

**HERRAMIENTA INTELIGENTE PARA EL
DIAGNÓSTICO DE FALLOS EN UNA UNIDAD DE CRACKING CATALÍTICO**

**GABRIEL ANDRES MAFFIOLD ARDILA
RAFAEL HERNÁN ROJAS GUALDRÓN**

**UNIVERSIDAD AUTONOMA DE BUCARAMANGA
ESCUELA DE CIENCIAS NATURALES E INGENIERIA
FACULTAD DE INGENIERIA MECATRONICA
BUCARAMANGA
NOVIEMBRE DE 2006**

**HERRAMIENTA INTELIGENTE PARA EL
DIAGNÓSTICO DE FALLOS EN UNA UNIDAD DE CRACKING CATALÍTICO**

**GABRIEL ANDRES MAFFIOLD ARDILA
RAFAEL HERNAN ROJAS GUALDRON**

Tesis

Este proyecto es presentado como requisito para optar el título de Ingeniero
Mecatrónico

DIRECTORES

**DR. ANTONIO FAUSTINO MUÑOZ MONER
DR. CARLOS FERNANDO AGUDELO**

**UNIVERSIDAD AUTONOMA DE BUCARAMANGA
ESCUELA DE CIENCIAS NATURALES E INGENIERIA
FACULTAD DE INGENIERIA MECATRONICA
BUCARAMANGA
NOVIEMBRE DE 2006**

CONTENIDO

*falta
paginación
normas?*

INTRODUCCIÓN

OBJETIVOS

		PAG
1	DISEÑO METODOLÓGICO	1
1.1	ESTADO DEL ARTE	1
1.2	METODOLOGÍA Y DISEÑO MECATRONICO	4
2	MARCO TEÓRICO	8
2.1	Detección Y Diagnostico De Fallos	8
2.2	Sistemas Expertos	12
2.3	Planta De Cracking	16
3	SISTEMA EXPERTO	21
3.1	Ingeniería del Conocimiento	21
3.2	Árboles De Fallos	24
3.3	Base De Conocimiento	29
3.3.1	Reglas Del Sistema Experto	29
3.3.2	Programación de las macros en Visual basic for applications Utilizando Microsoft Excel	33
3.4	Entorno Grafico	50
4	SISTEMA ELECTRÓNICO DE COMUNICACIÓN	53
5	PRUEBAS Y RESULTADOS	64

BIBLIOGRAFÍA

OBSERVACIONES, CONCLUSIONES Y RECOMENDACIONES

DEDICATORIA

Doy infinitas gracias...

A Dios, por el camino recorrido....

A mi hijo Dominic Santiago, por ser mi fuerza y templanza...

A mis padres y hermanos, por su amor y generosidad...

A mis amigos, Por darme apoyo y consuelo en el momento apropiado...

A la vida.... Por lo aprendido y aprehendido

Gabriel Andrés MaffioldArdila

AGRADECIMIENTOS

Agradecemos antes que todo a Dios que fue nuestro principal apoyo y quien nos brindo la fuerza espiritual para dar por finalizada esta etapa de nuestra carrera profesional.

A nuestros padres, que nos ofrecieron motivación, empeño y confianza para ayudarnos a salir adelante, no solo profesionalmente sino además como personas que se desenvolverán en una sociedad para beneficiar a los demás aportando soluciones optimas a los problemas y necesidades que con el tiempo se van dando a causa de la evolución de la perfección del mundo entero.

Agradecemos de una manera especial a nuestros directores: El Dr. Faustino Muñoz y el Dr. Carlos Agudelo, a nuestros asesores: Msc. Diego Tibaduiza, Dr. Juan Carlos García Díaz; quienes fueron un apoyo incondicional a través de cada una de las etapas de nuestra meta que aquí culmina.

También a todos aquellos que de una u otra manera estuvieron involucrados en el desarrollo circunstancial de esta investigación. Gracias por su colaboración y empeño ya que con ello aportan a sacar adelante procesos cognoscitivos, prácticos, reales y tangibles que sirvan al presente y futuro de esta sociedad.

LISTA DE FIGURAS

		Pág
Figura 1	Esquema sobre detección y diagnóstico de fallos basado en conocimiento	10
Figura 2	Estructura de un sistema experto	12
Figura 3	Planta de Cracking	17
Figura 4	Reactor	18
Figura 5	Torre Fraccionadora	19
Figura 6	Regenerador	20
Figura 7	Panel de control del operador	22
Figura 8	Alarmas del área de cracking	23
Figura 9	Circulación Errática de Catalizador	25
Figura 10	Flujo Inverso	26
Figura 11	Formación de Coke – Ensuciamiento	27
Figura 12	Pérdidas de Catalizador	28
Figura 13	Pantalla Inicial del Sistema Experto	50
Figura 14	Vector de Variables	51
Figura 15	Mensaje de Interrogación al Operador	51
Figura 16	Mensaje Exclamativo al Operador	52
Figura 17	Mensaje Informativo al Operador	52
Figura 18	Sistema Electrónico De Comunicación	53
Figura 19	Medidor de Caudal	54
Figura 20	Termocupla tipo j	54
Figura 21	Fieldpoint	55
Figura 22	Modulo FP-TC-120	55
Figura 23	Modulo FP-CTR-500	56
Figura 24	Ruta A Measurement & Automation	56
Figura 25	Dispositivos e interfaces	57
Figura 26	Reconocimiento Fieldpoint	57

Figura 27	Configuración Fieldpoint	58
Figura 28	Configuración Termocupla FP-TC-120	58
Figura 29	Ruta Acceso Labview	59
Figura 30	Rutina Ciclo For-Captura Pulsos Medidor de Caudal	60
Figura 31	Rutina Número De Pulsos Por Segundo	60
Figura 32	Rutina Conversión Pulsos A Caudal	61
Figura 33	Temperatura De Salida	61
Figura 34	Rutina Interfase Labview-Excel	62
Figura 35	Datos Pasados A Excel Desde Labview	63
Figura 36	Mensaje Inicialización PROLOG	64
Figura 37	Botón Alta Temperatura del Regenerador	65
Figura 38	Mensaje Baja Temperatura del Precalentador	65
Figura 39	Botón Circulación Errática del Catalizador	65
Figura 40	Mensaje de Pregunta sobre vibración en la Tubería	66
Figura 41	Mensaje de Pregunta sobre sonido de tren	66
Figura 42	Mensaje Recomendación al Operador	66
Figura 43	Botón Perdidas de Catalizador	67
Figura 44	Mensaje Recomendación al Operador 2	67
Figura 45	Estados de Operación Normales Planta Piloto	68
Figura 46	Prueba con valor de caudal normal	69
Figura 47	Mensaje no hay fallo en el flujo	69
Figura 48	Caída de flujo simulada	70
Figura 49	Problemas con el flujo	70
Figura 50	Temperatura de Salida Normal	71
Figura 51	Prueba con Valor de Temperatura Normal	71
Figura 52	Mensaje no hay fallos relaciones con formación de coke	72
Figura 53	Elevación de Temperatura Simulada	72
Figura 54	Problemas con la formación de coke	73

INTRODUCCIÓN

Actualmente, el control industrial gana una fuerza cada vez mayor, causando un efecto similar al que se observara con la máquina de vapor durante la revolución industrial de la Inglaterra de finales del siglo XVIII. Nos encontramos en un proceso que algunos han bautizado acertadamente con el nombre de segunda revolución industrial, debido a que se han venido presentando cambios profundos en los procesos de producción. Es por esto, que más que un objetivo es una obligación de los profesionales con el país, evitar que este se mantenga indiferente ante un cambio de tal magnitud que podría llevar a aumentar aun mas la brecha tecnológica que existe entre nosotros y quienes si han aceptado el nuevo reto.

La universidad ha decidido convertirse en parte de la solución, valiéndose para ello de la unión estratégica con las empresas del sector industrial de la región. Ha adoptado cada una de sus necesidades y se prepara para responder al desafío de hacerlas competitivas, especialmente en el momento en el que nos encontramos, en el cual la globalización obliga a las industrias a ser eficientes con el fin de poder competir con otras alrededor del mundo. La única forma de lograr este objetivo, es crear un vínculo entre la universidad y el sector productivo, promoviendo así la investigación y buscando juntos alcanzar un alto nivel en tecnología.

El desarrollo de la automatización industrial, ha llevado a que toda planta de una empresa importante haya sido automatizada y a que se haya implementado un control sobre todos sus procesos. ECOPETROL no es la excepción, teniendo controles eficientes y respuestas satisfactorias en cada una de sus plantas. Desafortunadamente, estos controles implementados, así como los que encontramos hoy en día en el mercado, dan excelente resultado para un funcionamiento normal de la planta, pero no están diseñados para responder de manera automática en escenarios de fallo. Esto representa una gran responsabilidad para el operario, el cual, si bien está altamente calificado y posee una gran experiencia en este tipo de anomalías, muchas veces se enfrenta a un bombardeo de alarmas que dejan de ser una solución y pasan a convertirse en un problema. El propósito de este proyecto es fusionar la capacidad de respuesta de un sistema computarizado con el conocimiento y la experiencia del operario, con el fin de lograr que el sistema de control detecte tempranamente los errores en las variables de control, y también utilice una base de conocimiento para saber cual es la causa de este y cual es el procedimiento a seguir.

OBJETIVOS

GENERAL

Desarrollar una herramienta que implementa un método para el diagnóstico de fallos en una unidad de cracking catalítico, valiéndose para esto de la creación de una base de conocimiento.

ESPECÍFICOS

- Recopilar una base de conocimiento que asocie las causas de falla típicas en una unidad de cracking catalítico, por medio del reconocimiento de los síntomas observables en las mismas.
- Desarrollar una herramienta que permita realizar la validación de las hipótesis planteadas para explicar los síntomas observados en los fallos.

1. DISEÑO METODOLOGICO

1.1 ESTADO DEL ARTE

La detección y el diagnóstico de fallos es un problema importante en los procesos de ingeniería. Es el componente principal de la administración de eventos anormales (AEM), el cual ha atraído gran atención recientemente. AEM trata con la detección temprana, diagnóstico y corrección de condiciones de fallos en un proceso. El diagnóstico y detección temprana de fallos en el proceso mientras la planta se encuentra aun operando en una región controlable pueden ayudar a evitar el avance de eventos anormales y reducir las pérdidas de productividad. Dado que las industrias petroquímicas pierden un estimado de 20 billones de dólares al año, han escogido AEM como el principal problema a resolver. Además, hay un considerable interés en este campo de practicantes industriales así como de investigadores académicos, contrario a hace unos años atrás. Existe una gran cantidad de teoría en los rangos en el diagnóstico de fallos en un proceso desde métodos analíticos hasta inteligencia artificial y aproximaciones estadísticas.

Desde una perspectiva de modelos, existen métodos que requieren modelos de procesos precisos, modelos semicuantitativos o modelos cualitativos. Por otro

lado, existen métodos que no asumen ningún tipo de información de modelos y recaen únicamente en los datos históricos del proceso. Además, dado el conocimiento del proceso, existen diferentes técnicas de búsqueda que se pueden aplicar al diagnóstico de desempeño. Existe una serie de metodologías y alternativas que a menudo se convierte en un reto para cualquier aspirante que no sea especialista en dichas técnicas. Algunas de estas ideas parecen muy distantes unas de las otras que un investigador no experto o un practicante se pregunta a menudo por la conveniencia de un método para su situación de diagnóstico. Aunque ha habido algunos excelentes resúmenes sobre este campo en el pasado, muchas veces se enfocaban en un campo en particular, así como los modelos analíticos, de esta extensa disciplina.

A nivel nacional, ECOPETROL con su programa de Salud Ocupacional, queriendo reducir el estrés del trabajo en una situación de condiciones anormales, ha iniciado la implementación de metodologías para el gerenciamiento de las alarmas de las plantas en sus refinerías, incluyendo el uso de herramientas inteligentes que ayuden al operador en la toma de decisiones durante una situación anormal. El Instituto Colombiano del Petróleo ha iniciado trabajos exploratorios para la investigación acerca del control de la planta en situaciones anormales de funcionamiento comenzando por la detección temprana de fallas con el fin de evitar llegar a una situación en la cual la planta pueda sufrir un daño permanente o se le imponga una responsabilidad excesivamente grande al operador produciéndole un nivel demasiado alto de tensión y angustia.

Se espera realizar una contribución no solo a ECOPELROL sino también a la industria en general por medio de la obtención de resultados satisfactorios que demuestren no solo la eficiencia del método sino también el ahorro que puede llegar a significar en el momento en que una planta se vea obligada a trabajar en modo de fallo. Es nuestro objetivo que en las futuras investigaciones que realicen estudiantes o investigadores en los próximos años encuentren en este proyecto un apoyo y no se vean obligados como nosotros a solamente referirse en el estado del arte a proyectos realizados por fuera del país.

1.2 METODOLOGIA Y DISEÑO MECATRONICO

En primer lugar, se realizó una interpretación de los objetivos teniendo en cuenta que el proyecto debe satisfacer las necesidades y expectativas de las dos instituciones patrocinadoras del mismo como lo son la Universidad Autónoma de Bucaramanga y el Instituto Colombiano del Petróleo. En la realización de dicha interpretación se encontró que, si bien normalmente los proyectos de control realizados en la universidad se hacen utilizando como plataforma el Matlab, el instituto tenía la necesidad de que esta herramienta fuera montada utilizando Microsoft Excel, debido a que es en donde se realiza la consulta de la base de datos en tiempo real que posee la planta. Siguiendo con la interpretación, encontramos que, si bien el instituto pedía que las pruebas se hicieran de forma teórica por medio de la introducción manual de vectores de fallos sacados del historial de datos de la planta, la universidad exigía la realización práctica de la toma de estos datos, por lo cual para la prueba se utilizaron dos variables controladas por medio del field point, lo cual será explicado en detalle más adelante. Se encontró también que la propuesta del instituto de realizar la herramienta utilizando el encadenamiento hacia atrás sería mucho más fácil de realizar utilizando el interprete de Prolog, el cual por defecto realiza este tipo de búsqueda basado en la lógica de predicados, además de tener la ventaja de ser de uso libre. Una vez elegidas las plataformas sobre las cuales iría montada la

herramienta, se procedió a buscar la forma de comunicar ambos programas, para lo cual se encontró una librería de Amzi que permitía trabajarlos como si fueran uno sólo.

En segundo lugar, se procede al cumplimiento metódico de cada uno de los objetivos. El primero, la recopilación de una base de conocimiento que asocie las causas de falla típicas en una unidad de cracking catalítico, por medio del reconocimiento de los síntomas observables en las mismas. Para esto, se realizó una investigación acerca del proceso de cracking catalítico en sí, basados en el manual de operación de la planta. Seguido de esto, se procedió a una recopilación de los fallos más comunes en la misma y a su consiguiente relación con sus causas y pasos a seguir cuando estos se presentan. Una vez hecho esto se procedió a su clasificación de acuerdo a la parte particular de la planta involucrada en dichos fallos y posteriormente a la creación de la base de conocimiento. Para la creación de la base de conocimiento se recurrió a la lógica de predicados y a un conjunto de reglas que garantizaban el correcto encadenamiento hacia atrás, el cual permitía la validación de las hipótesis planteadas como posibles causas de un estado de fallo. La base de conocimiento fue creada en Prolog, lo cual permite interrogar al intérprete sobre el grado de verdad de cualquiera de las hipótesis que se puedan tener como causas de fallo una vez actualizada la base de hechos. La base de hechos consiste en actualizar al sistema sobre el grado de verdad de los síntomas que se puedan estar o no presentando en la planta en un preciso momento. Una vez superada y probada la etapa de la base de conocimiento se

prosiguió al cumplimiento del segundo objetivo, el cual era desarrollar una herramienta que permitiera realizar la validación de las hipótesis planteadas para explicar los síntomas observados en los fallos. Para esto, se realizó una macro en Excel, la cual a su vez se encontraba dividida en dos partes. La primera, la actualización de la base de hechos, para lo cual el programa debía revisar cada una de las variables controladas de la planta y darle un grado de verdad a cada uno de los síntomas y enviárselos a Prolog. Para la revisión de dichos datos, se realizó una discretización de cada una de las variables de la planta, teniendo en cuenta sus límites y rangos de operación normal, para así poder afirmar con seguridad si se está presentando un síntoma en un instante de tiempo o no. En la actualización de la base de conocimiento aparece un nuevo problema, el cual consistía en el reconocimiento de síntomas no reconocibles para el sistema de control, pero tangibles para el operario. Para solucionar dicho inconveniente, se recurrió a la interrogación al operario sobre el grado de verdad de un síntoma particular en la planta, mediante un mensaje de pregunta en su panel de control. Una vez terminada la etapa de actualización de la base de hechos, se continúa con la etapa de interrogación al intérprete del Prolog. La idea es recibir una alarma de posible fallo en cierto lugar de la planta y realizar la verificación de las posibles hipótesis relacionadas con fallos en ese punto de la planta. Una vez realizada la respectiva interrogación existen dos posibilidades: una, que no se este presentando ningún fallo y sea una falsa alarma, o la otra, que se este presentando un fallo, en cuyo caso, el operario verá en su pantalla una recomendación al problema que se está presentando.

