado para tal fin lo denominó FPGA. La gran mayoría de los desarrolladores empezaron entonces a utilizar los FPGA generando una nueva cantidad de productos y aplicaciones.

Una FPGA es una estructura de celdas lógicas o módulos interconectados a deseo del diseñador, esto quiere decir que el usuario puede diseñar, programar y hacer sus propios cambios en su circuito cuando el lo cree conveniente, lo que permite la implementación de esta tecnología en sistemas en donde la rápida elaboración y la precisión sean necesarios.

Actualmente se cuentan con FPGAs que superan los 10 millones de compuertas¹, esto en un principio puede sonar exagerado, pero es evidente que este tipo de dispositivos posee una gran flexibilidad a la hora de realizar aplicaciones.

Existen dos tipos de FPGAs, las basadas en SRAM y las OTP (One-time Programmable), cada una de ellas contiene diferentes tipos de mecanismos de interconexión e implementación, pero la gran diferencia radica en que las primeras de ellas son reprogramables por el diseñador pero pierden su "programa" cada vez que el dispositivo es apagado, es por esto que se encuentran generalmente apoyadas en una memoria serial PROM, muy al contrario de sus hermanas la OTP, que mantiene el programa indefinidamente

Figura 13. Estructura de la FPGA



¹ Fuente www.xilinx.com

4.5 EVENTOS ALEATORIOS Y PROBABILIDAD

Un evento aleatorio es un evento el cual tiene una oportunidad de ocurrir, y la probabilidad es una medida numérica de esa oportunidad. La probabilidad es un número entre 0 y 1, ambos inclusive; los valores más altos indican una mayor oportunidad. Se escribe $P(A)$ para denotar la probabilidad de que un evento A ocurra; $P(A+B+\dots)$ para la probabilidad de que al menos uno de los eventos A, B, \dots ocurran; $P(AB\dots)$ para la probabilidad de que todos los eventos A, B, \dots ocurran; y $P(A|B)$ para denotar la probabilidad de que el evento A ocurra cuando se conoce que el evento B ocurrió, $P(A|B)$ se llama probabilidad condicional de A dado B . Los axiomas más importantes que gobiernan la probabilidad son:

$$P(A+B+\dots) \leq P(A)+P(B)+\dots$$

$$\text{y } P(AB) = P(A|B) \cdot P(B).$$

Si solamente uno de los eventos A, B, \dots puede ocurrir, ellos se llaman excluyentes. Si al menos uno de los eventos A, B, \dots tiene que ocurrir ellos se llaman completos.

Una función de probabilidad es una función que asigna a cada suceso A en C (Colección de todos los eventos) un número $P(A)$, llamado probabilidad del suceso A , la cual cumple los axiomas siguientes:

- i) $P(A) \geq 0$ para todo A en C ,
- ii) $P(\Omega) = 1$ donde Ω denota al suceso seguro,
- iii) Para toda sucesión numerable de sucesos disjuntos $A_1, A_2, \dots, A_n, \dots$

$$P\left(\sum_{n=1}^{\infty} A_n\right) = \sum_{n=1}^{\infty} P(A_n)$$

4.5.1 Posibilidad Matemática.

Posibilidad y probabilidad son dos términos muy parecidos utilizados en algunos casos de manera indiferente, pero la probabilidad es la asignación de una medida estadística cuantificable, la posibilidad no. Por eso se dice que todo en la vida es posible, pero no todo es probable.

La posibilidad ha de contar con aquello que le es entregado y con las circunstancias que limitan sus posibilidades y su libertad.

El término posibilidad se refiere a lo que satisface a las condiciones generales impuestas a un orden de realidad o que satisface las condiciones generales de la experiencia.

4.6 INSTRUMENTACIÓN Y SENSÓRICA

Las aplicaciones de la electrónica, presentes actualmente en diferentes aspectos de la vida cotidiana, no serían posibles sin los sensores.

Sin la capacidad que éstos ofrecen de medir las magnitudes físicas para su conocimiento o control, muchos dispositivos electrónicos no serían más que simples curiosidades de laboratorio.

La utilización de los sensores contrasta con la escasa bibliografía que se encuentra sobre ellos, en particular desde la perspectiva de la Ingeniería Electrónica.

4.6.1 Sistemas de Medida.

Se denomina sistema a la combinación de dos o más elementos, subconjuntos y partes necesarias para realizar una o varias funciones. En los sistemas de medida, es la asignación objetiva y empírica de un número a una propiedad o cualidad de un objeto o evento, de tal forma que la describa.

Es decir, el resultado de la medida debe ser independiente del observador (objetiva, basada en la experimentación (empírica), y de tal forma que exista una correspondencia entre las relaciones numéricas y las relaciones entre las propiedades descritas.

4.6.2 Elementos necesarios en instrumentación.

4.6.2.1 *Transductor:* Es el elemento encargado de modificar la naturaleza de la señal que proporciona el sensor para hacerla más fácilmente medible. En general se denomina transductor a todo dispositivo que convierte una señal de una forma física a una señal correspondiente, pero de otra forma física distinta. Es por lo tanto un dispositivo que convierte un tipo de energía en otro.

Dado que hay seis tipos de señales: mecánicas, térmicas, magnéticas, eléctricas, ópticas y moleculares (químicas), cualquier dispositivo que convierta una señal de un tipo en una señal de otro tipo debería considerarse un transductor, y la señal de salida podría ser cualquier forma física útil.

4.6.2.2 Sensor: Dispositivo que está en contacto con la variable que se mide. Un sensor es un dispositivo que a partir de la energía del medio donde se mide, da una señal de salida transducible que es una función de la variable de medida.

Sensor y transductor se consideran a veces como sinónimos, pero sensor sugiere un significado más extenso: la ampliación de los sentidos para adquirir un conocimiento de cantidades físicas que por su naturaleza y tamaño no pueden ser percibidas directamente por los sentidos. El transductor, en cambio sugiere que la señal de entrada y la de salida no deben de ser homogéneas.

4.6.2.3 Acondicionadores de señales: Los acondicionadores de señales, adaptadores o amplificadores, en sentido amplio son los elementos del sistema de medida que ofrecen, a partir de la señal de salida de un sensor electrónico, una señal apta para ser registrada o que simplemente permita un procesamiento posterior mediante un equipo o instrumento estándar. Consiste generalmente en circuitos electrónicos que ofrecen, entre otras funciones, las siguientes: amplificación, filtrado, adaptación de impedancias, y modulación y demodulación.

4.6.3 Características estáticas.

El comportamiento del sistema de medida viene condicionado por el sensor empleado.

Es por eso importante describir las características de los sensores, Sucede que en la mayoría de los sistemas de medida, la variable de interés varía tan lentamente que basta conocer las características estáticas del sensor, estas describen la actuación del instrumento en régimen permanente.

Ahora bien, las características estáticas incluyen también en el comportamiento dinámico del sensor, es decir, el comportamiento que se presenta cuando la magnitud medida varía a lo largo del tiempo. No obstante, se suele evitar su consideración conjunta por las dificultades matemáticas que entraña, y se procede a la distinción entre las características estáticas y las características dinámicas.

5. METODOLOGÍA

5.1 CLONACIÓN ARTIFICIAL EN INGENIERIA

La metodología de clonación aquí propuesta, permite la clonación de dispositivos como sensores y controladores, por lo que consiste en una metodología completa para el reemplazo de los elementos anteriormente mencionados. Este procedimiento se observa a continuación y se aprecia en la **Figura 14**.

- Creacion de los clusters

- Creacion de los cromosomas.

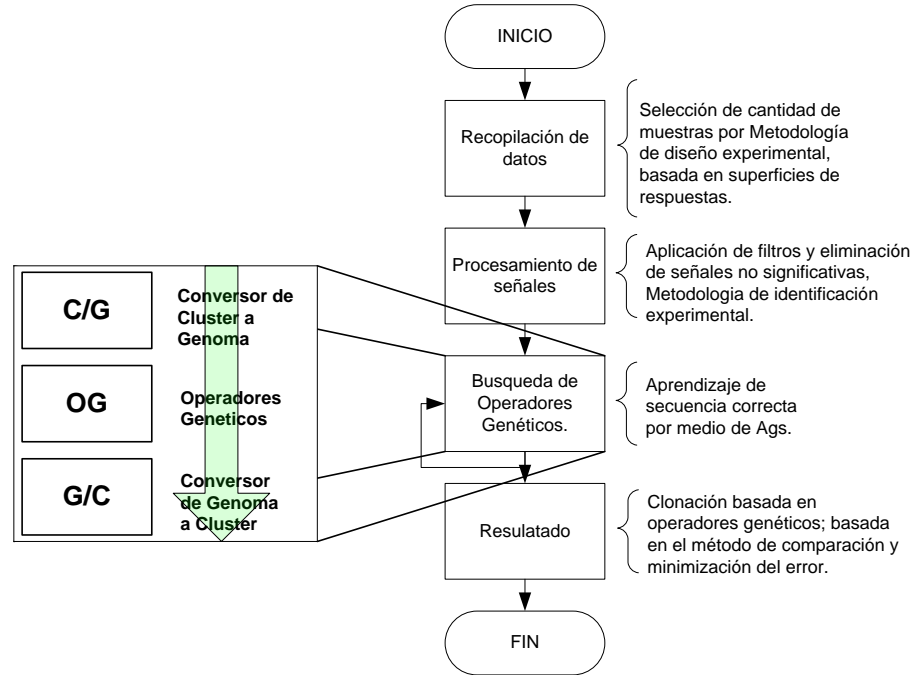
- Busqueda de consecuentes

- Identificar los operadores correctos
 - Crear algoritmo genetico que busque la secuencia.
 - Evaluar los individuos
 - Seleccionar los mejores (evaluados según error).
 - Reproducir y generar nueva poblacion.
 - Repetir ciclo optimizando el error.

- Identificación de valores de salida

- Defusificación

Figura 14. Metodología utilizada



El primer paso del proceso de clonación, consiste en la recopilación de datos, esta se fundamenta en la selección de una cantidad de muestras representativas del tipo de dispositivo a clonar, para colocar un ejemplo más claro, se puede tomar como referencia las variables (en el ejemplo de un sensor) representativas en el proceso, estas pueden ser seleccionadas con la ayuda del experto o utilizando técnicas de correlación para tal fin, seguido de esta selección, se procede a implementar el preprocesamiento de la señal, lo que permitirá trabajar con unas señales más "limpias" y coherentes a la realidad.

Realizado los dos primeros pasos, los cuales consisten más en una selección y preprocesamiento de las señales, se ejecuta la segunda etapa de clonación, el primer paso reside en crear los clusters para los valores de las entradas y salidas

(independiente del número de estas, lo que conlleva a ser una metodología multivariable), identificando señal por señal, entrada por entrada y salida por salida, los clusters más adecuados para cada uno de ellos.

