



UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO

FACULTAD DE INGENIERIA

SIMULACION INTEGRAL DE YACIMIENTOS  
DE GAS SECO; SISTEMA "SIMPREG"

**T E S I S**  
QUE PARA OBTENER EL TITULO DE:  
**INGENIERO PETROLERO**  
**P R E S E N T A :**  
**LUIS RODRIGO DIAZ TERAN ORTEGON**

DIRECTOR DE TESIS: M.I. NÉSTOR MARTÍNEZ ROMERO

**TESIS CON  
FALLA DE ORIGEN**





Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

# PAGINACIÓN DISCONTINUA



UNIVERSIDAD NACIONAL  
AVENIDA DE  
MEXICO

FACULTAD DE INGENIERIA  
DIRECCION  
60-1-681

**SR. LUIS RODRIGO DIAZ TERAN ORTEGON**  
Presente

En atención a su solicitud, me es grato hacer de su conocimiento el tema que propuso el profesor M. en I. Néstor Martínez Romero y que aprobó esta Dirección para que lo desarrolle usted como tesis de su examen profesional de Ingeniero Petrolero:

**SIMULACION INTEGRAL DE YACIMIENTOS DE GAS SECO SISTEMA "SIMPREG"**

- I INTRODUCCION
  - II MODELO DEL YACIMIENTO DE GAS, UNA FASE Y DOS DIMENSIONES
  - III MODELO DE FLUJO EN POZOS E INSTALACIONES DE RECOLECCION
  - IV CONCEPTOS FUNDAMENTALES PARA EL ACOPLAMIENTO DE LOS MODELOS NUMERICOS
  - V DESARROLLO DEL SISTEMA DE COMPUTO
  - VI RESULTADOS Y APLICACIONES
  - VII CONCLUSIONES Y RECOMENDACIONES
- NOMENCLATURA  
APENDICES  
REFERENCIAS

Ruego a usted cumplir con la disposición de la Dirección General de la Administración Escolar en el sentido de que se imprima en lugar visible de cada ejemplar de la tesis el título de ésta.

Asimismo, le recuerdo que la Ley de Profesiones estipula que se deberá prestar servicio social durante un tiempo mínimo de seis meses como requisito para sustentar examen profesional

Atentamente  
"POR MI RAZA HABLARA EL ESPIRITU"  
Cd. Universitaria D.F. a 24 de mayo de 2001  
EL DIRECTOR

  
ING. GERARDO FERRANDO BRAVO

GFB\*RLLR\*gtg  
R



**UNIVERSIDAD NACIONAL  
AUTÓNOMA  
DE MÉXICO**



**FACULTAD DE INGENIERÍA**

TESIS: "SIMULACIÓN INTEGRAL DE YACIMIENTOS  
DE GAS SECO; SISTEMA SIMPREG"

PRESENTADA POR: LUIS RODRIGO DÍAZ TERÁN ORTEGÓN

DIRIGIDA POR: M.I. NÉSTOR MARTÍNEZ ROMERO

**JURADO PARA EL EXAMEN PROFESIONAL**

PRESIDENTE: ING. EDUARDO G. LORETO MENDOZA

VOCAL: M.I. NÉSTOR MARTÍNEZ ROMERO

SECRETARIO: DR. FERNANDO SAMANIEGO VERDUSCO

1<sup>ER</sup> SUPLENTE: ING. MANUEL VILLAMAR VIGUERAS

2<sup>O</sup> SUPLENTE: M.I. JOSÉ MARTÍNEZ PÉREZ

**MÉXICO, D.F., CD. UNIVERSITARIA, MAYO DE 2002.**

# AGRADECIMIENTOS

Quiero expresar agradecimientos especiales al director de esta tesis, el M.I. Néstor Martínez Romero por la formación y el inestimable apoyo que me ha proporcionado a lo largo de mi carrera.

También quiero agradecer encarecidamente al Dr. Fernando Samaniego Verusco por su igualmente inestimable ayuda, además de su afabilidad y consideración en todo momento.

Extiendo mis agradecimientos a los ingenieros Manuel Villamar Viguera, José Martínez Pérez y Eduardo G. Loreto Mendoza por haber participado como sinodales de esta tesis y por sus valiosos comentarios y observaciones.

No puedo dejar de reconocer lo mucho que debo a las personas del Programa de Alto Rendimiento Académico por haberme inculcado un elevado ideal de excelencia, el cual me esmeré por alcanzar y lo tendré siempre como brújula durante toda mi vida profesional: En primer término, los Ingenieros Carlos Castillo Tejero y Pablo García y Colomé, coordinadores del programa en diferentes momentos; quienes fueron mis tutores, los Ingenieros Enrique del Valle Toledo, Héctor Luis Macías González y H. Ricardo Castrejón Pineda, así como Lidia Delgado, secretaria del programa.

Este trabajo lo dedico a mi padre Jorge Díaz Terán Capaceta y mis hermanos Jorge, Gabriela y Martha.

## RESUMEN

En el presente trabajo se describe el desarrollo de un simulador de yacimientos de gas seco con pozos e instalaciones de recolección en superficie llevando a cabo un acoplamiento entre ellos con una formulación explícita y condiciones de frontera de Dirichlet entre pozos y yacimiento. Se prestó atención a la interfaz gráfica con el usuario para la entrada y salida de la información y se implementó un algoritmo sencillo para graficar en tercera dimensión tanto al yacimiento como a la red de tuberías y pozos. Utilizando este simulador se investiga el comportamiento de convergencia del método y se ejemplifica cómo puede ser utilizado para resolver problemas de optimización, donde se requiera simular conjuntamente al yacimiento con la red de recolección.

# ÍNDICE

<b>RESUMEN</b>	<b>iv</b>
<b>ÍNDICE</b>	<b>v</b>
<b>ÍNDICE DE FIGURAS</b>	<b>viii</b>
<b>ÍNDICE DE TABLAS</b>	<b>xiv</b>
<b>ÍNDICE DE ALGORITMOS</b>	<b>xvii</b>
<b>I INTRODUCCIÓN</b>	<b>1</b>
<b>II MODELO DEL YACIMIENTO DE GAS SECO, UNA FASE Y DOS DIMENSIONES</b>	<b>6</b>
II.1 Modelo matemático del yacimiento	6
II.2 Modelo numérico del yacimiento	10
II.3 Resolución de las ecuaciones del modelo numérico	20
II.4 Solución de matrices pentadiagonales por el método LU	24
II.5 Representación de pozos en el yacimiento	28
<b>III MODELO DE FLUJO EN POZOS E INSTALACIONES DE RECOLECCIÓN</b>	<b>33</b>
III.1 Modelo del sistema de recolección	33
III.2 Algoritmo para el sistema de ecuaciones algebraicas no lineales del modelo de superficie	40
III.3 Flujo de gas a través de tuberías horizontales o ligeramente inclinadas	43
III.4 Flujo de gas a través de tuberías de producción	45



III.5 Flujo de gas a través de estranguladores _____	50
III.6 Compresión del gas _____	58
<b>IV CONCEPTOS FUNDAMENTALES PARA EL ACOPLAMIENTO DE LOS MODELOS NUMÉRICOS _____</b>	<b>60</b>
IV.1 Descripción del sistema yacimiento-pozos-sistema de recolección _____	50
IV.2 Algoritmo global _____	67
IV.3 Técnica iterativa _____	72
<b>V DESARROLLO DEL SISTEMA DE CÓMPUTO _____</b>	<b>78</b>
V.1 Partes del programa _____	78
V.2 Organización de la información _____	79
V.2.1 Organización de la información en el yacimiento _____	81
V.2.2 Organización de la información en la red superficial _____	83
V.2.3 Organización de la información de los estranguladores _____	85
V.2.4 Organización de la información de las terminaciones _____	86
V.2.5 Organización de la información de los pozos _____	87
V.2.6 Estructura "Programa de operaciones" _____	88
V.2.7 Estructura "Identificador de objetos" _____	90
V.2.8 Organización de la información de las condiciones de frontera en los puntos de entrega _____	92
V.2.9 Organización de la información de los parámetros de la corrida _____	93
V.2.10 Organización de la información de la historia de la corrida _____	95
V.2.11 Organización de la información de las propiedades PVT del gas _____	97
V.3 Interfaz con el usuario _____	102
V.3.1 Ventana principal _____	102
V.3.2 Edición del yacimiento _____	103
V.3.3 Interfaz gráfica de la red superficial _____	108
V.3.4 Edición del programa del desarrollo _____	120
V.3.5 Interfaz gráfica para introducir las condiciones de frontera _____	123
V.3.6 Interfaz de las propiedades PVT del gas _____	125
V.3.7 Interfaz de los parámetros de la corrida _____	130
V.3.8 Interfaz gráfica de la corrida del simulador _____	131
V.3.9 Generador de gráficas _____	134
V.3.10 Ventana de visualización en 3D de la corrida _____	135

V.4 Kernel de procesamiento de la información _____	138
V.4.1 Subrutina de resolución de la red _____	139
V.4.2 Subrutina de resolución del yacimiento _____	143
V.4.3 Subrutina del sistema yacimiento-pozos-sistema de recolección _____	149
V.5 Almacenamiento y recuperación de la información _____	156
<b>VI RESULTADOS Y APLICACIONES _____</b>	<b>165</b>
VI.1 Comportamiento de convergencia del método _____	165
VI.2 Ejemplo de un campo hipotético _____	184
VI.3 Ejemplo de optimización _____	198
<b>VII CONCLUSIONES Y RECOMENDACIONES _____</b>	<b>204</b>
VII.1 Conclusiones _____	204
VII.2 Recomendaciones _____	206
<b>NOMENCLATURA _____</b>	<b>209</b>
<b>BIBLIOGRAFÍA _____</b>	<b>213</b>
APÉNDICE A. VISUALIZACIÓN EN 3D DEL YACIMIENTO Y LA RED SUPERFICIAL _____	217
APÉNDICE B. CORRELACIONES PVT PARA GASES SECOS _____	241
APÉNDICE C. CONTROL AUTOMÁTICO DEL INTERVALO DE TIEMPO _____	255

# ÍNDICE DE FIGURAS

2.1.- Convención para la dirección y sentido de $i$ y $j$ , así como de E( Este), W (Oeste), N (Norte) y S (Sur) en el sistema cartesiano de la malla del yacimiento	14
2.2.- Los coeficientes de flujo en la dirección de celdas aledañas inexistentes son cero	18
2.3.- Matriz pentadiagonal de un yacimiento rectangular con ordenamiento secuencial por renglones	19
2.4.- matriz $A^{(k-1)}$ después de la operación $(k-1)$ -ésima y matriz $A^{(N-1)}$ después de $N-1$ operaciones; factorización LU	25
2.5.- Un yacimiento irregular causa anchos de banda variables	27
2.6.-Esquema ilustrativo del perfil de presiones dentro de una celda con un pozo	28
3.1.- Red de recolección representada por nodos y tubos	35
3.2.- Experimento numérico con el método de Newton-Raphson para resolver la ecuación de Szilas	55

3.3.- Tiempo de cálculo del método de Newton y de Bisección para la ecuación de Szilas en función de la tolerancia	56
3.4.-Tiempos relativos del método de Bisección con respecto al de Newton Raphson para resolver la ecuación de Szilas	56
4.1.- Esquema del sistema yacimiento-pozos-sistema de recolección	61
4.2.- Flujo de información entre los elementos del sistema yacimiento-pozos-red superficial utilizando condiciones de frontera de Dirichlet en los pozos	66
4.3.-Diagrama de bloques de la iteración global	67
5.1.-Partes del simulador	79
5.2.-Ejemplo de la organización de la información de las redes superficiales	84
5.3.- Índices de los objetos pozos a celdas del yacimiento y a nodos de la red superficial	88
5.4.- Programa de operaciones	89
5.5.- Ejemplo de cómo funciona el identificador de elementos	90
5.6.- Funcionamiento de la estructura de condiciones de frontera	93
5.7.- Ventana principal del simulador	102
5.8.- Malla de diseño del yacimiento. Las celdas de color rojo son celdas del yacimiento, mientras que las blancas no	104
5.9.- Malla de introducción de valores a las celdas del yacimiento. Las celdas blancas pertenecen al yacimiento, mientras que las grises no	104

5.10.- Ventana principal del diseño del yacimiento_____	105
5.11.- Flujo de información para la edición del yacimiento_____	106
5.12.- Simbología de objetos gráficos de la red superficial_____	108
5.13.- Ejemplo de una red de recolección_____	109
5.14.- Botones para la creación y eliminación de nodos y tubos_____	109
5.15.- Cuadro de diálogo de tuberías horizontales_____	117
5.16.- Cuadro de diálogo de edición de los pozos_____	117
5.17.- Modo selección del cuadro de diálogo de edición de los pozos_____	118
5.18.- Cuadro de diálogo de los estranguladores_____	118
5.19.- Cuadro de diálogo de la tubería de producción_____	119
5.20.- Cuadro de diálogo de las terminaciones_____	119
5.21.- Ventana General para nodos, tubos horizontales, tuberías de producción, terminaciones y estranguladores_____	120
5.22.- Introducción de redes y pozos al programa de operaciones_____	121
5.23.- Botones para editar el programa de operaciones_____	122
5.24.- Ventana para introducir las condiciones de frontera_____	124
5.25.- Ventana de Propiedades pseudocríticas y tabulares del gas_____	126
5.26.- Sección de valores tabulares y temperatura de referencia del gas_____	127

5.27.- Sección de propiedades pseudocríticas y densidad relativa y fracciones molares de componentes no hidrocarburos_____	128
5.28.- Sección de ajuste de polinomios a valores tabulares o correlaciones_____	129
5.29.- Ventana para la composición del gas_____	130
5.30.- Ventana para introducir los parámetros de la corrida_____	131
5.31.- Ventana de ejecución de la corrida_____	132
5.32.- Generador de gráficas_____	136
5.33.- Ventana de visualización de la corrida en 3D_____	137
6.1.- Yacimiento e instalaciones para el Experimento 1, 2 y 3_____	161
6.2.- Sucesión de gastos calculados para $J=0.02\{[(PCS/d)/(\psi^2/cp)]\}$ . Experimento 1_____	168
6.3.- Sucesión de gastos calculados para $J=0.05\{[(PCS/d)/(\psi^2/cp)]\}$ . Experimento 1_____	169
6.4.- Sucesión de gastos calculados para $J=0.1\{[(PCS/d)/(\psi^2/cp)]\}$ . Experimento 1_____	169
6.5.- Sucesión de gastos calculados para $P=3000[\text{psi}]$ . Experimento 2_____	171
6.6.- Sucesión de gastos calculados para $P=8000[\text{psi}]$ . Experimento 2_____	172
6.7.- Sucesión de gastos calculados para $P=15,000[\text{psi}]$ . Experimento 2_____	172
6.8.- Sucesión de gastos calculados sin estrangulador. Experimento 3_____	174
6.10.- Sucesión de gastos calculados para $D=64/64[\text{pg}]$ . Experimento 3_____	175
6.11.- Sucesión de gastos calculados para $D=32/64[\text{pg}]$ . Experimento 3_____	175

6.12.- Sucesión de gastos calculados para $D=20/64$ [pg]. Experimento 3	175
6.13.- Sucesión de gastos calculados para $D=16/64$ [pg]. Experimento 3	176
6.14.- Sucesión de gastos calculados para $D=15/64$ [pg]. Experimento 3	176
6.15.- Sucesión de gastos calculados para $D=14.4/64$ [pg]. Experimento 3	177
6.16.- Sucesión de gastos calculados para $D=13.6/64$ [pg]. Experimento 3	177
6.17.- Sucesión de gastos calculados para $D=13.6/64$ [pg], $\text{err}=5\%$ . Experimento 3	180
6.18.- Sucesión de gastos calculados para $D=20/64$ [pg]. Experimento 4	180
6.19.- Sucesión de gastos calculados para $D=16/64$ [pg]. Experimento 4	181
6.20.- Sucesión de gastos calculados para $D=15/64$ [pg]. Experimento 4	181
6.21.- Sucesión de gastos calculados para $D=14.4/64$ [pg]. Experimento 4	182
6.22.- Sucesión de gastos calculados para $D=13.6/64$ [pg]. Experimento 4	184
6.23.- Distribución de permeabilidades en X	185
6.24.- Distribución de permeabilidades en Y	185
6.25.- Distribución de porosidades	185
6.26.- Estado de la red 1/8	186
6.27.- Estado de la red 2/8	187
6.28.- Estado de la red 3/8	187

6.29.- Estado de la red 4/8	188
6.30.- Estado de la red 5/8	188
6.31.- Estado de la red 6/8	189
6.32.- Estado de la red 8/8	189
6.33.- Red (8/8) en vista areal	190
6.34.- Presión en el punto de entrega A	190
6.35.- Propiedades de los pozos	190
6.36.- Propiedades de los estranguladores	191
6.37.- Propiedades de las terminaciones	191
6.38.- Propiedades de los nodos	191
6.39.- Propiedades de las tuberías horizontales	192
6.40 Tiempos a los cuales deja de funcionar cada red	193
6.41.- Distribución de presiones para $t=200, 400, 600, 800, 1000$ y $1200[d]$	194
6.42.- Distribución de presiones para $t=1400, 1600, 1800, 2000, 2200$ y $2400[d]$	195
6.43.- Distribución de presiones para $t=2600, 2800, 3000, 3200, 3400$ y $3600[d]$	196
6.44.- Gasto total contra tiempo	196
6.45.- Presión media $[lb/pg^2]$	197
6.46.- Reserva contra tiempo	199



6.47.- Resultados de la primera serie de corridas	202
6.48.- Resultados de la segunda serie de corridas	202
A.1.- Observador de objetos en 3D	218
A.2.- Proyección de un punto en el plano de visualización	219
A.3.- Coordenadas X y Y del punto de intersección, según las coordenadas relativas del plano de visualización	220
A.4.-Escala de Colores	230
A.5.- Ejemplo de visualización de un yacimiento en 2D con el simulador SIMPREG 1.1	231
A.6.- Ejemplo de visualización de un yacimiento en 2D con el simulador SIMPREG 1.1	231
C.1.- Ejemplo del algoritmo C.1; puntos calculados	257
C.2.- Ejemplo del algoritmo C.1; incrementos en X y Y	258

# ÍNDICE DE TABLAS

2.1.- Datos que necesita el algoritmo 2.1 (iteración en el yacimiento)	21
2.2.- Incógnitas que calcula el algoritmo 2.1 (iteración en el yacimiento)	22
3.1.- Datos que necesita el algoritmo 3.1 (iteración en la red superficial)	40
3.2.- Incógnitas que calcula el algoritmo 3.1 (iteración en la red superficial)	41
4.1.- Datos que necesita el algoritmo 4.1 (propiedades de los elementos del sistema yacimiento-pozos-red superficial)	69
4.2.- Datos que necesita el algoritmo 4.1 (variables de la iteración pasada)	70
4.3.- Incógnitas que calcula el algoritmo 4.1 (iteración global)	70
5.1.- Compuestos que forman la mezcla gaseosa	100
5.2.- Tipos de mensajes de la interfaz de ejecución del simulador	133

5.3.- Tipos de archivos y extensiones de la entrada de datos	157
6.1.- Propiedades del yacimiento. Experimento 1	166
6.2.- Propiedades de la red. Experimento 1	166
6.3.- Variables del experimento 1	168
6.4.- Variables del experimento 2	171
6.5.- Variables del experimento 3	173
6.6.- Valores de presión para la primera serie de corridas	201
6.7.- Valores de presión para la segunda serie de corridas	201

# ÍNDICE DE ALGORITMOS

2.1.- Algoritmo para encontrar la distribución de presiones en el yacimiento y gastos en los pozos_____	22
3.1.- Algoritmo para encontrar las incógnitas en la red de recolección superficial_____	41
4.1.- Algoritmo de ajuste de gastos en los pozos calculados de el yacimiento y en la red superficial_____	71
4.2.- Método adaptativo para acelerar la convergencia de 4.1_____	76
5.1.- Añadir nodo en la red superficial_____	109
5.2.- Añadir tubo horizontal a la red superficial_____	111
5.3.- Creación de tuberías de producción, estranguladores y terminaciones en la red superficial_____	112
5.4.- Eliminación de Conexiones de un nodo a otro nodo_____	113

5.5.- Eliminación de tubos horizontales en la red	114
5.6.- Eliminación de tuberías de producción, estranguladores y terminaciones de la red	115
5.7.- Eliminación de nodos de la red	115
5.8.- Obtención del identificador de objetos en base al programa de operaciones	122
5.9.- Obtención de los puntos de entrega con base en el programa de operaciones y en el identificador de objetos	125
A.1.- Algoritmo de ordenamiento por distancias para resolver el problema de la visibilidad	223
C.1.- Algoritmo para control automático del incremento de tiempo	256

# CAPÍTULO I

## INTRODUCCIÓN

La administración de yacimientos petroleros tiene por objeto obtener el valor máximo económico de los hidrocarburos del subsuelo. Planear su explotación eficiente requiere de un conocimiento amplio de las características del yacimiento, tanto estáticas como dinámicas. Además son necesarios modelos matemáticos que describan los fenómenos de transporte tanto en los yacimientos como de los sistemas de extracción, recolección y proceso en superficie, para predecir el comportamiento de los primeros bajo diferentes esquemas de explotación y diseñar adecuadamente los segundos. Estos modelos generalmente se expresan en términos de ecuaciones diferenciales no lineales cuya solución, salvo contadas excepciones, sólo es posible obtenerlas mediante simuladores computacionales.

La simulación computacional de yacimientos es la ciencia y el proceso de resolución de los modelos numéricos por medio de programas de cómputo. El concepto de simulación computacional o numérica de yacimientos, que en ocasiones han llegado a ser sinónimos, surgió a principios de la década de los 60's, con la aparición de las primeras computadoras lo suficientemente potentes como para realizar eficientemente los cálculos requeridos y desde entonces, no ha dejado de evolucionar y desarrollarse a la par que se ha desarrollado la informática y que los problemas en la industria se han vuelto más complejos.

El desarrollo de los paquetes de simulación de yacimientos y de instalaciones de transporte y proceso está hoy día a cargo de institutos de investigación, universidades, departamentos de investigación y desarrollo de compañías operadoras, pero además y

sobretudo por las grandes compañías de servicios, que lo han hecho una de sus líneas de negocios, absorbiendo a las pequeñas compañías que fueron pioneras en el ramo.

Para el caso de México es muy importante tener profesionistas expertos en simulación, tanto para hacer un uso adecuado e inteligente de los productos ofrecidos por las grandes compañías, como para desarrollar productos a la medida de Petróleos Mexicanos y tratar de hacer menos dependiente al país de tecnología extranjera, que cuesta millones de dólares al año.

A este gran fin la presente tesis proporciona una modesta aportación tratando un tema viejo y sencillo de por sí (producción de gas seco), pero que será fructífero en experiencias y al final dará buena cuenta de cuanto se hubiere aprendido en ella.

### **¿Por qué simular conjuntamente al yacimiento y la red de recolección?**

Los estudios de simulación de yacimientos permiten identificar estrategias de desarrollo que puedan maximizar la recuperación económica de los hidrocarburos. Durante la vida de un campo, se tienen que confrontar numerosas decisiones, como son la localización y operación de pozos productores o inyectores, tiempos en que se perforen etc. Con la simulación se proponen varios escenarios para evaluar los costos y beneficios asociados a diferentes estrategias de desarrollo que compiten entre sí. Sin embargo, la sola modelación del yacimiento sin instalaciones no considera el efecto que ellas puedan tener en la rentabilidad de los programas de recuperación. Los costos de compresión, separación, inyección de fluidos y tratamiento de agua, pueden tener un efecto significativo en el éxito de la estrategia de explotación.

La modelación de las instalaciones superficiales es un componente importante en la planeación del desarrollo de los campos. A partir del modelo se obtiene información para el diseño de equipo de procesamiento de los fluidos producidos, separadores, compresores, bombas y controladores de flujo, entre otros. El modelo también asiste en el diseño de redes de tuberías para la recolección de los fluidos y su conducción a las unidades de procesamiento, o bien hacia pozos de reinyección. La simulación integral del yacimiento con las instalaciones da un enfoque más amplio a la planeación del desarrollo de campos.

## Breve reseña histórica del modelado yacimiento-instalaciones

Entre los primeros estudios del acoplamiento de modelos del yacimiento e instalaciones están algunos estudios de yacimientos de gas de Dempsey y cols.(1970). Después se reportaron extensiones de este modelo a yacimientos de aceite con Startzman y cols.(1977), donde un simulador de instalaciones superficiales calculaba la capacidad de la red y pasaba los gastos de los pozos a un simulador de yacimientos para que resolviera la siguiente iteración. Emmanuel y Ranney(1981) presentaron una formulación que consistía en 3 sistemas separados que resolvían el modelo del yacimiento, flujo en los pozos y las ecuaciones en la red superficial. El modelo de dicha red, sin embargo, estaba limitado a redes estructuradas como árboles, con una sola presión de entrega en un nodo.

Posteriormente hubo un aumento en la aplicación de modelos del yacimiento con restricciones en las en las instalaciones superficiales. Como ejemplos de estudios de campo están Wallace y Sproncen(1983) , Stanley y cols(1990), Stoist(1992) y Stoist y cols(1992), de rutinas de administración: Stackel y Brown(1981) y en el desarrollo de técnicas de optimización: Bohannon(1970) y Cain y Shehata(1981) . Un ejemplo del simulador eclipse acoplado a un simulador de superficie (Netopt) se da en Hegguler y cols(1997).

Muchos de esos modelos, por desgracia, adolecían de simplificaciones inconvenientes para reducir el tiempo de cálculo del programa, como es el tratamiento explícito de instalaciones superficiales, que puede causar errores muy grandes si las condiciones cambian muy rápidamente y el uso de curvas de comportamiento de afluencia (IPR) en vez de un simulador del yacimiento, con lo cual no se puede modelar adecuadamente el proceso de agotamiento ni la interferencia entre pozos.

Existen en la actualidad dos tipos de formulaciones, la implícita, que resuelve las ecuaciones del yacimiento y las instalaciones simultáneamente y la explícita, que resuelve por separado yacimiento e instalaciones con una "interfaz" (ecuación del índice de productividad, por ejemplo), que impone condiciones de frontera entre uno y otro.



## CAPÍTULO 1. INTRODUCCIÓN

Se ha desarrollado en los últimos años, además, el uso de la "descomposición en dominios" del yacimiento (descomponer al yacimiento en regiones, que se resuelven independientemente una de otra) para reducir los tiempos de cálculo (Schiozer y Aziz(1994).) Otros investigadores (Byer y cols.(1998) ) han desarrollado técnicas de "preacondicionamiento" (precondition) que es una alternación entre mallas gruesas y finas en el yacimiento que, con división en dominios cerca de los pozos, hace más eficiente el método en cuestión de tiempos de cálculo y exactitud.

Parece ser que la dirección que está siguiendo la investigación en el acoplamiento yacimiento-instalaciones, es en el diseño de código orientado a objetos y técnicas numéricas que hagan más eficiente los cálculos en paralelo con múltiples procesadores(Byer (2000)).