Una vez probada la herramienta en forma teórica como lo aconsejó el instituto, se prosiguió a realizar la adquisición de los datos a través del field point para cumplir con el requisito de la prueba física puesto por la facultad, para lo cual se adquieren los datos de los sensores y se exportan a Excel. Esto se explicará con detalle en un capítulo más adelante.

Realizadas las pruebas y habiendo certeza de la fiabilidad de la herramienta, se procede a la creación de la interfaz de usuario del sistema. La idea es lograr una herramienta práctica, la cual se limite a mostrar únicamente lo necesario, en nuestro caso, las recomendaciones que da el sistema al operario, o de considerarlo necesario, la interrogación a este sobre aquellos síntomas que no pueda controlar numéricamente. Como resultado obtenemos una herramienta que libera al operario de las molestas alarmas y le presenta una solución en lugar de un problema.

2. MARCO TEORICO

2.1 DETECCION Y DIAGNOSTICO DE FALLOS

La detección y diagnóstico de fallos, puede resultar compleja ya sea que se utilice información analítica o heurística. Es sabido que a una baja calidad de los sensores, aumenta la incertidumbre al disminuir la exactitud y la precisión en las mediciones por el lado de la información analítica. Si consideramos la información heurística, se puede apreciar que la incertidumbre es algo inherente en ella, pues depende del punto de vista del operador humano, que frecuentemente utiliza un razonamiento aproximado para la solución de problemas complejos, con un grado aceptable de eficacia.

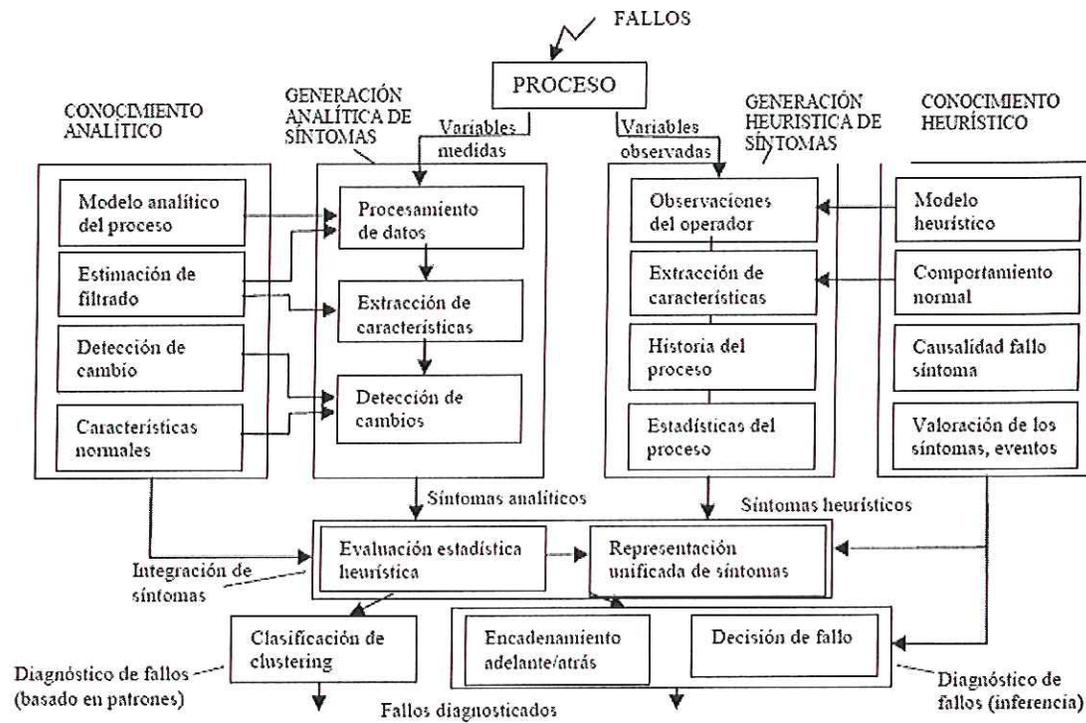
El conocimiento que podemos adquirir acerca del comportamiento de un sistema físico se fundamenta principalmente en la adquisición y valoración de dos tipos de información: la *cuantitativa*, que se adquiere a través de los diferentes instrumentos de medición de las variables que caracterizan el funcionamiento del sistema y la *cualitativa*, adquirida por humanos a través de los órganos sensoriales y procesada por el cerebro, que suministra información generalmente en forma lingüística. En el proceso de detección y diagnóstico, no se puede despreciar

ninguna clase de información, debido a que las dos son fundamentales para la generación de los síntomas de fallos.

El diagnóstico de fallos consiste en determinar el tipo, tamaño y localización de los fallos, así como su tiempo de detección, basado en los síntomas analíticos y heurísticos. Si no se dispone de un conocimiento total de los síntomas, se pueden aplicar métodos de clasificación, los cuales permiten relacionar los síntomas con los fallos. Para ello se utilizan métodos para clasificación geométrica y estadística o redes neuronales y lógica fuzzy. Pero si se dispone de conocimiento a priori de las causalidades de los síntomas de fallos, como pueden ser las redes causales, entonces, se usan estrategias de razonamiento de diagnóstico.

A continuación se muestra un esquema global sobre la detección y diagnóstico de fallos basado en el conocimiento. Las tareas principales pueden subdividirse en generación analítica de síntomas, generación heurística de síntomas y diagnóstico de fallos. En la generación de síntomas se encuentra incluida la detección de fallos que en este capítulo se estudiará con respecto a los modelos matemáticos de los procesos, de donde derivan los síntomas analíticos.

Figura 1. Esquema sobre la detección y diagnóstico de fallos basados en conocimiento.



En la generación analítica de síntomas, el conocimiento analítico del proceso debe producir información cuantificable. Para lograr esto, se deben procesar datos basados en variables medidas del proceso. Primero se generan los valores característicos por:

- *Revisión de los valores límite*, de las señales directamente mesurables. Los valores característicos son las tolerancias excedidas por las señales.

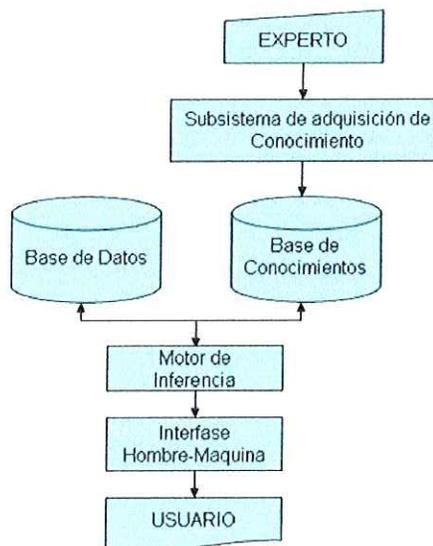
- *Análisis de señales* directamente medidas por el uso de modelos de señales como las funciones de correlación, espectros de frecuencia, autoregressive moving average (ARMA), valores característicos (variancias, amplitudes, frecuencias, o parámetros de modelo).
- *Análisis de proceso* donde se usan los modelos matemáticos del proceso junto con la estimación de parámetros, estimación de estados y los métodos de ecuaciones de paridad. Los valores característicos son parámetros, variables de estado o residuos.

En algunos casos, pueden extraerse características especiales de estos valores característicos como son: coeficientes del proceso físicamente definidos o residuos transformados o filtrados. Estas características son después comparadas con las características del proceso sin fallos. Luego se aplican métodos de detección de cambios y clasificación. Los cambios resultantes (discrepancias) en las señales medidas descritas anteriormente, se consideran como los *síntomas analíticos*.

2.2 SISTEMAS EXPERTOS

Los sistemas expertos son programas que reproducen el proceso intelectual de un experto humano en un campo particular, pudiendo mejorar su productividad, ahorrar tiempo y dinero, conservar sus valiosos conocimientos y difundirlos mas fácilmente. Deben ser capaces de explicar sus decisiones y el razonamiento subyacente. A veces será necesario trabajar con información incierta o incompleta. Algunas aplicaciones incluyen el diagnostico medico, la localización de fallas en equipo o maquinaria, etc.

Figura 2. Estructura de un Sistema Experto.



El conocimiento del experto se obtiene a través del subsistema de adquisición de conocimientos y se organiza en una base de conocimientos, y en función de la información disponible, se imita la forma de actuar del experto explorando en la base de conocimientos hasta encontrar la solución usando el motor de inferencia.

El subsistema de adquisición del conocimiento permite que se puedan modificar, añadir o eliminar conocimiento en el sistema experto. El sistema funcionará correctamente sólo si se mantiene actualizado su conocimiento y esto es muy necesario en entornos dinámicos. El subsistema de adquisición permite efectuar ese mantenimiento, anotando en la base de conocimientos los cambios que se producen. Es necesario estructurar de una forma adecuada los conocimientos, todo este conocimiento se almacena en lo que se conoce como la base de conocimientos.

Las bases de conocimiento incluyen el conocimiento específico en el dominio de la aplicación, hechos sencillos, reglas que describen relaciones o fenómenos, heurísticas, e ideas para la solución de problemas en el dominio específico. El motor de inferencia extrae conclusiones a partir de los datos simbólicos que están almacenados en las bases de hechos y de conocimiento.

En la interfaz Hombre-Maquina, el usuario puede consultar con el sistema a través de menús, gráficos, etc. y este le responde con resultados. También le puede mostrar la forma en que extrae las conclusiones a partir de los hechos.

Los *marcos (frames)* son estructuras de datos donde se almacenan información concreta de un cierto concepto e información relacional para completar la definición del concepto. Se representan por una serie de campos y los valores asociados a los mismos, pudiendo ser otro "frame". Las *redes semánticas* son representaciones gráficas del conocimiento, mediante nodos, que representan objetos o conjuntos de objetos, y arcos, que relacionan dichos objetos. Tanto los "frames" como las redes son representación descriptivas, difíciles de usar para representar el razonamiento, y la forma de representación más extendida son las *reglas* que representan la forma de razonar. Tienen la forma *If <condición> then <acción/conclusión>*. Es importante el orden de ejecución de las reglas, por ello se les suele dotar de prioridades. Una regla que activa otra se denomina metaregla, y se suelen usar para desarrollar progresivamente el conocimiento del experto.

El motor de inferencia que sirve para inferir conclusiones a partir de los hechos y las reglas. Todos los algoritmos son del tipo "pattern-matching", van disparando reglas a medida que se cumplen las condiciones. Se pueden diferenciar dos mecanismos de inferencia:

En el encadenamiento hacia adelante se extraen conclusiones a partir del cumplimiento de las condiciones de ciertas reglas que, a su vez, provocaran el cumplimiento de las condiciones en otras reglas hasta que no se cumplan en ninguna de ellas, mientras que en el encadenamiento hacia atrás se suponen

ciertas las conclusiones de una regla y, como consecuencia, se van disparando aquellas reglas que provocarían la regla original. El proceso acaba si no se cumplen las condiciones de las reglas o si se cumplen para todas las reglas.

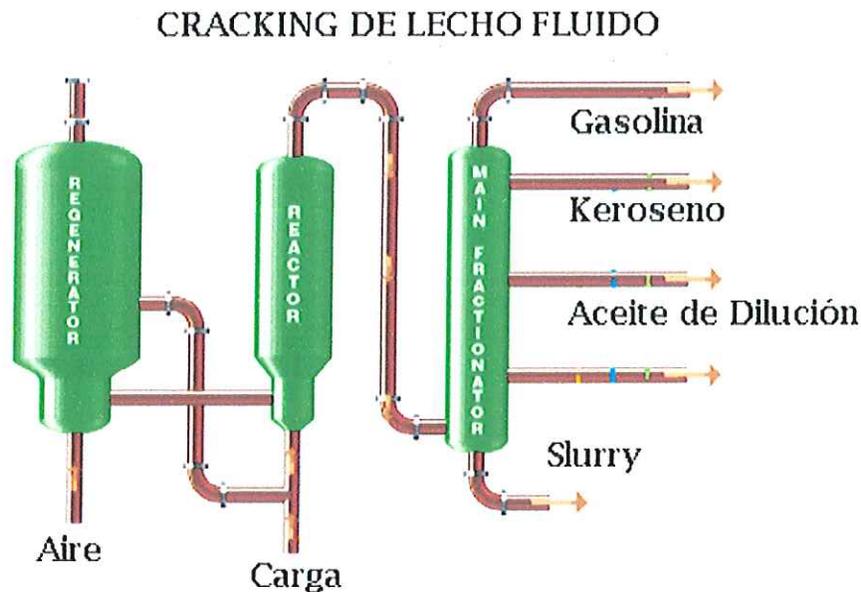
2.3 PLANTA DE CRACKING

El cracking catalítico es un proceso en el cual un catalizador en forma de microesferas craquea moléculas pesadas de hidrocarburo cuyo punto de ebullición es igual o superior a los 315°C a hidrocarburos livianos de cadena corta cuyo punto de ebullición se encuentra por debajo de los 221°C. Su finalidad no es otra que la de obtener la mayor cantidad de hidrocarburos livianos de gran aprecio para la industria, la mayoría de las cargas a las unidades de ruptura catalítica la constituyen gasóleos, aceites pesados como el DMOH y el DMO (Aceite demetalizado hidrogenado y Aceite Demetalizado, respectivamente.)

El cracking produce naftas de muy alta calidad. Las naftas de menor grado que se obtienen por destilación también pueden mejorarse por el proceso de reformación. Mediante este proceso, se modifica la estructura molecular por calentamiento bajo presión generalmente en presencia de un catalizador.

El cracking y la reforma hacen que la refinería pueda responder a los cambios que se producen en la demanda. Las personas a cargo de la programación de la producción se encargan de definir el ruteo de las distintas corrientes obtenidas en la destilación a través de los diversos procesos de conversión, para adecuar la cantidad y calidad de los productos finales, según la demanda.

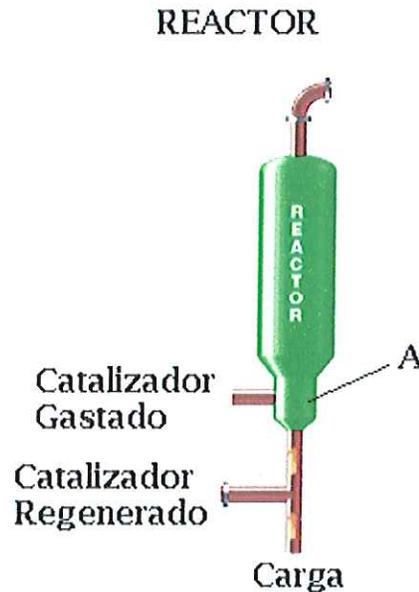
Figura 3. Planta de Cracking.



El Cracking Catalítico fluido es un proceso de conversión catalítica en el cual se convierte una carga de gasóleos en gas seco, LPG, butano, gasolina de alto octanaje, flux oil, aceite pesado de ciclo, alquitrán aromático y coke. Todos estos elementos son extraídos como productos, con excepción del coke, el cual es consumido internamente en el proceso para producir el calor requerido por la reacción de Cracking. El catalizador es típicamente una matriz de soporte sintético de sílica, alúmina o arcilla, a la cual se le ha adicionado una zeolita. Son fabricados en forma de un polvo fino con un tamaño promedio de partícula de 60 micrones, y aunque son sólidos, son mantenidos en suspensión o “fluidizados” en los recipientes y en la línea de transferencia por medio de una corriente gaseosa tal como los productos de la reacción, vapor o aire.