La tercera etapa, es la que tiene que ver más con el trabajo propio de la investigación, es la sección en donde se buscan los operadores genéticos, de ella se obtiene directamente el sensor o el controlador clonado, es un proceso iterativo y en el cual se pueden aplicar diversas técnicas, las cuales se explicaran en los apartados de este documento

Finalmente, el resultado obtenido con esta metodología, son funciones de salida (para problemas multiobjetivo) que contienen la información solicitada por el diseñador.

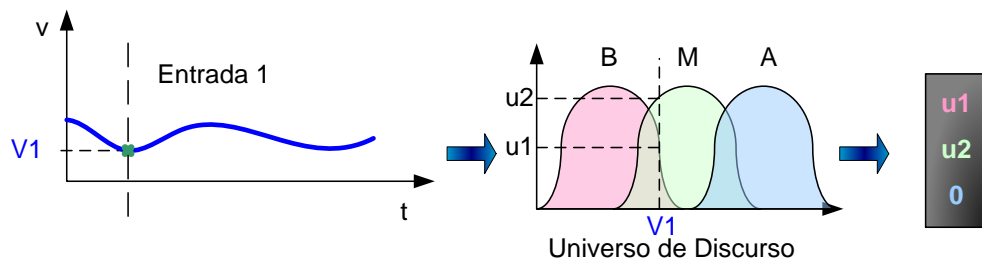
5.2 METODOLOGÍA TIPO GENÉTICO.

La primera metodología vista en esta investigación consiste en la implementación de "código genético" de cada variable, representado por los valores de pertenencia hallados previamente utilizando la técnica *"fuzzy c-mean"*, el cual permite la implantación de los valores de dicha variables en un código que consta de n genes, donde n representa la cantidad de clusters definidos para esa variable en específico.

El primer paso en la etapa de clonación consiste en crear los clusters para los valores de las entradas y salidas, esto contribuirá con la concepción de una metodología que pueda trabajar con problemas multiobjetivo.

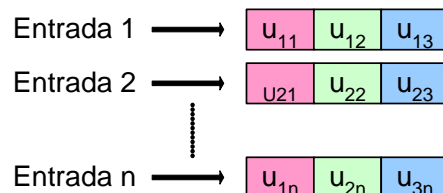
Para lograr este enfoque, es necesario, aplicar el “fuzzy c-mean” para encontrar los respectivos clusters de cada señal, estos clusters tienen una representación en conjuntos difusos, por lo que un valor $V1$ se puede representar en n valores de pertenencia, donde n es el número de clusters de la variable en mención. Este procedimiento se aprecia en la **Figura 15**.

Figura 15. Representación en clusters difusos.



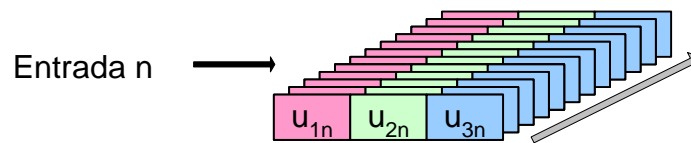
Este procedimiento se repite para las entradas del sistema y para todas las salidas, pasando de una representación por valor a una representación por grado de pertenencia en los clústeres, tal y como se aprecia en la **Figura 16**

Figura 16. Representación en código.



Para finalizar esta etapa, se convierte las señales en conjuntos de representaciones en clusters, lo que nos permitirá finalmente convertir todo el contenido de la señal en clusters, tal y como se aprecia en la **Figura 17**

Figura 17. Representación del código en el tiempo.



5.2.1 Creación de los cromosomas

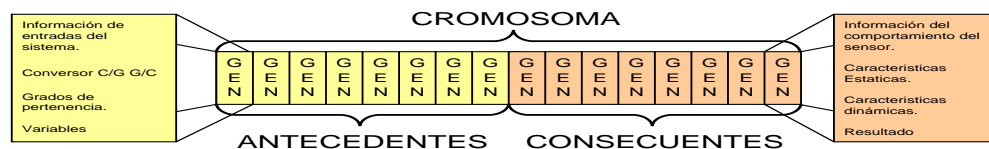
Para esta creación de cromosoma, fue implementada la visión dada por la referencia [Delgado98], en la cual se utiliza una división del cromosoma en antecedentes y consecuentes, los antecedentes corresponden a las entradas del sistema, es decir, todas las diferentes variables que influyen en la inferencia de la o de las variables de salida, en esta sección también se pueden encontrar la información codificada de los clusters, grados de pertenencia, tipos de conjuntos difusos, entre otros.

Los consecuentes del cromosoma contienen información que ha sido obtenida de los antecedentes, estas pueden ser, características estáticas y dinámicas, valor de salida propuesto, error estático y dinámico, por mencionar algunos.

La selección de los antecedentes y los consecuentes es libre y constituye una de las tareas más importantes para el desarrollo del proceso de clonación, esto es debido a que es en este punto del proceso, en donde se selecciona la información

relevante para el proceso de clonación, la cantidad y variedad de los denominados “genes” del cromosoma será por lo tanto una de las decisiones particulares en cada proceso, esta representación de antecedentes y consecuentes (AC) se aprecia en la **Figura 18**.

Figura 18. Codificación del cromosoma.



5.2.2 Codificación del cromosoma.

La codificación del cromosoma es uno de los pasos de mayor relevancia, esto es debido a la gran variedad de representaciones para los valores y estados de un sistema.

5.2.2.1 Codificación binaria: Es la más utilizada para realizar operaciones, esto debida gran facilidad a la hora de realizar operaciones genéticas (cruce, mutación), exige generalmente que el tamaño de los cromosomas crezcan cuando buscan la representación de varios valores con un alto grado de precisión, en el siguiente ejemplo se representa el número 36, el cual necesita 6 dígitos binarios para ser representado.

5.2.2.2 Codificación no binaria: Permite una lectura mas sencilla de los valores de cada “gen”, por lo que el tamaño del cromosoma se reduce considerablemente, exige un poco mas de atención a la hora de aplicar operadores genéticos en vista de no generar individuos incorrectos, en esta codificación se pueden encontrar valores enteros, caracteres (ideales para la representación fuzzy).

5.2.2.3 Codificación mixta: Incluye las dos codificaciones anteriormente mencionadas, y busca la flexibilidad de la codificación binaria en la implementación de los operadores genéticos y las bondades de la codificación no binaria en la disminución del tamaño del cromosoma, una representación de todas las posibles codificaciones se aprecian en la **Figura 18**.

Figura 19. Tipos de codificación.

Binaria	1	0	0	1	0	0
No binaria	A	B	C			
Mixta	A	B	10	1	0	

5.2.3 Búsqueda de consecuentes.

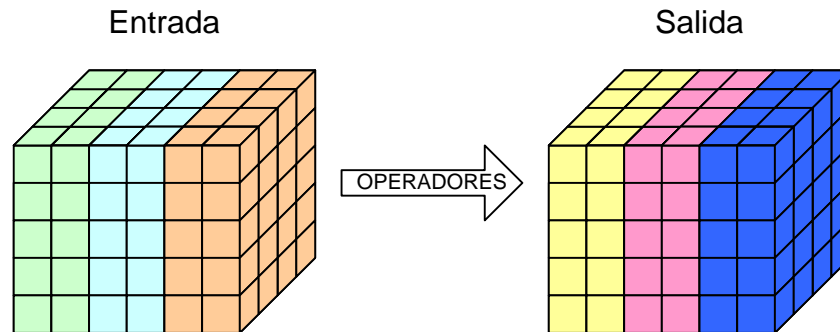
El procedimiento para encontrar los consecuentes es propio en cada sistema, pero es necesario saber, que esta información, como su nombre lo indica, es una consecuencial de la implementación de los antecedentes en el sistema, por lo que se debe tener especial cuidado en la elaboración de esta característica.

Un ejemplo de selección de consecuentes es la característica dinámica de las entradas, este se logra analizando previamente el comportamiento de las entradas y asignando un valor estimado a este comportamiento.

5.2.4 Identificación de operadores genéticos correctos.

Los operadores genéticos, serán los encargados de hacer interactuar los cromosomas previamente armados, estas operaciones convertirán los cromosomas de entrada en cromosomas de salida, estos cromosomas de salida deben corresponder a una estructura de cromosoma AC, para lograrlo, la metodología de clonación dispone de una etapa de aprendizaje utilizando un método de comparación con el patrón, donde, evidentemente el patrón a utilizar como referencia es la señal de salida deseada, el procedimiento deseado se aprecia en la siguiente .

Figura 20. Procedimiento de clonación.



Como es evidente en la **Figura** anterior, se tiene un conjunto de características denominadas entradas, las cuales, tras la aplicación de unos operadores genéticos, se debe convertir en el conjunto de características de la salida a emular, este es el procedimiento que se utilizara para el entrenamiento del sistema.

Para lograr este objetivo, los operadores genéticos modifican la información, esto nos genera una pregunta, ¿Cuál debería ser la secuencia correcta?, ¿Qué operadores genéticos se deben utilizar?, ¿Cuántos operadores son necesarios?

Para responder las preguntas anteriormente propuestas, es necesario implementar un algoritmo genético (AG), cuyo objetivo sea el encontrar una población óptima de secuencia de operaciones, las cuales serán evaluadas tomando en cuenta el valor deseado de salida y el número de operaciones genéticas (entre menos operaciones, mejor calificación tendrá el individuo)

5.2.5 Algoritmo genético.

El Algoritmo Genético, como se menciona anteriormente, es el encargado de realizar la búsqueda de la correcta secuencia de operadores genéticos, que lleven las entradas a la salida deseada, el procedimiento se ilustra en la 0. Las primeras etapas se enlistan a continuación.

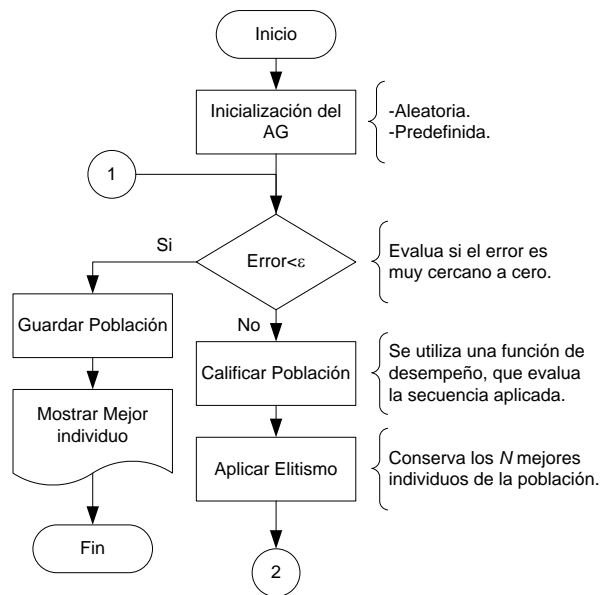
5.2.6 Inicialización del Algoritmo Genético.