### **El presente trabajo en el contexto del estado del arte**

Como se mencionó anteriormente, han tenido bastante desarrollo las técnicas numéricas y de programación para resolver eficientemente el acoplamiento. De todas las técnicas que existen en este trabajo sólo utilizamos una formulación explícita con condiciones de frontera de Dirichlet. No tratamos métodos más sofisticados y de hecho nuestro modelo es bastante sencillo (una sola fase, 2 dimensiones, flujo de gas seco por estranguladores y tuberías.)

### **Objetivo del trabajo**

El objetivo de esta tesis, más que hacer hincapié en una técnica determinada para resolver un problema, quiere presentar un ejemplo de cómo se puede desarrollar un simulador de yacimientos, tratando un rango muy diverso de temas, que en ocasiones podrían incluso escapar al interés primordial de la simulación. Se trata de la integración de un modelo numérico a partir de pequeños componentes que pueden analizarse por separado (yacimiento, tuberías horizontales, tuberías verticales, estranguladores, terminaciones, etc. ) Se tratan tópicos de la programación del modelo, como son la organización de la información para la programación, métodos para resolver sistemas de ecuaciones lineales, la interfaz con el usuario para introducir y mostrar la información de los datos y de la corrida, el guardado y recuperación de dicha información e incluso la manera de graficar en tercera dimensión para visualizar mejor la geometría y

## CAPÍTULO 1. INTRODUCCIÓN

configuración del modelo. Se trató y analizó, incluso, con el desempeño numérico del modelo y se investigaron posibles soluciones para mejorar y garantizar la convergencia.

Como se observa, el número de temas que se analizó fue grande y por tanto, la atención y esfuerzo se dispersó en varias direcciones y no se pudo abundar mucho en ninguna de ellas. No obstante, el presente trabajo tiene unidad y coherencia en lo que trata y aunque no lo hace con detalle en ningún tema en específico, la suma de todos ellos sí da un tema a esta tesis, que es la implementación de un programa de cómputo que tiene el fin de simular la explotación de un yacimiento de gas, desde la derivación del modelo matemático hasta la evaluación del desempeño numérico.

La aportación principal de este trabajo está dirigida a plantear las bases fundamentales en la construcción de simuladores integrales de yacimientos petroleros- sistemas de producción, que servirán como punto de partida para la comprensión de los estudiantes de ingeniería petrolera y el desarrollo de modelos más complejos durante sus clases.

## CAPÍTULO II

# MODELO DEL YACIMIENTO DE GAS SECO, UNA FASE Y DOS DIMENSIONES

### II.1 DESARROLLO DEL MODELO MATEMÁTICO PARA UN YACIMIENTO DE GAS SECO, DOS DIMENSIONES, UNA FASE

La simulación matemática de yacimientos es, en términos simples, la representación de los fenómenos de flujo en el yacimiento con ecuaciones matemáticas. La simulación numérica es representar al yacimiento con ecuaciones numéricas que sean una aproximación de las analíticas y la simulación computacional es el uso de programas de computadora para resolver las ecuaciones numéricas. Estos programas reciben el nombre de simuladores. Odeh(1969) da una buena introducción a lo que es la simulación de yacimientos, a la par que hace énfasis en que las ecuaciones que modelan el flujo de fluidos en medios porosos no son nuevas. El fundamento teórico de dichas ecuaciones (llamadas ecuaciones de difusividad de fluidos en medios porosos), aparece ya en el libro de Muskat(1949), que es un libro de trascendencia histórica en la ingeniería de yacimientos y una referencia obligada para quien desee ahondar en el tema. Para una deducción breve y concisa de la ecuación de difusividad para una o varias fases se remite al lector al primer capítulo del libro de simulación de Aziz y Settari(1979).

A continuación deduciremos la ecuación diferencial parcial, no lineal que rige el fenómeno de flujo de un gas real en un medio poroso, anisotrópico y no homogéneo, pues es el caso que nos atañe. Este desarrollo se basa en Al-Hussainy y cols(1966) y más concretamente en Lee y Wattenbarger (1996).

La ley de Darcy expresada en forma vectorial, proporciona la relación entre la velocidad aparente promedio de un fluido en un medio poroso y los gradientes de presión a los que está sujeto. En nuestro modelo consideraremos que hay flujo darciano entre las celdas y sólo se presenta flujo del tipo a alta velocidad del tipo Forshheimer en las vecindades de los pozos que, consideraremos como fuentes o sumideros, más que unas condiciones de frontera propiamente dichas.

La ecuación es:

$$\bar{v} = -\frac{1}{\mu(P,T)} K \nabla P, \quad (2.1)$$

donde K es el tensor de permeabilidades dado por:

$$K = \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_z \end{bmatrix} \quad (2.2)$$

Las permeabilidades en la ec. 2.2 son permeabilidades al gas en presencia de una saturación de agua congénita irreductible (la porosidad  $\phi$  puede considerarse el producto  $S_g \phi$ ). Además el gradiente del lado derecho de la ec. 2.1 queda en términos de la presión por considerarse el gradiente de presión hidrostático del gas despreciable, esto es, efectos gravitacionales despreciables.

Por otro lado la ecuación de continuidad, para un fluido (una sola fase) en un medio poroso, y considerando cada punto del volumen del medio como una fuente o sumidero es:

$$\nabla \cdot (\bar{v} \rho) = -\frac{\partial(\rho \phi)}{\partial t} + \frac{\partial R}{\partial V}. \quad (2.3)$$

El segundo término de la derecha de la ec. (2.3) representa la cantidad de materia, por unidad de tiempo, por unidad de volumen, que **entra** al sistema. Es decir:

$$\frac{\partial r\mathcal{E}}{\partial V} > 0 \text{ si se inyecta, y}$$

$$\frac{\partial r\mathcal{E}}{\partial V} < 0 \text{ si se produce.}$$

Este término da cuenta de la existencia de pozos que serán modelados como fuentes o sumideros. Al discretizar, sólo persistirá en aquellas celdas donde hubiese una terminación.

Hacen falta 3 ecuaciones para describir totalmente el flujo del gas en el yacimiento. La primera, que es una ecuación de movimiento, ec. (2.1). La segunda es la ecuación de conservación de la materia, ec. (2.3). Hace falta una tercera, una ecuación de estado para el fluido en cuestión. Esta ecuación de estado es la ley general de los gases reales, ec. 2.4:

$$\rho = \frac{PM_m}{z(P,T)R_v T} \quad (2.4)$$

Sustituyendo la ec. 2.1 en 2.3 resulta :

$$-\nabla \cdot (\rho K \nabla P) = -\frac{\partial(\rho\varphi)}{\partial t} + \frac{\partial r\mathcal{E}}{\partial V} \quad (2.5)$$

de donde ahora al sustituir 2.4 en 2.5:

$$-\nabla \cdot \left( \frac{PM_m}{zR_v T\mu} K \nabla P \right) = -\frac{\partial}{\partial t} \left( \frac{PM_m}{zR_v T} \varphi \right) + \frac{\partial r\mathcal{E}}{\partial V} \quad (2.6)$$

Es conveniente introducir ahora una transformación que simplifique la forma de la ecuación (2.6). Se puede hacer uso de una propiedad artificial de los gases llamada pseudopresión, introducida por primera vez por Al-Hussainy y cols(1966). La pseudopresión, denotada  $P_p$  está dada por:

$$P_p(P, T) = 2 \int_0^r \frac{P dP}{z(P, T) \mu(P, T)}. \quad (2.7)$$

De la expresión anterior puede deducirse que:

$$\nabla P_p = \frac{\partial P_p}{\partial P} \nabla P, \quad (2.8)$$

donde de acuerdo con la ec. 2.7:

$$\frac{\partial P_p}{\partial P} = \frac{2P}{z\mu}. \quad (2.9)$$

Sustituyendo la ecuación 2.9 en 2.8 y multiplicando el resultado por el tensor de permeabilidades K se obtiene:

$$\frac{1}{2} K \nabla P_p = \frac{P}{z\mu} (K \nabla P). \quad (2.10)$$

La ecuación 2.10 se sustituye en la ec. 2.6:

$$-\frac{M_m}{2R_u T} \nabla \cdot (K \nabla P_p) = -\frac{M_m}{R_u T} \frac{\partial}{\partial t} \left( \frac{P\varphi}{z} \right) + \frac{\partial r_{\text{ef}}}{\partial V}. \quad (2.11)$$

Ahora bien,

$$\frac{\partial r_{\text{ef}}}{\partial V} = -\frac{M_m P_{\text{atm}}}{T_{\text{atm}} R_u} \frac{\partial q_g}{\partial V}. \quad (2.12)$$

Sustituyendo 2.12 en 2.11 llegamos a la forma de la ecuación de difusividad que utilizaremos como modelo matemático, la cual en su forma en diferencias finitas es base del modelo numérico del yacimiento:

$$\frac{T_{\text{atm}}}{2TP_{\text{atm}}} \nabla \cdot (K \nabla P_p) = \frac{T_{\text{atm}}}{TP_{\text{atm}}} \frac{\partial}{\partial t} \left( \frac{P\varphi}{z} \right) - \frac{\partial q_g}{\partial V}. \quad (2.13)$$

## II.2 DESARROLLO DEL MODELO NUMÉRICO PARA UN YACIMIENTO DE GAS SECO, DOS DIMENSIONES Y UNA FASE

La ecuación (2.13) no ha podido ser resuelta analíticamente, ni siquiera para casos idealizados. La solución de la ec. (2.13) deberá ser forzosamente numérica. Nuestras suposiciones, además, hacen a la solución altamente dependiente de las condiciones iniciales y de frontera, la heterogeneidad y anisotropía del yacimiento así como el número de pozos y sus gastos.

Comenzaremos pues, encontrando una ecuación en diferencias finitas que sea una "aproximación" a la ecuación (2.13) en tanto que dichas diferencias, si se hacen cada vez más pequeñas, la ecuación numérica equivalente tenderá a ser la ecuación (2.13). En el libro de Aziz y Setari(1979), capítulo 3, se puede encontrar una introducción a las diferencias finitas, así como una rica bibliografía acerca de la teoría y aplicaciones.

Hasta este momento se había supuesto que tanto el tiempo como el espacio eran continuos, es decir que existía un número infinito de puntos para los cuales era posible evaluar la función pseudopresión  $P_p(x,y,t)$ , las permeabilidades  $K_x(x,y)$ ,  $K_y(x,y)$ , porosidad, etc. Es menester ahora dividir al volumen del yacimiento en un número finito de volúmenes de control, para los cuales se supondrá que las variables y parámetros arriba citados (es decir presión, permeabilidad, porosidad, etc.), tendrán valores constantes dentro de dichos volúmenes para un instante dado, e iguales al valor en los puntos colocados en los centros geométricos de cada uno de ellos. Desde este punto de vista, se conocerá el valor de la pseudopresión y la presión sólo en puntos determinados y no en un rango continuo en todo el yacimiento.

Para deducir la ecuación en diferencias finitas, se debe partir de la expansión en series de McLaurin de una función con el objeto de aproximar las derivadas con diferencias.

Para una función  $p(x)$ , si conocemos el valor de  $p(x_0)$  y todas sus derivadas en  $x=x_0$ , podemos aproximar el valor de  $p(x_0+\Delta x)$  y  $p(x_0 - \Delta x)$  con:

$$p(x_0 + \Delta x) = p(x) + \Delta x p'(x) + \frac{\Delta x^2}{2!} p''(x) + \frac{\Delta x^3}{3!} p'''(x) + \dots + \frac{\Delta x^n}{n!} p^n(x) + \dots \quad (2.14)$$

$$p(x_0 - \Delta x) = p(x) - \Delta x p'(x) + \frac{\Delta x^2}{2!} p''(x) - \frac{\Delta x^3}{3!} p'''(x) + \dots + (-1)^n \frac{\Delta x^n}{n!} p^{(n)}(x) + \dots \quad (2.15)$$

donde  $p^{(n)}(x)$  es la derivada n-ésima de p. La serie es exacta para un número infinito de términos. Sin embargo, si truncamos la serie en un número n de términos, introduciremos un error por truncamiento  $e_t$  expresado por:

$$e_t = \sum_{i=n+1}^{\infty} \frac{\Delta x^i}{i!} p^{(i)}(x) \quad (2.16)$$

Con las ecuaciones (2.14) y (2.15) se puede demostrar que las derivadas parciales de primer orden de la pseudopresión en función de las coordenadas X y Y, se pueden expresar como:

$$\left. \frac{\partial P_p^{n+1}}{\partial x} \right|_{i,j} = \frac{P_{p \ i+1,j}^{n+1} - P_{p \ i,j}^{n+1}}{\Delta x_{i,j}} + R_{X_{i,j}}^f \quad (2.17)$$

$$\left. \frac{\partial P_p^{n+1}}{\partial y} \right|_{i,j} = \frac{P_{p \ i,j+1}^{n+1} - P_{p \ i,j}^{n+1}}{\Delta y_{i,j}} + R_{Y_{i,j}}^f \quad (2.18)$$

donde los subíndices i y j denotan que las propiedades están siendo valuadas en los puntos  $x_i$  y  $y_j$ , los superíndices n+1 denotan el nivel de tiempo n+1 (se está siguiendo el esquema implícito de solución.) Los términos  $R_{X_{i,j}}^f$  y  $R_{Y_{i,j}}^f$  son los errores por truncamiento en las series (el superíndice f denota hacia adelante). Es difícil evaluar dichos errores, y en realidad se eliminan de la serie, quedando la relación:

$$\left. \frac{\partial P_p^{n+1}}{\partial x} \right|_{i,j} \approx \frac{P_{p \ i+1,j}^{n+1} - P_{p \ i,j}^{n+1}}{\Delta x_{i,j}} \quad (2.19)$$

$$\left. \frac{\partial P_p^{n+1}}{\partial y} \right|_{i,j} \approx \frac{P_{p \ i,j+1}^{n+1} - P_{p \ i,j}^{n+1}}{\Delta y_{i,j}} \quad (2.20)$$



Seguendo al libro de Aziz y Setari (1979) en su capítulo cuarto, las expresiones necesarias para evaluar  $\nabla \cdot (K \nabla P_p)$ , expresadas en diferencias finitas son:

$$\frac{\partial}{\partial x} \left( f(x, y) \frac{\partial P_p}{\partial x} \right) \approx \frac{1}{\Delta x^2} \left[ f_{i+\frac{1}{2}, j} (P_{p_{i+1, j}}^{n+1} - P_{p_{i, j}}^{n+1}) - f_{i-\frac{1}{2}, j} (P_{p_{i, j}}^{n+1} - P_{p_{i-1, j}}^{n+1}) \right], \quad (2.21)$$

$$\frac{\partial}{\partial y} \left( f(x, y) \frac{\partial P_p}{\partial y} \right) \approx \frac{1}{\Delta y^2} \left[ f_{i, j+\frac{1}{2}} (P_{p_{i, j+1}}^{n+1} - P_{p_{i, j}}^{n+1}) - f_{i, j-\frac{1}{2}} (P_{p_{i, j}}^{n+1} - P_{p_{i, j-1}}^{n+1}) \right], \quad (2.22)$$

donde  $f$  es cualquier función que dependa de  $X$  y  $Y$ . En el presente trabajo caso serán los términos de transmisibilidad que definiremos más adelante.

La parcial de cualquier función de la presión con respecto al tiempo, en el esquema implícito de solución se aproxima como:

$$\frac{\partial f(P)}{\partial t} \approx \frac{f(P)_{p_{i, j}}^{n+1} - f(P)_{p_{i, j}}^n}{\Delta t^n}. \quad (2.23)$$

Teniendo definida la forma de las derivadas parciales en diferencias finitas es posible ahora transformar la ecuación (2.13). Si tomamos la aproximación

$$dV = h(x, y) dx dy \approx h \Delta x \Delta y, \quad (2.24)$$

y además

$$\frac{\partial q_x}{\partial V} = \frac{q_{x, i, j}^{n+1}}{\Delta V} = \frac{q_{x, i, j}^{n+1}}{h \Delta x \Delta y}. \quad (2.25)$$

(2.13) queda escrita:

$$\frac{T_{aim}}{2TP_{aim}} \nabla \cdot (\Delta x \Delta y h K \nabla P_p) = \frac{T_{aim}}{TP_{aim}} \frac{\partial}{\partial t} \left( \frac{PV_p}{z} \right) - q_{x, i, j}^{n+1}. \quad (2.26)$$

Desarrollando al operador nabla:

$$\frac{T_{atm}}{2TP_{atm}} \left( \frac{\partial}{\partial x} \left( \Delta x \Delta y h k_x \frac{\partial P_p}{\partial x} \right) + \frac{\partial}{\partial y} \left( \Delta x \Delta y h k_y \frac{\partial P_p}{\partial y} \right) \right) = \frac{T_{atm}}{TP_{atm}} \frac{\partial}{\partial t} \left( \frac{PV_p}{z} \right) - q_{\kappa i,j}^{n+1}. \quad (2.27)$$

En la ecuación (2.27) se substituyó  $V_p = h \Delta x \Delta y \phi$ , que es el volumen poroso de la celda.

Expresando (2.27) en diferencias finitas:

$$\begin{aligned} & \frac{T_{atm}}{2TP_{atm}} \left[ \left( \frac{\Delta y h k_x}{\Delta x} \right)_{i+\frac{1}{2},j} \left( P_{\rho_{i+1},j}^{n+1} - P_{\rho_{i,j}}^{n+1} \right) - \left( \frac{\Delta y h k_x}{\Delta x} \right)_{i-\frac{1}{2},j} \left( P_{\rho_{i,j}}^{n+1} - P_{\rho_{i-1},j}^{n+1} \right) \right] + \dots \\ & \dots + \frac{T_{atm}}{2TP_{atm}} \left[ \left( \frac{\Delta x h k_y}{\Delta y} \right)_{i,j+\frac{1}{2}} \left( P_{\rho_{i,j+1}}^{n+1} - P_{\rho_{i,j}}^{n+1} \right) - \left( \frac{\Delta x h k_y}{\Delta y} \right)_{i,j-\frac{1}{2}} \left( P_{\rho_{i,j}}^{n+1} - P_{\rho_{i,j-1}}^{n+1} \right) \right] = \\ & \frac{T_{atm}}{TP_{atm} \Delta t^n} \left[ \left( \frac{PV_p}{z} \right)_{i,j}^{n+1} - \left( \frac{PV_p}{z} \right)_{i,j}^n \right] + q_{\kappa i,j}^{n+1}. \quad (2.28) \end{aligned}$$

En este momento es conveniente introducir los términos de transmisibilidad del medio. Estos están definidos como:

$$T_{Ei,j} = \left( \frac{\Delta y h k_x}{\Delta x} \right)_{i+\frac{1}{2},j}. \quad (2.29)$$

$$T_{Wi,j} = \left( \frac{\Delta y h k_x}{\Delta x} \right)_{i-\frac{1}{2},j}. \quad (2.30)$$

$$T_{Si,j} = \left( \frac{\Delta x h k_y}{\Delta y} \right)_{i,j+\frac{1}{2}}. \quad (2.31)$$

$$T_{N_i,j} = \left( \frac{\Delta x h k_y}{\Delta y} \right)_{i,j-\frac{1}{2}} \quad (2.32)$$

La convención para establecer la dirección y sentido de  $i$  y  $j$ , así como de E (Este), W (Oeste), N (Norte) y S (Sur) son de acuerdo a la figura 2.1:

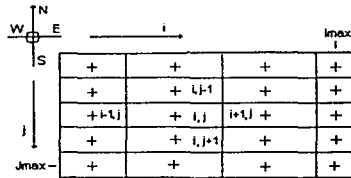


Figura 2.1.- Convención para las coordenadas cartesianas en el yacimiento

La interpretación física de las transmisibilidades viene de los términos que intervienen en ellas y de la ley de Darcy. Sus unidades son de área (área transversal), entre distancia (paralela al flujo) y permeabilidad. Es decir que son proporcionales al gasto volumétrico que atravesará un área con una permeabilidad determinada, recorriendo una distancia determinada para una caída de presión. El flujo mejora si el área es más grande o la permeabilidad es alta (que le es indiferente al término (Área transversal\* Permeabilidad) y disminuye, si para la misma caída de presión la distancia a recorrer es mayor. De ahí que la transmisibilidad mida con qué facilidad se "transmiten" o transportan los fluidos de una celda a otra, sin tomar en cuenta aún la viscosidad y otras características de los fluidos que se muevan por los poros.

Las transmisibilidades están evaluadas en los puntos  $(i,j+1/2)$ ,  $(i,j-1/2)$ ,  $(i+1/2,j)$  y  $(i-1/2,j)$ . El término  $1/2$  sólo significa que es algún promedio (aritmético, armónico, geométrico, logarítmico, etc.) para las propiedades entre la celda anterior (si es  $-1/2$ ) o la siguiente (si es  $+1/2$ ). En el modelo para el presente trabajo (que está basado en el capítulo 11 de libro de Lee y Wattenbarger(1996)), se toma la media geométrica para evaluar las propiedades en " $1/2$ ".

Se establecen las siguientes definiciones:

$$T_{1i,j} = \left( \frac{\Delta y_i h_{i,j} k_{x_{i,j}}}{\Delta x_i / 2} \right). \quad (2.33)$$

$$T_{2i,j} = \left( \frac{\Delta y_i h_{i+1,j} k_{x_{i+1,j}}}{\Delta x_{i+1} / 2} \right). \quad (2.34)$$

$$T_{3i,j} = \left( \frac{\Delta x_i h_{i,j} k_{y_{i,j}}}{\Delta y_j / 2} \right). \quad (2.35)$$

$$T_{4i,j} = \left( \frac{\Delta x_i h_{i,j+1} k_{y_{i,j+1}}}{\Delta y_{j+1} / 2} \right). \quad (2.36)$$

Las ecuaciones (2.29) y (2.31) pueden expresarse por:

$$T_{Ei,j} = \frac{T_{1i,j} T_{2i,j}}{T_{1i,j} + T_{2i,j}}. \quad (2.37)$$

$$T_{Si,j} = \frac{T_{3i,j} T_{4i,j}}{T_{3i,j} + T_{4i,j}}. \quad (2.38)$$

Por la simetría de estos coeficientes,

$$T_{wi,j} = T_{Ei-1,j}. \quad (2.39)$$

$$T_{Ni,j} = T_{Si,j-1}. \quad (2.40)$$

Si se incorporan las transmisibilidades a la ecuación (2.28), ésta queda:

$$\frac{T_{alm}}{2TP_{alm}} \left[ T_{E_{i,j}} (P_{p_{i+1,j}}^{n+1} - P_{p_{i,j}}^{n+1}) + T_{w_{i,j}} (P_{p_{i-1,j}}^{n+1} - P_{p_{i,j}}^{n+1}) + T_{S_{i,j}} (P_{p_{i,j+1}}^{n+1} - P_{p_{i,j}}^{n+1}) + T_{N_{i,j}} (P_{p_{i,j-1}}^{n+1} - P_{p_{i,j}}^{n+1}) \right] =$$

$$\frac{T_{alm}}{TP_{alm} \Delta t^n} \left( \left( \frac{PV_p}{z} \right)_{i,j}^{n+1} - \left( \frac{PV_p}{z} \right)_{i,j}^n \right) + q_{k_{i,j}}^{n+1}. \quad (2.41)$$

En las iteraciones globales, para cada nivel de tiempo se darán las presiones de fondo fluyendo de las celdas donde haya un pozo produciendo. En una sección posterior se discutirá la representación de los pozos en un simulador de yacimientos. Por el momento es conveniente definir aquí el concepto de índice de productividad para pozos de gas.

$$J'_{i,j}{}^{n+1} = \frac{q_{k_{i,j}}^{n+1}}{P_{p_{i,j}}^{n+1} - P_{p_{wf_{i,j}}}^{n+1}}$$

(2.42)

Nótese que la ecuación (2.42) utiliza pseudopresiones en vez de presiones. La  $P_{p_{wf_{i,j}}}^{n+1}$  es la pseudopresión de fondo fluyente. Existen diferentes ecuaciones para evaluar a J, de acuerdo a las características de la terminación, tipo de flujo (darciano, no darciano, etc.)

De (2.42) se puede sustituir el gasto en la ecuación (2.41) como:

$$q_{k_{i,j}}^{n+1} = J'_{i,j}{}^{n+1} (P_{p_{i,j}}^{n+1} - P_{p_{wf_{i,j}}}^{n+1}). \quad (2.43)$$

Por otro lado, el lado derecho de la ecuación (2.41) está en función de presiones y de  $z(P,T)$ , no de pseudopresiones. Se debe introducir una variable adicional, la variable  $\alpha$ , que permitirá linealizar la ecuación (2.41) una vez que ésta sea evaluada para una iteración en el yacimiento (para detalles vea la sección II.3).

$$\alpha_{i,j}^{n+1} = \frac{1}{\Delta t^n} \left( \frac{T_{alm}}{P_{alm} T} \right) \left( \frac{\left( \frac{V_p P}{z} \right)_{i,j}^{n+1}}{P_{p_{i,j}}^{n+1} - P_{p_{i,j}}^n} - \left( \frac{V_p P}{z} \right)_{i,j}^n \right). \quad (2.44)$$

Sustituyendo (2.43) y (2.44) en (2.41) se tiene:

$$\frac{T_{atm}}{2TP_{atm}} \left[ T_{Ei,j} (P_{\rho_{i+1,j}}^{n+1} - P_{\rho_{i,j}}^{n+1}) + T_{Wi,j} (P_{\rho_{i-1,j}}^{n+1} - P_{\rho_{i,j}}^{n+1}) + T_{Si,j} (P_{\rho_{i,j+1}}^{n+1} - P_{\rho_{i,j}}^{n+1}) + T_{Ni,j} (P_{\rho_{i,j-1}}^{n+1} - P_{\rho_{i,j}}^{n+1}) \right] =$$

$$= \alpha_{i,j}^{n+1} (P_{\rho_{i,j}}^{n+1} - P_{\rho_{i,j}}^n) + J'_{i,j}{}^{n+1} (P_{\rho_{i,j}}^{n+1} - P_{\rho_{wf,i,j}}^{n+1}). \quad (2.45)$$

Para llegar a una forma conveniente de (2.45) donde las pseudopresiones (incógnitas) queden del lado izquierdo, y los términos "conocidos" del lado derecho, se introducen los coeficientes de flujo, definidos por las ecuaciones siguientes:

$$a_{Ei,j} = \frac{T_{atm}}{2P_{atm}T} T_{Ei,j}, \quad (2.46)$$

$$a_{Wi,j} = \frac{T_{atm}}{2P_{atm}T} T_{Wi,j}, \quad (2.47)$$

$$a_{Si,j} = \frac{T_{atm}}{2P_{atm}T} T_{Si,j}, \quad (2.48)$$

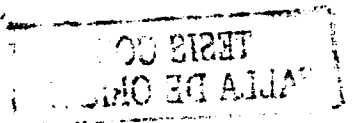
$$a_{Ni,j} = \frac{T_{atm}}{2P_{atm}T} T_{Ni,j}, \quad (2.49)$$

$$a_{Ci,j}^n = a_E + a_W + a_S + a_N + J'_{i,j}{}^n + \alpha_{i,j}^n. \quad (2.50)$$

Además, si se define el término independiente (que no dependa explícitamente de las pseudopresiones en el nivel de tiempo  $n+1$ ) como:

$$d_{i,j}^n = \alpha_{i,j}^n P_{\rho_{i,j}}^n - J'_{i,j}{}^n P_{\rho_{wf,i,j}}^n. \quad (2.51)$$

La ecuación (2.45) queda expresada finalmente:



$$-a_{Ei,j} P_{\rho_{i+1,j}}^{n+1} - a_{wi,j} P_{\rho_{i-1,j}}^{n+1} + a_{ci,j} P_{\rho_{i,j}}^{n+1} - a_{Si,j} P_{\rho_{i,j+1}}^{n+1} - a_{Ni,j} P_{\rho_{i,j-1}}^{n+1} = d_{i,j}^n \quad (2.52)$$

La ecuación (2.52) no es una sola ecuación, sino un conjunto de n ecuaciones, siendo n el número de celdas que haya en el yacimiento. Cualquiera de los coeficientes de flujo, excepto el central (dado por la ec. 2.50), será cero si no existe la celda inmediata a la ij en cuestión y en la dirección del coeficiente de flujo. Véase la figura 2.2.