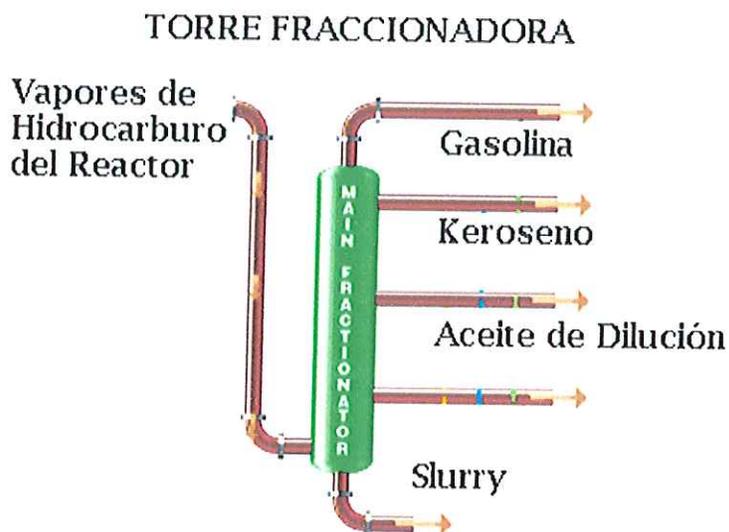
En este estado, el catalizador se comporta como un fluido y fluye entre el reactor y el regenerador.

Figura 4. Reactor.



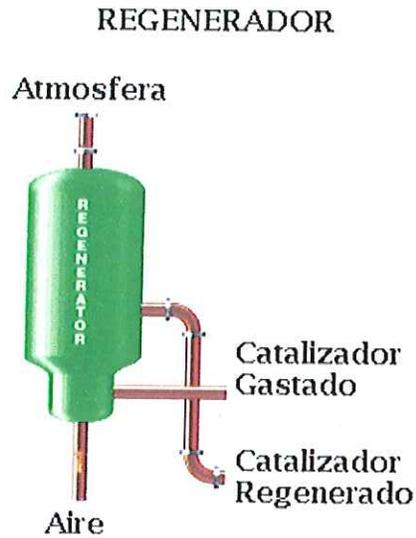
El Reactor opera entre 950 y 980°F y a aproximadamente 25 psig. La carga de gasóleos atomizada con vapor es inyectada en el fondo del elevador de carga a través de boquillas especialmente diseñadas Ver Figura 2 (A). Cuando el aceite se pone en contacto con el catalizador caliente en el elevador, se vaporiza y craquea hacia componentes más livianos. La mayoría de la conversión ocurre en el elevador de carga (80%), el craqueo adicional ocurre tanto en el lecho del reactor como en el recipiente mismo durante el proceso de separación del catalizador de los vapores de hidrocarburo.

Figura 5. Torre Fraccionadora.



Los vapores de aceite pasan a través de la cima del Reactor hacia la sección de fraccionamiento Figura 3, donde los productos son separados. El catalizador con el coke depositado sobre él en la reacción de cracking, cae hacia abajo a la sección de despojamiento. En esta zona el catalizador es despojado con vapor para removerle el aceite que aún permanece sobre la superficie del catalizador y los vapores de aceite arrastrados entre las partículas de catalizador. El catalizador entonces fluye a través de la U gastada hacia el regenerador.

Figura 6. Regenerador.



En el regenerador Figura 6 , el coke es quemado del catalizador con aire a temperatura entre 1250 – 1340 °F, en presencia de un promotor de combustión de CO, y presión aproximada de 25 psig. El catalizador regenerado caliente fluye sobre el pozo de rebose hacia la U de catalizador regenerado, regresa de nuevo al elevador de carga del Reactor.

El catalizador y los gases de chimenea pasan a través de los ciclones en los cuales se remueve la mayoría del catalizador arrastrado. El catalizador es regresado al lecho y los gases venteados a la atmósfera.

3. SISTEMA EXPERTO

3.1 INGENIERIA DEL CONOCIMIENTO

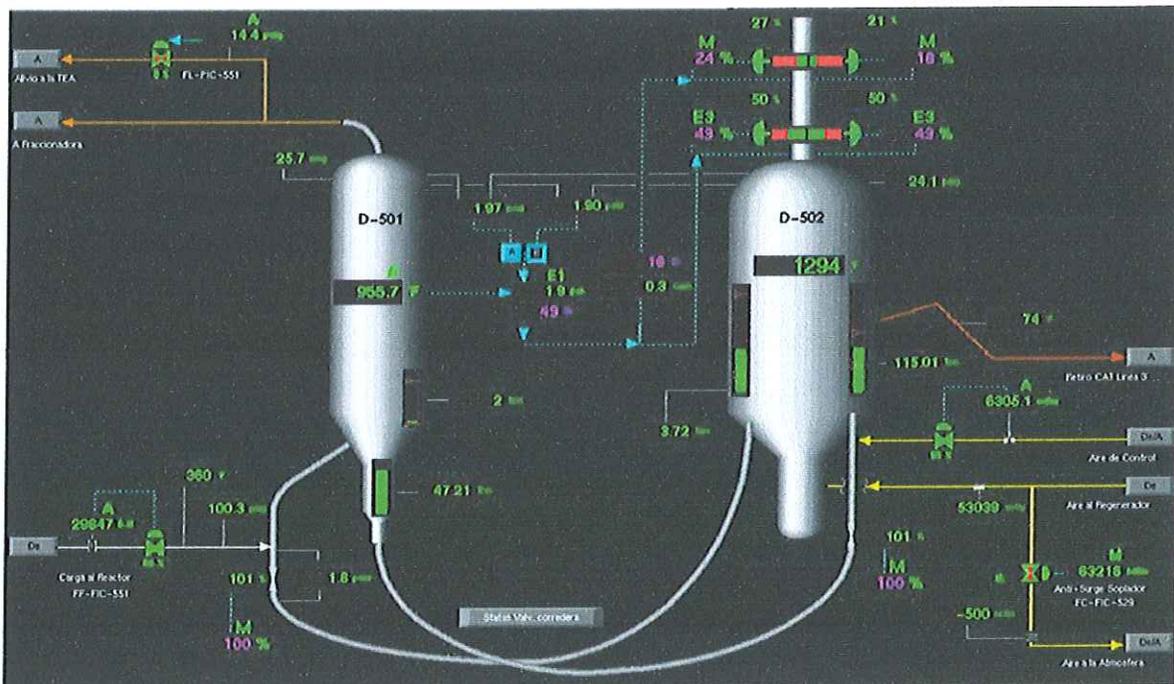
Una de las etapas mas importantes del proyecto fue la consecución de toda la información referente a la unidad de cracking catalítico Modelo 4, su funcionamiento, el nivel de monitoreo que tenia y los fallos típicos para la unidad.

Inicialmente, nos referimos a unos árboles de fallos típicos que el Instituto Colombiano del Petróleo nos facilito, después de haberlos leído y analizados observamos que algunos de los árboles debían ser ajustados para el modelo de planta que estábamos trabajando, ya que los árboles venían generalizados para las unidades de cracking existentes en el mercado. Con ayuda de los expertos en el ICP empezamos el análisis y con ayuda de un experto de cracking fuimos llegando a los árboles de fallos específicos para la unidad con la que trabajamos.

A continuación fue necesario conocer el monitoreo que se tenían en la planta de cracking modelo 4, aprovechando que el Instituto tenia 2 estudiantes en practica en la refinería de Cartagena, a través de llamadas telefónicas, emails y reuniones via Internet, nos facilitaron información sobre las variables que se monitorean en la planta, los valores típicos, los valores de fallo, Si generaban señal de alarma e

incluso obtuvimos una muestra del panel de control del operador donde nos muestra los valores y la ubicación de los instrumentos de la planta de cracking modelo 4 de la refinería de Cartagena.

Figura 7. Panel de control del operador.



En el dibujo siguiente podemos ver un fragmento de la archivo que se nos facilito de las variables que se controlan en la unidad de cracking catalítico modelo 4 de la refinería de Cartagena, este archivo nos muestra el nombre del instrumento, una descripción, las unidades en que mide, los limites superiores e inferiores de fallos y si generan alarma o no.

Figura 8. Alarmas del Área de Cracking.

ALARMAS DEL AREA DE CRACKING - DCS								
EN_H2= Habilita alarma Alto- Alto			EN_L1= Habilita alarma Bajo			EN_L2= Habilita alarma Baja		
NAME	DESCR	UNIT	EN_H2	HI_LIM2	EN_HI	HI_LIM1	EN_L1	LO_LIM1
FC-FI-515	Vap. 600 a C-501	lb/h	0	90	0	80	1	60000
FC-PDI-503	DIF SUC/DESC C-501	psid	1	33	1	32,5	0	-80
FC-FI-527	Aire PPAL al RG	scfm	0	90	0	80	1	41000
FC-LI-1A	Catalizador RX	Ton	0	90	1	25	1	0
FC-LI-501B	Pozo Rebose RG	%	0	90	1	90	1	25
FC-LI-1B	Catalizador RX	%	0	90	1	75	1	0
FC-LI-3B	Cataliz Lecho RG	Ton	0	90	1	150	1	75
FC-LI-501A	Pozo Rebose RG	Ton	0	90	1	8,50E+00	1	1,50E+00
FC-LI-3A	Cataliz Lecho RG	%	0	90	1	80	1	20
FC-LI-514	Nivel TK501 Alarma	%	0	110	1	100	0	25
FC-PI-5001	Reactor D-501	psig	0	90	1	26,5	0	-80
FC-PI-5002	Regenerador D-502	psig	0	90	1	26	0	-80
FC-TI-518	Temp. Reactor	°F	0	1320	1	990	1	935
FCFI501-FCFI529	Dif TX Aire C501	scfm	0	90	1	1000	1	-1000
FC-TI-509	Desc C-502	°F	0	90	1	520	0	-80

Gran parte de la discretización de las variables la hicimos con ayuda de la información obtenida, pero fue necesario realizar algunas llamadas a la refinería para hablar directamente con alguno de los operadores y hacerles preguntas sobre variables que no son medidas en la planta, como es el caso de la vibración en las tuberías. Estas variables que no nos fue posible discretizar con valores numéricos, les dimos valores booleanos, de esta manera, el operador solo deberá responder Si o No dependiendo de la alarma.

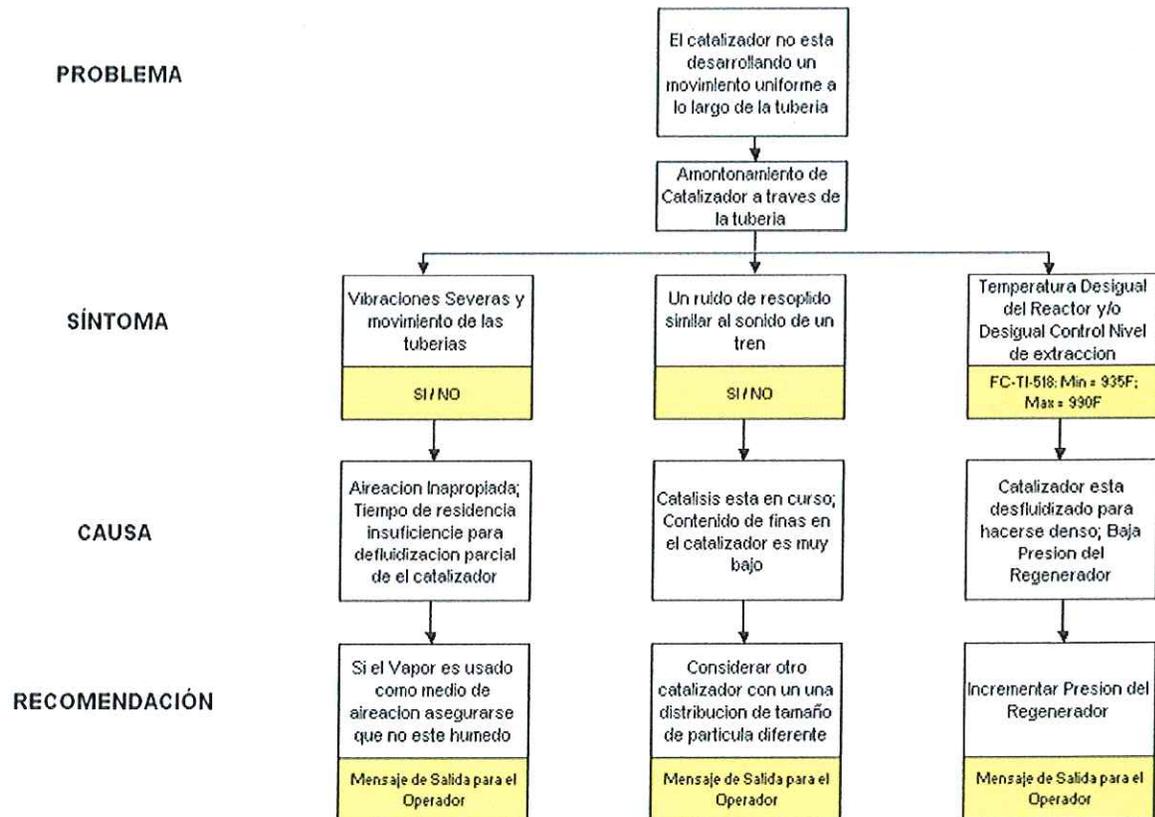
3.2 ARBOLES DE FALLOS

Después de haber realizado todo el trabajo de ingeniería del conocimiento, fuimos modificando los árboles de fallos de tal manera que se ajustaran a la unidad de cracking catalítico modelo 4 de la refinería de Cartagena, debido a que existen diferentes características mecánicas entre unidades de cracking catalítico fue necesario eliminar los árboles que relacionaban el flujo de catalizador con la válvula de corredera en las Ues , ya que la modelo 4 no controla el flujo de catalizador con las válvulas de corredera de las Ues, sino con las que están ubicadas a la salida de gases del regenerador.

CIRCULACION ERRATICA DE CATALIZADOR

Para el árbol de Circulación Errática de Catalizador, las variables VIBRACIONES SEVERAS/ MOVIMIENTO DE LAS TUBERIAS y RUIDO DE RESOPLIDO SIMILAR AL DE UN TREN, debido a que estas variables no las miden en la planta de cracking se le asigno un SI o No para que de esta manera el operador en el cuarto de control le pregunte al operador de campo si se están experimentando. Y La Variable TEMPERATURA DESIGUAL DEL REACTOR y/o trabajamos la FC-TI-518 y sus rangos son: Max 990F y min 935F.

Figura 9. Circulación Errática de Catalizador.

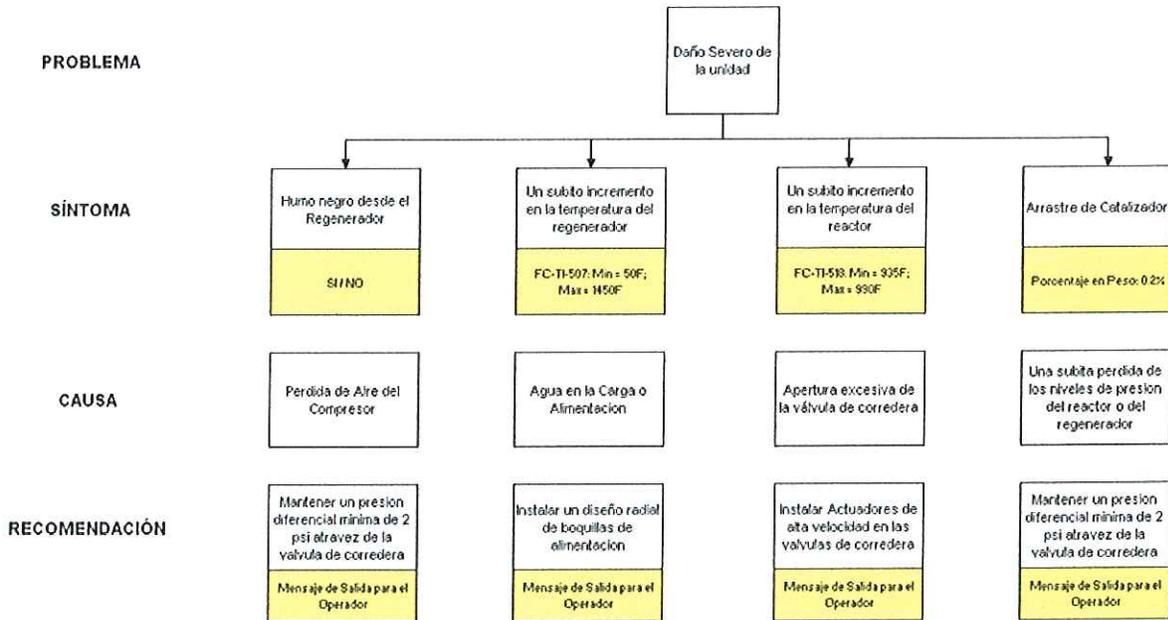


FLUJO INVERSO

La variable HUMO NEGRO DESDE EL REGENERADOR, debido a que esta variable no las miden en la planta de cracking se le asigno un SI o No para que de esta manera el operador en el cuarto de control le pregunte al operador de campo si se están experimentando. La variable SUBITO INCREMENTO EN LA TEMPERATURA DEL REGENERADOR con el instrumento FC-TI-507 y sus rangos: Max 1450F y Min 50F. La variable SUBITO INCREMENTO EN LA TEMPERATURA DEL REACTOR con el instrumento FC-TI-518 y sus rangos: Max

990F y min 935F. La variable ARRASTRE DE CATALIZADOR con la prueba de laboratorio se observa el porcentaje máximo que debe ser de 0.2%.