En esta etapa se crean los primeros individuos de la primera población, es recomendable generar individuos aleatoriamente, cabe aclarar que en algunos casos, el desarrollador puede tener indicios de la secuencia correcta de los operadores genéticos a aplicar, por lo que es conveniente, iniciar alguna porción de la población con valores dados por el usuario y dejar los faltantes individuos a un proceso de creación aleatoria.

5.2.7 Evaluar característica de rendimiento.

Se evalúa si la característica seleccionada ha disminuido o aumentado, dependiendo de cada caso, es por ejemplo en sensores clonados, es común utilizar la medida del error entre la medición real y la medición del sensor clonado como calificación para el individuo (secuencia de operadores genéticos).

Figura 21. Algoritmo del AG



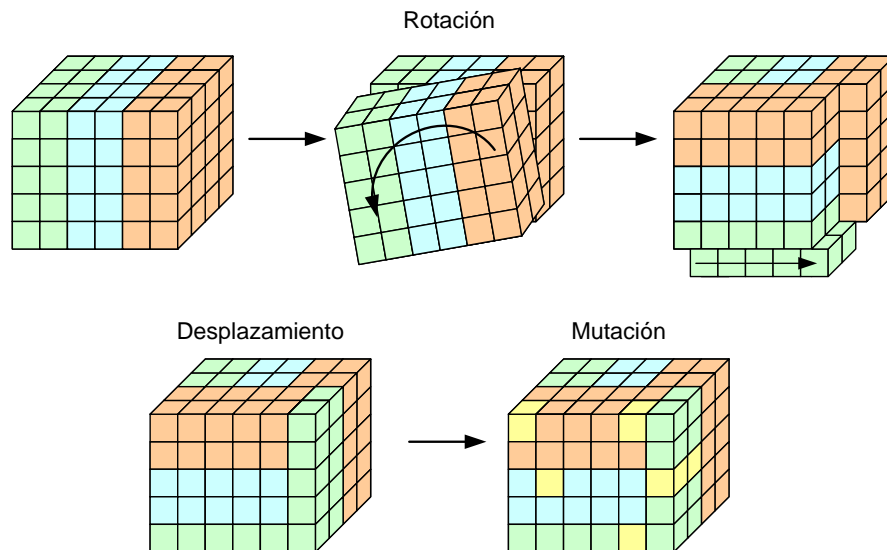
5.2.8 Calificación de la población.

Es la característica más importante de un AG, de esta depende la dirección de convergencia del AG, es evidente que si se desea buscar una cierta característica en los individuos, esta es la que debe ser premiada con una alta calificación (o baja, dependiendo si se busca maximizarla o minimizarla).

En la implementación de la metodología de clonación es importante utilizar la medida del error entre la señal deseada y la señal obtenida por el dispositivo clonado, al mismo tiempo, es importante añadir una segunda componente, la cual estará encargada de “premiar” a las soluciones con menores operadores genéticos y “castigar” a las que contienen mayor número de operadores; esta técnica, le permitirá al algoritmo disminuir el número de operadores, creando una secuencia más óptima y eficiente, pensando con antelación la implementación en hardware del dispositivo clonado.

Para conocer la “calificación” de cada individuo, es necesario aplicar la secuencia de operadores genéticos y comparar el resultado de estos con el resultado deseado, este procedimiento se aprecia en la **Figura 21**

Figura 22. Proceso de Clonación



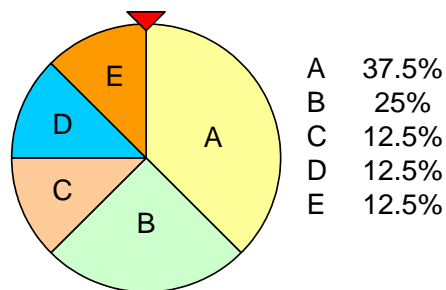
5.2.9 Elitismo

El elitismo es un operador de gran importancia pues permite conservar los N mejores individuos de cada población, asegurando el mejoramiento continuo del algoritmo, en el ejemplo específico de la clonación es necesario implementarlo para apreciar la disminución constante del error, al mismo tiempo esta característica le permite a la metodología disponer en cualquier momento de la ejecución del algoritmo, una respuesta o un acercamiento a la misma.

5.2.10 Selección de Padres

Para la metodología de clonación es necesario implementar una herramienta de fácil uso y de excelentes resultados, para esta etapa del proceso es utilizado la metodología de la “ruleta”, la consiste en ubicar proporcionalmente a su calificación a los individuos en una ruleta, por lo que, el individuo mejor calificado, tendrá las mejores opciones para ser elegido como futuro padre, el individuo B, tendrá también una buena cantidad de opciones y los 3 individuos restantes, tendrán menores posibilidades de cruzarse, este ejemplo se aprecia en la **Figura 23**.

Figura 23. Selección de padres.



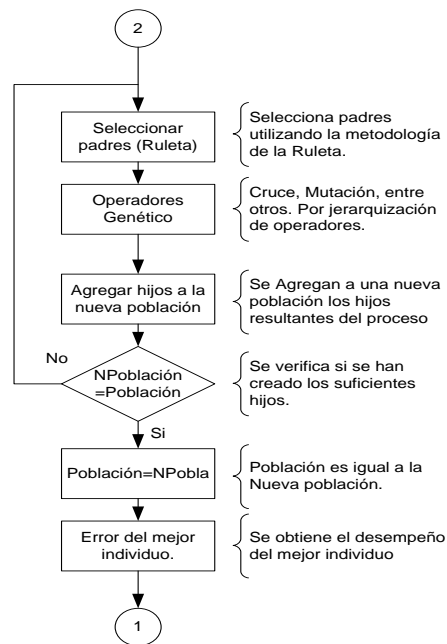
5.2.11 Operadores genéticos

Los operadores genéticos, son los encargados de hacer la combinación de los padres elegidos, entre ellos se utilizan los tradicionales cruce y mutación, cabe aclarar que este punto es propio de AG y no del modelo de clonación, estos operadores son los encargados de *evolucionar* las secuencias requeridas por la metodología de clonación, cabe recalcar nuevamente, que pueden existir los mismos operadores genéticos, tanto para el AG como para las secuencias de clonación, pero esto no es indispensable.

Inclusive se puede encontrar algún proceso de clonación que posea muchos operadores genéticos y que el AG que busca su optimización utilice solamente cruce y mutación.

Posterior de este paso, es lógico agregar estos nuevos individuos a la nueva población que será la encargada de reemplaza la población de estrategias actuales; El procedimiento antes mencionado (Selección de padres, Operadores Genéticos y Agregar hijos a la nueva población) se repite hasta obtener una cantidad nueva de secuencias, luego de alcanzado este valor, se procede a modificar al población actual por la población nueva, de esta nueva población se elige al mejor individuo y su respectivo rendimiento, este valor se guarda para volver a la etapa 1 de la 0. El procedimiento descrito en esta sección se aprecia en la **Figura 24**.

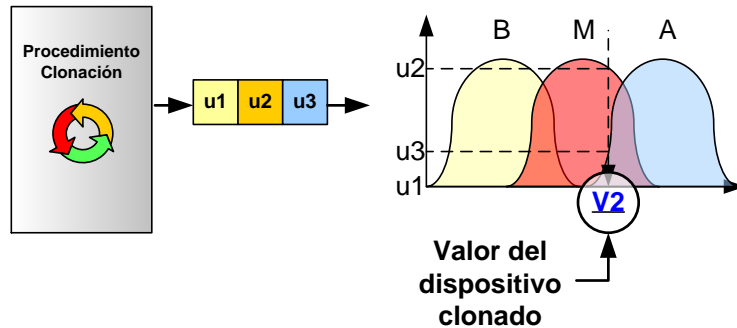
Figura 24. Continuación del Algoritmo del AG



5.2.12 Identificación de clusters de salida.

Posterior a la búsqueda utilizando AGs, es necesario implementar el mejor individuo (la mejor secuencia) obtenido por el AG como sensor clonado, al utilizar esta secuencia, se toman los datos de entrada, se aplican los operadores genéticos y se obtiene un conjunto de clusters de salida, a estos clusters le aplicamos un procedimiento de defusificación para tener el valor o los valores deseados de salida, cabe resaltar que el proceso de clonación también permite la obtención de otras características propias del sistema clonado y no únicamente un valor específico, puede encontrarse un conjunto de estos. Una breve ilustración se aprecia en la **Figura 25**.

Figura 25. Inferencia de clonación.



5.2.13 Dispositivo clonado.

El dispositivo clonado, ya no precisa de una etapa de aprendizaje, esta ya ha sido realizada en el procedimiento del AG (en la etapa de búsqueda de la correcta secuencia), por lo que se implementa directamente, reemplazando al dispositivo "padre", el cual se puede utilizar como elemento de referencia en una etapa primaria de implantación y como respaldo para el sistema en una etapa final de implantación.

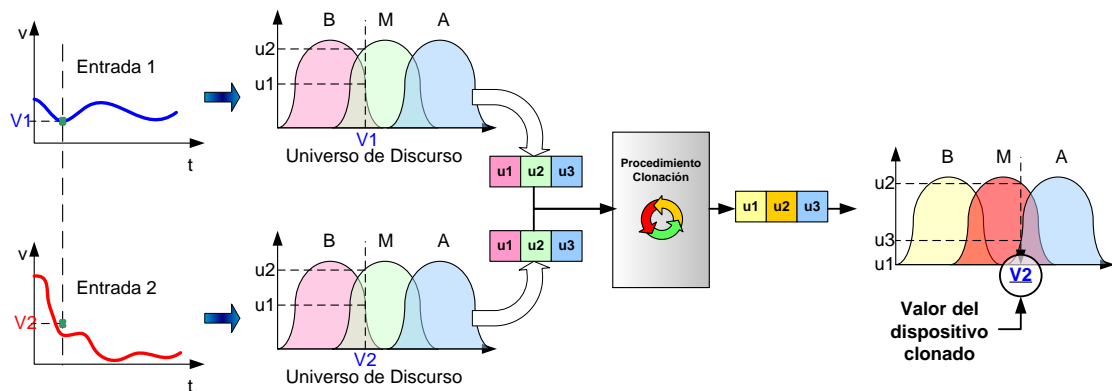
La 0, muestra un resumen de la operación del sistema clonado, en esta ilustración se aprecia, como las entradas son convertidas, gracias al "*Fuzzy c-mean*" en clusters difusos y estos a su vez son reflejados en conjuntos difusos.