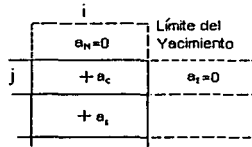


Figura 2.2.- Los coeficientes de flujo en la dirección de celdas aledañas inexistentes son cero

Como se puede observar de la ecuación (2.52), cada ecuación implica 5 términos, o más explícitamente, 5 celdas del yacimiento. El sistema de ecuaciones dado por la ecuación (2.52) puede expresarse en forma matricial:

$$A \bar{P}_p = \bar{d} \quad (2.53)$$

donde  $\bar{P}_p$  es un vector formado por las pseudopresiones y  $\bar{d}$  es el vector de términos independientes, ambos con elementos numerados y ordenados de manera secuencial.

La forma de la matriz A (y el orden de los elementos de los vectores, como se dijo), depende de la numeración que se utilice para identificar a cada celda del yacimiento y del ordenamiento de dichas celdas. Por ejemplo, para yacimientos rectangulares, con un orden secuencial por renglones (ver fig. 2.3) la matriz que se forma es pentadiagonal con los coeficientes  $a_c$  en la diagonal principal, los coeficientes  $a_e$  en la diagonal central derecha, coeficientes  $a_w$  en la diagonal central izquierda y coeficientes  $a_s$  y  $a_n$  en las diagonales separadas derecha e izquierda, respectivamente.

**TESIS CON FALLA DE ORIGEN**

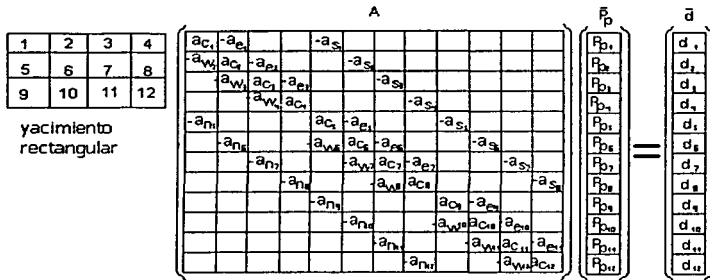


Figura 2.3.- Matriz pentadiagonal de un yacimiento rectangular, con ordenamiento secuencial por renglones.

El sistema de ecuaciones que resultan de la ecuación (2.52), o su forma matricial expresada por (2.53) no pueden resolverse en forma simple con respecto al vector de pseudopresiones en el tiempo  $n+1$ , puesto que tanto los coeficientes  $a_c$  (ecuación (2.50)) y los términos independientes  $d$  (ecuación (2.51)) dependen de las constantes  $\alpha$  (ecuación (2.44)) y  $J$  (índice de productividad, definido por (2.42)), que a su vez dependen de las presiones y pseudopresiones en el tiempo  $n+1$ , los gastos de los pozos (que no se conocen en el tiempo  $n+1$  y son parte de las incógnitas que se tienen que encontrar, además de las pseudopresiones).

De aquí queda claro que la ecuación (2.53) no es una ecuación lineal, puesto que tanto la matriz  $A$  como el vector  $d$  son funciones del vector  $P_p$ . En el libro Numerical Analysis de Burden y Faires(1989) se dan los fundamentos matemáticos (definiciones, teoremas y sus correspondientes demostraciones) y algoritmos de diferentes métodos numéricos para resolver sistemas de ecuaciones no lineales. Algunos de los métodos son el de sustituciones sucesivas, el método de Newton, el de Broyden y el del llamado "descenso más pronunciado", entre otros.

En el presente trabajo, para resolver la ecuación (2.53), se utiliza una especie de variante del Método de sustituciones sucesivas (nuestro modelo es exactamente el presentado por Lee y Wattenbarger (1996).) La característica sobresaliente de este método es que se suponen valores del vector  $P_p$ , calculándose con ellos la matriz  $A$  (es decir se calcula el vector  $\alpha$ , ver ec. (2.50)) y el vector  $d$  y con ellos se resuelve para  $P_p$  de (2.53). ( $P_p$  no está





despejado, diferencia por la cual se dice que el método es una variante del de sustituciones sucesivas); con este nuevo valor se calculan nuevamente  $A$  y  $d$  y se repite el ciclo, iterando hasta que los valores calculados de  $P_p$  converjan y cumplan una tolerancia especificada.

En este momento es conveniente establecer el algoritmo que el programa seguirá para resolver la ecuación (2.53), especificando claramente los datos, las incógnitas y la dependencia funcional de cada término necesario para los cálculos. Debe tenerse en mente que el objetivo de resolver la ecuación (2.53) es encontrar la distribución de presiones y pseudopresiones en el yacimiento, así como gastos en los pozos en el tiempo  $n+1$  dados la distribución de presiones (y pseudopresiones) del tiempo  $n$  y las presiones de fondo fluyendo en el nivel de tiempo  $n+1$  (esta condición está impuesta por la forma en que está unido el yacimiento a sus condiciones de frontera, que son los pozos unidos a la red superficial de recolección y cómo la iteración global calcula gastos y presiones en todos los componentes del sistema, lo cual se discutirá a detalle en el capítulo IV.)

## II.3 ALGORITMO PARA RESOLVER LAS ECUACIONES NUMÉRICAS DEL YACIMIENTO

Antes de formular el algoritmo, se enunciarán los datos y las incógnitas que son entradas y salidas, respectivamente.

Los coeficientes de flujo Este, Oeste, Sur y Norte sólo se necesitan calcular una vez en toda la iteración en el yacimiento y se les puede considerar como propiedades de las celdas.

Dentro de los datos se encuentran las presiones de fondo fluyendo en el nivel de tiempo  $n+1$ . Esto se debe a que los cálculos de la red superficial proporcionan como resultado las presiones en la cabeza de los pozos y los gastos de cada pozo; con estos datos se calculan las presiones de fondo fluyendo (cálculos que no competen al yacimiento) y posteriormente el simulador para el yacimiento debe de encontrar la nueva distribución de presiones y nuevos gastos en los pozos.

DATOS:

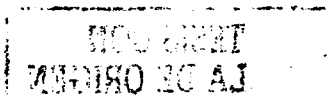


Tabla 2.1.- Datos del algoritmo 2.1.

Propiedades del Yacimiento	Coefficientes de Flujo (se deducen de las propiedades del yacimiento)	Datos de la iteración pasada o bien iniciales
1.- $h_{i,j} \forall (i, j) \in Yac.$	1.- $a_{Ei,j} \forall (i, j) \in Yac.$ ec. 2.46	1.- $P_{Pi,j}^n \forall (i, j) \in Yac.$
2.- $K_{Xi,j} \forall (i, j) \in Yac.$	2.- $a_{Wi,j} \forall (i, j) \in Yac.$ ec. 2.47	2.- $P_{i,j}^n \forall (i, j) \in Yac.$
3.- $K_{Yi,j} \forall (i, j) \in Yac.$	3.- $a_{Si,j} \forall (i, j) \in Yac.$ ec. 2.48	3.- $P_{Pwf,i,j}^n \forall (i, j) \in Yac.*$
4.- $\Delta X_{i,j} \forall (i, j) \in Yac.$	4.- $a_{Ni,j} \forall (i, j) \in Yac.$ ec. 2.49	4.- $q_{i,j}^n \forall (i, j) \in Yac.*$
5.- $\Delta Y_{i,j} \forall (i, j) \in Yac.$		5.- $V_{Pi,j}^n \forall (i, j) \in Yac.$
6.- $\phi_{i,j} \forall (i, j) \in Yac.$		
7.- $C_{f,i,j} \forall (i, j) \in Yac.$		
8.- $S_{i,j} \forall (i, j) \in Yac.*$		Parámetros de la iteración
9.- $D_{i,j} \forall (i, j) \in Yac.*$		1.- $tol$
10.- $T_{yac}$		2.- $\Delta t^n$
11.- $P_{atm}, T_{atm}$		

(\*) En las celdas donde no exista terminación estas propiedades o variables son siempre cero. La nomenclatura se presenta al final del trabajo.

## INCÓGNITAS

Tabla 2.2.- Incógnitas del algoritmo 2.1.

NECESARIAS PARA LA SIGUIENTE ITERACIÓN EN EL YACIMIENTO	INTERNAS O AUXILIARES EN LOS CÁLCULOS
1.- $P_{p,i,j}^{n+1} \forall (i, j) \in Yac.$ , ec.5.53	1.- $\alpha_{i,j}^{n+1} \forall (i, j) \in Yac.$ , ec. 2.44
2.- $q_{i,j}^{n+1} \forall (i, j) \in Yac.$ , ec.2.43	2.- $J_{i,j}^{n+1} \forall (i, j) \in Yac.$ , ec. 2.43
3.- $P_{i,j}^{n+1} \forall (i, j) \in Yac.$	3.- $\alpha_{c,i,j}^n \forall (i, j) \in Yac.$ , ec. 2.50
4.- $V_{p,i,j}^{n+1} \forall (i, j) \in Yac.$	4.- $d_{i,j}^n \forall (i, j) \in Yac.$ , ec. 2.51

## ALGORITMO 2.1

El siguiente algoritmo tiene por entradas las variables y parámetros enunciados como datos en la tabla (2.1) y proporciona como resultado las variables enlistadas como incógnitas en la tabla (2.2). Su objetivo es encontrar la distribución de presiones en el yacimiento y gastos en los pozos dados el estado anterior del yacimiento, las presiones de fondo fluyendo y un cambio o "delta de tiempo".

Se utiliza la notación vectorial para abreviar, entendiéndose que el vector contiene a los elementos de todas las celdas ordenados apropiadamente (ver secc. II.2.)

$$1.- \text{Suponer } \bar{P}_p^{n+1} \text{ y } \bar{P}^{n+1} = \bar{P}^{n+1}(\bar{P}_p^{n+1}).$$

2.- Calcular todos los elementos de  $\bar{q}^{n+1}$  con la ecuación

$$q_{s,i,j}^{n+1} = J_{i,j}^{n+1} \left( P_{p,i,j}^{n+1} - P_{pwf,i,j}^{n+1} \right). \quad (2.43)$$

Este paso debe omitirse si es la primera iteración del ciclo. En ésta se utilizan los valores de  $q_{i,j}^n$ .

3.- Calcular  $\bar{J} = \bar{J}(\bar{q}^{n+1})$  por medio de la ecuación (2.72) (ver sección II.5.)

4.- Calcular  $\bar{V}_p^{n+1} = \bar{V}_p(\bar{P}^{n+1})$  y  $\bar{z}^{n+1} = \bar{z}(\bar{P}^{n+1})$ .

5.- Calcular todos los elementos de  $\alpha$  con:

$$\alpha_{i,j}^{n+1} = \frac{1}{\Delta t^n} \left( \frac{T_{ann}}{P_{ann} T} \right) \left( \frac{V_p P}{z} \right)_{i,j}^{n+1} - \left( \frac{V_p P}{z} \right)_{i,j}^n. \quad (2.44)$$

6.- Calcular todos los elementos de  $\bar{a}_C$  con:

$$a_{C,i,j}^{n+1} = a_E + a_W + a_S + a_N + J'_{i,j}{}^{n+1} + \alpha_{i,j}^{n+1}. \quad (2.50)$$

7.- Calcular todos los elementos de  $\bar{d}$  con:

$$d_{i,j}^{n+1} = \alpha_{i,j}^{n+1} P_{p,i,j}^{n+1} - J'_{i,j}{}^{n+1} P_{pwf,i,j}^{n+1}. \quad (2.51)$$

8.- Formar o modificar la matriz A con los coeficientes de flujo laterales (éstos jamás se modifican) y los centrales recién calculados en el paso 5.

9.- Obtener  $\bar{P}_p^{n+1}$  resolviendo de

$$A\bar{P}_p = \bar{d}. \quad (2.52)$$

10.- Comparar los valores calculados de  $\bar{P}_p$  con los supuestos. Si el valor absoluto de todos y cada uno de los elementos de la diferencia de ambos vectores entra en la tolerancia especificada, terminar el ciclo, de lo contrario tomar  $\bar{P}_p$  calculado como supuesto y regresar al paso 2.

## II.4 SOLUCIÓN DE MATRICES PENTADIAGONALES POR ELIMINACIÓN LU

En el punto 9 del algoritmo 2.1 se requiere que se resuelva el sistema

$$A\vec{P}_p = \vec{d} . \quad (2.52)$$

donde la matriz A está formada por los coeficientes de flujo. Sin embargo nada se mencionó acerca del método que habría de seguirse para resolver dicha matriz.

Como se mencionó brevemente en la sección II.2, la forma de la matriz depende del ordenamiento de las celdas al ser puestas las pseudopresiones en el vector  $\vec{P}_p$ , adquiriendo características especiales que pueden usarse al aplicar algoritmos más eficientes que exploten ventajosamente tales características. Si el ordenamiento es secuencial por columnas o renglones la forma que tendrá la matriz será pentadiagonal, con ceros en los elementos que no pertenezcan a las diagonales.

En materia de selección de métodos para resolver matrices pentadiagonales que resultan de diferentes esquemas de ordenamiento, resulta muy conveniente remitir al lector al capítulo 8 del libro de Aziz y Setari(1979) . Como se ve en él, los métodos pueden clasificarse en directos o en iterativos. Los directos son factorización LU y matrices ralas. En los indirectos están el método de Jacobi, Gauss-Seidel, métodos de relajación, métodos aditivos-correctivos, etc. En el mismo libro de Aziz(1979) se proporciona una guía sobre la selección de los métodos. Se concluye que los métodos directos tienen la desventaja de requerir un gran espacio en memoria y sufrir de grandes errores de redondeo para problemas de flujo multifásico. Su gran ventaja es su estabilidad. Las desventajas de los métodos iterativos son que su eficiencia depende del problema y en ocasiones se vuelven inestables. Su ventaja es el poco espacio requerido en memoria. Más decisivo aún para la selección del método para nuestro caso es que para yacimientos pequeños (con menos de unas 1600 celdas o de 40X40), los métodos directos son más eficientes que los iterativos.

A la luz de esta información nos hemos decidido a utilizar el método de factorización LU, puesto que no se espera que los yacimientos para nuestro caso sean muy grandes. El alcance de nuestro simulador es académico, no servir como una herramienta profesional. Por el tiempo limitado no se implementó ningún método iterativo. Se seleccionó un método directo como el LU por su estabilidad.

Considérese una matriz cuadrada pentadiagonal  $A$  de  $N \times N$  con un "ancho de banda"  $B=2M+1$ , donde  $M$  es el número de columnas que hay desde la diagonal principal hasta el término más alejado diferente de cero en un mismo renglón; es decir, que si

$$A = \{a_{ij}\}, \quad (2.53)$$

entonces

$$a_{ij} = 0 \quad \text{si } |j - i| > M. \quad (2.54)$$

En la figura (2.3) se esquematiza una matriz cuadrada pentadiagonal de orden  $N=12$  y "distancia" a las diagonales laterales  $M=4$ .

La eliminación consiste en operaciones de renglones de la matriz para transformar a  $A$  en una matriz triangular superior. Se deben eliminar los elementos diferentes de cero que se encuentren en la misma columna por debajo del elemento de la diagonal principal del renglón  $k$ -ésimo, variando  $k$  de 1 hasta  $N-1$ . La operación  $k$ -ésima se define como:

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - a_{ij}^{(k-1)} \left[ \frac{a_{ij}^{(k-1)}}{a_{kk}^{(k-1)}} \right], \quad j = k, \dots, \min(k+M, N),$$

$$\text{para los renglones de } i = k+1 \text{ hasta } \min(k+M, N), \quad (2.55)$$

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} \quad \text{en cualquier otra parte.} \quad (2.56)$$

Después de la operación  $(k-1)$ -ésima la matriz lucirá como en la figura 2.4 a la izquierda:

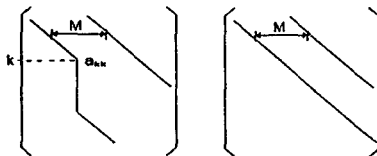


Figura 2.4.- Izquierda: matriz  $A^{(k-1)}$  después de la operación  $(k-1)$ -ésima. Derecha: matriz  $A^{(N-1)}$  después de  $N-1$  operaciones.

Después de  $N-1$  iteraciones la matriz  $A$  se transforma en

$$A^{(N-1)} \equiv U . \quad (2.57)$$

Nótese que no se ha formado explícitamente la matriz L. En vez de esto calculamos el vector

$$\bar{g} = L^{-1} \bar{d} . \quad (2.58)$$

Al realizar el mismo tipo de manipulaciones en el vector d del lado derecho de la ec. (2.52). Esto se lleva a efecto al calcular una secuencia de vectores  $\bar{d}^{(k)}$  junto con  $A^{(k)}$  haciendo:

$$d_i^{(k)} = d_i^{(k-1)} - d_k^{(k-1)} \left[ \frac{\alpha_{ij}^{(k-1)}}{\alpha_{kk}^{(k-1)}} \right], \text{ para } i=k+1, \dots, \min(k+M, N). \quad (2.59)$$

$$d_i^{(k)} = d_i^{(k-1)} \text{ en cualquier otra parte.} \quad (2.60)$$

Por lo tanto

$$\bar{d}^{(n-1)} = \bar{g} , \quad (2.61)$$

y la solución final se obtiene resolviendo:

$$UP_p = \bar{g} . \quad (2.62)$$

En el libro de Aziz y Setari se proporciona una fórmula del trabajo W en términos de número de multiplicaciones requeridas por el método para una matriz pentadiagonal de NXN con "distancia" a las diagonales laterales M.

$$W = (N - 2M + 1)[(M + 1)^2 + M] + \frac{M(M-1)(2M-1)}{3} + M(M-1) + (M+2)^2 - M - 9 . \quad (2.63)$$

En el presente trabajo la forma de ordenamiento es secuencial en la manera como se ilustra en la figura 2.3. El programa tiene la posibilidad de diseñar yacimientos con forma irregular. Esta forma irregular provoca trastornos en el ancho de la banda de la matriz

(denotado  $M$ ) y hace que éste sea variable, siendo en algunos renglones mayor y en otros menor.

Para poder utilizar el método LU en esta situación se toma  $M$  igual al ancho máximo encontrado en todos los renglones de la matriz. Es incorrecto tomar  $M$  variable como  $M(k)$  en (2.55) y (2.59) puesto que no hay garantía de que los elementos que originalmente eran cero a la izquierda y derecha de  $k-M$  y  $k+M$ , respectivamente, sigan siéndolo después de que el renglón haya sido modificado por alguna operación de (2.55). Lo que conviene hacer es sólo omitir el renglón  $i$  en la ecuación (2.55) si y sólo si

$$a_{ik}^{(k-1)} = 0. \quad (2.64)$$

De aquí resulta evidente que la eficiencia del método disminuye si el yacimiento es irregular, del tipo en que se ilustra en la figura 2.5:

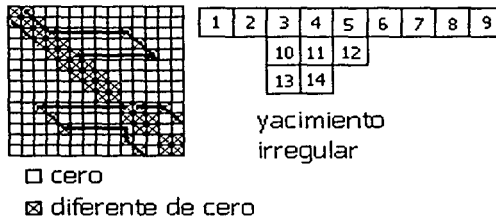


Figura 2.5.- Un yacimiento irregular causa anchos de banda variables

El método se vuelve ineficiente especialmente si hay pocos renglones en la matriz, con "anchos de banda" muy grandes con respecto al ancho promedio de los renglones de la matriz, tal como el yacimiento representado en la figura 2.5.

## II.5 REPRESENTACIÓN DE LOS POZOS EN LAS CELDAS DEL YACIMIENTO



Como resultado de la simulación de yacimientos, se obtienen las presiones medias en los puntos de la malla que supuestamente son la presión media de las celdas que rodean al punto. Sin embargo un pozo localizado dentro de alguna celda no puede utilizar dicha presión como presión de fondo fluente, pues en realidad hay todo un perfil de presiones dentro de la celda para llegar hasta el pozo, es decir que hay una caída de presión desde alguna presión "media" hasta la presión de fondo fluente (ver figura 2.6)

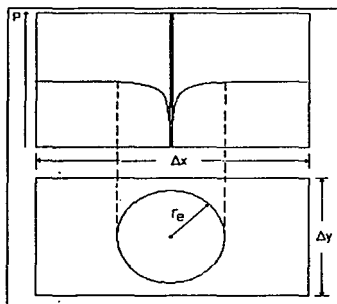


Figura 2.6.- Perfil de presiones dentro de una celda con un pozo

Es necesario obtener una ecuación que relacione la presión de la celda con la presión de fondo fluendo del pozo. Hay en la literatura una buena cantidad de ecuaciones que intentan resolver este problema (ver Demetre y Farouq(1994) para una revisión bibliográfica). Para el caso que nos atañe (flujo en 2 dimensiones), podríamos dividir a las ecuaciones en 2 grandes ramas, las que involucran a las presiones o propiedades de las celdas aledañas a la celda con el pozo y las que no. La mayoría de las ecuaciones utiliza el concepto de índice de productividad, definido por:

$$J = \frac{q}{P_{med} - P_{wf}} \quad (2.65)$$

Donde usualmente el gasto es gasto de aceite a condiciones estándar y Pmed es alguna presión media "del yacimiento". Suponiendo una distribución de presiones en función del radio a partir del pozo (ver fig. 2.6), esta presión media se evalúa a un radio r\_e, llamado radio de drene. Si se considera un pozo vertical, con penetración total, estado

estacionario (la presión media y el radio de drenaje no cambian) y gasto constante, entonces la ecuación de Darcy en coordenadas radiales nos da:

$$q = \frac{2\pi kh}{\mu B} \frac{P_{med} - P_{wf}}{\ln\left(\frac{r_e}{r_w}\right) + s} \quad (2.66)$$

Incluyendo ya los efectos de daño.

Comparando la ec. (2.66) con la (2.65) se deduce que para este caso:

$$J = \frac{2\pi kh}{\mu B \left( \ln\left(\frac{r_e}{r_w}\right) + s \right)} \quad (2.67)$$

En los primeros trabajos como los de van Poolen y cols.(1968), al aplicar la ecuación (2.66) para relacionar la presión de la celda con la del pozo (considerando un solo pozo en el centro de la celda), hicieron la suposición básica de que la presión media de (2.66) es la presión media areal de un círculo de igual área que la celda, es decir:

$$r_e = \sqrt{\frac{\Delta x \Delta y}{\pi}} \quad (2.68)$$

Posteriormente Peaceman(1978) demostró, con una serie de experimentos numéricos que para una celda cuadrada la presión de la celda era la presión evaluada a un radio igual a 0.2 veces el espaciamiento de la celda (radio efectivo, denotado como  $r_e$ ). Posteriormente él mismo(1982) dedujo la expresión para encontrar el radio efectivo del pozo para celdas rectangulares ( $\Delta x \neq \Delta y$ ), con permeabilidad anisotrópica en la celda:

$$r_o = 0.28 \frac{\left[ \left( \frac{k_y}{k_x} \right)^{\frac{1}{2}} \Delta x^2 + \left( \frac{k_x}{k_y} \right)^{\frac{1}{2}} \Delta y^2 \right]^{\frac{1}{2}}}{\left( \frac{k_y}{k_x} \right)^{\frac{1}{4}} + \left( \frac{k_x}{k_y} \right)^{\frac{1}{4}}} \quad (2.69)$$

La ecuación (2.69) es quizás la ecuación más popular para problemas areales con pozos verticales centrados en las celdas. Esta ecuación es la que ocupamos en nuestro simulador por su sencillez y por que no implica las presiones de otras celdas.

### Índice de productividad para pozos de gas

La ecuación (2.66) fue derivada considerando propiedades constantes en las cercanías del pozo para un líquido incompresible. Estas consideraciones no son adecuadas para el flujo de gas debido a su alta compresibilidad y dependencia de la viscosidad de la presión. Para obtener un índice de productividad para pozos de gas, Russell y cols.(1982) utilizaron primeramente la misma forma de la ecuación (2.66) pero con las presiones al cuadrado considerando las propiedades del gas constantes e iguales a su valor evaluado en la presión media entre el radio de drene y el radio del pozo. Posteriormente, al ser introducido el concepto de seudopresión por Al Hussainy y cols.(1966) se demostró que resolviendo la ecuación de difusividad con el cambio de variable de presión a seudopresión, la ecuación para flujo en estado estacionario tenía la misma forma que la ecuación (2.66), pero sustituyendo presiones por seudopresiones y suprimiendo la viscosidad. La ecuación resultante es:

$$q_r = \frac{2\pi khT_{aim}(P_{P_{med}} - P_{P_{wf}})}{P_{aim}T \left( \ln \left( \frac{r_e}{r_w} \right) + s \right)} \quad (2.70)$$

Si se incluye en la ecuación (2.70) el factor D de caída de presión por flujo turbulento debido a altas velocidades se obtiene:

$$q_g = \frac{2\pi khT_{am}(P_{r_{med}} - P_{r_{wf}})}{P_{am}T \left( \ln \left( \frac{r_e}{r_w} \right) + s + Dq_g \right)} \quad (2.71)$$

De aquí se obtiene la ecuación de índice de productividad para pozos de gas:

$$J = \frac{2\pi khT_{am}}{P_{am}T \left( \ln \left( \frac{r_e}{r_w} \right) + s + Dq_g \right)} \quad (2.72)$$

Con este índice de productividad se puede relacionar la pseudopresión de la celda con la pseudopresión de fondo fluyente, utilizando las ecuaciones analíticas para encontrar el radio efectivo (como la ec. (2.69)) y sustituyendo en (2.72)  $r_e$  por  $r_o$ . Finalmente obtener el gasto expresado como índice de productividad por la diferencia de pseudopresiones:

$$q_{g_{i,j}}^{n+1} = J'_{i,j}{}^{n+1} \left( P_{p_{i,j}}^{n+1} - P_{p_{wf,i,j}}^{n+1} \right) \quad (2.43)$$

Donde ya se han incluido los subíndices  $i,j$  para la celda y  $n$  para el nivel de tiempo.

### Tipo de condición de frontera entre pozos y las celdas del yacimiento

Dependiendo del tipo de condición de frontera en los pozos, de Newman (Gasto de la celda especificado) o de Dirichlet (presión de fondo fluyendo especificada), se requerirá involucrar a  $J$  en los cálculos de las pseudopresiones.