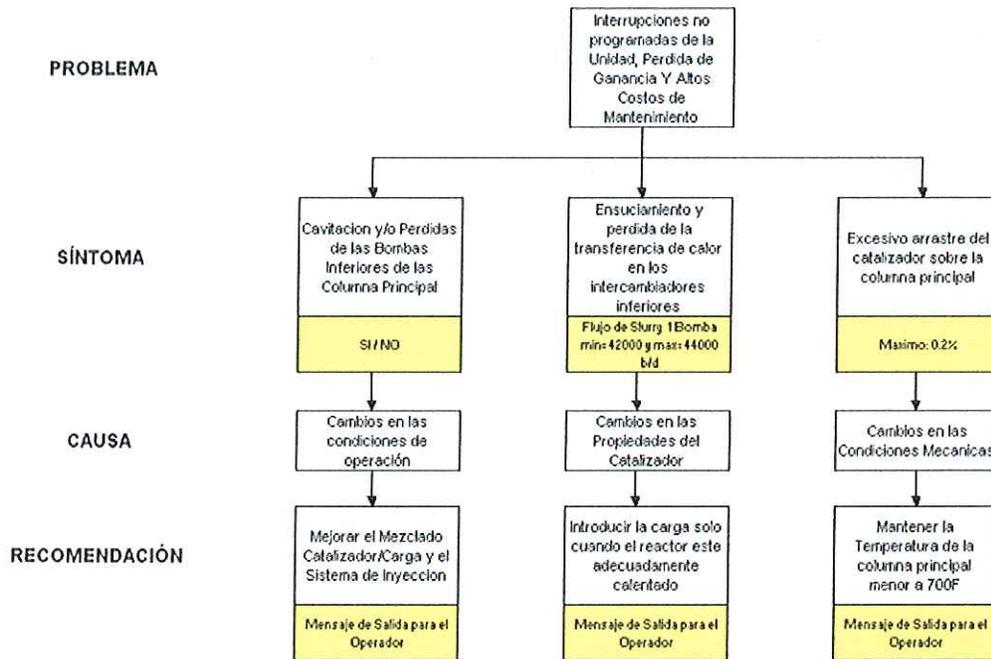
Figura 10. Flujo Inverso.



FORMACION DE COKE - ENSUCIAMIENTO

La variable CAVITACION Y PERDIDAS DE LAS BOMBAS COLUMNA PRINCIPAL, debido a que esta variable no las miden en la planta de cracking se le asigno un SI o No para que de esta manera el operador en el cuarto de control le pregunte al operador de campo si se están experimentando. La variable FLUJO DE SLURRY con el instrumento FF-FI-2 y sus rangos: Max 44000 b/d y Min 42000 b/d. La variable ARRASTRE DE CATALIZADOR con la prueba de laboratorio se observa el porcentaje máximo que debe ser de 0.2%.

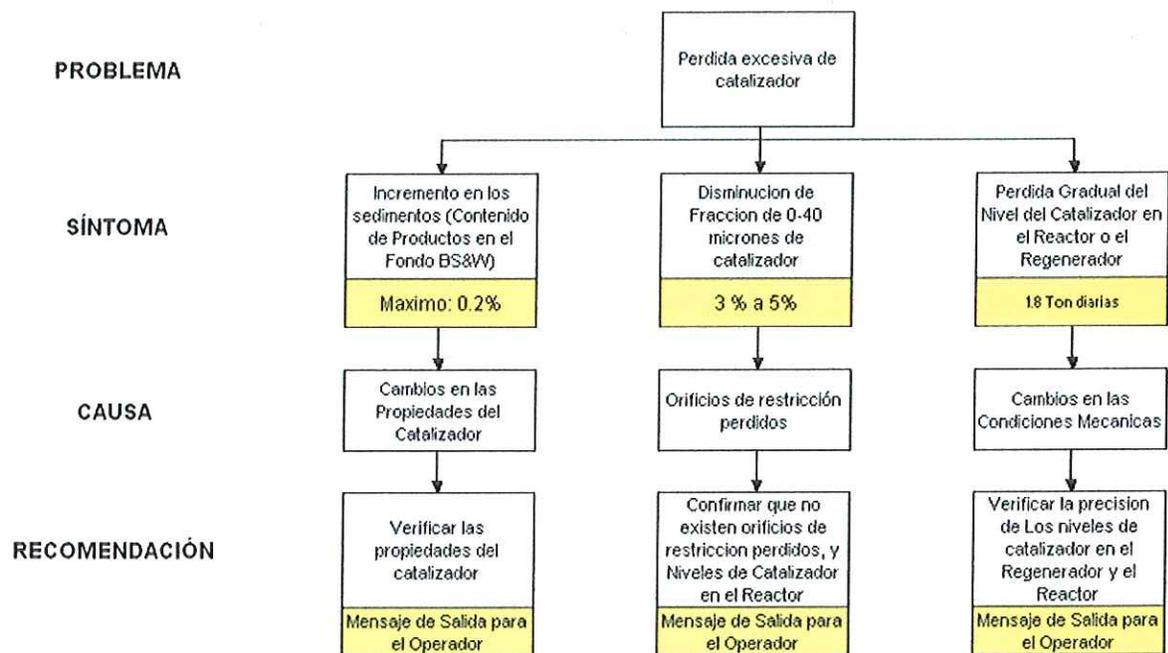
Figura 11. Formación de coke – ensuciamiento.



PERDIDAS DE CATALIZADOR

La variable PERDIDA GRADUAL DEL NIVEL DE CATALIZADOR se mide con la diferencia entre catalizador añadido y el extraído y el valor máximo de pérdidas es de 1.8 Toneladas diarias. La variable INCREMENTO EN LOS SEDIMENTOS con la prueba de laboratorio BS&W se observa el porcentaje máximo que debe ser de 0.2%. Y la variable DISMINUCION DE FRACCION DE 0-40 MICRONES DE CATALIZADOR con la prueba de laboratorio al catalizador y los porcentajes normales están entre 3 y 5%.

Figura 12. Pérdidas de Catalizador.



3.3 BASE DE CONOCIMIENTO

Toda la base de conocimiento se realizo en PROLOG, este intérprete nos permite indagar al sistema de una manera más natural como lo veremos a continuación:

Envía un mensaje confirmando que el intérprete de PROLOG se está ejecutando.

```
hello(Caller, Greeting) :-  
  
    strcat($Prolog Iniciado: $, Caller, S1),  
  
    strcat(S1, $, Amzil Prolog.$, Greeting).
```

Declaración de funciones AND, OR, IF, THEN

```
:-op(800,fx,if).  
:-op(700,xfx,then).  
:-op(300,xfy,or).  
:-op(200,xfy,and).
```

3.3.1 REGLAS DEL SISTEMA EXPERTO

Formacion de Coke

```
if formacion_de_coke and  
  
cavitacion_y_o_perdida_en_las_bombas_inferiores_de_la_columna_principal  
  
then cambios_en_las_condiciones_de_operacion.
```

If formacion_de_coke and
ensuciamiento_y_perdida_de_la_transferencia_de_calor_en_los_intercambiadores_inferiores **then** cambios_en_las_propiedades_del_catalizador.

If formacion_de_coke and
excesivo_arrastre_del_catalizador_sobre_la_columna_principal
Then cambios_en_las_condiciones_mecanicas.

Alta Temperatura del Regenerador

If alta_temperatura_del_regenerador and
niveles_altos_de_tierras_raras_en_el_catalizador
Then baja_temperatura_del_precalentador.

Circulacion Erratica de Catalizador

If circulacion_erratica_de_catalizador and
vibraciones_severas_y_movimiento_de_las_tuberias
Then aireacion_inapropiada.

If circulacion_erratica_de_catalizador and
un_ruido_de_resoplido_similar_al_sonido_de_un_tren
Then catalisis_esta_en_curso.

If circulacion_erratica_de_catalizador and
temperatura_desigual_del_reactor_y_o_desigual_control_nivel_de_extraccion
Then baja_presion_del_regenerador.

Flujo Inverso

If flujo_inverso and humo_negro_desde_el_regenerador

Then perdida_de_aire_del_compresor.

If flujo_inverso and un_subito_incremento_en_la_temperatura_del_regenerador

Then agua_en_la_carga_o_alimentacion.

If flujo_inverso and un_subito_incremento_en_la_temperatura_del_reactor Then

apertura_excesiva_de_la_valvula_de_corredera.

If flujo_inverso and arrastre_de_catalizador

Then

una_subita_perdida_de_los_niveles_de_presion_del_reactor_o_del_regenerador.

Perdidas de Catalizador

If perdidas_de_catalizador and incremento_en_los_sedimentos

Then cambios_en_las_propiedades_del_catalizador.

If perdidas_de_catalizador and

disminucion_de_fraccion_de_040_micrones_de_catalizador

Then

confirmar_que_no_existen_orificios_de_restriccion_perdidos_y_niveles_de_catalizador_en_el_reactor.

If perdidas_de_catalizador and

perdida_gradual_del_nivel_del_catalizador_en_el_reactor_o_el_regenerador

Then cambios_en_las_condiciones_mecanicas.

Codigo que configura al interprete para que realice la inferencia por el metodo de encadenamiento hacia atrás.

```
is_true(P) :- fact(P). is_true(P) :- if Condition then P, is_true(Condition).
```

```
is_true(P1 and P2) :- is_true(P1), is_true(P2). is_true(P1 or P2) :- is_true(P1);
```

```
is_true(P2).
```

3.3.2 PROGRAMACION DE LAS MACROS EN VISUAL BASIC FOR APPLICATIONS UTILIZANDO MICROSOFT EXCEL

El siguiente código carga el intérprete del PROLOG y crea la interfase entre este y excel.

```
Attribute VB_Name = "Interprete"  
Dim xlSheet As Worksheet  
  
Dim s As String, Term As Long  
Dim tf As Boolean  
  
Public Function Inicializar()  
  
Call partir  
Call InitLS("")  
Call LoadLS("C:\MATLAB7\work\cracking.xpl")  
tf = CallStrLS(Term, "hello($Cracking$, X).")  
Call GetArgLS(Term, 2, bSTR, s)  
rt = MsgBox(s, vbOKOnly)  
Set xlSheet = Worksheets("variables")
```

End Function

Public Function Cerrar()

Call CloseLS

End Function

Se configura la ubicación de la hoja de Excel con la que trabajamos y definimos el nombre de las hojas y variables con las que se trabajaron. Las siguientes líneas de código se encargan de discretizar los valores de las variables, darle un grado de verdad y actualizar la base de hechos del PROLOG, luego interroga al intérprete buscando hallar el antecedente o causa de la evidencia insertada en la base de hechos.

VERSION 1.0 CLASS

BEGIN

MultiUse = -1 'True

END

Attribute VB_Name = "Hoja2"

Attribute VB_GlobalNameSpace = False

Attribute VB_Creatable = False

Attribute VB_PredeclaredId = True

Attribute VB_Exposed = True

Dim xlSheet As Worksheet

Dim timer_enabled As Boolean

Dim timer_interval As Double

Dim interval As Double

Dim s As String, Term As Long

Dim tf As Boolean

Private Sub circ_errat_cataliza_Click()

'CIRCULACIÓN ERRÁTICA DE CATALIZADOR

Call Inicializar

Set xlSheet = Worksheets("variables")

fcti518v = xlSheet.Range("fcti518").Value

ndespojv = xlSheet.Range("ndespoj").Value

'ACTUALIZA BASE DE HECHOS

tf = CallStrLS(Term, "assert(fact(circulacion_erratica_de_catalizador)).")

If fcti518v >= 990 Or fcti518v < 935 Then

tf = CallStrLS(Term,

"assert(fact(temperatura_desigual_del_reactor_y_o_desigual_control_nivel_de_ext
raccion)).")

Else

If ndespojv > 48 Or ndespojv < 46 Then

```
tf = CallStrLS(Term,
"assert(fact(temperatura_desigual_del_reactor_y_o_desigual_control_nivel_de_ext
raccion)).")
Else
tf = CallStrLS(Term,
"retract(fact(temperatura_desigual_del_reactor_y_o_desigual_control_nivel_de_ex
traccion)).")
End If
End If
MSG = MsgBox("¿NOTA VIBRACIONES SEVERAS Y" & Chr(13) &
"MOVIMIENTOS DE LAS TUBERÍAS?", vbQuestion + vbYesNo)
If MSG = vbYes Then
tf = CallStrLS(Term,
"assert(fact(vibraciones_severas_y_movimiento_de_las_tuberias)).")
Else
tf = CallStrLS(Term,
"retract(fact(vibraciones_severas_y_movimiento_de_las_tuberias)).")
End If
MSG = MsgBox("¿NOTA UN RUIDO DE RESOPLIDOS" & Chr(13) & "SIMILAR AL
SONID DE UN TREN?", vbQuestion + vbYesNo)
If MSG = vbYes Then
```

```
tf = CallStrLS(Term,  
"assert(fact(un_ruido_de_resoplido_similar_al_sonido_de_un_tren)).")  
Else  
tf = CallStrLS(Term,  
"retract(fact(un_ruido_de_resoplido_similar_al_sonido_de_un_tren)).")  
End If
```

'INTERROGA AL INTERPRETE

```
tf = CallStrLS(Term, "is_true(aireacion_inapropiada).")  
If tf Then  
MSG = MsgBox("SI EL VAPOR ES USADO COMO MEDIO DE AIREACIÓN" &  
Chr(13) & "ASEGURARSE QUE NO ESTÉ HUMEDO", vbCritical + vbOKOnly)  
Else  
tf = CallStrLS(Term, "is_true(catalisis_esta_en_curso).")  
If tf Then  
MSG = MsgBox("CONSIDERAR OTRO CATALIZADOR CON" & Chr(13) & "UNA  
DISTRIBUCIÓN DE TAMAÑO DE PARTÍCULA DIFERENTE", vbCritical +  
vbOKOnly)  
Else  
tf = CallStrLS(Term, "is_true(baja_presion_del_regenerador).")  
If tf Then  
MSG = MsgBox("INCREMENTAR PRESIÓN DEL REGENERADOR", vbCritical +  
vbOKOnly)  
Else
```

```
MSG = MsgBox("NO HAY FALLOS RELACIONDAOS CON" & Chr(13) &  
"CIRCULACIÓN ERRÁTICA DEL CATALIZADOR", vbOKOnly + vbInformation)
```

```
End If
```

```
End If
```

```
End If
```

```
tf = CallStrLS(Term, "retract(fact(circulacion_erratica_de_catalizador)).")
```

```
Call Cerrar
```

```
End Sub
```

```
Private Sub Flujo_inverso_Click()
```

'FLUJO INVERSO

```
Call Inicializar
```

```
Set xlSheet = Worksheets("variables")
```

```
fcti507v = xlSheet.Range("fcti507").Value
```

```
fcti518v = xlSheet.Range("fcti518").Value
```

```
sedimentv = xlSheet.Range("sediment").Value
```

'ACTUALIZA BASE DE HECHOS

```
tf = CallStrLS(Term, "assert(fact(flujo_inverso)).")
```

```
MSG = MsgBox("¿VE HUMO NEGRO DESDE EL REGENERADOR?", vbQuestion  
+ vbYesNo)
```

```
If MSG = vbYes Then
```

```
tf = CallStrLS(Term, "assert(fact(humo_negro_desde_el_regenerador)).")
```

Else

tf = CallStrLS(Term, "retract(fact(humo_negro_desde_el_regenerador)).")