Posteriormente, se convierte la información de los conjuntos, de las características de las señales en los respectivos cromosomas, estos cromosomas, como se aprecia en "Creación de los cromosomas" contienen antecedentes y consecuentes.

Luego de poseer esta información en forma de “cromosoma”, se aplica la secuencia de operadores genéticos, la cual convierte la información de entrada en “cromosomas” de salida con la misma estructura de datos (antecedentes y consecuentes).

El conjunto de datos entregado por el proceso de clonación, es procesado por el sistema, una de las etapas importantes de este proceso es la defusificación de una parte del “cromosoma”, de tal forma que podamos tener el valor de la variable clonada en rangos del universo de discurso y no en términos de pertenencia a conjuntos borrosos.

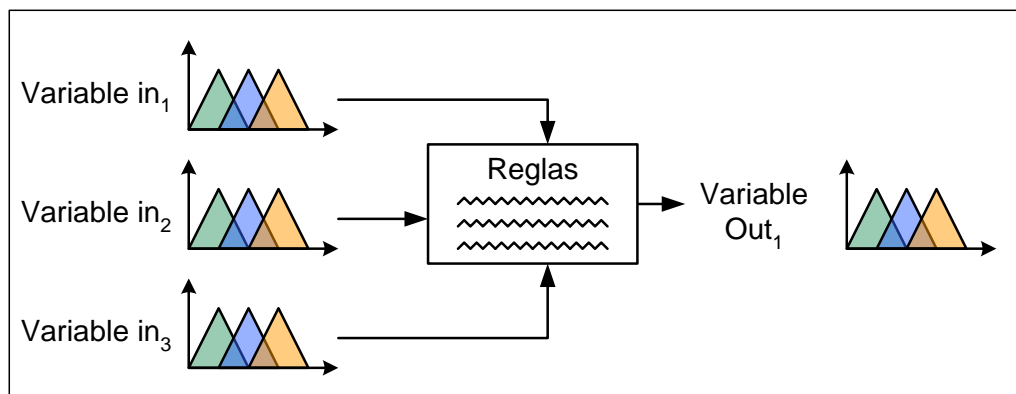
Figura 26. Procedimiento de clonación 2.



5.3 METODOLOGÍA FUZZY-GENÉTICO.

La implementación de esta metodología busca, la integración de las dos técnicas de inteligencia artificial, los ya analizados y discutidos algoritmos genéticos y la lógica fuzzy, esta metodología utiliza la lógica fuzzy como eje central de la metodología, el algoritmo genético se encarga de encontrar la cantidad de reglas y las posibles combinaciones entre ellas. Este planteamiento se puede apreciar en la **Figura 27**.

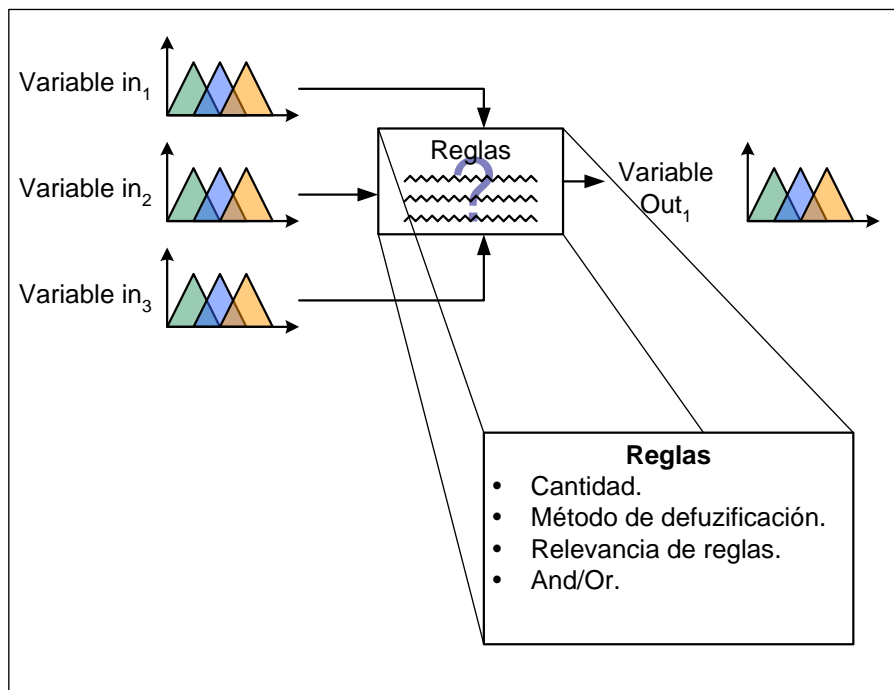
Figura 27. Reglas difusas.



Como se observa, los elementos más importantes en la lógica fuzzy es el conjunto de reglas y la definición de los conjuntos *fuzzy*, para esta última la metodología de clonación utiliza la herramienta de "fuzzy c-mean" que permite encontrar los conjuntos *fuzzy* que mas se acercan al comportamiento de la variable estudiada, y que no representa una selección caprichosa del diseñador; para el segundo componente, las reglas del sistema de inferencia *fuzzy*, no existe como tal una guía o metodología exacta para el planteamiento de las mismas, si bien existen aproximaciones en el tema de extracción de reglas difusas es necesario un proceso matemático riguroso y además se debe contar con una data lo

suficientemente amplia para implementar dicha metodología, por lo cual es importante evaluar el amplio universo de discurso para encontrar un sistema de inferencia *fuzzy* que cumpla con las características del controlador o sensor clonado. Este tipo de inferencias se puede apreciar en la **Figura 28**.

Figura 28. Contenido de reglas difusas.



5.3.1 Codificación del problema

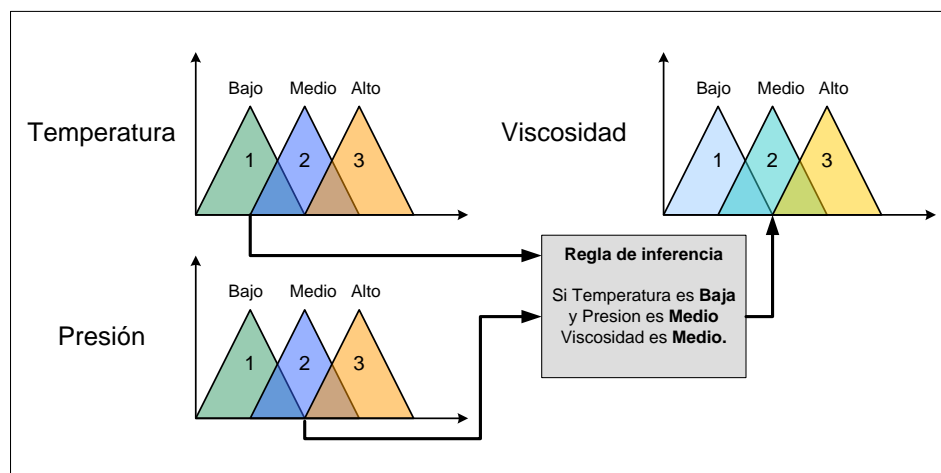
El primer paso para la realización de este tipo de metodología es la codificación del problema, en otras palabras es encontrar la forma de representar las variables significativas en un "código genético" para encontrar una solución óptima.

Como se introdujo anteriormente, las reglas difusas son el objetivo de nuestro algoritmo genético (hay que recordar que los conjuntos difusos se pueden inferir utilizando la técnica de "fuzzy c-mean"), por lo que debemos enlistar las características de estas reglas que queremos optimizar.

- Cuantas y cuales reglas.
- Tipo de defuzificación.
- Tipo de conexión.
- Relevancia o importancia de cada regla.

El primer paso consiste en adecuar el algoritmo para que pueda definir una sola regla, esto se logra definiendo en su código, los conjuntos de entrada (denominados antecedentes) y los conjuntos de salida (denominados consecuentes), los elementos que contienen una regla tipo "if" se aprecian en la **Figura 29**, en donde podemos observar claramente la información que contiene dicha regla, los antecedentes (Temperatura baja, Presión Medio), la forma de conexión ("y" que denota la presencia de las dos características anteriormente mencionadas) y un consecuente (Viscosidad Medio).

Figura 29. Ejemplo de regla difusa.



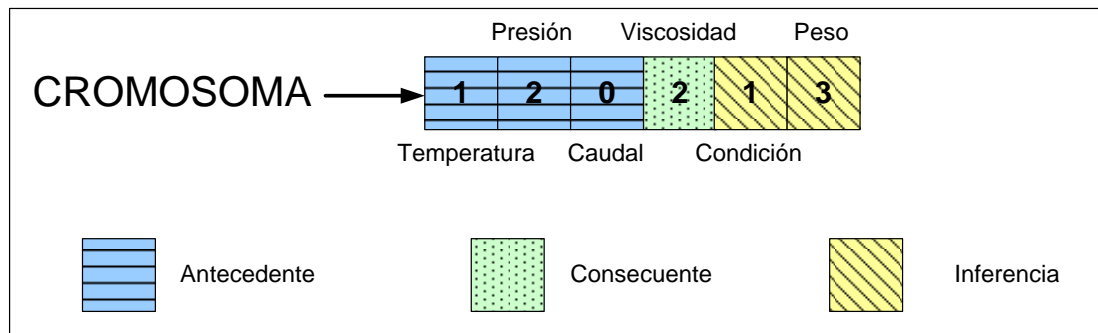
La primera característica que se debe modelar en el algoritmo genético es indudablemente la información indivisible de una regla difusa, sus antecedentes y sus consecuentes, este tipo de codificación se logra representando los conjuntos difusos no por sus etiquetas de lenguaje, sino por un número asignado, en orden ascendente en el universo de discurso a cada conjunto, este tipo de numeración se aprecia en el 0, este tipo de numeración se encuentra en la codificación que se observa en la 0, la cual representa la presencia de 3 variables de entrada, las cuales son temperatura (Baja), presión (medio) y caudal (0); en el mismo cromosoma, se encuentra la información que representa el conjunto difuso de salida (Medio) de la variable de salida seleccionada, que en este ejemplo es viscosidad.

Estando definidos los antecedentes y consecuentes de una regla difusa, es necesario concretar el método de conexión, en este ejemplo se usaran únicamente 2 tipos de conexión, and y or, estos se denotaran con los números 1 y 2 respectivamente como una parte importante del cromosoma, este gen, pertenece a la sección de inferencia de cada cromosoma.