Si la condición de frontera es la de Dirichlet, el índice de productividad entra en los coeficientes de flujo centrales, de acuerdo a la ecuación:

$$a_{C_{i,j}}^n = a_E + a_W + a_S + a_N + J'_{i,j}{}^n + \alpha_{i,j}^n \quad (2.50)$$

con el vector de términos independientes expresado como:

$$d_{i,j}^n = \alpha_{i,j}^n P_{\rho i,j}^n - J_{i,j}^n P_{\rho w f i,j}^n \quad (2.51)$$

Si la condición de frontera es de Newman, no es necesario calcular J para actualizar los coeficientes centrales de flujo en cada iteración del yacimiento, pues en este caso  $a_{c i,j}^n$  no depende de J:

$$(a_{c i,j}^{n+1})_k = a_E + a_W + a_S + a_N + (\alpha_{i,j}^{n+1})_k \quad (2.73)$$

y los elementos del vector de términos independientes están expresados como:

$$(d_{i,j}^n)_k = (\alpha_{i,j}^n P_{\rho i,j}^n)_k + q_{i,j}^{n+1} \quad (2.74)$$

Una vez que se hubiere obtenido la distribución de pseudopresiones en el yacimiento, se calcula J para el gasto  $q_{i,j}^{n+1}$  y se obtiene la presión de fondo fluyente para la celda con:

$$P_{\rho w f i,j}^{n+1} = P_{\rho i,j}^{n+1} - \frac{q_{i,j}^{n+1}}{J_{i,j}^{n+1}} \quad (2.76)$$

IN CONSIDERATION

OF THE PROVISIONS OF THE ACT  
RELATIVE TO THE REGISTRATION OF  
MARRIAGES

AND THE PROVISIONS OF THE ACT  
RELATIVE TO THE REGISTRATION OF  
DEATHS

AND THE PROVISIONS OF THE ACT  
RELATIVE TO THE REGISTRATION OF  
BIRTHS

AND THE PROVISIONS OF THE ACT  
RELATIVE TO THE REGISTRATION OF  
MARRIAGES

## CAPÍTULO III

# MODELO DE FLUJO EN POZOS E INSTALACIONES DE RECOLECCIÓN

### III.1 MODELO DEL SISTEMA DE RECOLECCIÓN

Se ha modelado en diversos trabajos a la red o sistema de recolección en superficie como un conjunto de tuberías conectadas entre sí y a diversos elementos como son, pozos, separadores, compresores, intercambiadores de calor, etc. En nuestro caso sólo incluiremos tuberías, compresores, pozos y puntos de entrega, en virtud de las siguientes consideraciones:

- En los pozos sólo se producen hidrocarburos en fase gaseosa, sin presencia de agua, ni condensados, ni crudo.
- El gas nunca se condensa en la red.
- No existen separadores en la red (son innecesarios por ser el gas seco).
- No existen en la red válvulas ni accesorio alguno que pudiera causar pérdidas menores, adicionales a las caídas de presión en los tubos.
- No existen intercambiadores de calor.

Para el flujo en las tuberías se considera que:

- El gas es una mezcla de hidrocarburos con posible presencia de CO<sub>2</sub> y H<sub>2</sub>S, con composición y propiedades termodinámicas constantes durante toda la vida del yacimiento.
- El gas se aleja de la ecuación de gases ideales y su comportamiento P-V-T, está completamente descrito por la ley de estados correspondientes y el factor de compresibilidad dado por la correlación de Standing y Katz.(1942).
- Para un tramo de tubería determinada, la temperatura permanece constante.
- El gas pierde presión al viajar a través de una tubería debido a la fricción con la tubería y "fricción interna" viscosa; su reología es de un fluido newtoniano con viscosidad dependiente de la presión y la temperatura.
- Existe una función matemática, llamada correlación de flujo de gas en tuberías, que relaciona la presión a la entrada, a la salida, el diámetro y la longitud del tubo con el gasto volumétrico a condiciones estándar, que pasa a través de la tubería.
- Existe una ecuación que modela los compresores relacionando a la presión de entrada, de salida, características del gas y potencia del compresor con el gasto que pasa a través de él.

Una red de recolección puede modelarse como un conjunto de tuberías que transportan gas, unidas unas con otras en puntos específicos llamados nodos. Cada nodo debe de cumplir con la ecuación de continuidad, es decir que la sumatoria de gastos másicos que entren al nodo por las tuberías debe ser igual a cero, puesto que se considera que en cualquier punto de la red hay estado estacionario y régimen permanente. Véase la fig. 3.1

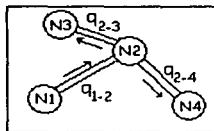


Figura 3.1.- Red de recolección representada por nodos y tubos. La sumatoria de gastos que entran y salen del nodo 2 debe ser igual a cero.

Ahora bien, la manera de modelar los pozos y puntos de entrega es también por medio de nodos, haciendo que cada nodo tenga la posibilidad adicional de tener un gasto



másico de entrada o de salida (este mismo desarrollo puede encontrarse en Stoner(1969)). Por tanto, la ecuación de continuidad expresada en palabras es:

$$\begin{array}{l} \text{Gasto que entra al} + \text{ Sumatoria de todos los gastos} = 0. \\ \text{Nodo} \qquad \qquad \qquad \text{que entran por tuberías} \\ \qquad \qquad \qquad \qquad \qquad \text{conectadas al nodo} \end{array} \quad (3.1)$$

El gasto que entra al nodo, si es negativo entrará a la red, mientras que si es positivo sale. Para los gastos de los pozos, se elige positivo si sale del nodo y negativo si entra.

Denotando por medio de  $Q_i$  al gasto en el nodo  $i$ -ésimo y  $q_{ij}$  como el gasto que va del nodo  $i$  al  $j$  entonces la ecuación (3.1) queda escrita:

$$Q_i + \sum_{j=1}^{n_i} q_{ij} = 0. \quad (3.2)$$

El número  $n_i$  en (3.2) denota el número de nodos que están unidos al nodo  $i$ . Por ejemplo, para el nodo 2 de la figura 3.1,  $i=2$ ,  $j$  toma los valores de 1,3 y 4. Cuando  $j=1$ ,  $q_{1 \rightarrow 2}$  es negativo, cuando  $j=3$  y  $j=4$ ,  $q_{2 \rightarrow 3}$  y  $q_{2 \rightarrow 4}$  son positivos, respectivamente.

Habíamos dicho que existía una ecuación relacionando el gasto que pasa a través de una tubería en función de la presión en el inicio, al final y otras características del tubo y del gas. Existen en la literatura, diferentes ecuaciones o correlaciones de flujo, como son la de Weymouth, Panhandle A, Panhandle B, Tian y Adewumi, Clinedinst, etc. Por el momento, nos basta saber que dichas ecuaciones tienen la forma general:

$$q_x = q_x(P_1, P_2, L, D, T, \delta_x, z(P, T), \mu_x, E, f(N_x, E)). \quad (3.3)$$

Donde  $P_1$  es la presión a la entrada del tubo (nodo origen) y  $P_2$  es la presión a la salida (nodo destino). En general, los conectores entre nodos pueden no ser sólo tuberías. Los compresores, según veremos en la sección III.6, pueden tratarse análogamente a las tuberías. Se proseguirá en el análisis considerando sólo tuberías, pero debe tenerse en mente que el modelo puede extenderse para incluir compresores y de hecho, así lo hemos hecho.

Si vemos que las variables de la función del gasto (ec. (3.3)) están fijas o son dependientes de la presión para un problema específico, podríamos decir que en la ecuación (3.2) sólo hay dos tipos de variables, que a saber son:

- Gastos en los nodos
- Presiones en los nodos

Reescribiendo (3.2), puede quedar expresada como:

$$Q_i + \sum_{j=1}^{n_i} q_{r_j}(P_i, P_j, L, D, T, \delta_r, z(P, T), \mu_r, E, f(N_r, \epsilon)) = 0. \quad (3.4)$$

En nuestro problema tenemos puntos de entrega con presión o gasto especificado, nodos (con gasto especificado igual a 0) y pozos con gasto especificado y presión como incógnita. Por tanto:

Para puntos de entrega:

- La presión es conocida y constante.
- El gasto es una incógnita de la red (gasto positivo, sale de la red)

Para nodos:

- El gasto es conocido e igual a cero.
- La presión es una incógnita de la red.

Para pozos:

- El gasto es conocido (gasto negativo, entra a la red)
- La presión es una incógnita de la red.

En forma vectorial se puede expresar la ecuación (3.4) como:

$$\begin{bmatrix} Q_1 + \sum_j^{n_1} q_x(P_1, P_j, L, D, T, \delta_x, z(P, T), \mu_x, E, f(N_R, \epsilon)) \\ Q_2 + \sum_j^{n_2} q_x(P_2, P_j, L, D, T, \delta_x, z(P, T), \mu_x, E, f(N_R, \epsilon)) \\ \vdots \\ Q_n + \sum_j^{n_n} q_x(P_n, P_j, L, D, T, \delta_x, z(P, T), \mu_x, E, f(N_R, \epsilon)) \end{bmatrix} = \bar{F}(\bar{X}) = \bar{0}. \quad (3.5)$$

Es decir que  $\mathbf{F}$  es una función vectorial de varias variables escalares.  $\mathbf{X}$  es el vector de incógnitas que hay que encontrar en la red. Por ejemplo, si tuviéramos la siguiente red:

- Nodo 1.- Pozo
- Nodo 2.- Pozo
- Nodo 3.- Nodo común y corriente
- Nodo 4.- Punto de entrega

Las incógnitas serían:

- $P_1$
- $P_2$
- $P_3$
- $Q_4$

Y el vector  $\mathbf{X}$  sería:

$$\mathbf{X} = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ Q_4 \end{bmatrix}. \quad (3.6)$$

Hay que notar que el orden de las incógnitas en el vector es arbitrario. Es posible resolver la ecuación (3.5) con el método de Newton-Raphson en su forma vectorial (Burden y cols. (1989)). Las ecuaciones de iteración del método son:

$$\bar{X}_{k+1} = \bar{X}_k + \Delta \bar{X}_k \quad (3.7)$$

$$J(\bar{X}_k) \Delta \bar{X}_k = -\bar{F}(\bar{X}_k) \quad (3.8)$$

donde J es la matriz jacobiana, dada por:

$$J(\bar{X}_k) = \begin{bmatrix} \frac{\partial F_1}{\partial X_1} & \frac{\partial F_1}{\partial X_2} & \dots & \frac{\partial F_1}{\partial X_n} \\ \frac{\partial F_2}{\partial X_1} & \frac{\partial F_2}{\partial X_2} & \dots & \frac{\partial F_2}{\partial X_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial X_1} & \frac{\partial F_n}{\partial X_2} & \dots & \frac{\partial F_n}{\partial X_n} \end{bmatrix} \quad (3.9)$$

De acuerdo a la definición de las funciones  $F_1, F_2, F_3, \dots, F_n$  definidas por (3.5), las parciales de la matriz jacobiana en (3.9) tienen las propiedades siguientes:

$$\frac{\partial F_i}{\partial P_j} = - \sum_{j \in (i,j) \in P} \frac{\partial q_r(P_i, P_j, L, D, T, \delta_r, z, \mu_r, E, f)}{\partial P_j} \quad (3.10)$$

$$\frac{\partial F_i}{\partial P_j} = - \frac{\partial q_r(P_i, P_j, L, D, T, \delta_r, z, \mu_r, E, f)}{\partial P_j} \quad \text{, si el nodo } j \text{ está conectado al nodo } i. \quad (3.11)$$

$$\frac{\partial F_i}{\partial P_j} = 0 \quad \text{, si el nodo } j \text{ no está conectado al nodo } i. \quad (3.12)$$

$$\frac{\partial F_i}{\partial Q_i} = 1, \quad (3.13)$$

$$\frac{\partial F_i}{\partial Q_j} = 0. \quad (3.14)$$

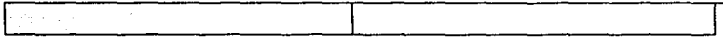
A continuación se presenta el algoritmo para encontrar las incógnitas de la red.

### III.2 ALGORITMO PARA EL SISTEMA DE ECUACIONES ALGEBRAICAS NO LINEALES DEL MODELO EN SUPERFICIE

Como procedimos en el capítulo del modelo del yacimiento, antes de formular el algoritmo, se enunciarán los datos y las incógnitas que son entradas y salidas, respectivamente, del proceso que lleva a cabo el algoritmo.

Tabla 3.1.- Datos del algoritmo 3.1

Propiedades de los tubos, en los nodos y del gas	Datos que son siempre conocidos para cualquier nivel de tiempo n+1
1.- $D_i \forall i \in \text{Tubos.}$	1.- $P_i^{n+1} \forall i \in P.E.$
2.- $L_i \forall i \in \text{Tubos.}$	2.- $q_j^{n+1} \forall j \in \text{Pozos.}$
3.- $E_i \forall i \in \text{Tubos.}$	3.- $q_j^{n+1} = 0 \forall j \in \text{Nodos.}$
4.- $T_{med_i} \forall i \in \text{Tubos.}$	Valores iniciales de las incógnitas (supuestos o calculados con base en información de los niveles de tiempo anteriores)
5.- $\varepsilon_i \forall i \in \text{Tubos.}$	
6.- $h_j \forall j \in (\text{Nodos} \cup \text{Pozos} \cup P.E.)$	1.- $q_j^{n+1}_{sup} \forall j \in P.E.*$
7.- $\delta_r$	2.- $P_i^{n+1}_{sup} \forall i \in \text{Nodos}*$
8.- $z(P,T)$	3.- $P_i^{n+1}_{sup} \forall i \in \text{Pozos}*$
9.- $\mu_r(P,T)$	



- Sup = supuesto

Tabla 3.2.- Incógnitas del algoritmo 3.1.

Presiones en los nodos y afuera de los estranguladores de los pozos; gastos en los puntos de entrega y en los pozos.

1.-  $q_j^{n+1} \forall j \in P.E.$

2.-  $P_i^{n+1} \forall i \in \text{Nodos}$

3.-  $P_i^{n+1} \forall i \in \text{Pozos}$

4.-  $q_i^{n+1} \forall i \in \text{Pozos}$

#### Algoritmo 3.1

- 1.-Especificar las propiedades de los elementos de la red y datos que son siempre conocidos para cualquier nivel de tiempo  $n+1$ .
- 2.-Proponer valores iniciales de las incógnitas. Formar con ellos el vector  $\bar{X}_0$  (una vez seleccionado el orden de las incógnitas en el vector, éste ya no puede cambiar)
- 3.-inicializar  $k=0$
- 4.-Calcular  $\bar{F}(\bar{X}_k)$  con:

$$F(\bar{X}_K) = \begin{bmatrix} Q_1 + \sum_j^n q_r(P_j, P_j, L, D, T, \delta_r, z(P, T), \mu_r, E, f(N_R, \epsilon)) \\ Q_2 + \sum_j^n q_r(P_{2i}, P_j, L, D, T, \delta_r, z(P, T), \mu_r, E, f(N_R, \epsilon)) \\ \vdots \\ Q_n + \sum_j^n q_r(P_{ni}, P_j, L, D, T, \delta_r, z(P, T), \mu_r, E, f(N_R, \epsilon)) \end{bmatrix} \quad (3.15)$$

5.- Calcular  $J(\bar{X}_K)$  con:

$$J(\bar{X}_K) = \begin{bmatrix} \frac{\partial F_1}{\partial X_1} & \frac{\partial F_1}{\partial X_2} & \dots & \frac{\partial F_1}{\partial X_n} \\ \frac{\partial F_2}{\partial X_1} & \frac{\partial F_2}{\partial X_2} & \dots & \frac{\partial F_2}{\partial X_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial X_1} & \frac{\partial F_n}{\partial X_2} & \dots & \frac{\partial F_n}{\partial X_n} \end{bmatrix} \quad (3.9)$$

6.- Resolver la ecuación matricial

$$J(\bar{X}_K) \Delta \bar{X}_K = -F(\bar{X}_K) \quad (3.8)$$

para el vector  $\Delta \bar{X}_K$  mediante algún método directo o iterativo.

7.- Efectuar la operación:

$$\bar{X}_{K+1} = \bar{X}_K + \Delta \bar{X}_K \quad (3.7)$$

8.- Si

$$|X_i^{K+1} - X_i^K| \leq tol \text{ para } i=1 \text{ hasta } n, \quad (3.16)$$

entonces terminar las iteraciones y devolver como valores de las incógnitas los elementos del vector  $\bar{X}_{K+1}$ , de lo contrario hacer  $k=k+1$  y regresar al paso 4.

### III.3 FLUJO DE GAS A TRAVÉS DE TUBERÍAS HORIZONTALES O LIGERAMENTE INCLINADAS

Como se mencionó previamente, para encontrar la caída de presión a través de una tubería horizontal o ligeramente inclinada es necesario tener una ecuación que relacione las características de la tubería, su inclinación, propiedades del gas y presión inicial y final con el gasto que pasa a través de ella. En la sección III.1 observamos que dicho gasto es una función de los siguiente parámetros:

$$q_x = q_x(P_1, P_2, L, D, T, \delta_x, z(P, T), \mu_x, E, f(N_R, \epsilon)) \quad (3.3)$$

Todas las ecuaciones que existen en la literatura (para una revisión ver Garaicochea y cols.(1991), Economides y cols.(1994), Ikoku(1980), Kumar(1980)) han sido derivadas de la primera ley de la termodinámica para sistemas abiertos, con flujo de gas real a través de un volumen de control (el interior del tubo). La diferencia que hay entre ellas se debe a consideraciones o simplificaciones particulares. No obstante, en casi todas se consideran propiedades promedio del gas a lo largo de todo el tubo. Muchas de estas ecuaciones tienen la forma:

$$q_x = C(L, D, T, \delta_x, z(P, T), \mu_x, E, f(N_R, \epsilon)) (P_1^2 - P_2^2)^{\frac{1}{2}} \quad (3.17)$$

Es decir, que el cuadrado del gasto es proporcional a la diferencia de los cuadrados de las presiones. En la ecuación (3.17) el subíndice 1 denota al nodo con mayor presión y 2 al nodo con menor presión. Si el flujo fuera de 2 a 1 se invierte el signo del gasto y el orden de las presiones en el paréntesis.

La forma de la ecuación (3.17) es conveniente para usarse en la ecuación (3.2). En este trabajo no se hicieron comparaciones entre diferentes correlaciones, ni el efecto que tienen en la simulación de el sistema acoplado. Se eligió programar tan sólo la correlación de Weymouth por ser la que da resultados cercanos a la de Clinedinst (ecuación al parecer muy exacta.)



**Correlación de Weymouth**

En la resolución de la red por el método de Newton utilizamos la correlación de Weymouth, en su forma especial para tuberías ligeramente inclinadas. La constante de la ecuación (3.17) resulta ser:

$$C(L, D, T, \delta_g, z(P, T)) = \frac{\left( 31.5027 \left[ \frac{\text{MSCF/d} \cdot ft^{\frac{1}{2}}}{\text{in}^{\frac{8}{3}} \cdot R^{\frac{1}{2}}} \right] \right)^{T_{\text{atm}}}}{P_{\text{ima}}} \left( \frac{D^{\frac{16}{3}}}{\delta_g Le T_{\text{med}} z(P_{\text{med}}, T_{\text{med}})} \right)^{\frac{1}{2}} \quad (3.18)$$

Como se observa, la constante C no es función de la viscosidad del factor de fricción ni del número de Reynolds. Por otro lado, en (3.18) aparece una variable denominada Le, que es la longitud equivalente de una tubería inclinada. Cuando el ángulo de inclinación (con respecto a la horizontal) es cero, la longitud equivalente tiende de la longitud real de la tubería. Esta longitud equivalente se define como:

$$Le = \frac{e^S - 1}{S} \quad (3.19)$$

S es una medida de la inclinación de la tubería. Se define como:

$$S = \frac{0.0375 \delta_g \Delta h}{Z(P_{\text{med}}, T_{\text{med}}) T_{\text{med}}} \quad (3.20)$$

Las unidades de las ecuaciones (3.19) y (3.20) no están claras; sin embargo utilizando la diferencia de alturas en [pie] y una temperatura media en [R] en (3.20) debe de obtenerse la longitud equivalente en [pie] de (3.19). (Obsérvese que para una diferencia de alturas igual a cero S también es cero y Le en este caso no está definida. No obstante, puede demostrarse que cuando S tiende a cero, Le tiende a L si se encuentra el límite de (3.19) cuando S->0, usando la regla de l'Hôpital.)

La correlación completa es:

$$q_g = \frac{\left( 31.5027 \left[ \frac{\text{MSCF/d} \cdot f_T^{\frac{1}{2}}}{\text{in}^{\frac{8}{3}} \cdot R^{\frac{1}{2}}} \right] \right) T_{\text{atm}}}{P_{\text{ima}}} \left( \frac{D^{16}}{\delta_g L e T_{\text{med}} z (P_{\text{med}} T_{\text{med}})} \right)^{\frac{1}{2}} \cdot (P_1^2 - e^S P_2^2)^{\frac{1}{2}} \quad (3.21)$$

La presión final de la tubería ( $P_2$ ) está afectada por  $e^S$ . Así el gasto calculado con (3.21) es menor cuando la tubería va cuesta arriba ( $S > 0$ ) y es mayor cuando la tubería va cuesta abajo ( $S < 0$ ).

La presión media de la tubería es la media geométrica, definida como:

$$P_{\text{med}} = \frac{2}{3} \frac{P_1^3 - P_2^3}{P_1^2 - P_2^2} \quad (3.22)$$

El factor de fricción  $z$  debe evaluarse a esta presión media.

### III.4 FLUJO DE GAS A TRAVÉS DE TUBERÍAS DE PRODUCCIÓN

Los pozos son el medio físico que comunica al yacimiento con la superficie. Cualquier alteración que se desee hacer en las condiciones iniciales del sistema roca-fluidos del yacimiento debe hacerse a través de los pozos, desde explotar al yacimiento así como inyectar fluidos para proyectos de recuperación secundaria y mejorada. En la producción de gas (y en general de aceite y cualquier fluido del yacimiento) el análisis de las caídas de presión desde la zona de disparos hasta la boca del pozo es de gran importancia, ya que es aquí donde se consume la mayor parte de la energía de presión de los fluidos, además de haber también importantes caídas de temperatura debido a la expansión de los mismos.

Existen un número de correlaciones para calcular la caída de presión en un intervalo de tubería vertical o inclinada. Puede consultarse para una revisión de ellas el libro de Ikoku(1980). Todas parten de la ecuación de continuidad, la primera ley de la

termodinámica para sistemas abiertos y una ecuación de estado para gases reales. La diferencia de una con otra suele ser las consideraciones que toman para el flujo.

Dentro de las principales correlaciones podemos citar :

- Cullender y Smith
- Sukkar y Cornell
- Tian y Adewumi

Las ecuaciones de Cullender y Smith y de Tian y Adewumi consideran cambios en la temperatura del gas, aunque pueden ser utilizadas también para alguna temperatura constante promedio y esto es precisamente la consideración que tomamos en nuestro modelo, tanto en las tuberías de producción como en las tuberías horizontales en superficie. Discutiremos aquí sólo la ecuación de Tian y Adewumi por ser la única que se implementó en el programa.

### **Correlación de Tian y Adewumi(1994)**

La correlación de Tian y Adewumi(1994) es una ecuación semiteórica para gases secos, basada en la primera ley de la termodinámica para sistemas abiertos y la ecuación de estado de gases reales de Benedict-Webb-Rubin. Tiene la ventaja de haber sido integrada analíticamente desde el inicio hasta el fin de la tubería, así que no es necesario, a diferencia de la correlación de Cullender y Smith o de Sukkar-Cornell de integrar numéricamente, usando el método trapecial o la regla de Simpson. Tiene también la ventaja de servir tanto para tuberías inclinadas y verticales como horizontales. Al parecer es comparable en resultados a la correlación de Sukkar-Cornell, de menor exactitud que Cullender-Smith, pero los autores dan pruebas de que muchas veces se introduce un mayor error al no considerar el cambio de energía cinética que considerar  $z$  y  $T$  constantes, así que puede dar algunas ventajas balanceando exactitud con tiempo de cálculo.

La ecuación que proponen los autores en unidades consistentes es:

$$\left( \frac{zR_u T}{2M_m g \sin \alpha} + \frac{d}{f} \right) \ln \left( \frac{\frac{f \rho u}{2DA^2} + \frac{M_m^2 g P_2^2 \sin \alpha}{z^2 R_u^2 T^2}}{\frac{f \rho u}{2DA^2} + \frac{M_m^2 g P_1^2 \sin \alpha}{z^2 R_u^2 T^2}} \right) - \frac{d}{f} \ln \left( \frac{P_2^2}{P_1^2} \right) + L = 0. \quad (3.23)$$

donde:  $P_1$  y  $P_2$  son la presión a la entrada y salida de la tubería, respectivamente,  $\rho u$  es el gasto másico de la tubería y  $z$  es el factor de compresibilidad evaluado a la temperatura media  $T$  de la tubería y presión media  $P$ , que se definen:

$$T = \frac{T_2 + T_1}{2} \quad (3.24)$$

$$P = \frac{P_2 + P_1}{2}$$

Con esta  $z$  se calcula el número de Reynolds medio (calculando la viscosidad a las condiciones medias). El factor de fricción  $f$  se calcula con este número de Reynolds y la rugosidad relativa.

Rearreglando la ecuación (3.23), haciendo  $P_1 = P_{wf}$  y  $P_2 = P_h$  queda:

$$P_{wf}^2 = \frac{z^2}{\eta_3} \exp \left\{ \frac{\ln \left( \frac{P_{wf}^2}{P_h^2} \right) \frac{\eta_2}{f} + \eta_1}{\eta_6 z + \frac{\eta_2}{f}} \right\} \left( \eta_1 f + \frac{\eta_2}{z^2} \right) - \eta_1 f. \quad (3.25)$$

donde:

$$\rho u = \left( 4.007635933 \times 10^{-4} \left[ \frac{\text{kg} \cdot \text{d}}{\text{s} \cdot \text{MSCF}} \right] \right) \delta_k q_k. \quad (3.26)$$

$$\eta_1 = \left( 94.58699252 \times 10^6 \left[ \frac{\text{in}^5}{\text{m}^5} \right] \right) \frac{\rho u^2}{2DA^2}. \quad (3.27)$$

$$\eta_2 = \left( 181.7364023 \left[ \frac{\text{kg}^2 \cdot \text{R}^2 \cdot \text{in}^4 \cdot \text{mol}^2}{\text{s}^2 \cdot \text{m}^5 \cdot \text{lb}^2 \cdot \text{g}^2} \right] \right) \frac{M_m^2 P_h^2 \sin \alpha}{T^2}. \quad (3.28)$$

$$\eta_3 = \left( 181.7364023 \left[ \frac{\text{kg}^2 \cdot \text{R}^2 \cdot \text{in}^4 \cdot \text{mol}^2}{\text{s}^2 \cdot \text{m}^5 \cdot \text{lb}^2 \cdot \text{g}^2} \right] \right) \frac{M_m^2 \sin \alpha}{T^2}. \quad (3.29)$$

$$\eta_5 = \left( 0.0254 \left[ \frac{m}{in} \right] \right) D, \quad (3.30)$$

$$\eta_6 = \left( 235.417374 \left[ \frac{m \cdot kg}{kmol \cdot R} \right] \right) \frac{T}{M_m \sin \alpha}, \quad (3.31)$$

$$\eta_7 = \left( 0.3048 \left[ \frac{m}{ft} \right] \right) L. \quad (3.32)$$

La ecuación (3.23) es implícita con respecto a  $P_{wh}$ . La manera de resolverla es evaluar todas las constantes, de  $\eta_1$  a  $\eta_7$ , suponer una  $P_{wh}$  inicial, obtener  $P_{wh}^2$  de la ecuación (3.23) y ahora emplear el valor calculado como el supuesto en la iteración siguiente, hasta que se cumpla la tolerancia deseada (Método de sustituciones sucesivas.)