End If

If fcti507v < 50 Or fcti507v > 1450 Then

tf = CallStrLS(Term,

"assert(fact(un_subito_incremento_en_la_temperatura_del_regenerador)).")

Else

tf = CallStrLS(Term,

"retract(fact(un_subito_incremento_en_la_temperatura_del_regenerador)).")

End If

If fcti518v >= 990 Or fcti518v < 935 Then

tf = CallStrLS(Term,

"assert(fact(un_subito_incremento_en_la_temperatura_del_reactor)).")

Else

tf = CallStrLS(Term,

"retract(fact(un_subito_incremento_en_la_temperatura_del_reactor)).")

End If

If sedimentv > 0.002 Then

tf = CallStrLS(Term, "assert(fact(arrastre_de_catalizador)).")

Else

tf = CallStrLS(Term, "retract(fact(arrastre_de_catalizador)).")

End If

'INTERROGA AL INTERPRETE

```
tf = CallStrLS(Term, "is_true(perdida_de_aire_del_compresor).")
```

```
If tf Then
```

```
MSG = MsgBox("MANTENER UNA PRESIÓN DIFERENCIAL MÍNIMA" & Chr(13)  
& "DE 2 psi A TRAVÉS DE LA VÁLVULA DE CORREDERA", vbCritical +  
vbOKOnly)
```

```
Else
```

```
tf = CallStrLS(Term, "is_true(agua_en_la_carga_o_alimentacion).")
```

```
If tf Then
```

```
MSG = MsgBox("INSTALAR UN DISEÑO RADIAL DE" & Chr(13) & "BOQUILLAS  
DE ALIMENTACIÓN", vbCritical + vbOKOnly)
```

```
Else
```

```
tf = CallStrLS(Term, "is_true(apertura_excesiva_de_la_valvula_de_corredera).")
```

```
If tf Then
```

```
MSG = MsgBox("INSTALAR ACTUADORES DE ALTA VELOCIDAD" & Chr(13) &  
"EN LAS VÁLVULAS DE CORREDERA", vbCritical + vbOKOnly)
```

```
Else
```

```
tf = CallStrLS(Term,
```

```
"is_true(una_subita_perdida_de_los_niveles_de_presion_del_reactor_o_del_regen  
erador).")
```

```
If tf Then
```

```
MSG = MsgBox("MANTENER UNA PRESIÓN DIFERENCIAL MÍNIMA" & Chr(13)
& "DE 2 psi A TRAVÉS DE LA VÁLVULA DE CORREDERA", vbCritical +
vbOKOnly)
```

```
Else
```

```
MSG = MsgBox("NO HAY FALLOS RELACIONDAOS CON" & Chr(13) &
"CIRCULACIÓN ERRÁTICA DEL CATALIZADOR", vbOKOnly + vbInformation)
```

```
End If
```

```
End If
```

```
End If
```

```
End If
```

```
tf = CallStrLS(Term, "retract(fact(flujo_inverso)).")
```

```
Call Cerrar
```

```
End Sub
```

```
Private Sub Formacion_coke_Click()
```

'FORMACIÓN COKE - ENSUCIAMIENTO

```
Call Inicializar
```

```
Set xlSheet = Worksheets("variables")
```

```
slurryv = xlSheet.Range("slurry").Value
```

```
fffi2v = xlSheet.Range("fffi2").Value
```

```
sedimentv = xlSheet.Range("sediment").Value
```

'ACTUALIZA BASE DE HECHOS

```
tf = CallStrLS(Term, "assert(fact(formacion_de_coke)).")
```

```
MSG = MsgBox("¿HAY CAVITACIÓN DE LAS BOMBAS INFERIORES" & Chr(13)  
& "DE LA COLUMNA PRINCIPAL?", vbQuestion + vbYesNo)
```

```
If MSG = vbYes Then
```

```
tf = CallStrLS(Term,
```

```
"assert(fact(cavitacion_y_o_perdida_en_las_bombas_inferiores_de_la_columna_p  
rincipal)).")
```

```
Else
```

```
tf = CallStrLS(Term,
```

```
"retract(fact(cavitacion_y_o_perdida_en_las_bombas_inferiores_de_la_columna_p  
rincipal)).")
```

```
End If
```

```
If slurryv < 42000 Or slurryv > 44000 Then
```

```
tf = CallStrLS(Term,
```

```
"assert(fact(ensuciamiento_y_perdida_de_la_transferencia_de_calor_en_los_inter  
cambiadores_inferiores)).")
```

```
Else
```

```
If fffi2v < 25000 Then
```

```
tf = CallStrLS(Term,
```

```
"assert(fact(ensuciamiento_y_perdida_de_la_transferencia_de_calor_en_los_inter  
cambiadores_inferiores)).")
```

Else

tf = CallStrLS(Term,

"retract(fact(ensuciamiento_y_perdida_de_la_transferencia_de_calor_en_los_inter
cambiadores_inferiores)).")

End If

End If

If sedimentv > 0.002 Then

tf = CallStrLS(Term,

"assert(fact(excesivo_arrastre_del_catalizador_sobre_la_coumna_principal)).")

Else

tf = CallStrLS(Term,

"retract(fact(excesivo_arrastre_del_catalizador_sobre_la_coumna_principal)).")

End If

'INTERROGA AL INTERPRETE

tf = CallStrLS(Term, "is_true(cambios_en_las_condiciones_de_operacion).")

If tf Then

MSG = MsgBox("MEJORAR EL MEZCLADO DEL CATALIZADOR-CARGA" &
Chr(13) & "Y EL SISTEMA DE INYECCIÓN ", vbCritical + vbOKOnly)

Else

tf = CallStrLS(Term, "is_true(cambios_en_las_propiedades_del_catalizador).")

If tf Then

```
MSG = MsgBox("INTRODUCIR LA CARGA SOLO CUANDO EL REACTOR" &  
Chr(13) & "ESTÉ ADECUADAMENTE CALENTADO", vbCritical + vbOKOnly)
```

```
Else
```

```
tf = CallStrLS(Term, "is_true(cambios_en_las_condiciones_mecanicas).")
```

```
If tf Then
```

```
MSG = MsgBox("MANTENER LA TEMPERATURA DE LA" & Chr(13) &  
"COLUMNA PRINCIPAL MENOR A 700 F", vbCritical + vbOKOnly)
```

```
Else
```

```
MSG = MsgBox("NO HAY FALLOS RELACIONDAOS CON" & Chr(13) &  
"FORMACIÓN DE COKE O ENSUCIAMIENTO", vbOKOnly + vbInformation)
```

```
End If
```

```
End If
```

```
End If
```

```
tf = CallStrLS(Term, "retract(fact(formacion_de_coke)).")
```

```
Call Cerrar
```

```
End Sub
```

```
Private Sub Perdidas_catalizador_Click()
```

```
'PERDIDAS DE CATALIZADOR
```

```
Call Inicializar
```

```
Set xlSheet = Worksheets("variables")
```

```
micronesv = xlSheet.Range("micrones").Value
```

perd_catv = xlSheet.Range("perd_cat").Value

sedimentv = xlSheet.Range("sediment").Value

'ACTUALIZA BASE DE HECHOS

tf = CallStrLS(Term, "assert(fact(perdidas_de_catalizador)).")

If sedimentv > 0.002 Then

tf = CallStrLS(Term, "assert(fact(incremento_en_los_sedimentos)).")

Else

tf = CallStrLS(Term, "retract(fact(incremento_en_los_sedimentos)).")

End If

If micronesv < 0.03 Or micronesv > 0.05 Then

tf = CallStrLS(Term,

"assert(fact(disminucion_de_fraccion_de_040_micrones_de_catalizador)).")

Else

tf = CallStrLS(Term,

"retract(fact(disminucion_de_fraccion_de_040_micrones_de_catalizador)).")

End If

If perd_catv > 2 Then

tf = CallStrLS(Term,

"assert(fact(perdida_gradual_del_nivel_del_catalizador_en_el_reactor_o_el_regenerador)).")

Else

```
tf = CallStrLS(Term,  
"retract(fact(perdida_gradual_del_nivel_del_catalizador_en_el_reactor_o_el_regen  
erador)).")
```

```
End If
```

'INTERROGA AL INTERPRETE

```
tf = CallStrLS(Term, "is_true(cambios_en_las_propiedades_del_catalizador).")
```

```
If tf Then
```

```
MSG = MsgBox("VERIFICAR LAS PROPIEDADES" & Chr(13) & "DEL  
CATALIZADOR", vbCritical + vbOKOnly)
```

```
Else
```

```
tf = CallStrLS(Term,
```

```
"is_true(confirmar_que_no_existen_orificios_de_restriccion_perdidos_y_niveles_d  
e_catalizador_en_el_reactor).")
```

```
If tf Then
```

```
MSG = MsgBox("CONFIRMAR QUE NO EXISTEN ORIFICIOS DE  
RESTRICCION" & Chr(13) & " PERDIDOS Y NIVELES DE CATALIZADOR EN EL  
REACTOR", vbCritical + vbOKOnly)
```

```
Else
```

```
tf = CallStrLS(Term, "is_true(cambios_en_las_condiciones_mecanicas).")
```

```
If tf Then
```

```
MSG = MsgBox("VERIFICAR LA PRESICIÓN DE LOS NIVELES DE" & Chr(13) &  
"CATALIZADOR EN EL REGENERADOR Y REACTOR", vbCritical + vbOKOnly)
```

Else

MSG = MsgBox("NO HAY FALLOS RELACIONDAOS CON" & Chr(13) &
"PÉRDIDAS DE CATALIZADOR", vbOKOnly + vbInformation)

End If

End If

End If

tf = CallStrLS(Term, "retract(fact(perdidas_de_catalizador)).")

Call Cerrar

End Sub

Private Sub temp_regenerador_Click()

'ALTA TEMPERATURA DEL REGENERADOR

Call Inicializar

Set xlSheet = Worksheets("variables")

terrorv = xlSheet.Range("terror").Value

'ACTUALIZA BASE DE HECHOS

tf = CallStrLS(Term, "assert(fact(alta_temperatura_del_regenerador)).")

If terrorv >= 0.0325 Then

tf = CallStrLS(Term,

"assert(fact(niveles_altos_de_tierras_raras_en_el_catalizador)).")

Else

```
tf = CallStrLS(Term,  
"retract(fact(niveles_altos_de_tierras_raras_en_el_catalizador)).")  
End If
```

'INTERROGA AL INTERPRETE

```
tf = CallStrLS(Term, "is_true(baja_temperatura_del_precalentador).")  
If tf Then  
MSG = MsgBox("BAJA TEMPERATURA DEL PRECALENTADOR", vbCritical +  
vbOKOnly)  
Else  
MSG = MsgBox("NO HAY FALLOS RELACIONDAOS CON" & Chr(13) & "BAJA  
TEMPERATURA DEL PRECALENTADOR", vbOKOnly + vbInformation)  
End If  
tf = CallStrLS(Term, "retract(fact(alta_temperatura_del_regenerador)).")  
Call Cerrar  
End Sub
```

La siguiente porcion de codigo se encarga de partir el vector con los datos y organizarlos en las filas de excel.

```
Attribute VB_Name = "Parte"  
Sub partir()  
Worksheets("variables").Activate  
Cells(2, 1).Select
```

```
For Each BB In Selection.Cells
BB.Select
BB.Activate
MB = Split(BB, ",")
TAM = UBound(MB)
For NN = 0 To TAM
With ActiveCell
.Offset(0, NN + 1).Value = MB(NN)
End With
Next NN
Next BB
Worksheets("presentacion").Activate
End Sub
```

3.4 ENTORNO GRAFICO

El entorno grafico fue realizado totalmente en Excel, aprovechando las capacidades del mismo.

Figura 13. Pantalla Inicial Sistema Experto.



La etapa anterior del proceso lo que hace es mirar el comportamiento de las variables en el tiempo, la idea general del proyecto realizar una discretización y se le asigne un grado verdad, una vez se haya detectado el fallo se procede a dar una recomendación al operador.

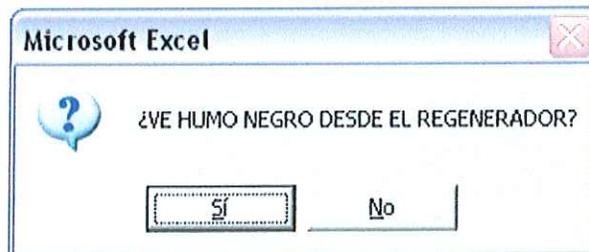
Figura 14. Vector de Variables.

terrar	fcti518	ndespoj	fcti507	sediment	slurry	ffi2	micrones	perd_cat
3%	960	47	1000	0.10%	43000	26000	0.04	1.8
Porcentaje en peso de tierras raras	Temperatura del reactor	Nivel despojadora	Temperatura del Regenerador	BS & W	Flujo de Slurry	Alarma de Flujo de Slurry	Disminucion de Fracciones	Perdida de Catalizador

La figura anterior muestra el vector donde se le asignara un valor específico para un escenario de fallo, este escenario es suministrado por el sistema PAI que es el que monitoriza y almacena la información de las variables en la refinería de Cartagena.

El sistema tambien interroga al operador por medio de la siguiente grafica, donde el operador deberá dar clic para confirmar la respuesta.

Figura 15. Mensaje de Interrogación al Operador.

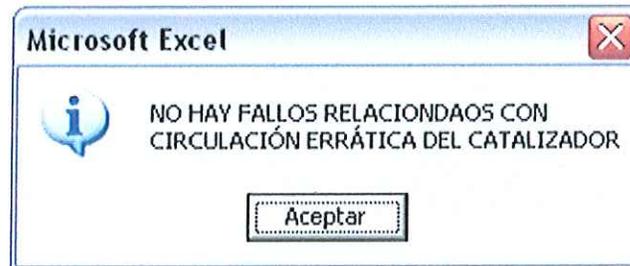


Tambien muestra mensajes informativos para el operador como ya sea que encuentre un fallo o no.

Figura 16. Mensaje Exclamación al Operador.



Figura 17. Mensaje Informativo al Operador.

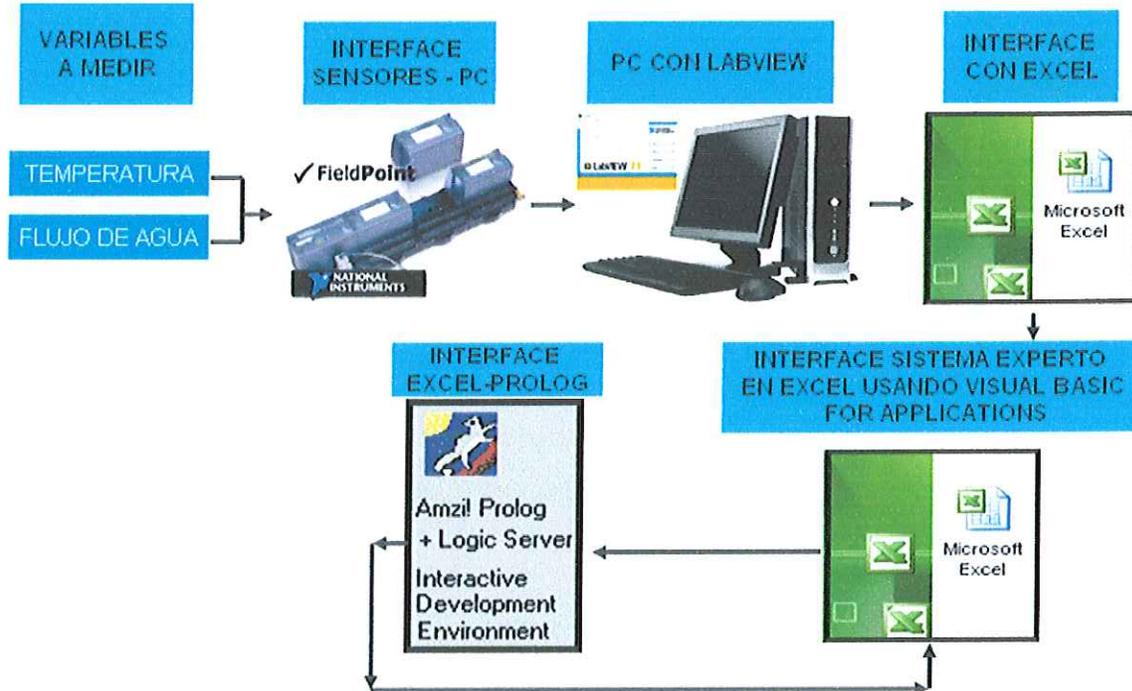


4. SISTEMA ELECTRONICO DE COMUNICACIÓN

El Sistema de comunicación es la componente electrónica que nos permitió conectar los dispositivos de monitoreo como ya sean sensores de temperatura, de flujo, etc, al Sistema Experto programado en Microsoft Excel usando Visual Basic para Aplicaciones.