Luego de definir el tipo de conexión que es implementado, es necesario permitir la definición de reglas que se consideran de mayor importancia, precisar al igual que se hace en las redes neuronales un "peso" para cada regla, con el fin de darle mayor relevancia a ciertos tipos de comportamientos del sistema difuso, esta información es agregada, al igual que el tipo de conexión a la sección de inferencia del cromosoma, siendo este un valor que varía en magnitud (entre mayor sea, mayor es el "peso" de esta regla), para este ejemplo se utilizará un valor real entre 1 y 10.

Finalmente, el cromosoma por cada regla contiene $n+m+2$ genes, donde n representa las variables de entrada y m las variables de salida, un ejemplo de esta codificación utilizando la regla de la 0 se puede apreciar en la **Figura 30**.

Figura 30. Cromosoma de metodología.

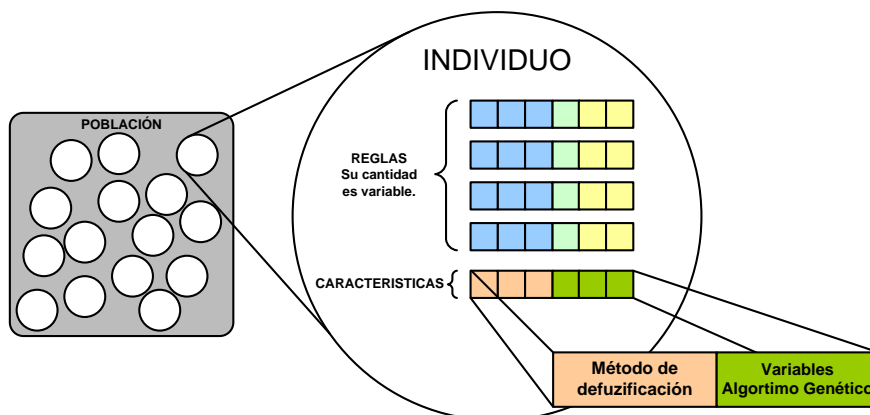


Al ser este una porción del problema (una regla difusa no define totalmente un sistema difuso) es necesario añadir una cantidad mayor de reglas, la gran incógnita es definir cuantas reglas son las necesarias para el correcto funcionamiento del sistema, por eso, el algoritmo genético incluye una estructura flexible, generando reglas aleatoriamente desde 0 hasta un tope máximo definido como R en cada individuo, este tope se debe elegir teniendo en cuenta las posibles combinaciones Pc que se obtienen para un sistema difuso, esto se puede apreciar en la \square ; en donde V_n simboliza la cantidad de conjuntos difusos de una variable lingüística, y S cumple el papel similar pero para la variable de salida. Es debido a esta amplia cantidad de posibles reglas que se le debe permitir al algoritmo, ir buscando el número óptimo de reglas para el problema, premiando a los individuos que con un mismo desempeño poseen una menor cantidad de reglas.

- $Pc = S \cdot (V_1) \cdot (V_2) \cdot \dots \cdot (V_n)$

Finalmente, para completar la información genética de cada individuo es necesario añadir otro tipo de cromosoma, en donde se pueda incluir información propia del método de defusificación, este gen se añade como una variable lingüística al cromosoma, su metodología de mutación y cruce se analizará posteriormente, por último se desarrolla un espacio para añadir variables propias del algoritmo genético, estas últimas variables son útiles para que el mismo algoritmo genético busque sus óptimos en porcentajes de reproducción y porcentajes de mutación entre otros, un ejemplo de este tipo de codificación se encuentra en la **Figura 31**.

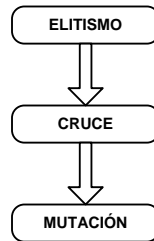
Figura 31. Población e individuos del AG



5.3.2 Operadores genéticos.

Al implementar una codificación estructurada, se presentan interrogantes en la implantación de los operadores genéticos para esta metodología. Como es conocido, los operadores genéticos permiten el intercambio de información relevante y la evolución hacia un resultado óptimo; por ende, es necesario implementar operadores genéticos que permitan y garanticen el intercambio de dicha información los operadores genéticos aplicados se aprecian en la **Figura 32**.

Figura 32. Operadores genéticos utilizados.

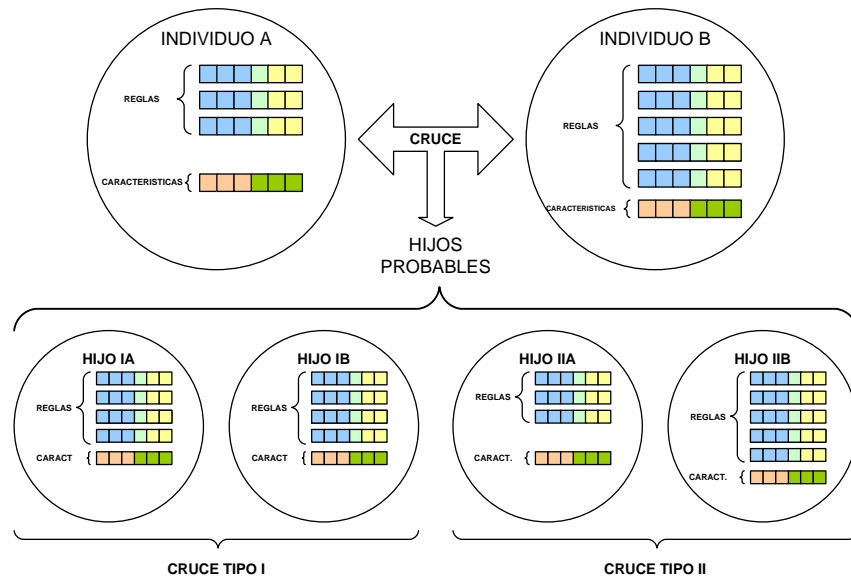


Como se aprecia, el primer operador genético utilizado es el operador del elitismo, este operador permite que cada generación sea igual o mejor que la anterior (en términos de función de evaluación), esto se logra manteniendo un porcentaje de los mejores individuos de la población actual, para este caso en específico se utiliza un porcentaje de elitismo del 10%.

El segundo operador utilizado es el operador de cruce, este está fijado en un 70%, lo cual permite que en un 30% de las interacciones los individuos "pasen" sin intercambiar información y que en un 70% de las veces, los individuos intercambien su información genética. Para lograr el intercambio de información genética entre dos individuos, se tienen dos tipos de cruce, el I y el II, el primero de ellos busca que el número de reglas sea igual al promedio de reglas de la suma de cada individuo, y el segundo tipo de cruce, mantiene el número de reglas iguales a las de sus padres, esto se aprecia mejor en la **Figura 33** y en la 0

$$RCruceI = \frac{RA + RB}{2}$$
$$RCruceII = RA \text{ ó } RB$$

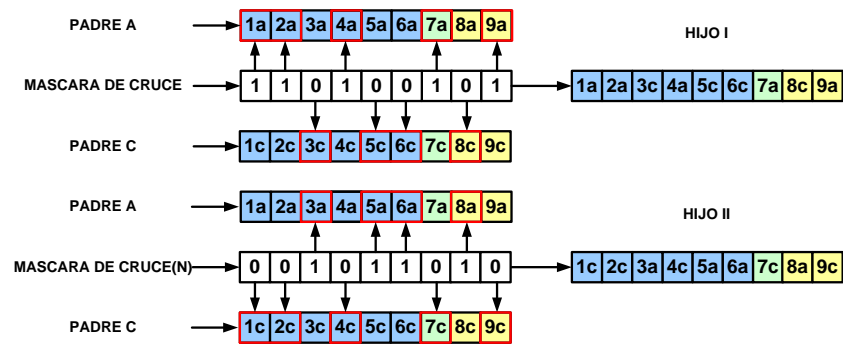
Figura 33. Tipos de cruce



Solucionado el problema de la selección del número de reglas para cada individuo tipo hijo, es hora de profundizar en el tema del intercambio de información genética.

5.3.2.1 Cruce Fuzzy: El cruce, en este algoritmo genético en particular, utiliza una "mascara" de intercambio de genes, la cual es generada aleatoriamente, para permitir una mezcla de características genéticas, los antecedentes, los consecuentes y el método de inferencia de cada cromosoma, el número de cromosomas, los cuales forman los genotipos, el tamaño de los genotipos es elegido, utilizando el promedio de los padres o manteniendo el número de los mismos.

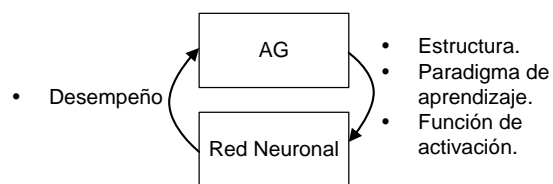
Figura 34. Cruce Fuzzy



5.4 METODOLOGÍA NEURONAL.

El fin de introducir una metodología neuronal busca que el dispositivo pueda aprender de ejemplos presentados con anterioridad y que responda de una manera fiel a los datos del pasado, con el fin de que responda de una manera óptima a futuros datos. En la **Figura 35** se aprecia la interacción que tiene el algoritmo genético y la red neuronal.

Figura 35. Algoritmo Neuro-Genético.

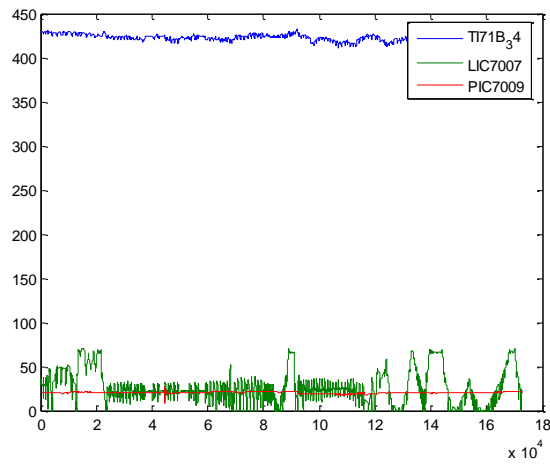


El proceso de esta metodología, no dista mucho de los dos anteriormente mencionados, se busca que el algoritmo genético busque (hay que recordar que los algoritmos genéticos son técnicas de búsqueda) 3 tipos fundamentales de variables, las cuales son, la estructura de la red, el paradigma de aprendizaje y la función de activación.