Esta ecuación se utiliza por las ventajas siguientes:

- Considera una temperatura media constante, lo que permite que  $z(P,T)$  sea evaluada con un polinomio  $z=Pol(P)$ ,  $T=T_{med}=cte$ .
- Da resultados excelentes aun cuando no se divida la tubería en varios tramos, como lo requiere la correlación de Cullender-Smith, que hace los cálculos más tardados.

A continuación se da el pseudocódigo para encontrar la presión de fondo fluvente con la correlación de Tian y Adewumi:

```
'Pth[1b/pg2], Pwf_ini[1b/pg2], qg[MPCS/d], l[pie], T[R], dr[adm], alfa[grados], D[pg]
'Eps[pg], Ppc[1b/pg2], Tpc[R], nit[adm], tol[1b/pg2]
```

Función Pwf\_Tian\_Adewumi (Pth como Doble, Pwf\_ini como Doble, qg como Doble, l como Doble, T como Doble, dr como Doble, alfa como Doble, D como Doble, Eps como Doble, Ppc como Doble, Tpc como Doble, nit como Entero, tol como Doble) como Doble

```
Dim G_m como Doble
Dim A como Doble
Dim Plc como Doble
Dim Plc_ant como Doble
Dim Pm como Doble
```

Dim zv como Doble

Dim fv como Doble

Dim sen\_alfa como Doble

Dim Mm como Doble

Dim Mugv como Doble

Dim Nre como Doble

Dim E1,E2,E3,E4,E5,E6,E7,E8,E9,E10 As Double

Mm = dr \* Mmaire

sen\_alfa = Sin(alfa \* 0.01745329252)

Plc = Pwf\_ini ^ 2

A = 0.7853981634 \* D ^ 2

G\_m = 0.0004007635933 \* dr \* qg

E1 = 94586992.53 \* G\_m ^ 2 / (2 \* D \* A)

E2 = 181.7364023 \* Mm ^ 2 \* Pth ^ 2 \* sen\_alfa / T ^ 2

E3 = 181.7364023 \* Mm ^ 2 \* sen\_alfa / T ^ 2

E5 = 0.0254 \* D

E6 = 235.417374 \* T / (Mm \* sen\_alfa)

E7 = 0.3048 \* 1

E8 = 50127.54113 \* G\_m / D

E9 = Eps / (D \* 3.7065)

E10 = (Eps / D) ^ 1.1098 / 2.8257

Para i = 1 hasta nit

Plc\_ant = Plc

Pm = (Pth + Plc ^ 0.5) / 2

zv = z\_Gopal(Pm / Ppc, T / Tpc)

Mugv = Viscosidad\_Lee (Pm, T, zv, dr)

Nre = E8 / Mugv

fv = (-0.8685889638 \* Log(E9 - 2.19110252 / Nre \* Log(E10 + 5.8506 / Nre ^ 0.8981))) ^ -2

Plc = zv ^ 2 / E3 \* (-E1 \* fv + (E1 \* fv + E2 / zv ^ 2) \* Exp((Log(Plc / Pth ^ 2) \* E5 / fv + E7) / (E6 \* zv + E5 / fv)))

Ka = 0.999

Plc = (1 - Ka) \* Plc\_ant + Ka \* Plc

Si Abs(Plc\_ant - Plc) <= tol \* Abs(Plc\_ant ^ 0.5 + Plc ^ 0.5) Entonces

Pwf\_Tian\_Adewumi = Plc ^ 0.5

Salir Función

Fin del si

Siguiente i

Pwf\_Tian\_Adewumi = -1 'Señal de que no convergió  
'al número de iteraciones deseadas

Fin de la Función

### III.5 FLUJO DE GAS A TRAVÉS DE ESTRANGULADORES

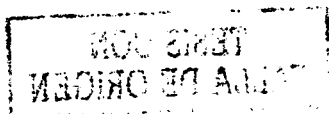
Un estrangulador es un aparato que restringe el flujo de los fluidos del yacimiento hacia la superficie. Va colocado justo después de la cabeza del pozo y antes de la línea de descarga. El estrangulador crea una caída de presión de la cabeza a la línea de flujo, creando así una contrapresión que influye a la presión de fondo fluyendo y por tanto el gasto del yacimiento a los pozos. Se puede encontrar una introducción al tema en el libro de los ingenieros Garaicochea y cols.(1991) y también en el de Economides y cols.(1994) . Una breve revisión bibliográfica para los casos más generales de flujo multifásico se encuentra en Schoitzer(1994).

Los estranguladores desempeñan un papel importante en modelos de simulación simultánea de yacimientos con instalaciones superficiales. La introducción de estranguladores afecta la interacción entre estos dos modelos, como las investigaciones y resultados de Schoitzer(1994) lo muestran.

De acuerdo al orden que se seguirá al hacer las iteraciones en el modelo, es necesario encontrar la presión en la cabeza de los pozos (denotado  $P_{th}$ ), dados un gasto y una presión al principio de la línea de flujo ( $P_e$ ), características del estrangulador (diámetro y coeficiente de descarga), temperatura en la cabeza ( $T_{th}$ ) del pozo o temperatura media a través del estrangulador (según lo requiera el modelo), densidad relativa del gas, índice adiabático a la temperatura en cuestión y sus propiedades pseudocríticas.

Conceptualmente, es similar el flujo de gas a través de estranguladores al de tuberías, con la diferencia de que la longitud del estrangulador es despreciable y las altas velocidades a través del estrangulador llegan a ser mayores o iguales a las del sonido, por lo que se desarrolla un flujo sónico o crítico (Garaicochea y cols.(1991), Economides y cols.(1994)).

Las correlaciones existentes para modelar el flujo de gas a través de estranguladores toman en cuenta esta característica, haciendo que el gasto esté en función de las variables arriba citadas y sea constante cuando el cociente de presiones  $P_e/P_{th}$  sea menor al cociente o razón crítica de presiones, dado por:



$$\left(\frac{P_e}{P_{th}}\right)_c = \left(\frac{2}{k+1}\right)^{\frac{k}{k-1}}. \quad (3.33)$$

De acuerdo a Schoizer(1994), las caídas de presión en los estranguladores son muy sensibles a pequeños cambios en los gastos, especialmente para diámetros de estrangulador pequeños. En el capítulo VI investigaremos esta sensibilidad y los efectos que tiene en el comportamiento de la convergencia. Para las correlaciones de flujo a través de estranguladores, el mismo autor recomienda lo siguiente:

- La correlación debe ser precisa para los dos tipos de flujo (crítico y subcrítico.)
- Debe predecir con precisión la frontera entre ellos.
- Debe haber una transición suave entre los dos tipos de flujo.

### Correlación de Szilas(1975)

Una correlación que cumple con los criterios arriba mencionados y tiene la ventaja de no requerir el cálculo del factor de compresibilidad del gas, es la ecuación de Szilas(1975):

$$q_g = \left( 3.505 \left[ \frac{MSCF / d \cdot 64^2 \cdot R^{1/2}}{in^2} \right] \right) D_{64}^2 \frac{P_{th}}{P_{atm}} C_d \left[ \frac{k}{\delta_g T_{th} (k+1)} \left( \left( \frac{P_e}{P_{th}} \right)^{\frac{2}{k}} - \left( \frac{P_e}{P_{th}} \right)^{\frac{k+1}{k}} \right) \right]^{\frac{1}{2}}. \quad (3.34)$$

donde  $q_g$  es el gasto de gas en [MPCS/d],  $D_{64}$  es el diámetro del estrangulador en 64avos de pulgada,  $C_d$  es el coeficiente de descarga (mayor a 0 y menor o igual a 1),  $P_{th}$  es la presión en la cabeza del pozo,  $P_e$  es la presión al comienzo de la línea de flujo,  $k$  es el índice adiabático del gas a la temperatura  $T_{th}$ .

La ecuación (3.34) es válida cuando

$$\left(\frac{P_e}{P_{th}}\right) \leq \left(\frac{P_e}{P_{th}}\right)_c. \quad (3.35)$$

Para  $P_e/P_{th}$  mayor a la  $(P_e/P_{th})_c$  se utiliza la ecuación (3.34) sustituyendo  $P_e/P_{th}$  por  $(P_e/P_{th})_c$ :



$$q_r = \left( 3.505 \left[ \frac{MSCF / d \cdot 64^2 \cdot R^{1/2}}{in^2} \right] \right) D_{64}^2 \frac{P_{th}}{P_{atm}} C_d \left[ \frac{k}{\delta_r T_{th} (k+1)} \left( \left( \frac{P_e}{P_{th}} \right)_C^{\frac{2}{k}} - \left( \frac{P_e}{P_{th}} \right)_C^{\frac{k+1}{k}} \right) \right]^{\frac{1}{2}} \quad (3.36)$$

En necesario rearreglar la ecuación (3.34) para obtener de ella la presión en la cabeza ( $P_{th}$ ). La ecuación (3.34) queda implícita con respecto a  $P_{th}$  y es necesario resolverla por medio de algún método numérico. A continuación se da un algoritmo conveniente para obtener la  $P_{th}$  utilizando la ecuación de Szilas.

1.- Calcular  $P_{th}$  suponiendo que hay flujo crítico con la ecuación (3.36), es decir:

$$P_{th} = \frac{q_r P_{atm}}{\left( 3.505 \left[ \frac{MSCF / d \cdot 64^2 \cdot R^{1/2}}{in^2} \right] \right) D_{64}^2 C_d} \left[ \frac{k}{\delta_r T_{th} (k+1)} \left( \left( \frac{P_e}{P_{th}} \right)_C^{\frac{2}{k}} - \left( \frac{P_e}{P_{th}} \right)_C^{\frac{k+1}{k}} \right) \right]^{\frac{1}{2}} \quad (3.37)$$

donde  $(P_e/P_{th})_C$  se calcula con la ecuación (3.33).

2.- Probar si la  $P_{th}$  calculada cumple con la ecuación (3.35). En caso de que no y  $P_{th}$  calculada sea mayor a  $P_e$ , tómesese la  $P_{th}$  calculada como valor inicial para resolver la ec. (3.34) iterativamente. Si  $P_{th}$  calculada es menor o igual a  $P_e$ , se toma este último como valor inicial de  $P_{th}$ .

3.- Resolver la ecuación (3.34) con el método de bisección, regla falsa, Newton-Raphson o el método de sustituciones sucesivas.

### Criterio de convergencia

El autor del presente trabajo realizó un análisis de la convergencia y número de iteraciones requeridas por diferentes métodos numéricos para resolver la ecuación (3.37). En general, el número de iteraciones necesario para llegar a una tolerancia dada es función del conjunto de parámetros de la ec. (3.36) (diámetro del estrangulador, Temperatura, densidad relativa del gas, coeficiente de descarga), del gasto, del índice adiabático y, muy importante, del valor inicial de  $P_{th}$  al comenzar a iterar.

Algunos resultados son los siguientes:

- Método de sustituciones sucesivas (sin la técnica iterativa de Coats y cols[1968]): Le toma del orden de 30 a 50 iteraciones para converger. En ocasiones es inestable.
- Método de Bisección: Le lleva del orden de 8 a 20 iteraciones. Estas cifras son funciones exactas de la tolerancia. No es recomendable cuando la solución queda cerca de las  $(P_{th})_c$ , por partir en mitades el intervalo  $[(P_{th})_c, P_e]$
- Método de la regla falsa: Le lleva de 70 a 200 iteraciones converger.
- Método de Newton: Le lleva del orden de 3 a 22 iteraciones convergir.

De todos los métodos mencionados en el punto anterior, por experiencia propia del autor del presente trabajo, el que tiene mayor velocidad de convergencia es el de Newton Raphson. Por esta razón fue el que se seleccionó para este trabajo. Co él se debe resolver la ecuación:

$$F(P_{th}) = q_r - \left( 3.505 \left[ \frac{MSCF / d \cdot 64^2 \cdot R^{1/2}}{in^2} \right] \right) D_{64}^2 \frac{P_{th}}{P_{nm}} C_d \left[ \frac{k}{\delta_r T_{th} (k+1)} \left( \left( \frac{P_e}{P_{th}} \right)^{\frac{2}{k}} - \left( \frac{P_e}{P_{th}} \right)^{\frac{k+1}{k}} \right) \right]^{\frac{1}{3}} = 0 \quad (3.38)$$

Al parecer se puede acelerar la convergencia modificando la ecuación de iteración del método multiplicando por un factor al cociente  $F/F'$ :

$$P_{th} = P_{th} - \alpha \frac{F(P_{th})}{F'(P_{th})} \quad (3.39)$$

Un valor al parecer aceptable de  $\alpha$  parece ser 1.9. Con esto se logra entrar a la tolerancia en menos iteraciones. Desgraciadamente los valores óptimos de  $\alpha$  son función de los parámetros mencionados más arriba y de no seleccionarse adecuadamente, la velocidad de convergencia se deteriora, comienza a ser inestable o bien diverge.

A partir de la primera parte del segundo término de la ec. (3.38) se define una constante  $C1$  (en  $MPCS/d/(lb/pg^2)$ ) como:

$$C_1 = \left( 3.505 \left[ \frac{MSCF / d \cdot 64^2 \cdot R^{1/2}}{in^2} \right] \right) \frac{D_{gs}^2 C_d}{P_{am}} \left[ \frac{k}{\delta_r T_{th} (k+1)} \right]^{1/2}. \quad (3.40)$$

De las ecuaciones (3.38) y (3.40) se deduce que el número de iteraciones (denotado  $N_{it}$ ) requerido por el método de Newton es función de  $C_1$ , de  $P_e$ , de  $k$ ,  $q_0$  y  $P_{th}$  inicial. Matemáticamente:

$$(N_{it})_{NEWTON} = N_{it} \left( C_1, P_e, k, q_0, \frac{P_{thini}}{P_e} \right). \quad (3.41)$$

$P_{th}/P_e$  es el inverso del cociente de presiones; está acotado por:

$$1 \leq \frac{P_{th}}{P_e} \leq \left( \frac{2}{k+1} \right)^{\frac{k}{1-k}}. \quad (3.42)$$

Dados un  $P_e$ ,  $k$  y  $P_{th}/P_e$  iniciales, se puede evaluar el número de iteraciones para diferentes  $C_1$  y  $q$ . Para ciertos valores de  $C_1$  grandes y  $q$  pequeños, se tiene problema con la diferencia (ec. 3.28):

$$\left[ \left( \frac{Pe}{P_{th}} \right)^{\frac{2}{k}} - \left( \frac{Pe}{P_{th}} \right)^{\frac{k+1}{k}} \right]^{1/2}. \quad (3.43)$$

ya que para esos casos el número dentro del paréntesis se hace negativo. La interpretación física es que, siendo  $C_1$  una cantidad que mide la facilidad con que fluye el gas a través del estrangulador, para  $q$  pequeños y  $C_1$  grandes la caída de presión a través del estrangulador tiende a ser cero, y esto de alguna manera causa la anomalía con la diferencia (3.43). En estos casos, en cuanto se detecte el error (que puede ser previsto evaluando el signo de la cantidad (3.43) antes de evaluar la raíz cuadrada) la subrutina debe terminar y devolver un valor de  $P_{th}$  igual a  $P_e$ .

Esta conclusión está respaldada por un experimento numérico que realizó el autor del presente trabajo en hojas de Excel, donde se comparan los valores de  $P_{th}$  calculados con el método de Newton-Raphson y Bisección. En el experimento se formó una matriz con valores de  $C_1$  en el eje de las ordenadas y  $q$  el de las abscisas y siendo cada elemento



Un segundo experimento apoya todavía más la conclusión de que el método de Newton-Raphson es el mejor en términos exactitud y velocidad de convergencia. Se determinaron los tiempos de cálculo del método de Newton y el método de bisección para varias tolerancias y un gran número de combinaciones de  $C_1$  y  $q$  en la región de convergencia. Posteriormente se calculó, un promedio (estadístico) de los tiempos para todas las combinaciones de  $C_1$  y  $q$  para una tolerancia especificada. Los resultados se muestran en las figuras 3.3 y 3.4. Sólo se reportan estos dos métodos ya que el de regla falsa y el de sustituciones sucesivas tuvieron problemas para la convergencia.

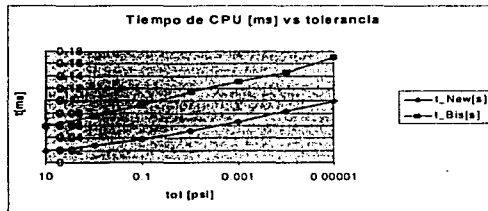


Figura 3.3: Tiempo de cálculo del método de Newton y de Bisección para la ecuación de Szilas en función de la tolerancia.

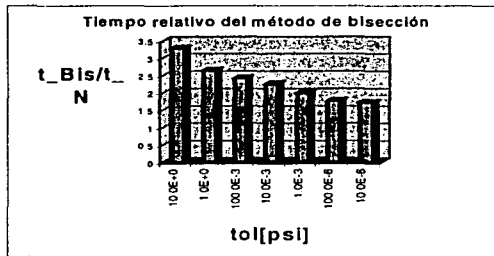


Fig 3.4: Tiempos relativos del método de Bisección con respecto al de Newton Raphson para resolver la ec. 3.16

Este segundo experimento puso en relieve los resultados siguientes:

- Para ambos métodos los tiempos de cálculo son casi proporcionales al logaritmo de la tolerancia, lo que hace que se logre más precisión para un

incremento reducido de tiempo de cálculo. Un valor de tolerancia entre 0.1 y 0.01 parece adecuado para el problema.

- El método de bisección para tolerancias grandes se tarda el triple que el de Newton, mientras que para tolerancias pequeñas se tarda 1.5 veces.

Por último, proporcionamos el pseudocódigo de Visual Basic para encontrar la presión en la cabeza del pozo utilizando la correlación de Szilas. Se indican en los comentarios las unidades que tiene cada uno de los argumentos. La presión que devuelve la función está en [lb/pg²].

```

'qg[MPCS/d],Pe[lb/pg2],dr [adm], T[R],K [adm], D[pg/64],Cd[adm],tol[lb/pg2],nit[adm]
Función Pth_Szilas(qg como Doble, Pe como Doble, dr como Doble, T como Doble, K como
Doble, D como Doble, Cd como Doble, tol como Doble, nit como Entero) como Doble
Dim Pth como Doble
Dim Coc_c como Doble
Dim Coc_s como Doble
Dim cte como Doble

Dim Pth_ant como Doble
Dim Pec como Doble
Dim Pthc como Doble
Dim Dif como Doble
Dim E1 como Doble
Dim E2 como Doble

cte = qg * 14.69445752 / (3.505 * D ^ 2 * Cd) * (dr * T * (K - 1) / K) ^ 0.5
Coc_c = (2 / (K + 1)) ^ (K / (K - 1))
Pth = cte * (Coc_c ^ (2 / K) - Coc_c ^ (1 + 1 / K)) ^ -0.5
Coc_s = Pe / Pth
Si Coc_s < Coc_c Entonces
    Pth_est_Szilas1 = Pth
    Salir Función
Fin del Si
Si Pth <= Pe Entonces Pth = Pe * 1.001
cte = -(3.505 * D ^ 2 * Cd) / 14.69445752 * (dr * T * (K - 1) / K) ^ -0.5
Para i = 1 hasta nit
    Pth_ant = Pth
    Coc_s = Pe / Pth
    E1 = Coc_s ^ (2 / K)
    E2 = Coc_s ^ ((K + 1) / K)

    Si E1 < E2 Entonces
        Pth_est_Szilas1 = Pe
        Salir Función

```

```

    Fin del Si
    Dif = (E1 - E2) ^ 0.5
    Pth = Pth - (qg / cte + Pth * Dif) / (Dif + (-2 / K * E1 + (K + 1) / K
    * E2) / Dif)
    Si Abs(Pth_ant - Pth) <= tol Entonces
        Pth_est_Szilas1 = Pth
        Salir Función
    Fin del Si
    Siguiente i
        Pth_Szilas = -1
    Fin de la Función

```

### III.6 COMPRESIÓN DEL GAS

Como se mencionó en la sección III.4, las caídas de presión a través de la tubería de producción son grandes. Estas caídas, aunadas a la declinación que tiene el yacimiento por agotamiento natural, rápidamente hacen que éste sea incapaz de mantener la presión deseada en la cabeza de algunos o de todos los pozos (presión de la línea de recolección, por ejemplo.) La producción declina rápidamente y se hace necesaria la perforación de nuevos pozos para mantenerla. Sin embargo, una manera de mantener la producción y cumplir con los requerimientos de presión de entrega, es empleando compresores en puntos determinados de la red de recolección. Los compresores elevan la presión del gas y le dan energía para que venza la resistencia que las líneas de recolección y/o transporte le oponen. Con esto es posible desplazar el gas a distancias muy grandes, tanto como estaciones de compresión y/o potencia haya.

En nuestro modelo se hace necesario incluir compresores, pues cualquier análisis realista de la declinación de un yacimiento de gas debe considerar la compresión para prolongar la vida productiva de los pozos, antes o después de que éstos sean incapaces de hacer llegar el gas con presión suficiente a las líneas de recolección o bien a troncales de varias líneas. Incluso, al considerar dichos compresores, es posible simular ya no tan sólo la producción y recolección del gas, sino también su transmisión e incluso su distribución, pues ya no hay restricción en las distancias que puede recorrer el gas, siempre y cuando se le dote de energía suficiente.

La ecuación que utilizamos en el presente trabajo está derivada de un análisis termodinámico de un gas que experimenta un proceso de compresión adiabático. Muchos autores han utilizado en sus análisis ecuaciones ligeramente diferentes, que se

conforman mejor al tipo de compresor en particular que se esté utilizando. Dichas ecuaciones suelen ser provistas por el fabricante y en su mayoría sólo cambian por constantes que toman en cuenta diferentes pérdidas de energía (eficiencias.) En nuestro caso consideramos el caso más sencillo que es el de la compresión adiabática de un gas real con una sola eficiencia, de la máquina.

Puede demostrarse, que si durante la compresión adiabática del gas se mantiene la ecuación

$$Pv^k = cte, \quad (3.44)$$

donde  $k$  es el índice adiabático del gas y  $v$  es el volumen específico del mismo, la potencia necesaria para comprimir un gasto constante de gas (expresado a c.s.) está expresada como:

$$W\& = \left( 8.072429247 \times 10^3 \left[ \frac{Hp \cdot d}{R \cdot MSCF} \right] \right) \frac{k}{k-1} q_r Tz(P_1, T) \left[ \left( \frac{P_2}{P_1} \right)^{\frac{k-1}{k}} - 1 \right], \quad (3.45)$$

donde  $W\&$  es la potencia,  $P_1$  es la presión a la entrada (o de succión) y  $P_2$  es la presión a la salida.

Despejando el gasto de la ecuación (3.45) e introduciendo la eficiencia  $\eta$  del compresor se obtiene:

$$q_r = \frac{\left( 12,387.84472 \left[ \frac{R \cdot MSCF}{d \cdot Hp} \right] \right) \eta (k-1) W\&}{k Tz(P_1, T) \left[ \left( \frac{P_2}{P_1} \right)^{\frac{k-1}{k}} - 1 \right]}. \quad (3.46)$$

Con (3.46) se puede calcular el gasto que pasa a través de un compresor con una potencia y eficiencia especificadas. Esta ecuación nos permite tratar a los compresores en la red análogamente a las tuberías: Con presiones de entrada y salida, propiedades del gas y características del "conector" de los nodos se determina el gasto que pasa de uno a otro, sólo que en este caso la presión de salida es mayor que la de entrada y las características propias del aparato son diferentes.



## CAPÍTULO IV

# CONCEPTOS FUNDAMENTALES PARA EL ACOPLAMIENTO DE LOS MODELOS NUMÉRICOS

### IV.1 DESCRIPCIÓN DEL SISTEMA YACIMIENTO-POZOS-RED DE RECOLECCIÓN

En los capítulos pasados se trató a cada uno de los componentes del sistema por separado, es decir el yacimiento con condiciones iniciales y de frontera impuestas por elementos ajenos a él, el flujo a través de tuberías verticales, flujo a través de estranguladores y flujo a través de redes de recolección. Ahora se procederá a integrar todos los elementos analizados separadamente para formar un sistema formado por todos estos componentes, que modele la explotación de un yacimiento de gas seco desde el yacimiento hasta los puntos de entrega.

Un volumen de fluido, al viajar desde la cavidad de los poros de la roca hasta el punto de entrega experimentará caídas de presión progresivas del yacimiento hasta el pozo (incluidos los efectos de daño y aceleración cerca del pozo), del extremo inferior de la tubería de producción hasta el estrangulador, a través del estrangulador y del estrangulador hasta el punto de entrega. La figura 4.1 muestra un esquema de los elementos que conforman al sistema yacimiento-pozos-red de recolección

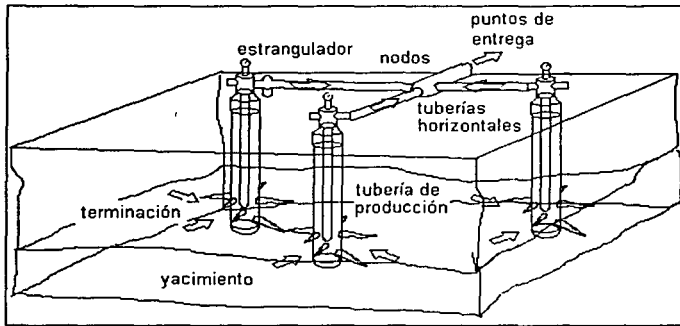


Fig. 4.1.- Esquema del sistema yacimiento-pozos-sistema de recolección

El modelo pretende calcular estos cambios de presión que experimenta el gas en su recorrido, teniendo cuenta rigurosa de interferencia entre pozos en el yacimiento y la declinación de la producción, a diferencia del análisis nodal con ecuaciones de comportamiento de afluencia (Brown y Lee (1995)).

Los elementos del sistema son:

- Yacimiento
- Terminaciones (considerando daño)
- Tuberías de producción
- Estranguladores
- Tuberías horizontales que conforman la red de recolección
- Nodos que unen a las tuberías horizontales
- Puntos de entrega

No se considerarán restricciones al flujo adicionales como válvulas y otros aditamentos. En realidad el modelo deja fuera separadores (por tratarse de gas seco que se considera que jamás se condensa), intercambiadores de calor y cualesquiera otros aditamentos superficiales.

Se puede modelar el uso de compresores fijando la presión de los puntos de entrega igual a la presión de entrada de los compresores.