Por razones de seguridad no se pudo probar el sistema experto en la planta de Cracking Modelo 4 de la refinería de Cartagena, para ello fue necesario recrear un minisistema de monitoreo en la planta de piloto de la universidad utilizando la infraestructura y los equipos que tenemos. En la Figura 17 se ve el orden en que el sistema adquiere la información de los sensores y los envía al Sistema Experto.

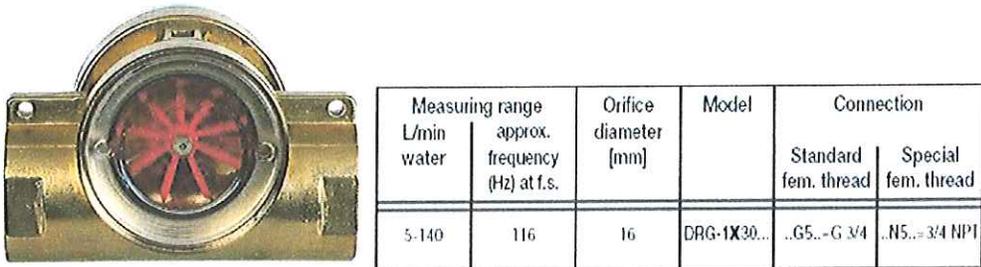
Figura 18. Sistema Electrónico De Comunicación.



VARIABLES

Las variables que utilizamos para probar nuestro sistema experto en la planta piloto fueron Un Termocupla Tipo J y Un medidor de caudal.

Figura 19. Medidor de Caudal.



El medidor de caudal a medida que el flujo avanza a través de ella envía una serie de pulsos, estos deben ser convertidos en caudal utilizando una función matemática, aprovechando que anteriormente ya habíamos caracterizado y hallado la función para este sensor en una de las materias vistas en la universidad, simplemente implementamos el código programado en LabView.

Figura 20. Termocupla tipo j.

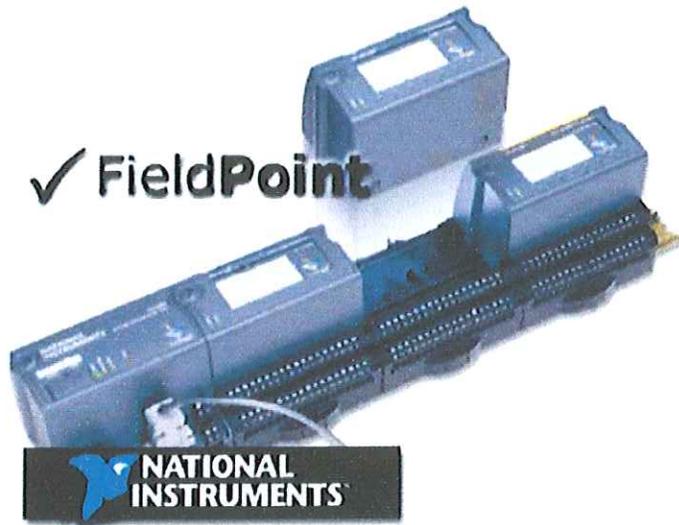


La termocupla Tipo J, conocida como la termocupla hierro – constantán. El hierro es el conductor positivo, mientras que para el conductor negativo se recurre a una aleación de 55 % de cobre y 45 % de níquel (constantán). El rango de Temperatura de la termocupla es de -200 a 900 Grados Celsius.

FIELDPOINT

La conexión de los sensores al computador se hace por medio de un Bus de Campo que en nuestro caso fue un FieldPoint de National Instruments.

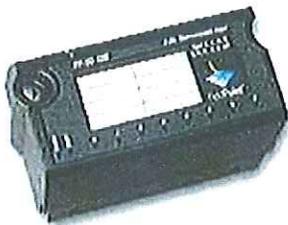
Figura 21. Fieldpoint.



El FieldPoint tiene características de monitoreo especiales, viene por módulos especializados, fue necesario utilizar el modulo FP-TC-120 y el FP-CTR-500.

Figura 22. Modulo FP-TC-120.

8-Channel Thermocouple Module for FieldPoint

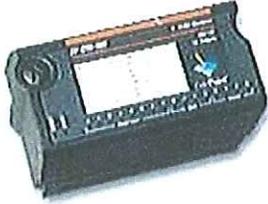


- Measures thermocouple or millivolt signals
- Software-configurable input range or thermocouple type, per channel
- -40 to 70° C operating range
- HotPnP (plug and play) operation
- 50/60 Hz noise rejection
- 8 differential inputs; 16-bit resolution

El modulo FP-TC-120 es el que nos permite medir la temperatura, la termocupla esta conectada directamente a uno de sus canales de entrada, en nuestro caso es el Canal 4.

Figura 23. Modulo FP-CTR-500.

FP-CTR-500

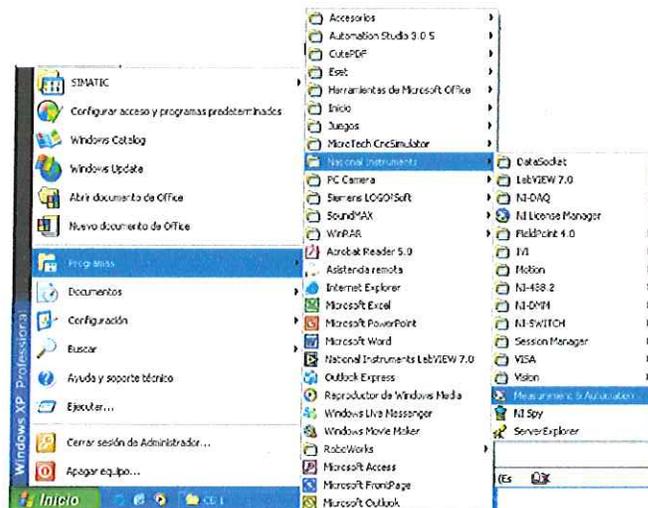


- 8 independent counters, four gate inputs, four counter outputs
- -40 to 70 C operating range
- Input/output ranges
- Isolated to 2,500 Vrms breakdown 250 Vrms working voltage
- HotPnP (plug and play) operation

El modulo FP-CTR-500 es el que nos permite contar los pulsos que nos envía el medidor de caudal, está esta conectado directamente a uno de sus canales de entrada, en nuestro caso es el Canal 1. La comunicación entre el Pc y el FieldPoint se lleva acabo via RS-232.

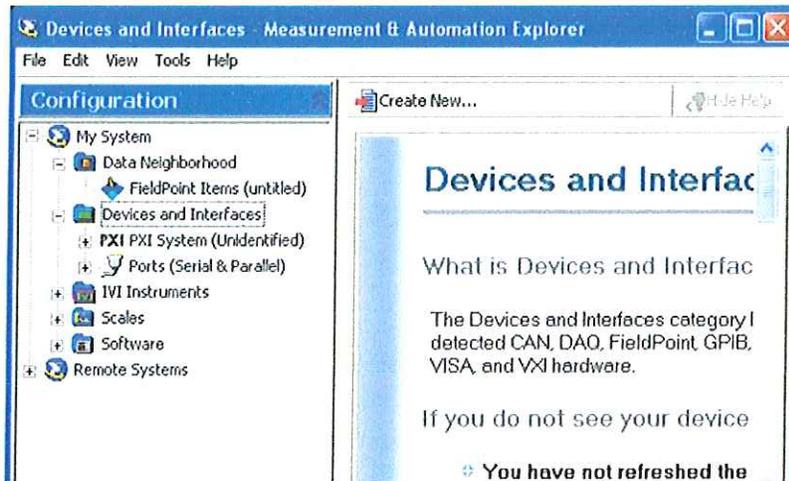
Aunque como la implementación del sistema lo realizamos en el Laboratorio de Automatización fue necesario conectarnos con el FieldPoint via RS-485 aprovechando que anteriormente ya se había realizado esa interfase con el FieldPoint ubicado en la planta piloto. Para poder manejar el Fieldpoint primero es necesario reconocer el Fieldpoint para ello entramos en Measurement & Automation:

Figura 24. Ruta A Measurement & Automation.



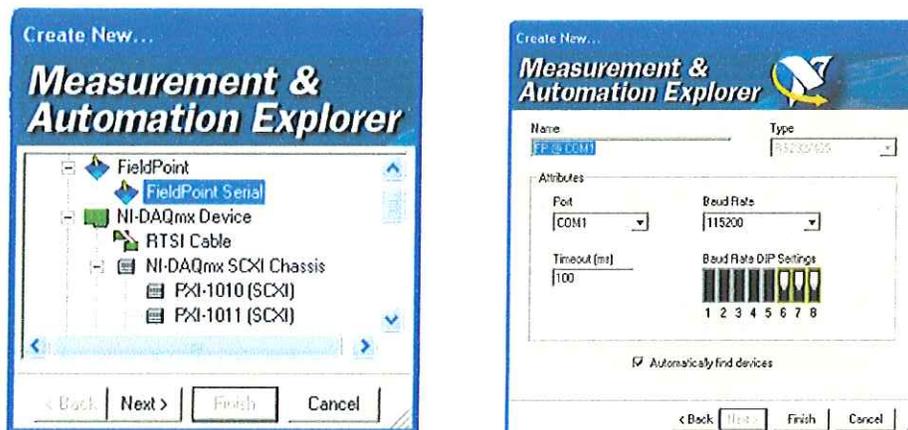
Aparece la ventana siguiente, entonces le damos clic en DEVICES AND INTERFACES y le damos clic en CREATE NEW.

Figura 25. Dispositivos e interfaces.



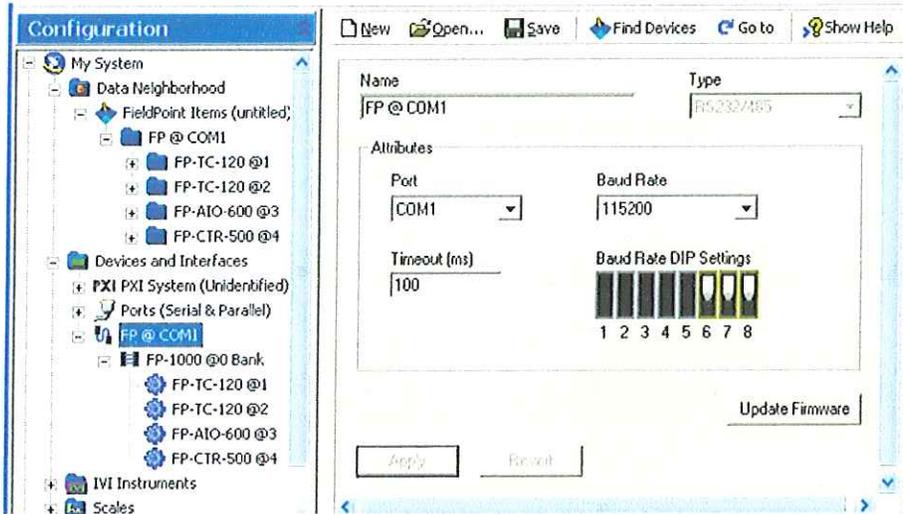
Una nueva ventana emerge y le damos clic en FIELDPOINT SERIAL y después clic en NEXT, la siguiente ventana nos muestra el puerto por donde esta conectado el FieldBus que en este caso es COM1, podemos configurar la velocidad de conexión, pero por defecto esta en 115200, ahora le damos clic en Finish.

Figura 26. Reconocimiento Fieldpoint.



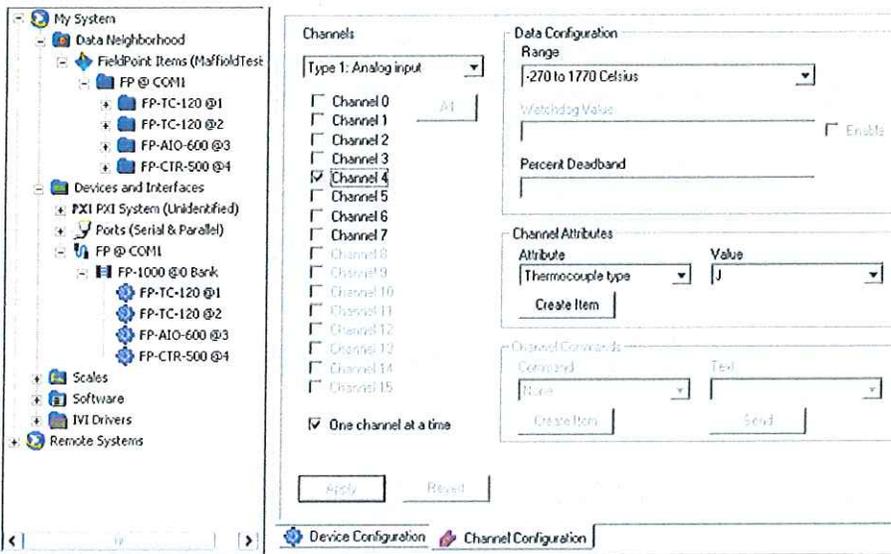
Muestra una nueva ventana donde aparece reconocido el FieldPoint y las expansiones que tienen conectadas. Lo podemos ver donde dice FP @ COM1 en la parte inferior sigue como FP-TC-120@1 y así sucesivamente.

Figura 27. Configuración Fieldpoint.



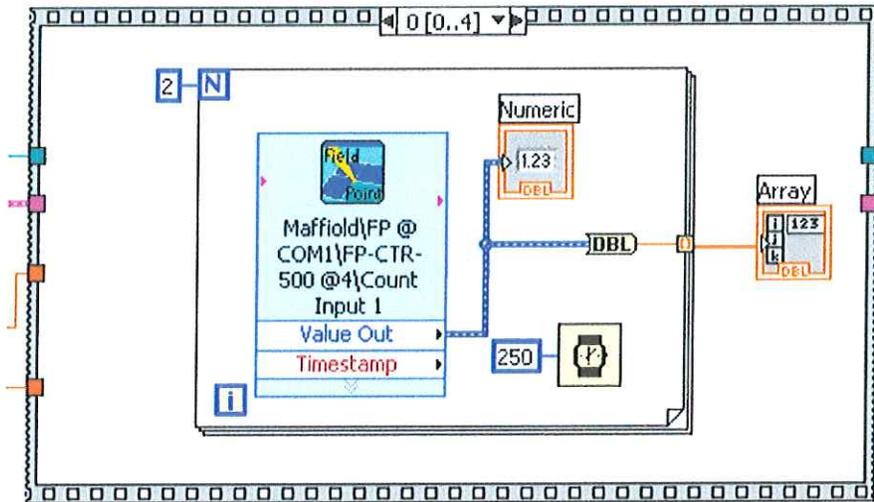
Damos clic en FP @ COM1, escogemos FP-TC-120@1 y después clic en Channel Configuration, en esta pestaña se configura el Rango de Medición y el Tipo de Termocupla.

Figura 28. Configuración Termocupla FP-TC-120.



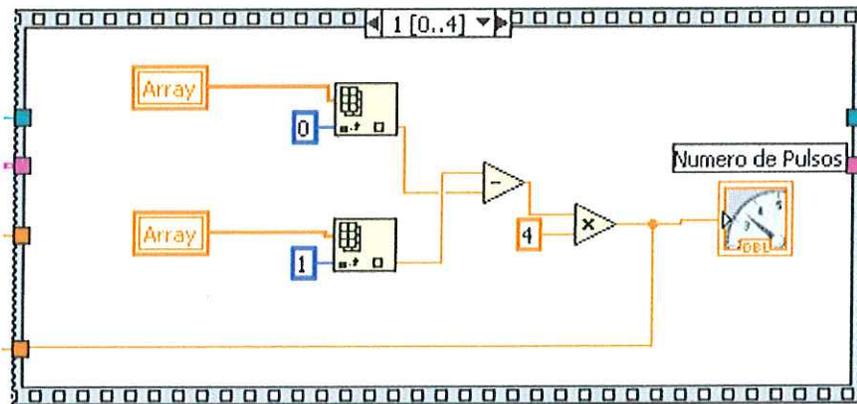
FieldPoint, para esto agregamos el FieldPoint y le agregamos el archivo Maffield.iak que creamos anteriormente, y capturamos el valor cada 250ms, y lo mostramos y a su vez lo almacenamos en un Array. Ver Figura 30.