6. IMPLEMENTACIÓN DE LA METODOLOGIA.

6.1 SIMULACIÓN DEL SENSOR

Figura 36. SEÑALES DE ENTRADA

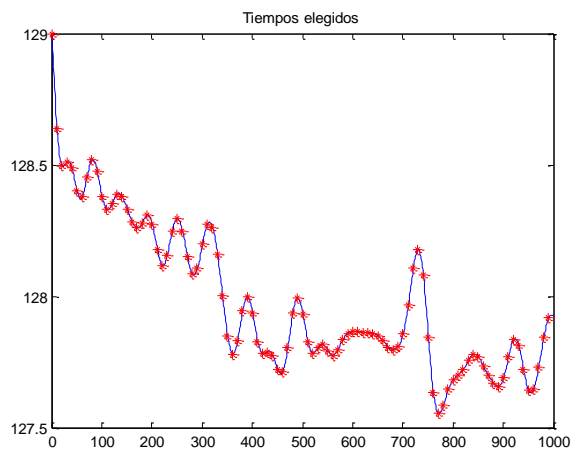


TEMPERATURA

NIVEL

PRESION

Figura 37. SEÑAL DE SALIDA



VISCOCIDAD

Figura 38. SEÑAL DE SALIDA Vs CLONADA

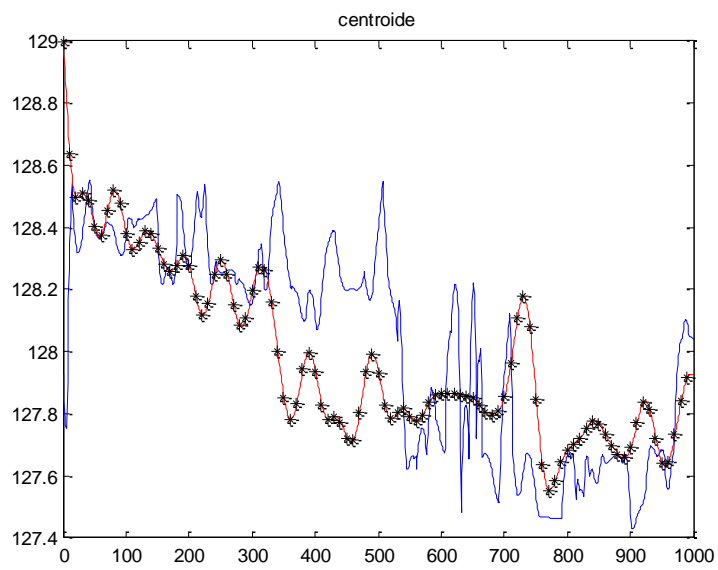


Figura 39. MARGEN DE ERROR

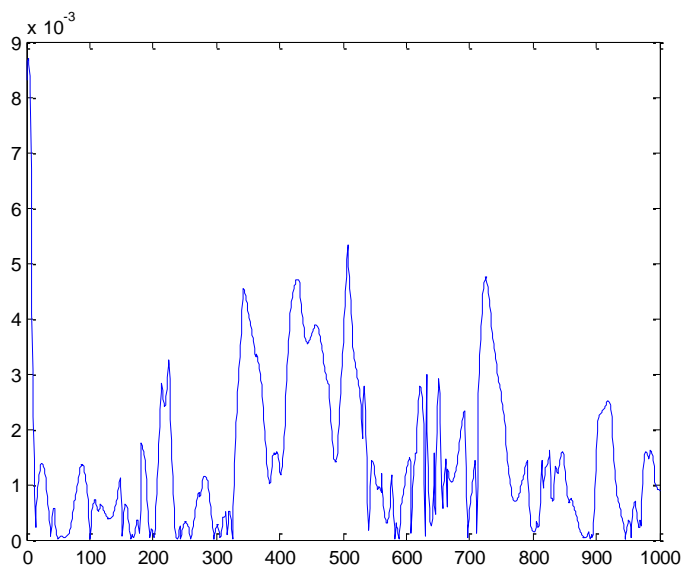


Figura 40. EVOLUCION DE LA CALIFICACION

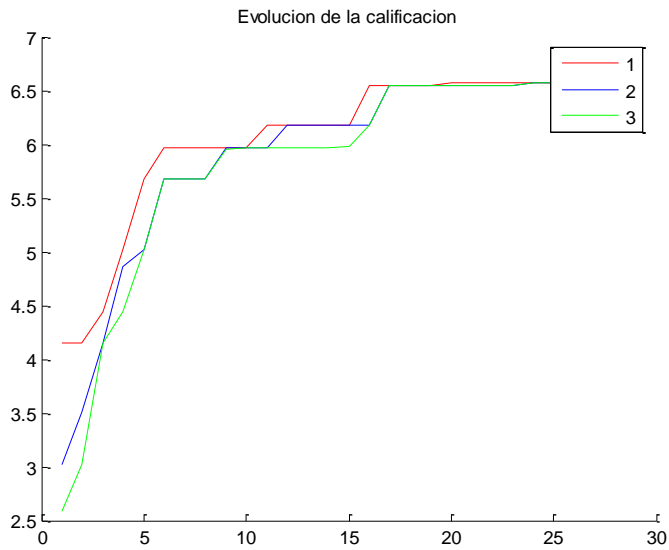
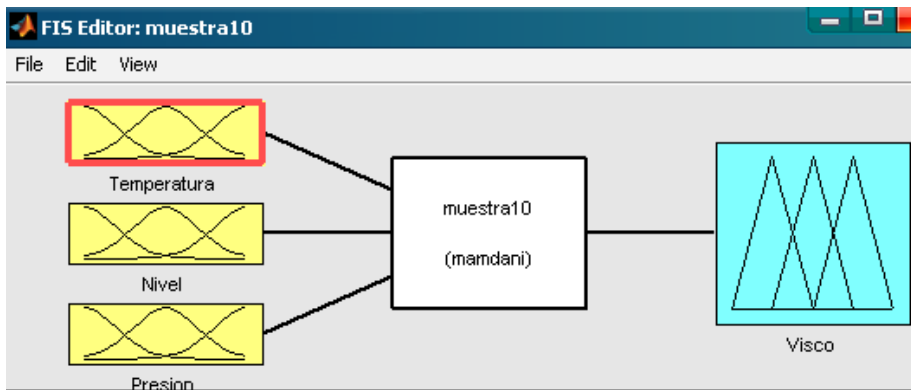


Figura 41. MODELO CONTROLADOR FUZZY

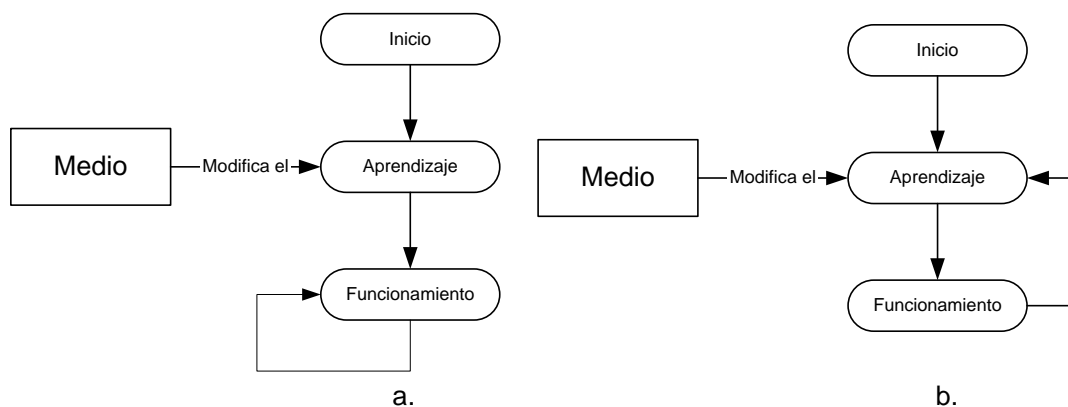


6.2 HARDWARE EVOLUTIVO.

El hardware evolutivo, es una herramienta necesaria para la implementación de la clonación artificial en ingeniería, las razones que fundamentan esta afirmación son varias, una de las más importantes, radica en la necesidad de aprendizaje del sistema, es evidente que el equipo desarrollado, sea sensor o controlador, va a funcionar por una cantidad de tiempo indeterminado, que en la mayoría de los casos se espera que sea un tiempo prolongado. Debido a esta situación es necesario prever que las condiciones, en las que fue "educado" el dispositivo, cambian o evolucionan, adicionando nuevas variables al proceso, lo que requeriría una adaptación del clon a su nuevo ambiente.

La adaptación que es requerida, no se puede lograr utilizando la metodología que se aprecia en el 0(a) en donde se observa que el aprendizaje sólo ocurre en un primer momento y que el proceso de ejecución o funcionamiento no es modificado en ninguna etapa. La siguiente concepción es permitirle al dispositivo la "reeducación" por medio de un aprendizaje que no necesariamente sea constante, pero si periódicamente, lo que facilitará la adaptación a nuevos cambios en el medio en el cual el clon trabaja, esta metodología se observa en la **Figura 42**

Figura 42. Tipos de funcionamiento



Para la implementación de un dispositivo o clon que aprenda periódicamente es posible que se haga de dos formas, "off-line" o "on-line", la primera de ellas consiste en detener el funcionamiento del clon, llevarlo a un laboratorio o unidad de aprendizaje e introducirle los nuevos parámetros vía software o hardware, el gran problema de esta concepción es que ciertamente se inducirán tiempos muertos en el funcionamiento del clon, es decir, el dispositivo estará fuera de funcionamiento cada vez que sea necesario (o el mismo dispositivo lo pida) un reaprendizaje, la totalidad de este tiempo será dada por la rapidez con la cual los encargados de realizar esta labor la cumplan, incluyendo factores humanos al proceso de aprendizaje, específicamente a los tiempos de los mismos.

En el aprendizaje "On-line", pasa todo lo contrario, el dispositivo, activa su función de aprendizaje cada cierto periodo de tiempo y lo ejecuta paralelamente a su funcionamiento, evitando el tener que detener el proceso en el cual el clon forma parte, posterior a un tiempo de aprendizaje, el clon puede modificar su estructura (Hardware evolutivo) para ya sea permitir la entrada de una nueva configuración que el mismo pueda suplir, o modificar totalmente su estructura.

En este caso en particular se desea implementar el uso del aprendizaje "On-line", para lo cual se ha estudiado muy de cerca el uso de células madres electrónicas, que al igual que sus homologas en la biología, estas células pueden convertirse en cualquier otro tipo de células dentro del cuerpo y a replicarse en una cantidad aun indeterminada de veces, lo que ha conllevado a los investigadores a interesarse en este de comportamiento y en ahondar en su estudio y evidentemente iniciar todo tipo de debates en el tema; afortunadamente las células madres que en esta investigación se utilizan distan sustancialmente de la polémica ética y moral, pero aportan una valiosa información para el desarrollo de sistemas de alta tecnología, cerrando una nueva brecha entre la ciencia biológica y la ciencia tecnológica.