Las consideraciones generales del modelo son :

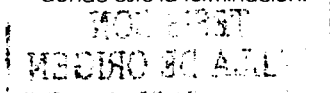
### Yacimiento

- El yacimiento está saturado por agua irreductible inmóvil y gas seco de composición constante, cuyo comportamiento PVT está descrito por la ley general de gases reales y su coeficiente de compresibilidad está dado por la correlación de Standing y Katz.
- El yacimiento tiene fronteras irregulares y tiene celdas de tamaño variable con una altura dada constante para cada celda.
- No existe flujo en la dirección vertical del yacimiento.
- La permeabilidad horizontal en X y Y es una permeabilidad media, que tomará en cuenta de intercalaciones lutíticas.
- El flujo en el yacimiento es isotérmico.
- Las condiciones de frontera del yacimiento son las de yacimiento cerrado. No existe entrada de agua ni ningún otro fluido al yacimiento.
- El único mecanismo de producción es la expansión del sistema roca-fluidos, aunque usualmente será despreciable el índice de empuje de la expansión de la roca en comparación con la del gas.

### Terminaciones

- En las celdas donde exista terminación habrá flujo representado por la ecuación de Forsheimer, con un coeficiente de aceleración D, especificado como dato.
- El radio de drene para cada pozo se considerará menor al  $\Delta x$  y al  $\Delta y$  de la celda

...donde esté la terminación:



- La presión de fondo fluyendo es solamente función de la presión media de la celda donde está la terminación, el gasto del pozo y propiedades de la roca para esa celda (que definen al pseudoíndice de productividad.)
- Existe para cada terminación un factor de daño  $S$ , que es la sumatoria de todos los daños que pudiere haber, es decir daño real, por terminación, por inclinación del pozo, penetración parcial, etc.
- Se produce por tuberías de producción solamente. No hay flujo por el espacio anular.

### **Tuberías y estranguladores**

- Existe régimen permanente tanto en tuberías de producción, como en estranguladores y tuberías horizontales. Durante un intervalo de tiempo dado, los gastos en los puntos de entrega, estranguladores, y tuberías de producción son constantes, y para la siguiente iteración dan un salto súbito a valores diferentes (típicamente menores).
- Tanto en tuberías horizontales, como en estranguladores y tuberías de producción se supondrá temperatura constante para cada elemento. Estas temperaturas quedarán especificadas antes de la simulación y son parte de los datos o información necesaria.
- Al cerrar pozos, el cierre se hará en el fondo del aparejo de producción y no en la cabeza del pozo. No hay por tanto efectos de almacenamiento.
- En los estranguladores el flujo puede ser crítico o subcrítico. Bajo ninguna circunstancia puede haber flujo en sentido contrario al del yacimiento hacia superficie.

### **Puntos de entrega**

- Cualquier nodo que no sea tipo pozo y esté unido a una sola tubería será un punto de entrega, de donde se extraerá gas de la red.
- Las presiones en los puntos de entrega son constantes o bien están en función del tiempo, si es que se impone una condición de presión.

- Los gastos en los puntos de entrega son constantes o bien están en función del tiempo, si es que se impone la condición de gasto.

Al inicio de la explotación, el yacimiento tiene una distribución de presiones y se considera que los pozos, si es que estaban cerrados, comienzan a producir en el instante en que inicia la simulación el yacimiento. Durante la simulación se irán añadiendo nuevos pozos, nuevas terminaciones y nuevos troncales a la red de recolección superficial (de acuerdo al programa de operaciones.) Se deja producir al yacimiento de acuerdo a los diámetros de estrangulador especificados y condiciones de frontera en los puntos de entrega. Estas condiciones en dichos puntos, conjuntamente con las características de todos y cada uno de los elementos del sistema y distribución inicial de presiones en el yacimiento, determinan y fijan los gastos que aporta cada pozo, durante un intervalo de tiempo, a la red superficial y los gastos que tiene cada punto de entrega.

### **Esquemas para el acoplamiento de los modelos yacimiento-instalaciones y condiciones de frontera**

Antes de proseguir es necesario revisar los esquemas que existen para acoplar los modelos. Los tipos de esquema pueden ser implícitos o explícitos (Byer (2000)):

- Esquema explícito: En éste las ecuaciones del yacimiento y las de la red de recolección (incluyendo en ésta a los pozos) se resuelven cada una por separado y hay una "interfaz" entre una y otra, expresada por una ecuación que relacione las presiones de los nodos con las presiones de fondo fluyente o gastos de los pozos. En el presente trabajo ésta es la ecuación del índice de productividad. En este esquema se dice que yacimiento e instalaciones forman dos dominios separados (un dominio está formado por una región en el espacio en la cual, al discretizarse, su conjunto de ecuaciones generadas por sus puntos se resuelve de manera independiente de las ecuaciones de puntos de otras regiones.) Se resuelve alternativamente entre un dominio y otro y sólo la interfaz es la que los une.
- Esquema implícito: En éste se resuelven de manera simultánea las ecuaciones del yacimiento y las de las instalaciones. Se mezclan todas las ecuaciones para

formar un solo conjunto. Se dice que instalaciones y yacimiento forman un solo dominio.

### Condiciones de Frontera:

Las condiciones de frontera entre las instalaciones y el yacimiento puede ser de dos tipos:

- Condiciones de frontera de Newman: Con estas condiciones se especifica el gasto a las celdas del yacimiento que sean sumideros (donde exista una terminación) y el yacimiento, con estas restricciones impuestas debe encontrar la nueva distribución de presiones para el intervalo de tiempo siguiente. Como resultado de estas condiciones de frontera el yacimiento "entrega" a la red superficial presiones de fondo fluente.
- Condiciones de frontera de Dirichlet: Con estas condiciones de frontera entre instalaciones y yacimiento, se especifica la presión de fondo fluente de las celdas con terminación y es necesario que el yacimiento encuentre los gastos y la nueva distribución de presiones para el nuevo intervalo de tiempo.

En el presente trabajo se utilizó una formulación explícita con condiciones de frontera de Dirichlet. El algoritmo 4.1, que se presentará en este mismo capítulo, es una manera de llevar a cabo el acoplamiento con esta formulación y condición de frontera.

### Flujo de Información entre un elemento y otro

La función que desempeña en la iteración global cada elemento del sistema en este trabajo es la siguiente (Formulación explícita, condiciones de frontera de Dirichlet):

- Yacimiento: Definidos una distribución de presiones de la iteración anterior y presiones de fondo fluendo, proporciona como resultado gastos en los pozos y la nueva distribución de presiones en las celdas, misma que se utilizará en la iteración siguiente.
- Tuberías horizontales: Dados unos gastos y presión en la cabeza del pozo, calcula presiones de fondo fluente.

- Estranguladores: A partir de un gasto y presión a la salida del estrangulador ( $P_e$ ) calcula la presión antes del estrangulador, que se asigna como presión en la cabeza del pozo ( $P_{th}$ ).
- Red superficial: Con los gastos de los pozos y presiones en los puntos de entrega, calcula las presiones antes de los estranguladores.

El siguiente esquema ilustra de manera simplificada el flujo de información.

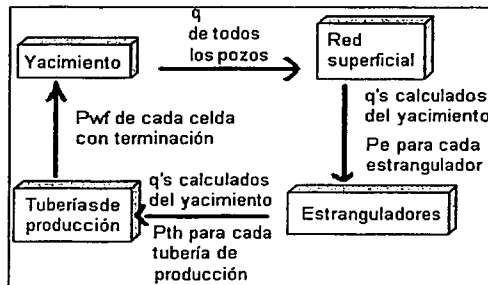


Fig. 4.2 Flujo de información entre los elementos del sistema yacimiento-pozos-red superficial, utilizando condiciones de frontera de Dirichlet en los pozos

La figura 4.2 parece mostrar un ciclo donde no hay cambios en el yacimiento y siempre calcula los mismos gastos y las mismas presiones de fondo fluyendo; sin embargo, se deben introducir a la red superficial unos gastos supuestos para que se inicie el ciclo. Los gastos que calcula el yacimiento son generalmente distintos a los que se suponen. Es de hecho, como se verá adelante, propósito de este intercambio de información crear un ciclo que permita comparar gastos supuestos y gastos calculados ("gastos de equilibrio", ver Schoizer (1994)) por el yacimiento, hasta que converjan estos últimos y haya una coherencia entre las caídas de presión en las celdas del yacimiento y presiones de fondo fluyendo con los gastos de gas necesarios para causar dichas caídas de presión, con la condición adicional de que al producir los pozos a esos gastos, el gas, después de haber pasado por la red superficial, llegue a los puntos de entrega con la presión que se especificó para ese intervalo de tiempo (o visto de manera inversa, dada una presión en

los puntos de entrega, calcular los gastos en los pozos y en dichos puntos y la nueva distribución de presiones.)

## IV.2 ALGORITMO GLOBAL

Un algoritmo que resuelva el problema de simular la explotación de yacimientos de gas o de aceite tomando en cuenta pozos y red superficial no es nuevo. El primer artículo en presentar el método numérico más simple (y el que nosotros seguimos,) precisamente para yacimientos de gas es el de Dempsey y cols.(1971). Para yacimientos de aceite el primer estudio apareció en Emmanuel y Ranney(1981). Estudios recientes con técnicas mucho más avanzadas para el caso general de flujo multifásico en el yacimiento y en superficie son los de Schoizer y Aziz(1994) y Byer y cols.(1998), (1999).

La figura 4.3 muestra las subiteraciones de la iteración global, y ésta a su vez, junto con el almacenamiento de la información para ese intervalo de tiempo, dentro del ciclo que va desde el tiempo 1 hasta el tiempo final de la simulación.

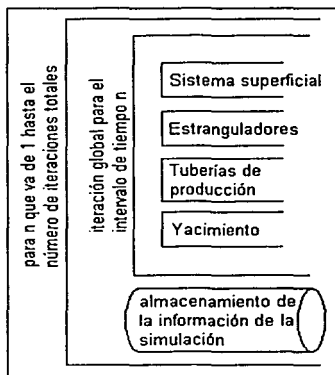


Figura 4.3 Diagrama de bloques de la iteración global

Este algoritmo es el más sencillo de todos: en él se incluye al yacimiento dentro del "balance" de gastos entre el yacimiento y la red. En algoritmos más complejos, tendientes a reducir el tiempo de cálculo sacando al yacimiento del balance, se emplean técnicas como la descomposición del yacimiento en "subdominios" y "preacondicionadores" (Byer



y cols.(1998), (1999), Schiozer y Aziz (1994)). Estas técnicas escapan a los alcances de esta tesis.

Refiriéndonos a la fig. 4.3, hay que notar que cada bloque, exceptuando el almacenamiento de la información después de un tiempo  $n$ , es una subiteración que con datos o información provenientes de otros bloques o de iteraciones globales pasadas(ver fig. 4.2), está buscando la convergencia:

- La subiteración del Sistema superficial corresponde al algoritmo 3.1 presentado en el capítulo III .
- La subiteración de los estranguladores corresponde al algoritmo del método de Newton-Raphson para resolver la ecuación de Szilas, discutida en el capítulo 3.
- La subiteración de tuberías de producción corresponde a los algoritmos de flujo a través de tuberías verticales discutidos en el capítulo 3.
- La subiteración en el yacimiento corresponde al algoritmo 2.1, presentado en el capítulo II.

Como se ha procedido en capítulos anteriores, se enuncian primero los datos e incógnitas del algoritmo y luego se presenta el algoritmo.

Tabla 4.1.- Datos del Algoritmo 4.1

Propiedades del Yacimiento	Propiedades de la red	Propiedades de los estranguladores
1.- $h_{i,j} \forall (i,j) \in Yac.$	1.- $D_i \forall i \in Tubos\_hor.$	1.- $D_{64} \forall i \in Est.$
2.- $K_{x,i,j} \forall (i,j) \in Yac.$	2.- $L_i \forall i \in Tubos\_hor.$	2.- $T_{thi} \forall i \in Est.$
3.- $K_{y,i,j} \forall (i,j) \in Yac.$	3.- $E_i \forall i \in Tubos\_hor.$	3.- $C_{di} \forall i \in Est.$
4.- $\Delta X_{i,j} \forall (i,j) \in Yac.$	4.- $T_{medi} \forall i \in Tubos\_hor.$	Propiedades del gas seco
5.- $\Delta Y_{i,j} \forall (i,j) \in Yac.$	5.- $\varepsilon_i \forall i \in Tubos\_hor.$	
6.- $\varphi_{i,j} \forall (i,j) \in Yac.$	6.- $h_j \forall j \in (Nodos \cup Pozos \cup P.E.)$	1.- $\delta_g$
7.- $C_{f,i,j} \forall (i,j) \in Yac.$	Propiedades de las tuberías de Producción	2.- $P_{rc}$
8.- $S_{i,j} \forall (i,j) \in Yac.$	1.- $D_i \forall i \in TP.$	3.- $T_{rc}$
9.- $D_{i,j} \forall (i,j) \in Yac.$	2.- $L_i \forall i \in TP.$	4.- $z(P,T)$
10.- $T_{yac}$	3.- $E_i \forall i \in TP.$	5.- $\mu_g(P,T)$
11.- $P_{atm}, T_{atm}$	4.- $T_{medi} \forall i \in TP.$	6.- $k(P,T)$
12.- $a_{\varepsilon,i,j} \forall (i,j) \in Yac. \text{ ec. 2.46}$	5.- $\varepsilon_i \forall i \in TP.$	Parámetros de la iteración
13.- $a_{w,i,j} \forall (i,j) \in Yac. \text{ ec. 2.47}$	6.- $\alpha_{inc,i} \forall i \in TP.$	1.- $tol_{yac}$
14.- $a_{s,i,j} \forall (i,j) \in Yac. \text{ ec. 2.48}$	Propiedades de las terminaciones	2.- $tol_{Re dSup}$
15.- $a_{N,i,j} \forall (i,j) \in Yac. \text{ ec. 2.49}$	1.- $D_i \forall i \in Term.$	3.- $tol_{It,Global}$
	2.- $S_i \forall i \in Term.$	4.- $\Delta t^n$
	3.- $r_{wi} \forall i \in Term.$	

Tabla 4.2.- Valores de las variables supuestas, de la iteración pasada o bien iniciales.

Variables iniciales o de la iteración pasada	Valores conocidos de la red	Valores supuestos
1.- $P_{r,i,j}^n \forall (i, j) \in Yac.$ 2.- $P_{i,j}^n \forall (i, j) \in Yac.$ 3.- $V_{p,i,j}^n \forall (i, j) \in Yac.$	1.- $P_i^{n+1} \forall i \in P.E.$ 2.- $q_j^{n+1} = 0 \forall j \in Nodos.$	1.- $q_j^{n+1} \sup \forall j \in P.E.*$ 2.- $P_i^{n+1} \sup \forall i \in Nodos*$ 3.- $P_i^{n+1} \sup \forall i \in Pozos *$ 4.- $q_i^{n+1} \sup \forall i \in Pozos *$

\*Sup=Supuesto

Tabla 4.3.- Incógnitas del algoritmo 4.1.

Distribución de propiedades en el yacimiento	Presiones en la red y en los pozos	Gastos en la red y en los pozos
1.- $P_{r,i,j}^{n+1} \forall (i, j) \in Yac.$ 2.- $P_{i,j}^{n+1} \forall (i, j) \in Yac.$ 3.- $V_p^{n+1} \forall (i, j) \in Yac.$	1.- $P_i^{n+1} \forall i \in Nodos$ 2.- $P_r^{n+1} \forall i \in Est$ 3.- $P_{ih}^{n+1} \forall i \in Est$ 4.- $P_{wf}^{n+1} \forall i \in T.P.$	1.- $q_j^{n+1} \forall j \in P.E.$ 2.- $q_i^{n+1} \forall i \in Tubo\_hor$ 3.- $q_i^{n+1} \forall i \in Pozos$
Propiedades en la cercanía de los pozos		
1.- $J_i^{n+1} \forall i \in Term$ 2.- $r_{oi}^{n+1} \forall i \in Term$		

**Algoritmo 4.1**

- 1.-Introducir todos los valores de las tablas 4.1 y 4.2
- 2.-Inicializar  $k=0$
- 3.- $K=k+1$
- 4.-Ejecutar el algoritmo 3.1; obtener los resultados listados en la tabla 3.2

Tabla 3.2

Presiones en los nodos y afuera de los estranguladores de los pozos; gastos en los puntos de entrega y en los pozos.
<ol style="list-style-type: none"> <li>1.- <math>(q_j^{n+1})_k \forall j \in P.E.</math></li> <li>2.- <math>(P_i^{n+1})_k \forall i \in \text{Nodos}</math></li> <li>3.- <math>(P_r^{n+1})_k \forall i \in \text{Pozos}</math></li> <li>4.- <math>(q_i^{n+1})_k \forall i \in \text{Pozos}</math></li> </ol>

- 5.-Resolver la ecuación de Szilas con el método visto en la sección III.5 ; Obtener las presiones en la cabeza de los pozos  $(P_{ir}^{n+1})_k$  para todos los pozos.
- 6.-Calcular las presiones de fondo fluyente  $(P_{wf}^{n+1})_k$  con alguna correlación, como la de Tian y Adewumi o Cullender y Smith.
- 7.- Ejecutar el algoritmo 2.1 para obtener la distribución de presiones en el yacimiento y gastos de los pozos, acordes a las  $(P_{wf}^{n+1})_k$  calculadas en el paso 4º; tabla 2.2.

Tabla 2.2

DISTRIBUCION DE PROPIEDADES EN EL YAC. Y GASTOS EN LOS POZOS	INTERNAS O AUXILIARES EN LOS CALCULOS DEL YACIMIENTO
1.- $P_{i,j}^{n+1} \forall (i, j) \in Yac.$ , ec.5.53	1.- $\alpha_{i,j}^{n+1} \forall (i, j) \in Yac.$ , ec. 2.44
2.- $q_{i,j}^{n+1} \forall (i, j) \in Yac.$ , ec.2.43	2.- $J_{i,j}^{n+1} \forall (i, j) \in Yac.$ , ec. 2.43
3.- $P_{i,j}^{n+1} \forall (i, j) \in Yac.$	3.- $a_{c,i,j}^n \forall (i, j) \in Yac.$ , ec. 2.50
4.- $V_{\rho_{i,j}^{n+1}} \forall (i, j) \in Yac.$	4.- $d_{i,j}^n \forall (i, j) \in Yac.$ , ec. 2.51

8.- Comparar los gastos de los pozos obtenidos en el yacimiento con los gastos de los pozos introducidos como supuestos a la red superficial. Si el valor absoluto de todos y cada uno de los elementos de la diferencia de ambos vectores, es menor que la tolerancia especificada, terminar el ciclo, de lo contrario calcular los nuevos gastos de los pozos para la red superficial como una función de los gastos recién obtenidos en el yacimiento y los que se habían supuesto. Regresar al paso 3.

### IV.3 TÉCNICA ITERATIVA

En las iteraciones del algoritmo 4.1 se necesitan introducir valores para los gastos de los pozos para que al final de cada iteración se obtengan gastos recalculados de éstos. Se mencionó además en el paso 8 del mismo algoritmo, que los valores de los gastos que se introducen en la iteración siguiente son función de los recién obtenidos. Esta relación funcional, que llamaremos G, no está preestablecida y el diseñador del simulador debe proponerla.

Si se considera que, para cada pozo individual, e ignorando a los otros pozos, el nuevo gasto obtenido es una función del viejo gasto procesado por los pasos 3, 4 y 5 del algoritmo 4.1 (aunque esto, en rigor, no es cierto, puesto que los gastos de los otros pozos causan interferencia en el yacimiento, lo cual modifica a los gastos del pozo en cuestión), puede expresarse como:

$$q^*_i = f(q_{i,k}) \text{ para } i=1 \text{ hasta no. de pozos.} \quad (4.1)$$

Donde  $q^*_i$  denota el gasto recién obtenido para el pozo  $i$ . La función  $G$  más sencilla es la que hace el nuevo gasto supuesto igual al recién calculado:

$$q_{i,k+1} = G(q_{i,k}) = f(q_{i,k}) \quad (4.2)$$

Esta ecuación (4.2) tiene la forma:

$$x_{k+1} = f(x_k), \quad (4.2)$$

que es la fórmula del método de sustituciones sucesivas. Coats(1968) analiza cualitativamente la convergencia de este método y llega a resultados muy útiles para este trabajo que fueron usados para diseñar un método adaptativo que acelera la convergencia del algoritmo 4.1 .

El problema de la convergencia de la ec. (4.2) es obtener una serie de valores  $x_k$  que se acerquen a una solución finita

$$x_{CONV} = \lim_{k \rightarrow \infty} x_k . \quad (4.3)$$

si es que el límite existe. Para este valor al cual converge  $x$  (denotado  $x_{CONV}$ ),

$$x_{CONV} = f(x_{CONV}). \quad (4.4)$$

Si representamos como  $\epsilon_k$  al error en la iteración  $k$ -ésima, tenemos por definición:

$$x_k = x_{CONV} + \epsilon_k \quad (4.5)$$

Sustituyendo de acuerdo a 4.5 para  $x_k$  y  $x_{k+1}$  en 4.2 tenemos:

$$x_{CONV} + \epsilon_{k+1} = f(x_{CONV} + \epsilon_k). \quad (4.6)$$

Desarrollando 4.6 en series de Taylor alrededor del punto  $x_{CONV}$  . tenemos que:

$$f(x_k) = \sum_{n=0}^{\infty} \frac{f^{(n)}(x_{CONV})}{n!} ((x_{CONV} + \varepsilon_k) - x_{CONV})^n = \sum_{n=0}^{\infty} \frac{f^{(n)}(x_{CONV})}{n!} \varepsilon_k^n. \quad (4.7)$$

Si  $\varepsilon_k$  es pequeño, puede truncarse la serie en el segundo término, para obtener:

$$f(x_k) = f(x_{CONV}) + f'(x_{CONV})\varepsilon_k \quad (4.8)$$

Sustituyendo 4.8 y 4.4 en 4.6 queda:

$$x_{CONV} + \varepsilon_{k+1} = x_{CONV} + \varepsilon_k f'(x_{CONV}), \quad (4.9)$$

que es lo mismo que:

$$f'(x_{CONV}) = \frac{\varepsilon_{k+1}}{\varepsilon_k}. \quad (4.10)$$

Es decir que, si (4.2) converge, la razón de los errores en  $k+1$  y en  $k$  es constante e igual a la razón de cambio de la función  $f$  con respecto a  $x$ , valuada en el valor al cual converge  $x$ . Esta relación no siempre se cumple a la perfección por que, aunque  $\varepsilon_k$  sea pequeña, los términos que hemos ignorado de (4.7) pueden tener algún peso; sin embargo la experimentación que se explicará en el capítulo VI muestra que (4.10), con pequeñas variaciones, es válida.

La condición de convergencia se obtiene a partir de (4.10), y es que los errores absolutos sucesivos sean menores a los anteriores, es decir:

$$|f'(x_{CONV})| < 1. \quad (4.11)$$

Para los casos en que  $|f'(x_{CONV})|$  sea cercana o mayor 1, la convergencia se puede acelerar con otra forma de la función  $G$ , en que se ponderan el valor inicial supuesto y el obtenido:

$$x_{k+1} = G(x_k) = \omega f(x_k) + (1 - \omega)x_k. \quad (4.12)$$

donde  $\omega$  es un factor de poderación. Haciendo un manejo análogo al que se hizo para obtener (4.10), con base en (4.6), se obtiene:

$$\frac{\varepsilon_{k+1}}{\varepsilon_k} = \omega f' + 1 - \omega. \quad (4.13)$$

El valor óptimo de  $\omega$  que resultará en la la velocidad de convergencia máxima de (4.12), es aquél que haga el error cero:

$$\frac{\varepsilon_{k+1}}{\varepsilon_k} = 0. \quad (4.14)$$

con

$$\omega = \frac{1}{1 - f'}. \quad (4.16)$$

Las ecuaciones (4.10) y (4.16) permiten una serie de "observaciones cualitativas", según explica Coats(1968):

- Si  $0 < f' < 1$  entonces (4.10) converge y la convergencia es monotónica. Se dice que la convergencia es monotónica cuando la serie de valores  $x_k$  se aproximan a  $x_{CONV}$  por un solo lado. En este caso todos los  $\varepsilon_k$  tienen el mismo signo. La ecuación (4.16) muestra que para este caso la  $\omega$  óptima será mayor a la unidad.
- Si  $f' > 0$  entonces la ecuación (4.10) divergirá, pero se puede obtener la convergencia con una  $\omega$  negativa.
- Si  $-1 < f' < 0$  entonces la convergencia tendrá un comportamiento oscilatorio, es decir que los errores  $\varepsilon_k$  tendrán signos alternados y la omega óptima que acelerará la convergencia estará entre 0.5 y 1.
- Finalmente, si  $f' < -1$  4.10 divergirá, pero se puede obtener la convergencia con (4.12) con una  $\omega$  entre 0 y 0.5.

Las ecuaciones (4.10) y (4.16) son en verdad útiles. Coats no lo menciona, pero el autor del presente trabajo encontró que, en teoría de cumplirse (4.10) a la perfección, se puede obtener el valor convergido de  $x_{CONV}$  en sólo 3 iteraciones, ya que de las definiciones de  $\varepsilon_k$ ,  $\varepsilon_{k+1}$ ,  $\varepsilon_{k+2}$  sustituidas en la ec. (4.10) aplicada a  $k$  y  $k+1$  se obtiene:



$$\frac{\varepsilon_{k+2}}{\varepsilon_{k+1}} = \frac{\varepsilon_{k+1}}{\varepsilon_k} = f'(x_{CONV}); \quad (4.17)$$

$$\frac{x_{k+2} + x_{CONV}}{x_{k+1} + x_{CONV}} = \frac{x_{k+1} + x_{CONV}}{x_k + x_{CONV}}. \quad (4.18)$$

Despejando  $x_{CONV}$  de (4.18):

$$x_{CONV} = \frac{x_{k+1}^2 - x_{k+2} \cdot x_k}{-x_{k+2} + 2x_{k+1} - x_k}. \quad (4.19)$$

Este valor obtenido con (4.19) no es exactamente el valor convergido, pero da muy buena aproximación de  $x_{CONV}$  y además permite estimar la  $\omega$  óptima. Teniendo una  $\omega$  nos es ya posible utilizar (4.12) que converge mucho más rápido que (4.10).

Con base en estas propiedades numéricas del método de sustituciones sucesivas, se introdujo el siguiente método para acelerar la velocidad de convergencia del algoritmo 4.1:

### Algoritmo 4.2

1.- Hacer 3 iteraciones del algoritmo (4.1) con:

$$q_{ik+1} = G(q_{ik}) = f(q_{ik}). \quad (4.2)$$

2.- Calcular los valores estimados para los gastos convergidos de cada pozo con la ec. (4.19):

$$q_i^{n+1}{}_{CONV} = \frac{q_{i2}^2 - q_{i3} \cdot q_{i1}}{-q_{i1} + 2q_{i2} - q_{i3}}. \quad (4.20)$$

3.- Estimar los errores  $\varepsilon_1$  y  $\varepsilon_2$  para cada pozo con base en:

$$\varepsilon_{1i} = q_i^{n+1}{}_{CONV} - q_i^{n+1}. \quad (4.21)$$

$$\varepsilon_{2i} = q_i^{n+1} \text{ CONV} - q_i^{n+1} \text{ 2} \cdot \quad (4.22)$$

4.-Calcular  $f'(q_i^{n+1} \text{ CONV})$  para cada pozo:

$$f'(q_i^{n+1} \text{ CONV}) = \frac{\varepsilon_{1i}}{\varepsilon_{2i}} \cdot \quad (4.23)$$

5.-Calcular  $\omega$  para cada pozo:

$$\omega_i = \frac{1}{1 - f'(q_i^{n+1} \text{ CONV})} \cdot \quad (4.24)$$

6.- Ajustar todos los gastos de los pozos a los valores estimados convergidos calculados con (4.20).