Figura 30. Rutina Ciclo For-Captura Pulsos Medidor de Caudal.



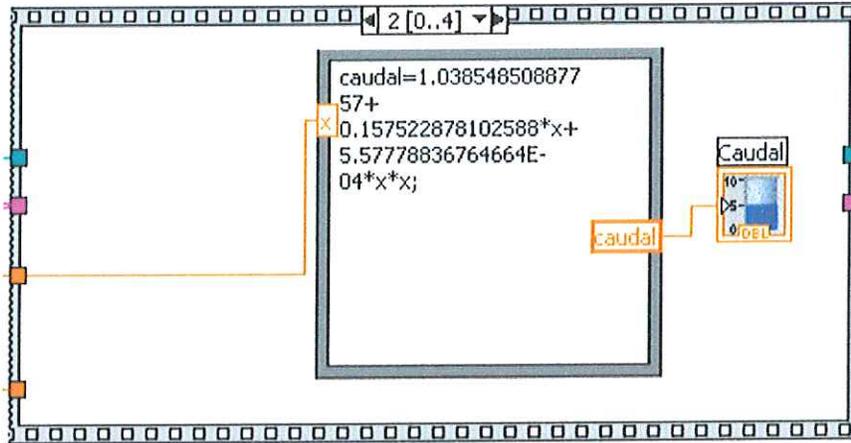
En el Array se almacenan 2 valores con un intervalo de 250 ms, Mayor y menor se muestran en unos indicadores. La diferencia de estos es el numero de pulsos cada 250ms, en el siguiente Frame, lo que hicimos fue hacer la diferencia y multiplicarlo por 4 para así tener el numero de pulsos por segundo. Y lo mostramos en **Numero de Pulsos por Segundo**. Ver Figura 31.

Figura 31. Rutina Número De Pulsos Por Segundo.



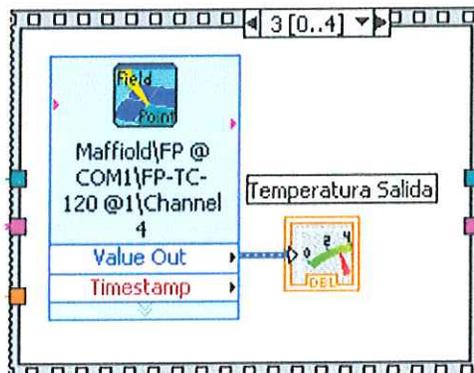
En el Frame que sigue, hacemos un **FORMULA NODE** y colocamos de entrada el Número de pulsos que para nosotros es **X**, y la salida es el caudal, y agregamos la ecuación que hallamos con la regresión no lineal. Y mostramos en un indicador el nivel de caudal actual y lo graficamos. Ver Figura 32.

Figura 32. Rutina Conversión Pulsos A Caudal.



A continuación obtenemos el valor de la temperatura por medio de la Termocupla instalada en el Canal 4 del modulo FP-TC-120. Ver Figura 33.

Figura 33. Temperatura De Salida.



El próximo frame y el mas importante es el que permite la conexión entre Microsoft Excel y LabView, para esto fue necesario instalar la toolbox de Labview que se llama Report Generation For Microsoft. Esta librería permite escribir directamente

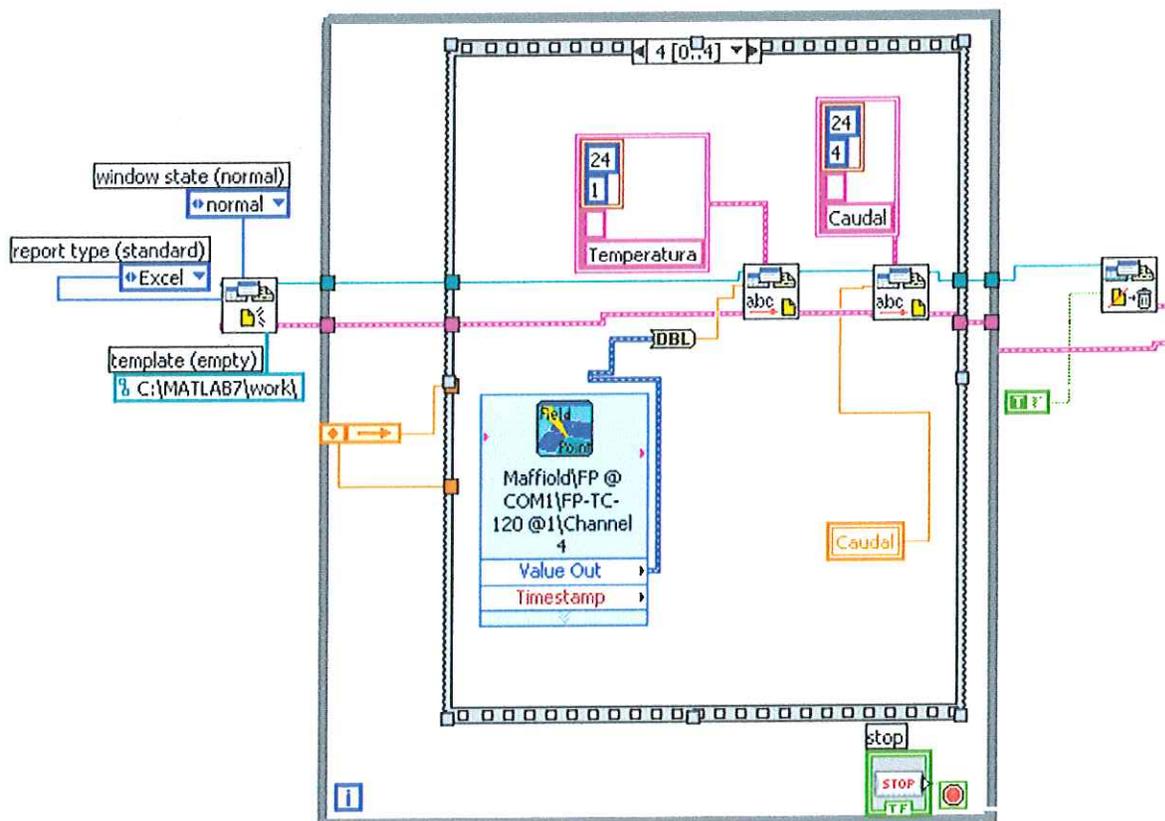
sobre alguna celda específica de Excel. A mano izquierda de la figura 34, vemos

un icono , lo primero que se direcciona es la hoja de Excel donde se va a escribir, en nuestro caso esta en la carpeta C:\MATLAB7\work, Segundo el tipo de reporte para nosotros Excel y finalmente el estado de la ventana al cargar la hoja

de Excel ya sea maximizada, minimizada o normal. El icono , envía el dato a la hoja de Excel, para ello le ponemos nombre a una de las celdas y le damos la coordenada de la misma, El dato de temperatura se obtiene directamente del Fieldpoint y el dato del caudal se obtiene la variable generada en el frame anterior.

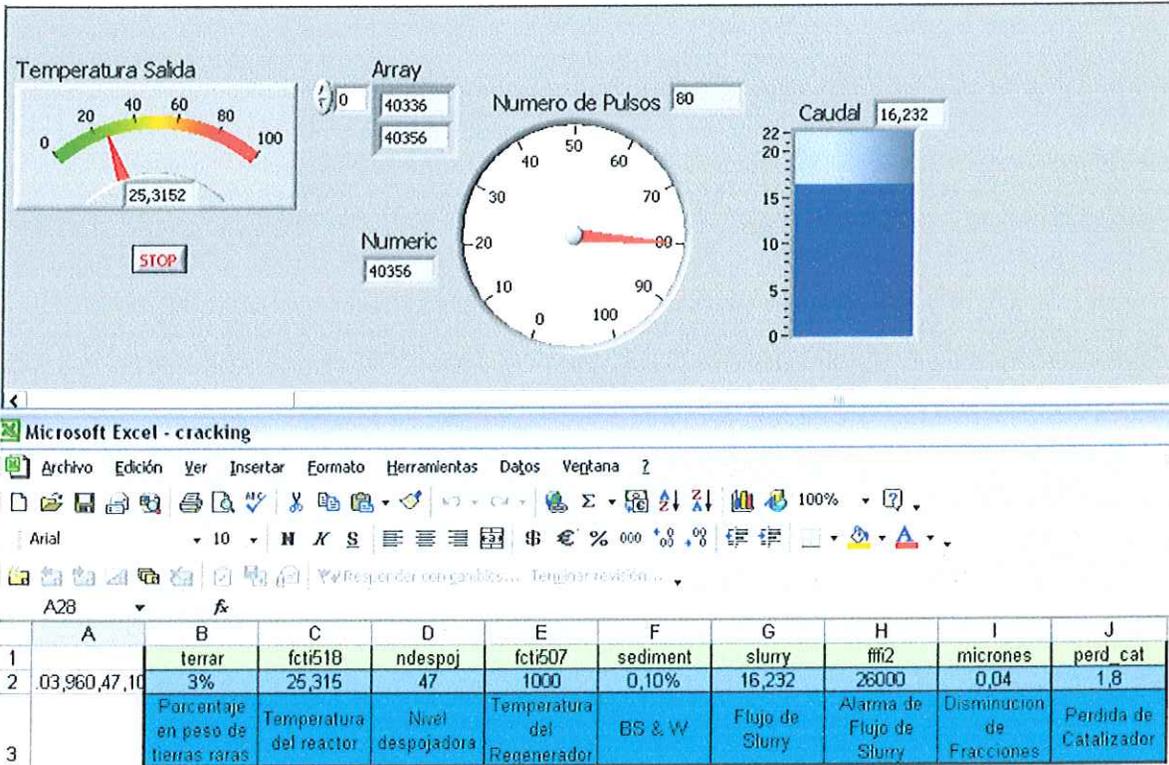
El icono , permite cerrar la hoja de cálculo. Las líneas verdes representan la conexión con la hoja de Excel y la rosada si se llega a presentar algún error.

Figura 34. Rutina Interfase Labview-Excel.



Finalmente en la figura siguiente vemos como el dato obtenido en Labview es mostrado en Microsoft Excel. La Temperatura de Salida en la pantalla superior aparece como Temperatura del Reactor en la Hoja de Excel, y El Caudal en Flujo de Slurry.

Figura 35. Datos Pasados A Excel Desde Labview.

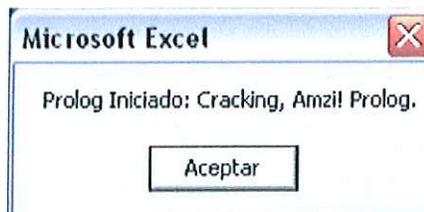


Con esto finalizamos el modulo de comunicaci3n, ya que estamos leyendo los valores del FieldPoint en Excel y ahora si se puede probar el sistema experto ya con los datos que vamos obteniendo de la planta piloto. En el Capitulo de Pruebas se mostrara con mas detalle las pruebas efectuadas ya con el modulo de comunicaci3n funcionando como lo haría la planta de Cracking Catalítico Modelo 4 de Cartagena en Tiempo Real.

5. PRUEBAS Y RESULTADOS

La primera prueba que efectuamos fue la de la apertura del intérprete, pero inicialmente tuvimos un error que fue no copiar todos los archivos en una misma carpeta, pero finalmente logramos abrir obteniendo la pantalla de ejecución del PROLOG.

Figura 36. Mensaje Inicialización PROLOG.



La segunda prueba fue colocar el porcentaje en peso de tierras raras en 4% por encima del límite que es 3.25% . Este valor lo cambiamos en el cuadro de variables. Y después de hacer clic sobre el icono alta temperatura del regenerador.

Figura 37. Botón Alta Temperatura del Regenerador.

**Alta
temperatura
del
regenerador**

Obtuvimos el siguiente mensaje de alerta para el operador.

Figura 38. Mensaje Baja Temperatura del Precaentador.



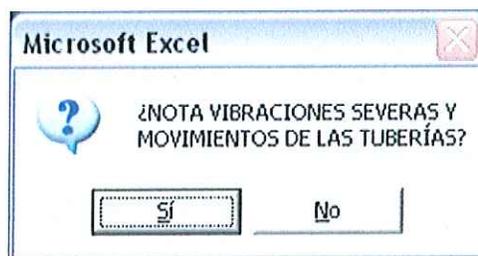
La tercera prueba fue dando clic sobre el icono de circulación errática de catalizador

Figura 39. Botón Circulación Errática del Catalizador.

**Circulación
errática del
catalizador**

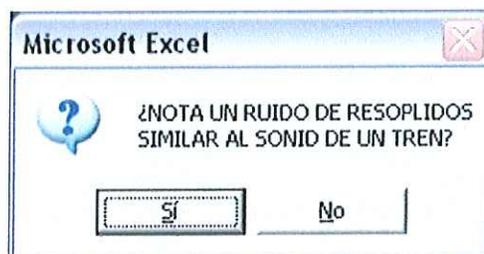
y contestando Si a la pregunta

Figura 40. Mensaje de Pregunta sobre Vibración en la Tubería.



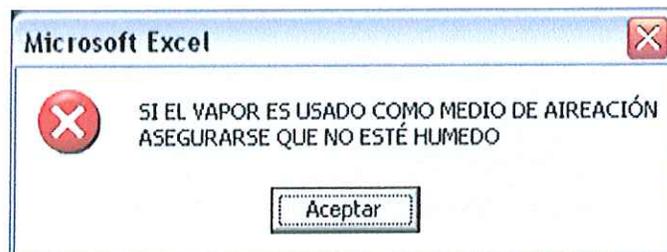
Y contestando No a la siguiente

Figura 41. Mensaje de Pregunta sobre Sonido de Tren.



La recomendación obtenida por el sistema es la siguiente

Figura 42. Mensaje Recomendación Operador



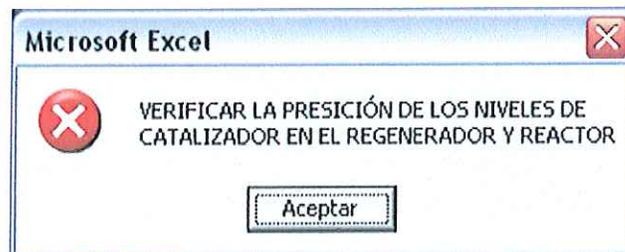
Y la prueba final que realizamos al sistema fue la de cambiar el valor de pérdida típico de catalizador a 2.1 Ton. Al hacer clic sobre el icono de Pérdidas de Catalizador

Figura 43. Botón Pérdidas de Catalizador



Obtenemos el mensaje de advertencia para el operador.