La célula madre que en este documento se desarrolla, es una unidad de procesamiento lógico digital, la cual, debido a su estructura puede modificar su comportamiento, gracias a la inclusión de una entrada denominada entrada de mutación, esta célula madre a diferencia de su homóloga en la naturaleza, no es capaz de replicarse a si misma, esta habilidad es reemplazada por la habilidad que poseerá el software para exigir la generación de nuevas células madres.

Para la implementación de este paradigma, es necesario contar con elementos que permitan una rápida y flexible configuración en hardware, para lograrlo utilizamos cualquier tipo de dispositivo lógico programable, en este caso en específico se utiliza un FPGA (Field Programmable Gate Array).

6.3 PROCESO DE CLONACIÓN DEL SENSOR

Dentro de la línea de estudio de circuitos lógicos digitales es importante conocer los operadores que intervienen en ellos, lo cual permitirá la homologación de funciones de una célula madre a un circuito electrónico.

El diseño de circuitos digitales, entre los paradigmas ya propuesto se conocen los diseños de compuerta AND y OR y sus correspondientes inversores, NAND y NOR, con estos operadores básicos se puede diseñar cualquier clase de los circuitos lógicos existentes (OR, AND, XOR, NOT), por lo que estas 2 compuertas se pueden llamar las compuertas base de toda la lógica digital.

Centrando la atención en las compuertas NAND y NOR, la característica más importantes de estos operadores es que uno o cualquiera de los dos es el resultado de negar o invertir las entradas de señal del otro es por esto que el diseño del circuito evolutivo se enfocará en la implementación de estas dos compuertas.

La idea de emular el comportamiento de los sistemas biológicos a resultado en muchos campos de la tecnología, para este diseño se tomará como base las células madres.

Para este diseño se implementara una FPGA SPARTAN3 de XILINX que es muy comercial y de fácil acceso. El primer paso consiste en modelar la célula madre en la FPGA, debido a la sencillez del ejemplo, se trabaja en la modalidad "*schematic*" del software proporcionado por la compañía desarrolladora, esta visualización nos ayuda a observar y analizar de una mejor manera la célula madre.

Posterior a esta selección es necesario implementar una compuerta NOR y compuerta NAND dentro del mismo circuito, y en la otra las compuertas OR y NOR, en este caso en especial, se trabajarán compuertas de 2 entradas, para lograr el funcionamiento del circuito como célula madre, se debe incorporar una 3 entrada, la cual funcionará como operador lógico mutable entre la NAND y la NOR. el circuito se puede apreciar en la **Figura 42** y la OR y la NOR en la **Figura 43**.

Figura 43. Hardware Evolutivo NAND-NOR

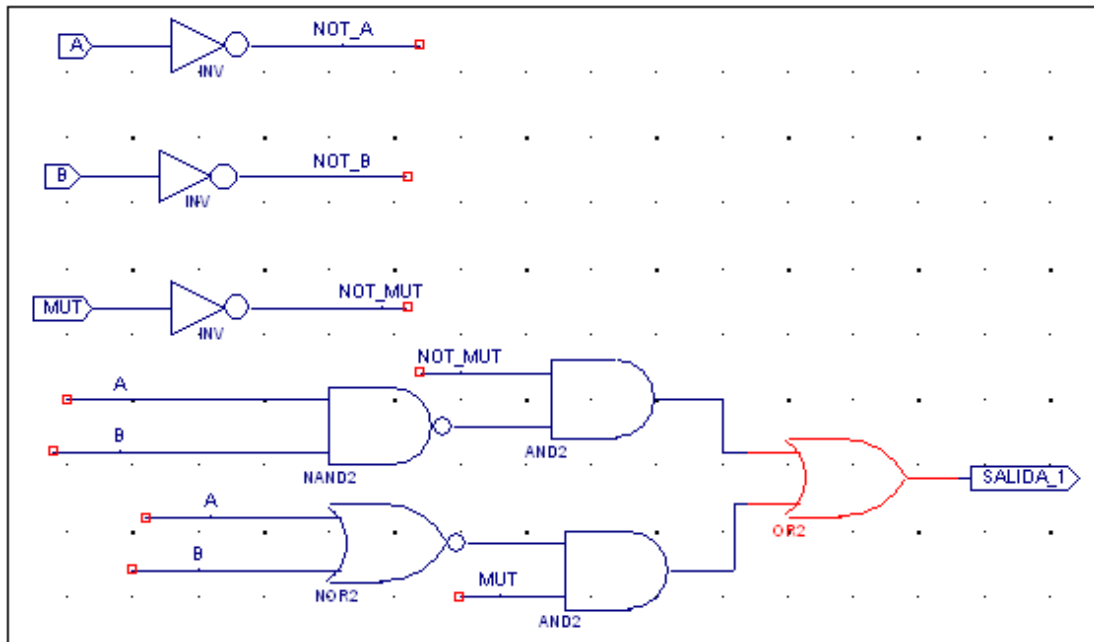
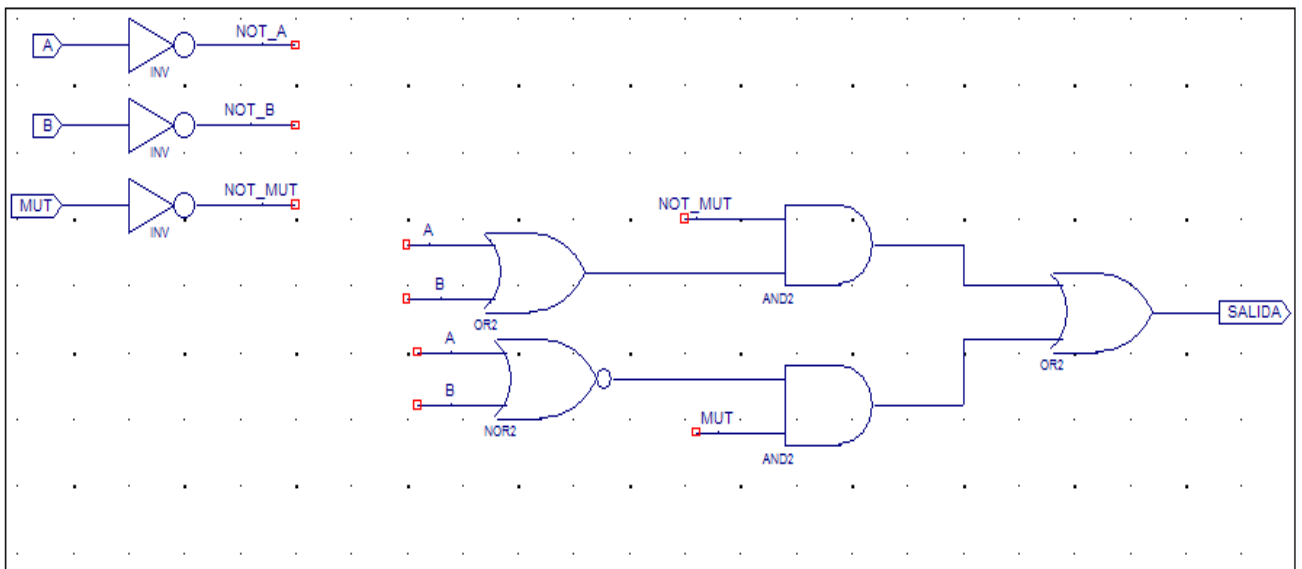


Figura 44. Hardware Evolutivo OR-NOR



Como se puede observar, la célula madre puede trabajar tanto como NAND-NOR o OR-NOR, dependiendo de su entrada de operador lógico mutable, lo que permite, al implementar una amplia cantidad de estas células, el desarrollo de una alta variedad de aplicaciones, así como igual número de arreglos lógicos. Esto se aprecia en la siguiente tabla.

Tabla 1. Operaciones Célula madre

A	B	Mut	Salida OR-NOR	Salida NAND-NOR
0	0	0	0	1
0	1	0	1	1
1	0	0	1	1
1	1	0	1	0
0	0	1	1	1
1	0	1	0	0
0	1	1	0	0
1	1	1	0	0

Figura 45. Correlación entre las entradas y las salidas.

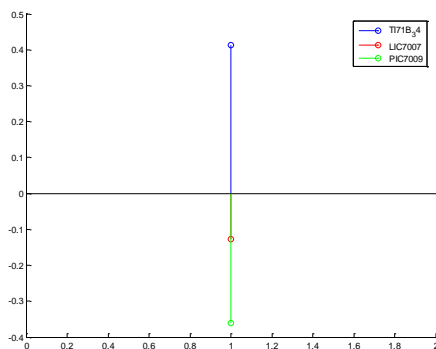


Figura 46. Clusters Para Temperatura

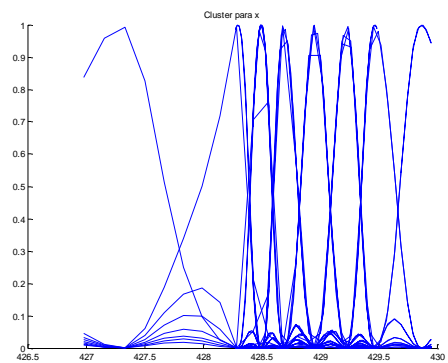


Figura 47. Clusters para Presión

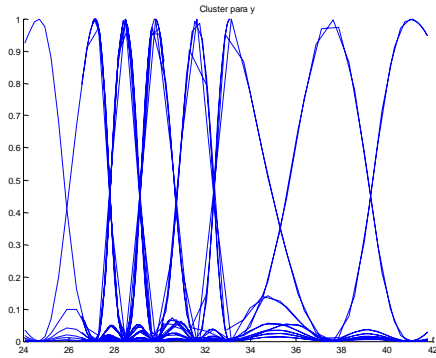


Figura 48. Cluster para Nivel

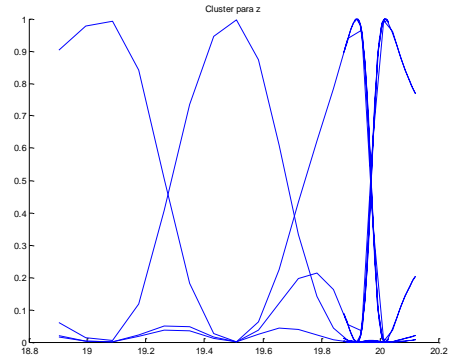


Figura 49. Entrada de temperatura sin filtrar

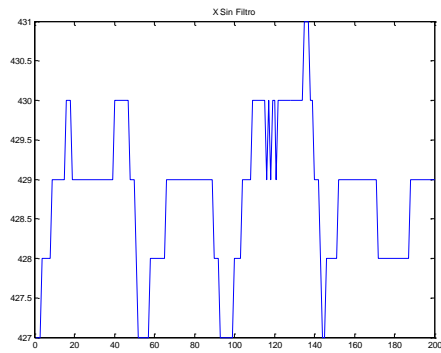


Figura 50. Entrada de temperatura Filtrada.