7.-Continuar iterando con el algoritmo (4.1), con los gastos supuestos para cada pozo calculados con la ec. (4.25):

$$q_{ik+1} = \omega_i f(q_{ik}) + (1 - \omega) q_{ik} \cdot \quad (4.25)$$

# CAPÍTULO V

## DESARROLLO DEL SISTEMA DE CÓMPUTO

### V.1 PARTES DEL PROGRAMA

El sistema de cómputo lleva a cabo las operaciones necesarias para resolver las ecuaciones numéricas presentadas en los capítulos 1, 2, 3 y 4. Los algoritmos y métodos vistos en esos capítulos se llevan a cabo con funciones y subrutinas programadas en Visual Basic 6.0. Podríamos dividir sucintamente en 3 partes nuestro programa:

- Interfaz con el usuario.
- Kernel de procesamiento de la información.
- Sistema de almacenamiento y recuperación de la información.

#### **Interfaz con el usuario**

En este trabajo hemos denominado Interfaz del usuario al conjunto de ventanas y objetos gráficos para la entrada de información, que procesará o almacenará el simulador en archivos. También tiene la función de desplegar los resultados generados por el procesamiento de la información y lo almacenado en archivos.

### Kernel de procesamiento de la información

Hemos denominado el Kernel o núcleo al conjunto de subrutinas que se encargan de llevar a cabo cálculos (esto es, ejecutar los algoritmos), para resolver la ecuación de difusividad del yacimiento, encontrar la distribución de gastos y presiones en la red y balancear los gastos del yacimiento con los de la red. No están directamente relacionados con la entrada de datos y la información que requieren puede obtenerse de la interfaz o de la lectura de archivos. Durante la ejecución de estas subrutinas se genera nueva información (la información de la corrida), que puede mostrarse y/o almacenarse.

### Almacenamiento y recuperación de la información

La tercera categoría de subrutinas está formada por aquéllas encargadas de crear archivos de información de los editores de la interfaz y de la historia de la corrida. Se encargan también de la lectura de estos archivos y asignan la información a variables.

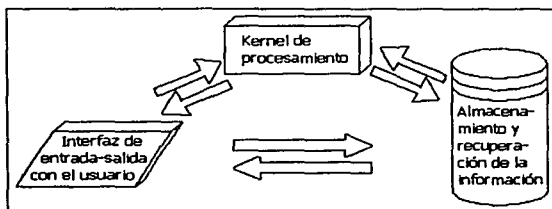


Figura 5.1.- Partes de un simulador

## V.2 ORGANIZACIÓN DE LA INFORMACIÓN

### Programación estructurada

El estilo de programación del simulador "SIMPREG" es estructurado, es decir con estructuras o tablas de información que define el usuario y después asigna a variables. Estas variables pueden ser invocadas por un sólo nombre, no obstante contienen valores

de "propiedades" múltiples (variables definidas en la tabla). Por ejemplo, el tipo definido por el usuario "Tipo\_gas" enseudocódigo podría ser:

```
Define Tipo Tipo_gas
```

```
    Ppc como doble
```

```
    Tpc como doble
```

```
    Dg como doble
```

```
Fin del Tipo
```

En este ejemplo se está indicando que una variable de tipo "Tipo\_gas" tiene propiedades de presiónseudocrítica (Pp), temperaturaseudocrítica (Tpc) y densidad específica (Dg), todas de doble precisión.

De esta manera, información que tiene relación entre sí puedè agruparse y tratarse como una unidad.

Las subrutinas y funciones del programa hacen operaciones con estas estructuras, crean variables de este tipo, o modifican la información de alguna de ellas definida previamente. Las estructuras son "pasadas" como argumentos a la subrutina o función. Por ejemplo, en la función que calculará el factor de compresibilidad de un gas con base en las propiedades de un gas, su presión y su temperatura, en vez de estar declarada como:

```
Función z(Ppc como doble,Dg como doble, Tpc como doble, P como doble,T  
como doble) como doble
```

Puede quedar declarada de la manera:

```
Función z(G como Tipo_gas, P como doble,T como doble) como doble.
```

De esta forma, en la variable G se está "pasando" ya como argumentos a la función tanto a la presiónseudocrítica del gas, como su temperaturaseudocrítica e incluso a la densidad específica.

### **Programación orientada a objetos**

En la programación orientada a objetos se tiene esta misma idea de agrupar lógicamente a un grupo de variables. Sin embargo la programación orientada a objetos posee muchas más propiedades como son el encapsulamiento, la herencia, polimorfismo, etc. Ciertamente desde hace tiempo se viene usando la programación orientada a objetos para simular sistemas de recolección y transporte. Uno de los primeros reportes es de Takacs y Turzo(1993). Los últimos avances en materia de simulación de yacimientos haciendo uso de la programación orientada a objetos ha estado marcada por Nogaret(1996) y más importante para nuestro caso de acoplamiento yacimiento-instalaciones superficiales, Byer y cols.(1998) Todos ellos utilizan técnicas numéricas y de programación muy por encima de nuestro trabajo, pero pueden ser revisados como bibliografía para tener idea del estado del arte de la simulación con programación orientada a objetos.

En nuestro caso, de hecho, no utilizamos la programación orientada a objetos. En ocasiones el tipo de variables declaradas como estructuras y a las estructuras mismas las llamaremos por comodidad objetos, aunque propiamente no lo sean.

Tenemos, pues, organizada la información que procesa el simulador en estructuras. A continuación se hace un listado en pseudocódigo de ellas. Cabe mencionar que este listado no es completo, sino que contiene sólo las propiedades más importantes y omite detalles que no son relevantes sino para la codificación verdadera.

## **V.2.1 ORGANIZACIÓN DE LA INFORMACIÓN EN EL YACIMIENTO**

La idea básica de crear un objeto "yacimiento" es que éste esté compuesto de un conjunto (arreglo) de objetos celdas, cada una conteniendo información individual para cada celda del yacimiento. Las propiedades de las celdas serán  $\Delta x$ ,  $\Delta y$ , altura de la celda, permeabilidad en x, permeabilidad en y, compresibilidad de la roca y porosidad, entre otras.

Así el objeto celda está definido como:

Tipo Celda	
Dx como doble	'Delta X de la celda en [pie]
Dy como doble	'Delta Y de la celda en [pie]
h como doble	'Altura de la celda en [pie]
Kx como doble	'Permeabilidad de la celda en X en [mD]
Ky como doble	'Permeabilidad de la celda en Y en [mD]
Cr como doble	'Compresibilidad de la roca en la celda en [pg <sup>2</sup> /lb]
fi como doble	'Porosidad de la roca en la celda
Pozo_sn como booleano	'Bandera que indica si en la celda existe una terminación o no
q como doble	'Gasto en el pozo en la celda, si lo hubiere [SCF/d]
Ppwf como doble	'Seudopresión de fondo fluyente del pozo en la celda, si lo hubiere [pg <sup>2</sup> ]
J como doble	'Índice de productividad del pozo en la celda, si lo hubiere [(lb/pg <sup>2</sup> ) <sup>2</sup> /cp] .
Fin del Tipo	

El yacimiento contiene un arreglo de celdas, o malla, además de otras propiedades:

Tipo yacimiento	
cel() As tipo celda	'Arreglo (malla) de todas las celdas del yacimiento
T como doble	'Temperatura del yacimiento
gas_yac como Gas	'gas del yacimiento (ver la definición del tipo_gas más adelante.)
Prof como doble	'Profundidad en [pie]
Fin del tipo	

## V.2.2 ORGANIZACIÓN DE LA INFORMACIÓN EN LA RED SUPERFICIAL

Los elementos básicos con los que se construye una red en superficie son nodos y tubos. En el capítulo 3 se explicó que había una diferencia entre puntos de entrega (nodos en los cuales sale gas de la red), pozos (nodos en los cuales entra gas a la red) y nodos propiamente dichos. En código estos 3 tipos de nodos son en realidad un tipo nodo, y su identidad como punto de entrega, nodo conector o pozo, está indicada en su propiedad `Id`.

La idea o filosofía básica de la división de la información en la red es tener un arreglo de objetos nodos, por un lado, y un arreglo de objetos tubos por otro. Los objetos nodos tienen propiedades tales como presión, gasto (usada por puntos de entrega y pozos) y altura con respecto a algún nivel de referencia. Los objetos tubos tienen información tal como diámetro interior del tubo, longitud, temperatura media de flujo, y eficiencia de flujo para cada tubo. Tomando un nodo individual, éste está unido a otros nodos por medio de tubos determinados (un tubo por cada unión). El objeto tubo contiene, pues, además de las propiedades arriba descritas, un arreglo de objetos "conexión". Este objeto conexión indicará el índice (en el arreglo de los nodos) del nodo `j` al que se conecta el nodo `i`, y el índice del objeto tubo por medio del cual lo hace. Los nodos tienen un arreglo de estos objetos (objetos conexión) entre sus propiedades, por que la conexión puede ser múltiple.

Veamos la definición del objeto nodo:

```
Tipo nodo
  h como doble          'altura con respecto de un nivel de referencia [pie]
  id como entero        '1=pozo;2=nodo conector;3=Punto de entrega
  p como presión        'Presión en el nodo [lb/pg2]
  q como doble          'Gasto que se extrae o se inyecta a la red en el
                        nodo[MPCS/d]
  con_tubo() As conexión 'Arreglo de objetos de tipo conexión
Fin del Tipo
```

El objeto tubo (`tipo_tubo`) es:



Tipo tipo\_tubo

- D como Doble 'diámetro en [pg]
- L como Doble 'Longitud en [pie]
- E como Doble 'Eficiencia
- T como Doble 'Temperatura media de flujo en [R]

Fin del Tipo

Y la definición del tipo Conexión es:

Tipo conexión

- destino como entero 'número del nodo destino
- No\_tubo como entero 'número de tubo que hace la conexión

Fin del Tipo

La figura 5.2 esquematiza cómo estaría ligada la información para un ejemplo de red superficial:

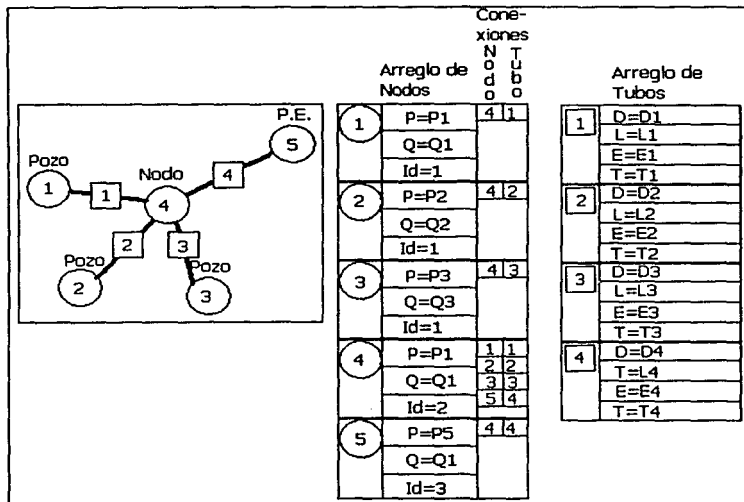


Figura 5.2.- Ejemplo de la organización de la información de las redes superficiales.

Ahora bien, es conveniente agrupar al arreglo de nodos y al arreglo de tubos bajo una sola estructura. Esta estructura contendrá toda la información de la red superficial, sin incluir estranguladores. Estos están incluidos en la estructura `Tipo_pozo` la cual tiene apuntes o conexiones a objetos de esta red, así como al yacimiento.

```

tipo Red_Superficial
    arr_nodos() como nodo           'arreglo de nodos
    arr_tubos () como tipo_tubo     ' arreglo de tubos horizontales
Fin del Tipo

```

### Tratamiento de los compresores

La manera de implementar los compresores en el programa fue tratarlos exactamente como tuberías de arreglo `arr_tubos`. Con esto es posible hacer el mismo tipo de análisis sólo que con compresores a veces en vez de tuberías. Puesto que los compresores, empero, tienen otro tipo de variables para calcular el gasto que pasa a través de ellos, se hizo un doble uso de variables de la estructura `tipo_tubo`. Cuando el "tubo" se trate de un compresor, la longitud  $L$  será la potencia [Hp], el diámetro  $d$  será el índice adiabático del gas a la temperatura  $T$ ,  $T$  seguirá siendo  $T$  (temperatura [R]) y finalmente  $\epsilon$  será la eficiencia del compresor. Es obvio que sin algún tipo de identificador en la estructura `tipo_tubo` no se podrá discernir entre un tubo y un compresor así como qué ecuación usar para el gasto ( ec. 3.21 ó 3.46 ), así que se adicionó una propiedad `Id` (identificador) a dicha estructura. Si `Id=1` el "tubo" será una tubería. Si `Id=2` el "tubo" será un compresor.

De acuerdo a este identificador (`Id`) se seleccionan las subrutinas para calcular los gastos y derivadas parciales de los gastos con respecto a las presiones de las ecuaciones (3.5), (3.9), (3.10) y (3.11.)

## V.2.3 ORGANIZACIÓN DE LA INFORMACIÓN DE LOS ESTRANGULADORES

Los objetos estranguladores están definidos en la estructura `tipo_estrangulador`. Esta contiene información necesaria para realizar los cálculos de caída de presión, diferente de los datos del gasto o presión a la salida del estrangulador, que varían constantemente.

El índice adiabático es función de la temperatura del gas en la cabeza del pozo. Esta temperatura, en rigor, es dependiente del gasto y características de la tubería de producción, sin embargo, los alcances de esta tesis sólo consideran temperatura constante en la cabeza de los pozos e independiente a cualquier variable del sistema.

Las propiedades del objeto estrangulador son diámetro del estrangulador en 64avos de pulgada, temperatura media del gas en el estrangulador, índice adiabático y coeficiente de descarga, ambos adimensionales:

```
Tipo tipo_estrangulador
  D64 como Doble 'Diámetro en 64avos de pulgada
  Tth como Doble 'Temperatura en la cabeza del pozo [R]
  K como Doble 'Índice adiabático a Tth
  Cd como Doble 'Coeficiente de descarga del est. [adm]
Fin del Tipo
```

## V.2.4 ORGANIZACIÓN DE LA INFORMACIÓN DE LAS TERMINACIONES

El objeto "terminación" contiene información que propiamente pertenecería a otros objetos como las celdas del yacimiento, sin embargo, todas estas propiedades (factor de daño, radio del pozo, factor de flujo turbulento y radio de drene del pozo) no tienen significado sino donde está terminado un pozo. Es por esto que fue conveniente agruparlas en una sola estructura bajo un mismo nombre. El objeto terminación se define como:

```
Tipo terminación
  D como Doble 'Factor de flujo no darciano de flujo turbulento, en [d/MPCS]
  S como Doble 'factor de daño (adimensional)
  Ro como Doble 'Radio de drene del pozo en [pie] (Se recalcula en cada iteración)
  rw como Doble 'radio del pozo, en [pie]
Fin del Tipo
```

## V.2.5 ORGANIZACIÓN DE LA INFORMACIÓN DE LOS POZOS

Las estructuras pozos contienen propiedades que atañen a la tubería de producción, trayectoria del pozo (vertical o inclinada) terminación y estrangulador. Esto es así debido a que cada estructura pozo, además de poseer propiedades únicas, tiene como propiedades a otras 3 estructuras: una estructura tubo, una estructura terminación y una estructura estrangulador. Esto obedece a que por cada pozo se tiene una terminación una tubería y un estrangulador. Las estructuras estranguladores, o las terminaciones, carecen de sentido si no van asociados a un pozo que produzca. Hubiera sido posible, empero, agrupar a los estranguladores dentro de las estructuras nodos, pero se hubiera desperdiciado espacio en memoria, puesto que no todo nodo es pozo. Por otro lado se hubiera podido incluir a las estructuras terminación en las estructuras celdas del yacimiento, pero esto también hubiera resultado en desperdicio de memoria, puesto que no toda celda contiene una terminación. Se ve entonces que la opción de diseño que se tomó fue la adecuada; el espacio utilizado en memoria para información de terminaciones y estranguladores es justo el necesario para el número de pozos que se tengan en un momento dado.

Todo pozo está unido a una celda del yacimiento y a un nodo tipo pozo en la red superficial. Esto se representa en la estructura pozo por medio de índices. Cada estructura pozo tiene un índice del lugar en el arreglo de estructuras tipo celda que ocupa la celda en que está su terminación y tiene un índice del lugar en el arreglo de estructuras tipo nodo del nodo (tipo pozo) al que se conecta su estrangulador.

La definición de la estructura pozo es :

Tipo tubo\_vertical

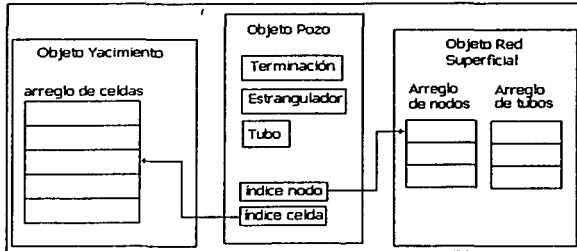
```

i_yac como entero 'Índice de la celda(en el arreglo celdas del yacimiento)
                    en la cual se encuentra la terminación
i_nodo como entero 'Índice del nodo(en el arreglo nodos de la red superficial)
                    en la cual se encuentra la terminación
Pth como doble 'Presión de fondo fluyente
tubo como tipo_tubo 'tubería de producción
alfa como Double 'ángulo de inclinación [°]
estrangulador como tipo_estrangulador 'estrangulador
term como tipo_terminación 'terminación

```

Fin del Tipo

El siguiente esquema ilustra las conexiones de la estructura pozo con la estructura yacimiento y la estructura red\_superficial:



5.3.- Índices de las estructuras pozos a celdas del yacimiento y a nodos de la red superficial.

## V.2.6 ESTRUCTURA "PROGRAMA DE OPERACIONES"

Al desarrollarse un yacimiento se comienza con un primer pozo, un segundo, tercero, etc. a la vez que al sistema de recolección se le añaden troncales. En la terminología de este simulador de yacimientos llamaremos "programa de operaciones" a la información que especifica el momento, posición espacial y demás características en que se añade una terminación o nuevo pozo al modelo del yacimiento. Para nuestro caso este programa debe incluir además la información de nuevos troncales en la red superficial para conducir la producción de los nuevos pozos.

El enfoque que se utilizó para resolver el problema de tener que añadir o cambiar a diferentes tiempos algunos pozos y elementos de la red, así como terminaciones fue hacer un arreglo de estructuras redes paralelo a un arreglo de arreglo de tubo\_vertical. En un momento dado, una red del arreglo (con pozos, nodos, etc... todo especificado) es la que funciona. En el momento en que haya cualquier cambio a esta red, lo que se hace es cambiar de red (avanzar el índice en el arreglo) aunque se repitan una y otra vez especificaciones de nodos, etc. Se pudo haber hecho también por medio de palabras

claves ("keywords") que durante la corrida ordenaran la creación o eliminación de estructuras de una estructura único de red, pero programáticamente es más complejo.

La estructura "programa de operaciones" es:

```
Tipo Programa_operaciones
  Redes () como Red_Superficial
  arr_pozos() como arr_tub_vert
Fin del Tipo
```

Donde arr\_tub\_vert es un arreglo de tubo\_vertical:

```
Tipo arr_tub_vert
  Arr_tubos() As tubo_vertical
Fin del Tipo
```

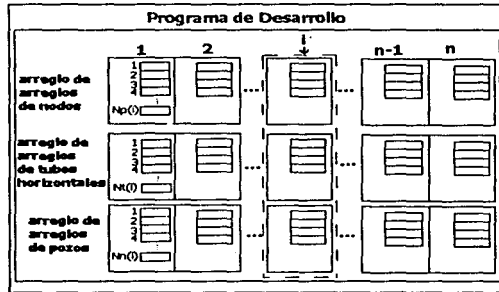


Figura 5.4.- Programa de operaciones

Como se observa en la definición de Programa\_operaciones, no se especifican los tiempos a los cuales han de entrar en uso las diferentes estructuras Red\_Superficial y arr\_tub\_vert, sólo el orden. Esta labor se ha relegado a otra estructura llamada Identificador\_de\_Objetos. Su razón de ser surge del hecho de que la configuración de la red y los pozos puede cambiar arbitrariamente y nada garantiza que, digamos, el nodo No. i siga siendo el nodo i en otra red del arreglo Redes. Debe haber algún objeto que identifique quién es quién en cada red de acuerdo a un identificador único. Con este identificador (que puede ser un número o un nombre en cadena de caracteres) se puede

"dar seguimiento" al objeto en los diferentes elementos del arreglo de Red\_Superficial, de no ser así, descuidadamente podríamos hablar, digamos, del gasto del pozo 1 en  $t=20[d]$  y en  $t=1500[d]$ , sin tomar en cuenta la eventualidad de que el pozo 1 se hubiere cerrado en  $t=600[d]$  y el que era el pozo 2 es el pozo 1 a partir de ese momento.

### 5.2.7 ESTRUCTURA "IDENTIFICADOR DE OBJETOS"

Para identificar cualquier elemento se utiliza la estructura `identificador_elemento`. El "elemento" en sí, puede ser un nodo, un tubo horizontal (`tipo_tubo`) o un pozo (`tubo_vertical`.) Cada tipo de elemento puede ser "rastreado" en los arreglos correspondientes de nodos (`Programa_operaciones.Redes(i).arr_nodos`.) de tubos horizontales (`Programa_operaciones.Redes(i).arr_tubos`) o de pozos (`Programa_operaciones.arr_pozos(i).Arr_tubos`), según sea el caso. La definición de `identificador_elemento` es:

```
Tipo identificador_elemento
  Nombre_elemento como cadena
  Ini como entero
  fin como entero
  Id() como entero 'va de ini a fin
Fin del Tipo
```

El `Nombre` del elemento es el identificador único. `Id` es un arreglo de que va desde `Ini` (inicio) hasta `Fin` y contiene los índices en los arreglos ( de nodos, de tubos horizontales o de pozos, según sea el caso) en que existe un elemento de nombre `Nombre`. La figura 5.5 muestra con un ejemplo cómo funciona el `identificador_elemento`.

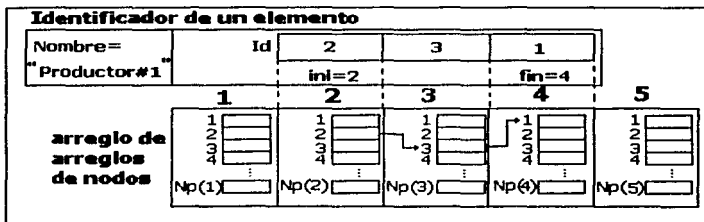


Figura 5.5.- Ejemplo de cómo funciona el `identificador_elemento`

En el ejemplo de la figura 5.5 existe un nodo llamado "Productor #1". Su existencia va desde  $ini=2$  hasta  $fin=4$ . En  $i=2$  existe en la red superficial como el nodo 2, en  $i=3$  existe como el nodo 3 y en  $i=4$  existe como el nodo 1. Nótese que el arreglo  $id$  corre paralelo al arreglo de arreglos de nodos desde  $ini$  hasta  $fin$ , con la misma numeración, incluso si su primer elemento ( $ini$ ) no es 1 ni el último coincide con el último elemento del arreglo de arreglos de nodos.

El tipo `Identificador_de_Objeto`s contiene, finalmente, 3 arreglos de identificadores, uno para los nodos en el arreglo de nodos, otro para los tubos horizontales en el arreglo de tubos horizontales y otro para los pozos en el arreglo de pozos. Contiene además un arreglo de tiempos que corre paralelo a los arreglos de arreglos del programa de operaciones (`Programa_operaciones.`)

```
Tipo Identificador_de_Objeto
    arr_inodos() como identificador_elemento
    arr_itubosh() como identificador_elemento
    arr_itubosv() como identificador_elemento
    tiempos() como doble 'en días
    i_tiempo_actual como entero
Fin del Tipo
```

Cada  $tiempos(i)$  es el tiempo, en días, en que **deja de utilizarse** la red  $i$ . El tiempo en que comienza a utilizarse la red  $i$  es  $tiempos(i-1)$ . Una excepción es  $i=1$ , cuyo  $tiempos(i-1)$  no existe en el arreglo y se define como 0 días. Así, la primer red se utiliza desde  $t=0[d]$  y la última red (red  $n$ )deja de utilizarse en  $tiempos(n)$ . El índice  $i\_tiempo\_act$  sirve para llevar un registro de la red del programa de operaciones que se encuentre en uso. Se revisa que el tiempo en que se encuentre la corrida sea menor a `Identificador_de_Objeto.tiempos(Identificador_de_Objeto.i_tiempo_actual)`. En el momento que lo rebase, se lleva a cabo un "cambio de red" al hacer:

```
Identificador_de_Objeto.i_tiempo_actual = Identificador_de_Objeto.i_tiempo_actual + 1.
```



### V.2.8 ORGANIZACIÓN DE LA INFORMACIÓN DE LAS CONDICIONES DE FRONTERA EN PUNTOS DE ENTREGA

Las condiciones de frontera en los puntos de entrega (nodos sumideros) puede especificarse como presión en función del tiempo o bien gasto en función del tiempo. Cada punto de entrega se identifica en la estructura `Identificador_de_Objeto`s por un `identificador_elemento`. Sin embargo, a menos que se haga una búsqueda directa en el programa de operaciones (`programa_operaciones`), no se puede saber, sólo con la información del identificador de objetos (`Identificador_de_Objeto`s) qué nodos son puntos de entrega. Por otro lado necesitamos ligar o "sujetar" por algún medio la información de qué gasto o presión tienen determinados tiempos dichos puntos de entrega. Esto podría hacerse añadiendo a la estructura `Identificador_de_Objeto`s arreglos u otras estructuras como propiedades para que ahí se guardara la información, sin embargo no todo nodo es punto de entrega, así que se desperdiciaría espacio en memoria. La mejor solución, al parecer, es tener una estructura (que llamaremos `condiciones_de_fronteras`) que contenga la información de gastos o presiones en función del tiempo y tenga un índice al `identificador_elemento` del nodo punto de entrega en el `Identificador_de_Objeto`s (esto es `Identificador_de_Objeto`s `.arr_inodos()`). Con esto se logra tener el espacio adecuado para almacenar la información de las condiciones de frontera además de poder asignar los gastos o presiones pertinentes a los nodos tipo punto de entrega correctos en las redes del programa de operaciones.

La definición de `condiciones_de_fronteras` es:

```
Tipo Condiciones_de_Frontera
    arr_cond() como Tipo_Cond_Nodo
Fin del Tipo
```

Como se ve, la estructura es un arreglo de `Tipo_Cond_Nodo`, que son los que poseen la información individual de los gastos o presiones para cada punto de entrega (referido en el arreglo de identificadores de nodo del identificador de objetos por su propiedad `I_in`)

Tipo Tipo\_Cond\_Nodo

I\_in como entero 'lugar que ocupa en el arreglo arr\_inodos del programa de operaciones

P\_o\_Q como booleano 'verdadero=Presión[lb/pg<sup>2</sup>], falso=Gasto[MPCS/d]

Val() como Doble 'Valores de gasto o presión, según indique P\_o\_Q

tiempo() como Doble 'Tiempo en que TERMINA de usarse el P o Q en i.