Figura 44. Mensaje Recomendación Operador 2



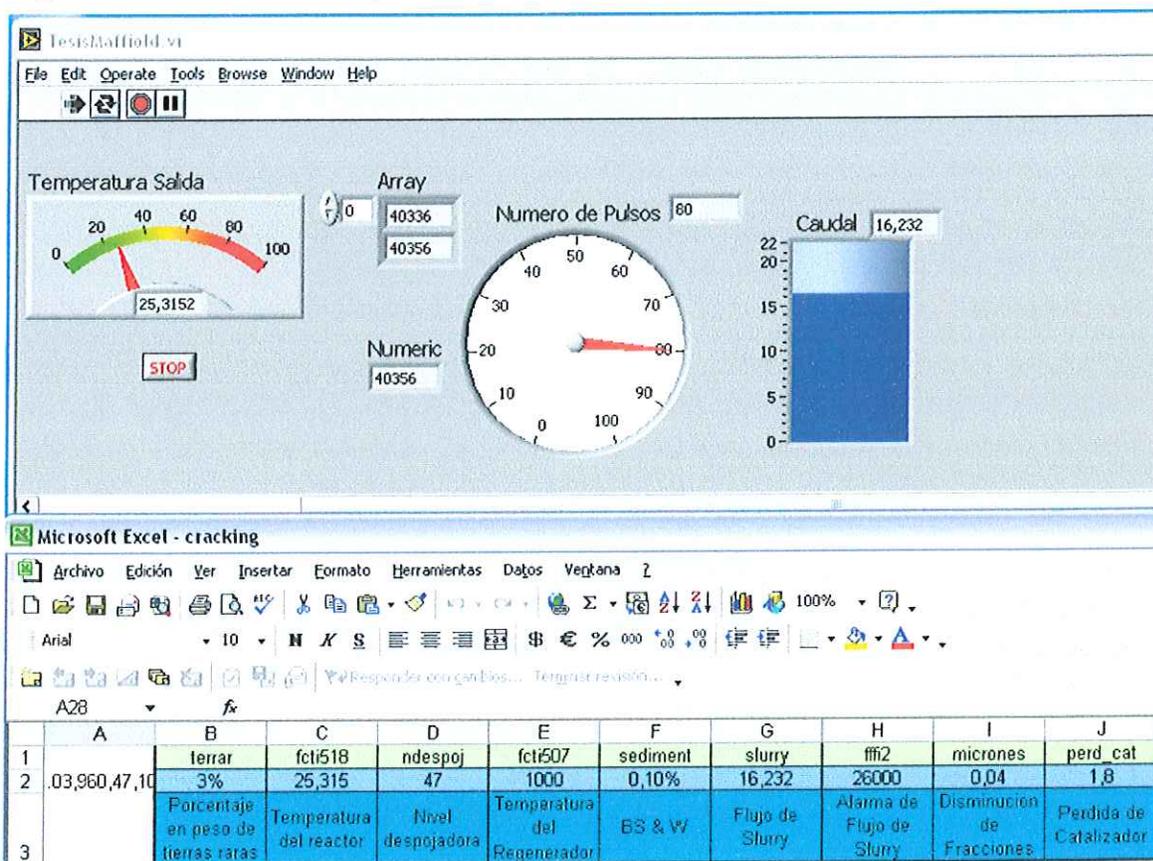
Después de haber realizado las pruebas anteriores al sistema llegamos a la conclusión de que el sistema se está comportando de la manera que habíamos planeado con anterioridad dando como resultado un sistema experto que funciona basado en 2 partes principales: la primera consiste en la verificación de las variables discretas para confirmar o negar la veracidad de un hecho; la segunda parte consta de un motor de inferencia que trabaja por medio de la lógica de predicados y que tiene como objetivo relacionar el consecuente confirmado en una

primera etapa con su respectivo antecedente para poder dar un mensaje de recomendación que facilite la tarea del operario.

Las primeras pruebas que ejecutamos fueron agregando el valor de los datos manualmente en la hoja de Excel, pero en la vida real los datos llegarían directamente de la planta para ello fue que realizamos la interconexión entre LabView y Excel, así obtenemos los datos de la planta piloto en tiempo real y podemos así probar el sistema experto.

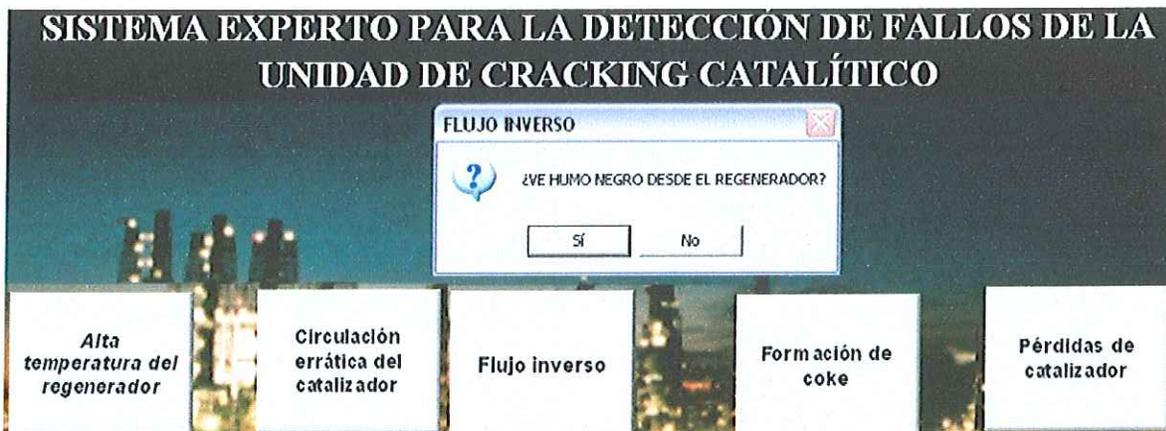
La primera prueba que realizamos era tener condiciones de operación normales para el flujo y la temperatura. Los rangos de valores normales para el flujo son de 16 y 18 Galones, y para la temperatura de 23 a 30 ° Celsius. En la figura 45, vemos la temperatura en 25,3152 y el caudal en 16,232.

Figura 45. Estados de Operación Normales Planta Piloto



A continuación vamos a probar el sistema experto haber si reporta algún error en la planta, para ellos damos clic en FLUJO INVERSO, y después le decimos NO a la pregunta.

Figura 46. Prueba con valor de caudal normal.



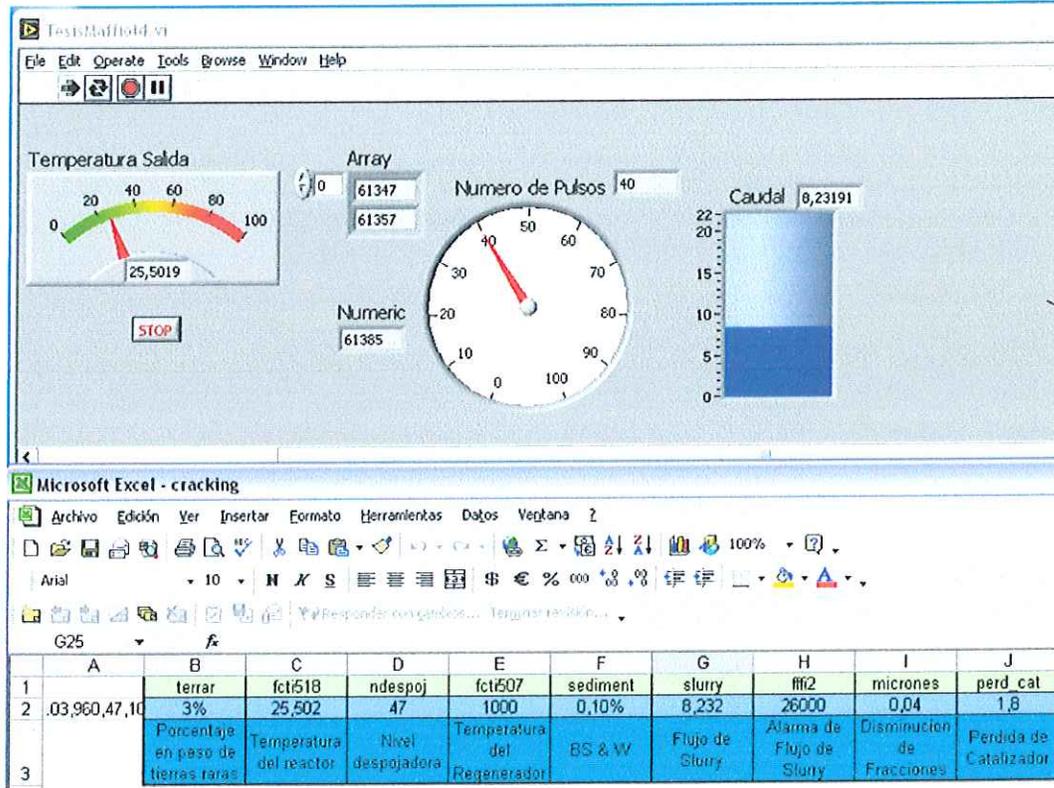
Entonces, el sistema devuelve el siguiente mensaje

Figura 47. Mensaje no hay fallo en el flujo.



A continuación vamos a realizar una caída de flujo simulada cerrando el paso de agua hasta 8 Galones. En la Figura 48 podemos ver el caudal a cuanto ha caído.

Figura 48. Caída de flujo simulada.



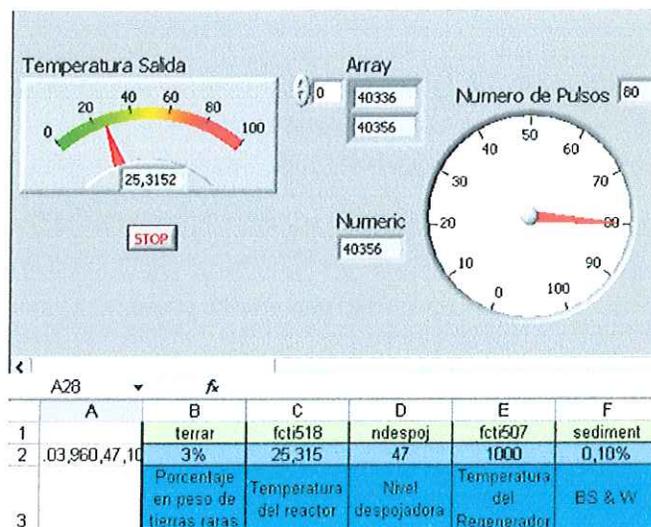
Ahora realizamos el mismo procedimiento anterior pero esta vez el mensaje mostrado por el sistema experto es el siguiente:

Figura 49. Problemas con el flujo.



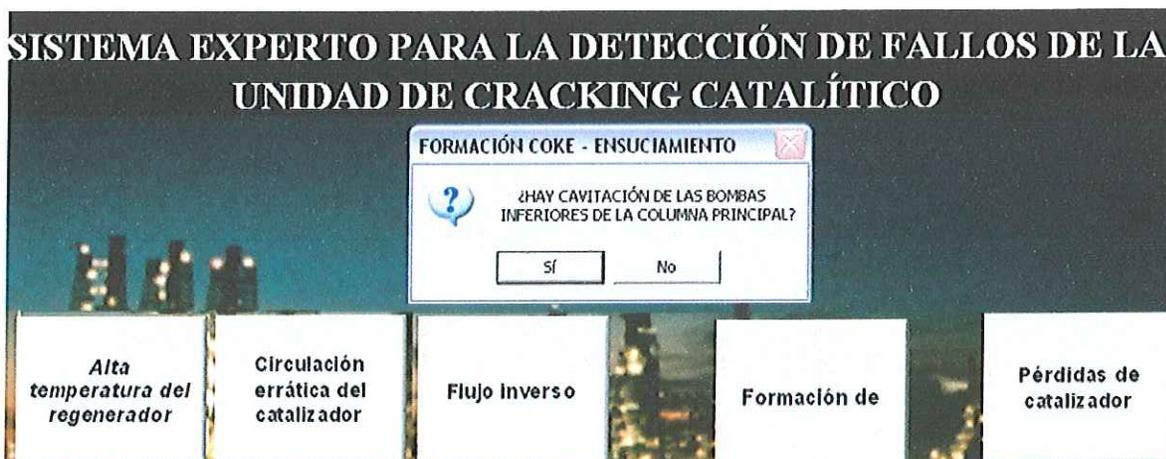
La siguiente prueba fue probando el sistema con temperatura de salida normal como vemos en la Figura 50, La temperatura se encuentra en 25,3152 Grados Celsius.

Figura 50. Temperatura de Salida Normal.



Ahora vamos a ejecutar el sistema experto en condiciones normales de temperatura, damos clic en FORMACION COKE y después al mensaje emergente le damos clic en No.

Figura 51. Prueba con Valor de Temperatura Normal.



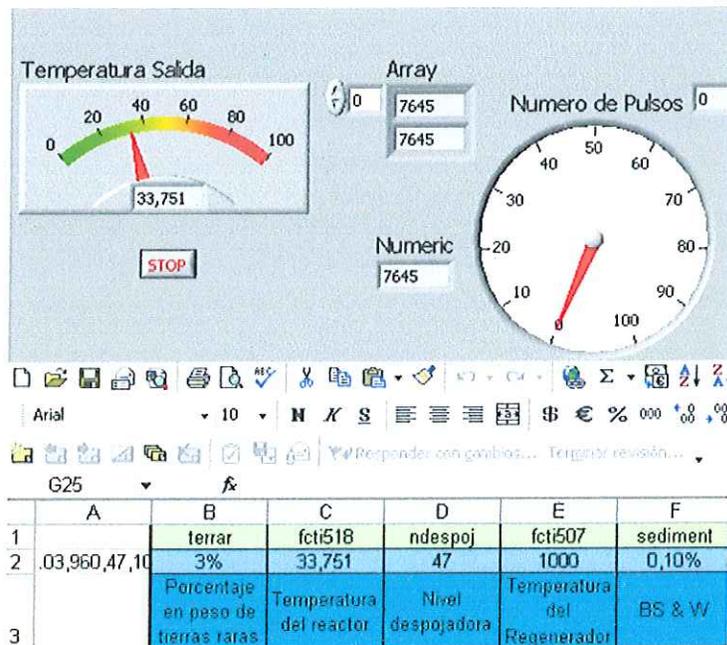
Entonces, el sistema devuelve el siguiente mensaje

Figura 52. Mensaje no hay fallos relaciones con formación de coque.



A continuación vamos a realizar una elevación de la temperatura simulada calentando la termocupla hasta 33 ° Celsius. En la Figura 53 podemos ver cuanto ha aumentado el valor de la temperatura.

Figura 53. Elevación de Temperatura Simulada.



Finalmente realizamos el procedimiento anterior y comprobamos el nuevo mensaje de salida que podemos ver en la Figura 54.

Figura 54. Problemas con la formación de coque.



Al final de esta prueba nos sorprendió la efectividad del sistema incluso con información obtenida en línea como se haría en una planta real. Los mensajes de alerta son generados por los árboles de fallos que anteriormente mostramos.

BIBLIOGRAFÍA

[Sadeghbeigi, 2000] Sadeghbeigi, Reza. *Fluid Catalytic Cracking Handbook. Design, Operation and Troubleshooting of FCC Facilities.* 2000

[Ecopetrol, 2005] Gerencia Refinería de Cartagena. *Manual de Cracking.* 2005

[Ogata K, 1998] Ogata, K. *Ingeniería de Control Moderna Tercera Edición,* Prentice Hall, 1998

[Ramírez J, 2003] Ramirez, J. *Métodos de Inteligencia Artificial en Diagnóstico,* Universidad Politécnica de Valencia, 2003

[Morant F, 2003] Morant, F. *Failure diagnosis of a cement kiln using expert systems,* Universidad Politécnica de Valencia, 2003

OBSERVACIONES, CONCLUSIONES Y RECOMENDACIONES

Se creó una base de conocimiento, en la cual se hizo una clasificación de los fallos más comunes de acuerdo a las partes de la planta en donde estos tienen lugar, para ser utilizada como motor de inferencia en la toma de decisiones, al contar con la experiencia de un operario. Para esto, se realizó una exhaustiva investigación acerca del proceso de cracking catalítico, la cual incluyó sus características de operación normal, sus características en modo de fallos, las variables controladas que posee la planta y sus fallos más comunes. Para la implementación de esta base de conocimiento se recurrió a la lógica de predicados utilizando como plataforma el intérprete de Prolog, el cual no sólo presentaba la ventaja de poseer por defecto un método de búsqueda por encadenamiento hacia atrás, el cual era el planteado en los objetivos de este proyecto, sino que además permite interrogar sobre el grado de verdad de una hipótesis basado en los hechos que se le hayan insertado como ciertos.

Se realizó además, una discretización de las variables de la planta debido a que la herramienta no trabajará en el tiempo, sino previo aviso de una alarma de posible fallo recibida del sistema de control, el cual tiene en cuenta valores continuos. Para esto se tomó cada una de las variables controladas de la planta, se revisó el historial de datos tanto en modo normal como en modo de fallos para así crear

rangos de operación normal para cada una de ellas y determinar los valores máximos y mínimos que pueden tomar. Luego de realizado esto, se creó una macro en Microsoft Excel, programada por medio del Visual Basic for Applications la cual se encarga de controlar en que estado se encuentra cada una de las variables de la planta, además de insertar hechos a la base de Prolog y de interrogar al intérprete sobre el grado de verdad de las hipótesis relacionadas con dichas variables que no se encuentran en estado normal de operación.

Para realizar una prueba física de este proyecto se utilizó la planta piloto, realizando las respectivas variaciones al programa, obteniendo un resultado positivo. Lo mejor que nos deja esta experiencia es la creación de un posible proyecto para los estudiantes que vienen detrás de nosotros, ya que el control por medio de un sistema experto de la planta piloto parece un desafío interesante, más ahora que el control industrial se ha convertido en uno de los campos de acción más importantes de un ingeniero mecatrónico y los sistemas expertos en una de las herramientas con las que cuenta para desempeñar a cabalidad esta labor.