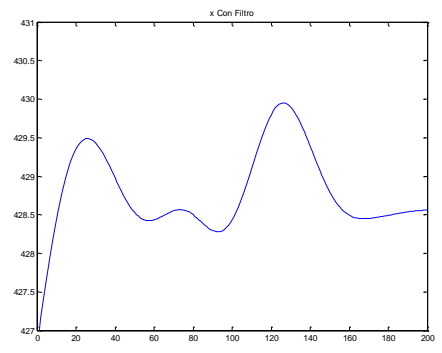


Figura 51. Entrada de Presión sin filtrar

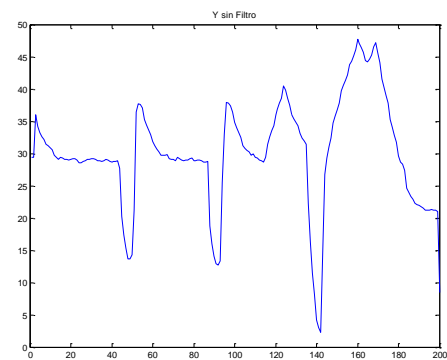


Figura 52. Entrada de Presión filtrada

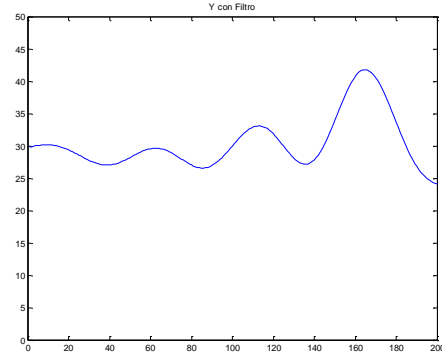


Figura 53. Entrada Nivel Sin filtrar.

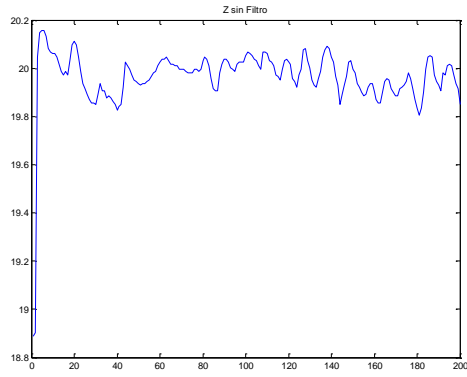
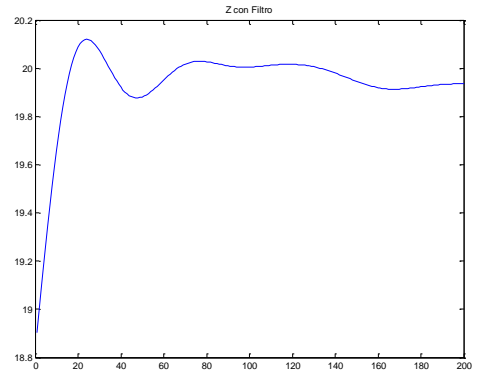


Figura 54. Entrada Nivel filtrada



7. CONCLUSIONES

- Partiendo de la necesidad de adquirir nuevos métodos y técnicas que permitan reemplazar el funcionamiento de un sensor real instalado, se logró diseñar un sensor clonado capaz de inferir por medio de comportamientos lógicos y algoritmos genéticos, las señales de entrada tomadas por el sensor de viscosidad, para este caso en específico la Temperatura, Presión y Nivel, para lograr así, mejorar la calidad de su información y obtener una respuesta de alta confiabilidad.
- El diseño de los diagramas estructurales de los componentes del proceso, se basó en la búsqueda de mejoras en la exactitud de los datos de salida, por esta razón, la unión entre la adquisición de datos, y la aplicación de algoritmos genéticos sobre la cadena de bits arrojada por el sensor en línea, mejora altamente la exactitud en la variable de salida.
- Se diseñó en MATLAB un ambiente gráfico que evalúa los valores de las variables de entrada del sensor, dando solución generando el valor de la variable de salida con una alta exactitud y precisión en su respuesta.
- Para la implementación de técnicas de clonación artificial, es indispensable el uso de herramientas programables (Ej FPGA).

- El algoritmos genético, por ser un algoritmo de búsqueda optimizado, encuentra soluciones satisfactorias para el proceso de clonación.
- La FPGA por ser una herramienta de ultima generación, es una gran ayuda por todos los dispositivos electrónicos que posee.
- El cluster difuso, permite encontrar los rangos de pertenencia de las variables lingüísticas propias de los sistemas difusos.

8. RECOMENDACIONES

Para la futura implementación del sensor virtual en la planta de Viscosidad de la Empresa Colombiana de Petróleos, ECOPEL, de Barrancabermeja, sugerimos la instalación del sistema de algoritmos genéticos por medio de dispositivos lógicos programables de campo, FPGAs.

Verificar el sistema de adquisición de datos y la red de campo utilizada.

BIBLIOGRAFIA

- Altrock C. Fuzzylogic and neuro-fuzzy technologies in appliances. Embedded Systems Conference, <http://www.esc.com>, USA, 1999.
- Arellano Cardenas Oliverio. Procesamiento y clasificación de señales Utilizando lógica difusa. Centro de investigación y estudios avanzados del I.p.n.
- Departamento de ingeniería eléctrica. Tesis Doctoral. Octubre 2000
- Richard J. Hathaway and James C. Bezdek. Fuzzy c-Means Clustering of Incomplete Data. SMCB, 31(5):735--744, 2001.
- Carr Joseph J. Sensors and circuits. Prentice Hall.1993.
- Cavalcanti Pacheco Marco Aurélio. Núcleo de pesquisa em inteligência computacional aplicada. Universidade Católica do Rio de Janeiro. 2002.
- Chiang Luciano Sánchez. Diseño conceptual de productos mecatrónicos. Departamento de Ingeniería Mecánica y Metalúrgica. P. Universidad Católica de Chile.2003.
- Chipperfield Andrew. Flemingh Peter .Pohlheim Hartmut . Fonseca Carlos. Genetic Algorithm TOOLBOX For Use with MATLAB. Department of Automatic control and systems engineering. University of Sheffield. Version 1.2. 2005
- Chiu. Stephen L. Extracting Fuzzy Rules from Data for Function Approximation and Pattern Classification. Fuzzy Information Engineering: A Guided Tour of Applications, John Wiley & Sons, 1997.
- Choque Guillermo A. Inteligencia artificial, perspective y realizaciones. Universidad mayor de San Andres, Facultad de ciencias de la computación. La Paz, Bolivia 2002.
- Cogle Christopher R, Guthrie Steven M, Sanders Ronald C , Allen William L. An overview of stem cell research and regulatory issues. Mayo Clinic Proceedings. Rochester:2003.Tomo78, Nº 8.

- Inteligencia Artificial y Minirobots. Delgado Alberto. Ecoe Ediciones 1998.
- Dresser Rebecca. Stem Cell Research: the bigger picture. Perspectives in Biology and Medicine. Chicago: Spring 2005. Tomo48
- Angel Garcia Baños. Computación evolutiva y algoritmos genéticos. Universidad del Valle. 2001
- Goldberg David E. Genetic algorithms in search, optimization and machine learning, Addison-Wesley. 1992.
- Bioinformática, simulación, vida artificial e inteligencia artificial. Lahoz-Beltrá Rafael. Diaz de Santos. 2004
- Loyo De Sardi Jaqueline Algoritmos Genéticos y Optimización en la Investigación de Operaciones. Caso: Problema de Optimización con Objetivos Múltiples Expresado como un Modelo de Programación No – Lineal por Metas. Departamento de Computación, Facultad de Ciencias y Tecnología, Universidad de Carabobo, Valencia, Venezuela. Febrero 2002.
- Maneiro Malavé Ninoska. Algoritmos genéticos aplicados al problema cuadrático de asignación de facilidades. Departamento de Investigación Operativa, Escuela de Ingeniería Industrial, Universidad de Carabobo, Valencia. Venezuela. Febrero 2002.
- Martín del Brio, Bonifacio. Redes neuronales y sistemas difusos. Alfaomega Grupo Editor. 2002.
- Maxfield, Clive. Genetic algorithms: Programs that boggle the mind. EDN. Boston. Vol. 42. Mar 3, 1997.
- Moro Sancho Quiliano Isaac . Aplicación de las redes neuronales a la predicción meteorológica. Tesis de Maestría, Departamento de Ingeniería de Sistemas y Automática. Universidad de Valladolid - España, julio 1992.
- Muñoz, A.F., The technology of advanced control of applied artificial cloning to highly precise Mechatronic Systems. WSEAS, 2004.
- Sensorica e instrumentación, Mecánica de Alta precision. Muñoz Antonio F. Pueblo y educación.

- Ortiz Jr Francisco; Simpson James R ; Pignatiello Jr Joseph J ; Hered Alejandro. A Genetic Algorithm Approach to Multiple-Response Optimization
- Journal of Quality Technology; Oct 2004; 36, 4; ABI/INFORM Global pg. 432
- Parnell Karen, Mehta Nick, Programmable Logic Design Quick Start Hand Book. 4 edicion. Xilinx. 2003.
- Passino, k.m, Fuzzy Control. Addison Wesley. USA. 1997. 475p.
- Saggiani G M; G Caligiana; F Persiani. Multiobjective wing design using genetic algorithms and fuzzy logic. Proceedings of the Institution of Mechanical Engineers; Apr 2004.
- Sahni Vishal. Pyara V.Prem. An Embryonic Approach to Reliable Digital Instrumentation Based on Evolvable Hardware. IEEE Transactions on instrumentation and measurement. Diciembre 2003 .
- Santos José, Duro Richard. Evolución Artificial y Robótica Autónoma. Alfaomega 2005.
- Simons G.L. Introducción a la inteligencia artificial. Ediciones Santos. 1987.
- Teng. T.K., J.S. Shieh, C.S. Chen. Genetic algorithms applied in online autotuning PID parameters of a liquid-level control system. Transactions of the Institute of Measurement and Control. London: Dec 2003. Vol. 25.
- Williams Sam. Unnatural selection. Technology Review. Cambridge: Feb 2005. Tomo 108, N° 2.
- Walbridge Charles T. Genetic Algorithms: What Computers Can Learn from Darwin Technology Review; Jan 1989; 92, 1; ABI/INFORM Global pg. 46
- Zilouchian Ali, Jamshidi Mo. Intelligent Control Systems Using Soft Comuting Methodologies.CRC Press. 2001. ISBN 0-8493-1875-0.

ANEXOS