Fin del Tipo

La siguiente figura ilustra cómo funciona la estructura condiciones\_de\_fronteras:

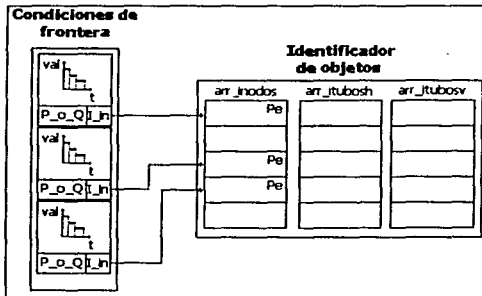


Figura 5.6.- Funcionamiento de la estructura de condiciones de frontera

## V.2.9 ORGANIZACIÓN DE LA INFORMACIÓN DE LOS PARAMETROS DE LA CORRIDA

En el simulador se utilizan varios parámetros que determinan el tiempo de cálculo de la corrida y la exactitud de sus resultados, así como el número máximo de iteraciones y correlaciones a usarse. Todos ellos están agrupados en una sola estructura llamada Parámetros. Esta estructura contiene:

- El cambio máximo de presión de las celdas del yacimiento [ $\text{lb/pg}^2$ ] por iteración.
- Cambio máximo de gasto de los pozos por iteración.
- % de error en el balance de materia (por iteración, en contraposición al acumulativo)
- Tolerancia en el balance de gastos (modelo acoplado yacimiento-instalaciones) [MPCS/d]
- Tolerancia del método de Newton en el yacimiento [ $(\text{lb/pg}^2)^2/\text{cp}$ ].
- Tolerancia del método de Newton en las instalaciones superficiales, ya sea en [ $\text{lb/pg}^2$ ] o en [MPCS/d].
- Tolerancia en el cálculo de la presión de fondo fluyente en las tuberías de producción [ $\text{lb/pg}^2$ ]
- Tolerancia en el cálculo de la presión en la cabeza de los pozos en los estranguladores [ $\text{lb/pg}^2$ ]
- Número máximo de iteraciones para el balance de gastos yacimiento-instalaciones.
- Número máximo de iteraciones para el método de Newton en el yacimiento.
- Número máximo de iteraciones para método de Newton en las instalaciones superficiales.
- Número máximo de iteraciones para el cálculo de la presión de fondo fluyente.
- Número máximo de iteraciones para el cálculo de la presión en la cabeza de los pozos.
- Correlación a usarse para calcular el radio de drene de los pozos.
- Correlación a usarse para calcular la presión de fondo fluyente.
- Correlación a usarse para calcular la presión en la cabeza de los pozos.

Su definición es:

Tipo Parámetros\_corrida

DPLIM Como doble 'cambio máximo en la presión

DQLIM Como doble 'Cambio máximo en el gasto de los pozos

EBMM Como doble 'Error INSTANTANEO de BM máximo tolerado

Tol\_Gasto Como doble 'tolerancia para los gastos en la iteración global

Tol\_Yac Como doble 'Tolerancia para las Pp en el yac  
 Tol\_RS Como doble 'Tolerancia para la red superficial  
 Tol\_TP Como doble 'Tolerancia para las tuberías de producción  
 Tol\_Est Como doble 'Tolerancia para los estranguladores  
  
 nit\_yac Como entero '# máx. de iteraciones en el yac  
 nit\_RS Como entero '# máx. de iteraciones en la red superficial  
 nit\_TP Como entero '# máx. de iteraciones en la Tub. de producción  
 nit\_Est Como entero '# máx. de iteraciones en el estrangulador  
 nit\_Global Como entero '# máx. de iteraciones en el balance de gastos  
  
 Corr\_Pozo Como entero 'Correlación que se usará para calcular ro.  
 Corr\_est Como entero 'Correlación que se usará para calcular Pwh  
 Corr\_TP Como entero 'Correlación que se usará para calcular Pwf  
 t\_rep() Como doble 'tiempos de reporte

Fin del Tipo

## V.2.10 ORGANIZACIÓN DE LA INFORMACIÓN DE LA HISTORIA DE LA CORRIDA

El simulador obtiene para diferentes tiempos las variables listadas en la tabla 4.3. Estos valores deben ser almacenados en una serie de arreglos. La manera como el simulador SIMPREG los almacena es introduciéndolos a una sola estructura, y posteriormente la información almacenada en esta estructura es guardada en archivo, si el usuario lo desea, o bien usada por las subrutinas de la interfaz para que la historia de la corrida sea mostrada gráficamente. Una variable de este tipo es el resultado principal que arroja la subrutina que ejecuta el algoritmo 4.1.

Antes de definir esa estructura, es necesario presentar tres estructuras que son propiedades de la primera:

- Objeto elemento\_presiones: Es un arreglo de presiones, ya sea de las celdas, de nodos, etc.

```
Tipo elemento_presiones
  p() como Doble 'presiones en [lb/pg2]
Fin del Tipo
```

- Objeto elemento\_gastos: Es un arreglo de gastos a través de tuberías o bien de pozos.

```
Tipo elemento_gastos
  q() como doble 'gastos en [MPCS/d]
Fin del Tipo
```

- Objeto elemento\_reporte\_pozos: Contiene tres arreglos. Uno de presiones de fondo fluyente, otro de presiones en la cabeza de los pozos y otro de gastos. Su finalidad es contener, para un tiempo dado, las presiones de fondo fluyendo y en las cabezas de los pozos, así como los gastos de todos los pozos.

```
Tipo elemento_reporte_pozos
  Pwf() como Doble 'Presiones de fondo fluyente [lb/pg2]
  Pth() como Doble 'Presiones en las cabezas de los pozos [lb/pg2]
  q() como Doble 'Gastos en los pozos [MPCS/d]
Fin del Tipo
```

Estas tres estructuras son la definición del tipo de algunas propiedades de la estructura historia\_corrida.

Tipo historia_corrida	
I_red() como entero	'Índice de la red que está funcionando
Pres_nodos() As elemento_presiones	'arreglo de Presiones en los nodos.
rep_pozos() As elemento_reporte_pozos	'arreglo del "estado" de los pozos.
Gastos_PE() As elemento_gastos	' arreglo de los gastos en los puntos de entrega.

Gastos_tubos() As el_gastos_car	'arreglo de los gastos en los tubos horizontales.
Pres_yac() As elemento_presiones	'arreglo de la distribución de presiones en el yacimiento.
tiempo() como Doble	'arreglo de tiempos [d].
Gasto_total() como Doble	'Sumatoria de gastos de los pozos [MPCS/d]
P_med() como Doble	'Presión media del yacimiento [lb/pg <sup>2</sup> ]
Reserva() como Doble	'Reserva de gas @ c.s. [MMPCS]
EBM() como Doble	'%Error de balance de materia acumulado
EBMI() como Doble	'%Error de balance de materia instantáneo
Delt_CPU() como Sencillo	'Tiempo CPU/iteración global
No_it() como Entero	'no de iteraciones en la iteración global
No_it_Yac() como Entero	'no it. medio del Yac
No_it_Red() como Entero	'no it. medio de la red

Fin del Tipo

Todos los arreglos en la estructura `historia_corrida` van paralelos, alineados con el arreglo `tiempo`. La idea es que para un tiempo  $k$  la variable declarada como esta estructura dé una "fotografía" del yacimiento, pozos y sistema de recolección al tiempo `historia_corrida.tiempo(k)`. De esta manera es posible, en la interfaz gráfica, representar la corrida como una película simplemente mostrando el estado del sistema para el tiempo  $k$ , haciendo variar  $k$  desde 0 hasta el número total de tiempos almacenados.

## V.2.11 ORGANIZACIÓN DE LAS PROPIEDADES PVT DEL GAS

El simulador del yacimiento utiliza directamente en sus cálculos a la pseudopresión ( $P_p$ ) y el factor de compresibilidad ( $z$ ). Por otro lado, el simulador de superficie utiliza en sus cálculos al mismo factor  $z$  y además a la viscosidad del gas (El factor de volumen no se ocupa como tal en dichos cálculos.) Estas propiedades pueden ser especificadas por tablas de datos (por ejemplo obtenidas de análisis PVT) o por correlaciones. Si son por correlaciones, pueden ser obtenidas a su vez de dos maneras diferentes; especificando la presión pseudocrítica, temperatura pseudocrítica y densidad relativa del gas o bien

especificando su composición. Este segundo caso, empero, llega a reducirse al primero, pues de la composición se obtienen las propiedades pseudocríticas y la densidad relativa. Con estas propiedades se "alimentan" las correlaciones de viscosidad, factor de compresibilidad, etc., para que sean evaluadas a diferentes presiones y temperaturas. La ecuación de estado que siempre se utiliza es la ley general de gases reales (ec. 2.4).

El simulador del yacimiento, por otro lado, no calcula las propiedades durante la corrida (por ejemplo la pseudopresión en función de la presión,) sino que realiza previamente un ajuste de polinomios a curvas de las propiedades y es a ellos a quienes evalúa durante la corrida. Esto con el propósito de evitar tener que evaluar correlaciones cuyo tiempo de cálculo puede ser muy superior al del polinomio. No se implementó la búsqueda en tablas por que se pensó en un principio que era más eficiente el uso de polinomios. Sin embargo la experiencia mostró que se requieren polinomios de hasta 6° o 7° grado tener exactitud en todo el rango del ajuste (especialmente en los polinomios de presiones en función de las pseudopresiones.) Evaluar un polinomio de alto grado es costoso en términos de multiplicaciones al calcular las potencias de orden superior. En cambio la búsqueda en tablas, si es que los puntos de presión en la tabla son equidistantes, sólo se requiere una multiplicación para encontrar el intervalo donde "cae" el valor deseado e interpolar linealmente entre los dos puntos que forman los extremos del intervalo. Su exactitud depende del número de datos de presión que se proporcionen y por ser muchas de las curvas suaves, éste número no necesita ser extremadamente grande.

La estructura que contenga información PVT del gas deberá poseer toda la información necesaria para cualquier caso posible de los arriba mencionados. Podemos dividir estas propiedades en las siguientes categorías:

- Propiedades de la composición del gas:
  - Fracciones molares de compuestos hidrocarburos individuales hasta los hexanos (sólo parafinas.)
  - Fracciones molares de compuestos no hidrocarburos.
  - Fracción molar de la fracción pesada C7+.
  - Propiedades de la fracción pesada.
- Propiedades "globales" de la mezcla:
  - Presión pseudocrítica.

- Temperatura pseudocrítica.
- Densidad relativa (al aire.)
- Tablas de valores especificadas por el usuario:
  - Tabla de valores de  $z$  vs presión .
  - Tabla de valores de Viscosidad vs presión.
  - Tabla de valores de pseudopresión vs presión.
- Polinomios de ajuste de propiedades:
  - Temperatura de referencia a la cual fueron ajustados los polinomios.
  - Coeficientes del polinomio de ajuste de  $z=z(P)$ .
  - Coeficientes del polinomio de ajuste de  $Visc=Visc(P)$ .
  - Coeficientes del polinomio de ajuste de  $P_p=P_p(P)$ .
  - Coeficientes del polinomio de ajuste de  $P=P(P_p)$  (transformación inversa.)

Antes de proceder a enunciar la estructura general de propiedades PVT (llamada `Info_PVT`) definiremos las estructuras `Gas` y `Heptanos` más, que son propiedades de `Info_PVT`, y explicaremos algunas particularidades de la manera en que se registra la composición del gas.

### Estructura gas

La estructura `Gas` caracteriza completamente a una mezcla gaseosa multicomponente si se obtienen por correlaciones la viscosidad y el factor de compresibilidad en función de sus propiedades pseudorreducidas y su densidad relativa.

Tipo Gas

<code>Pc</code> como Doble	'Presión pseudocrítica, en [lb/pg <sup>2</sup> ]
<code>Tc</code> como Doble	'Temperatura pseudocrítica en [R]
<code>dr</code> Como doble	'Densidad específica a condiciones estándar (adimensional)

Fin del Tipo



### Estructura heptanosmás

Por otro lado, si se utiliza la composición del gas, será necesario especificar las propiedades de la fracción pesada (C7+) que queda registrada en la siguiente estructura:

#### Tipo Heptanosmás

Mm como Doble 'Masa molecular aparente de la fracción C7+

dr como Doble 'Densidad en edo. Líquido relativa al agua

Pc como Doble 'Presión pseudocrítica [lb/pg<sup>2</sup>]

Tc como Doble 'Temperatura pseudocrítica [R]

Fin del Tipo

### Convención de ordenamiento de los compuestos

Como se vio arriba, esta estructura contiene la composición del gas, entre muchas otras cosas. Se ha establecido una convención del orden en que las fracciones molares de los compuestos aparecen en un arreglo de la estructura Compo (de composición). Como las propiedades pseudocríticas y masas moleculares de los compuestos son constantes y conocidas, no se almacenan en la estructura Info\_PVT; éstas se obtienen de una función que las devuelve de acuerdo, precisamente, al índice asociado a cada compuesto.

Tabla 5.1.- Compuestos que forman la mezcla gaseosa.

Índice o lugar	Compuesto
1	Metano
2	Etano
3	Propano
4	n-Butano
5	Isobutano
6	n-Pentano
7	Isopentano
8	Neopentano
9	n-Hexano
10	2-Metilpentano
11	3-Metilpentano
12	Neohexano

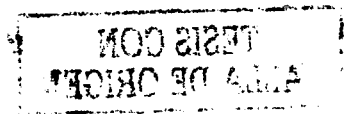
13	2,3-Dimetilbutano
14	Nitrógeno
15	Sulfuro de Hidrógeno
16	Dióxido de Carbono

### Estructura info\_pvt

Su definición, finalmente es:

#### Tipo Info\_PVT

Compo(1 a 17) como Doble 'Fracciones molares de los compuestos de la tabla 5.1  
 ban\_compo(1 To 17) como Booleano 'Banderas para indicar qué compuestos están  
 presentes en la mezcla  
 GasY como Gas 'Propiedades pseudocríticas y densidad relativa  
 Hept\_más como Heptanosmás 'Propiedades de la fracción pesada C7+  
 P1 como Doble 'Presión mínima para el ajuste de polinomios  
 P2 como Doble 'Presión máxima para el ajuste  
 T como Doble 'temperatura [R] de referencia  
 arrP() como Doble 'Datos de tabla de presiones  
 arrZ() como Doble 'Datos de tabla de factor de compresibilidad  
 arrVisc() como Doble 'Datos de tabla de viscosidad  
 arrBg() como Doble 'Datos de tabla de Factores de volumen  
 arrPp() como Doble 'Datos de tabla de pseudopresiones  
 ban\_corrZ como Booleano 'Bandera para indicar si se usa correlación o tabla para Z  
 ban\_corrMu como Booleano 'Bandera para indicar si se usa correlación o tabla para  
 la viscosidad  
 ban\_corrPp como Booleano 'Bandera para indicar si se usa correlación o tabla para  
 el factor de volumen.  
 pol\_m\_P() Como doble 'Polinomio de ajuste de la pseudopresión vs presión  
 pol\_P\_m() Como doble 'Polinomio de ajuste de la presión vs pseudopresión  
 pol\_z\_P() Como doble 'Polinomio de ajuste del fact. De comp. vs presión  
 pol\_Visc\_P() Como doble 'Polinomio de ajuste de la viscosidad vs presión  
 Fin del Tipo



## V.3 INTERFAZ CON EL USUARIO

### V.3.1 VENTANA PRINCIPAL

La ventana principal del simulador es básicamente un menú (fig. 5.7) formado por botones que permite al usuario entrar a las diferentes secciones del programa:

- Edición del yacimiento.
- Edición de la red de recolección.
- Propiedades PVT del gas.
- Parámetros de la corrida.
- Visualización en 3D de la corrida.
- "Generador" de gráficas.

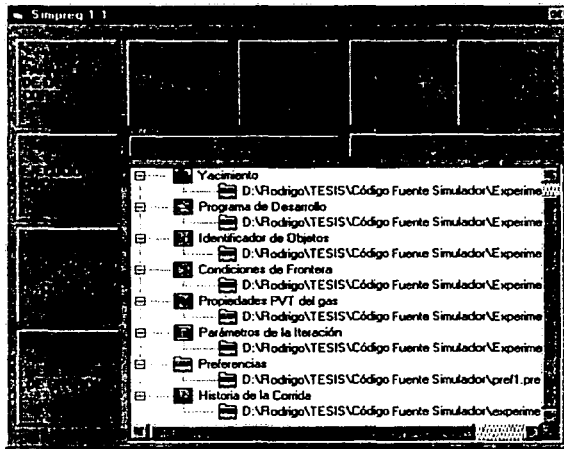


Figura 5.7.- Ventana principal del simulador

En la misma ventana (figura 5.7) existe un objeto de estructura de árbol con la función de mostrar los nombres y rutas de archivos abiertos. Sólo puede haber un archivo abierto de

cada tipo. Estos tipos de archivos los presentaremos en la sección V.3 y por el momento tan sólo los mencionaremos:

- Información del yacimiento
- Programa de operación del campo
- Identificador de objetos (ver sección V.2)
- Condiciones de frontera (ver sección V.2)
- Propiedades PVT del gas
- Parámetros de la corrida
- Preferencias del usuario
- Historia de la corrida (una vez que se ha corrido el simulador)

El botón "Abrir Proyecto" permite cargar el conjunto de archivos mencionados arriba que conforman un proyecto. La idea de agrupar los archivos en una unidad llamada proyecto es evitar que el usuario tenga que abrir por separado archivos que estén relacionados unos con otros.

El botón "Guardar proyecto" permite guardar los nombres de los archivos del proyecto (extensión .prj) pero no guarda los archivos mismos. Estos deben ser guardados desde su editor correspondiente.

En esta misma sección discutiremos cada uno de estos editores que permite trabajar gráficamente con la información, guardarla, recuperarla y ordenar ejecutar las corridas del simulador.

### V.3.2 EDICIÓN DEL YACIMIENTO

El programa permite diseñar una malla cartesiana (2D), con valores de presión, porosidad, permeabilidad en la dirección X, permeabilidad en la dirección Y, altura y compresibilidad de la formación para cada celda.

Este diseño se hace por medio de mallas cuyas celdas (que representan celdas del yacimiento) pueden contener cantidades para indicar que alguna propiedad la celda del yacimiento tiene determinado valor o bien tener de algún color para indicar que la celda de la malla pertenece o no al yacimiento.

Las figuras 5.8 y 5.9 muestran dos tipos de mallas; la primera de diseño del yacimiento y la segunda de introducción y edición de valores en las celdas.

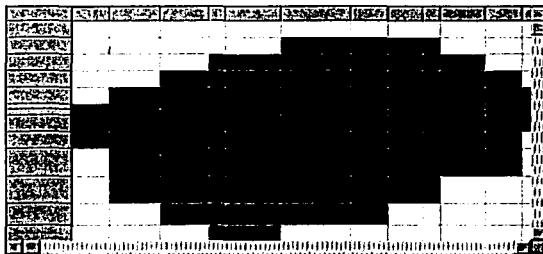
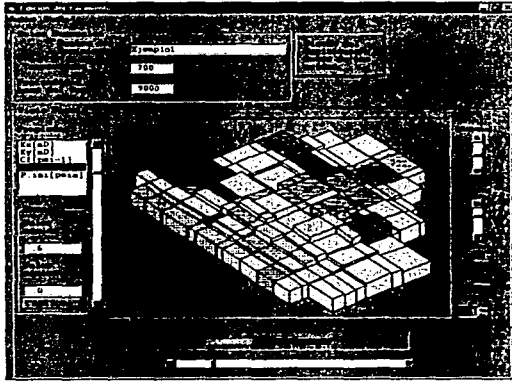


Figura 5.8.- Malla de diseño del yacimiento. Las celdas de color rojo son celdas del yacimiento, mientras que las blancas no.

10322.5806451613	27741.335483811	21935.4838703677	21935.483870
10322.5806451613	27741.335483811	21935.4838703677	21935.483870
10322.5806451613	27741.335483811	21935.4838703677	21935.483870
10322.5806451613	27741.335483811	21935.4838703677	21935.483870
10322.5806451613	27741.335483811	21935.4838703677	21935.483870
10322.5806451613	27741.335483811	21935.4838703677	21935.483870
10322.5806451613	27741.335483811	21935.4838703677	21935.483870
10322.5806451613	27741.335483811	21935.4838703677	21935.483870
10322.5806451613	27741.335483811	21935.4838703677	21935.483870
10322.5806451613	27741.335483811	21935.4838703677	21935.483870

Figura 5.9.- Malla de introducción de valores a las celdas del yacimiento. Las celdas blancas pertenecen al yacimiento, mientras que las grises no.

La idea de estas dos mallas es que el usuario pueda hacer que cambie de color de determinadas celdas de la primera malla, diseñando así el contorno irregular del yacimiento. Una vez hecho esto, se procede a la segunda malla para introducir los valores de las propiedades (porosidad, permeabilidad, etc.) de las celdas que pertenezcan al yacimiento. Después de diseñar e introducir valores de las celdas del yacimiento se puede hacer una vista previa tridimensional de la geometría y los valores de propiedades introducidos a la malla. Para ver detalles acerca de cómo se grafica en 3D el lector puede ver la sección de graficación en 3D. La pantalla de vista previa, que podríamos considerar como principal se muestra en la figura 5.10:



5.10.- Ventana principal del diseño del yacimiento

Como se aprecia en dicha ventana, en ella se introduce el nombre del yacimiento, profundidad a la base de la estructura y temperatura del yacimiento. En la figura 5.10 se muestra un yacimiento heterogéneo (se grafica porosidad en la escala de colores.)

Programáticamente, para llevar a cabo este diseño y edición del yacimiento se requieren dos objetos gráficos que interactúan con dos estructuras. Los objetos gráficos son la malla de diseño del yacimiento y la de introducción o edición de los valores de las propiedades. Las estructuras con las que interactúan son una variable global de tipo yacimiento y un arreglo bidimensional, que en un momento definiremos y tiene la función de contener información momentánea capturada de los objetos gráficos y que será después copiada a la variable tipo yacimiento.

Tipo arreglo\_de\_celdas

n() como entero 'Bandera para identificar si la celda i,j pertenece o no al yacimiento.

cel() como entero 'arreglo lineal de tipo Celda

Fin del Tipo

Tipo arreglo\_bidimensional\_de\_celdas

arr() como arreglo\_de\_celdas 'arreglo de arreglos lineales de celdas (matriz)

n\_cel como entero 'número de celdas en el yacimiento

End Type

El propósito de la estructura `arreglo_de_celdas` es servir de base para que se puedan formar arreglos de arreglos de celdas. Contiene 2 vectores que corren paralelos; `cel` tiene la información de una fila de celdas en el yacimiento y `n` es un vector de banderas (como no toda celda pertenece al yacimiento, es necesario el vector de banderas para indicarlo.)

La estructura `arreglo_bidimensional_de_celdas`, es una estructura que intercambia información con las mallas arriba descritas y la variable de tipo `yacimiento`. Contiene un arreglo de un arreglo de celdas, es decir una matriz de celdas y banderas asociadas a estas últimas para indicar si la celda pertenece o no al yacimiento.

Parecería trivial e innecesario a primera vista el uso de este arreglo, si toda información concerniente al yacimiento puede ser introducida directamente a la estructura declarada con este nombre, sin embargo, la constante creación y eliminación de celdas en la edición de la malla crea complejidades para la conservación de la información ya introducida representada en memoria (los objetos gráficos no la pueden ni la deben retener o "recordar"); se tendrían que eliminar e insertar celdas directamente en la estructura `yacimiento` y para esto se tendría que hacer una búsqueda para saber qué celda se eliminó o cual se va a crear en su arreglo lineal. El arreglo bidimensional proporciona una manera adecuada de retener esta información mientras se lleva a cabo la edición y una vez que el usuario confirme la versión de yacimiento que desea para la corrida, transfiere su información a la variable tipo `yacimiento` y luego desaparece.

El flujo de información entre `yacimiento`, arreglo bidimensional y mallas puede esquematizarse de la siguiente manera:

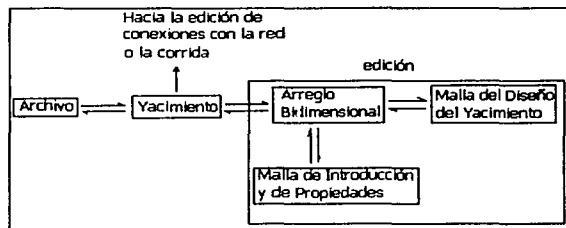


Figura 5.11.- Flujo de información para la edición del yacimiento.

En esta figura (figura 5.11) se ha incluido la lectura y guardado en archivo para mostrar todos los componentes involucrados en la introducción de información al yacimiento.

Las flechas son, programáticamente, subrutinas que copian información de una estructura a otra o de un objeto a una estructura y viceversa. Se listan a continuación sus nombres, con objeto de ilustrar mejor los conceptos de la figura 5.11, pero no sus parámetros ni sus definiciones puesto que contienen detalles que no son relevantes sino para la codificación verdadera, además de que se sobreentiende su función y las estructuras/objetos principales que involucran.

- `Mete_arr_bid_en_Yac`: Copia la información contenida en un `arreglo_bidimensional_de_celdas` a una variable tipo `yacimiento`.
- `Mete_Yac_en_arr_bid`: Copia la información contenida en una variable tipo `yacimiento` a un `arreglo_bidimensional_de_celdas`.
- `Mete_Malla_Prop_en_arr_bid`: Copia la información de la malla de propiedades a un `arreglo_bidimensional_de_celdas`.
- `Mete_arr_bid_en_Malla_Prop`: Copia la información de un `arreglo_bidimensional_de_celdas` a la malla de propiedades.
- `Mete_Malla_Dis_en_arr_bid`: Copia la información de la malla de diseño a un `arreglo_bidimensional_de_celdas`.
- `Mete_arr_bid_en_Malla_Dis`: Copia la información de un `arreglo_bidimensional_de_celdas` a la malla de diseño.

En la sección de almacenamiento y recuperación de la información se discutirán las subrutinas para guardar y recuperar la información del yacimiento.



### V.3.3 INTERFAZ GRÁFICA DE LA RED SUPERFICIAL

#### V.3.3.1 Objetos gráficos

Como se explicó en la sección de la organización de la información, para representar la información de la red superficial, incluyendo nodos, tuberías horizontales, estranguladores, tuberías de producción y terminaciones, se tienen una serie de estructuras para representar la red en memoria. La creación de instancias de estas estructuras (es decir variables, declaradas con esas estructuras,) así como la edición de la información que contienen, requiere de objetos gráficos en los cuales el usuario pueda introducir información y pueda ver de una manera gráfica la red que está diseñando. La idea primordial para lograr esto es tener variables con arreglos dinámicos, cuyo número de elementos pueda ser alterado por subrutinas. Cada vez que se quiera un nuevo nodo, por ejemplo, se redimensiona el arreglo de  $n$  nodos en  $n+1$  nodos, sin borrar la información previa. Con esto los objetos gráficos que el usuario manipula crean eventos que llaman a subrutinas que modifican el tamaño de los arreglos o bien asignan valores a sus propiedades.

Se tiene un arreglo de objetos gráficos (objetos image de Visual Basic) que representa nodos en la red y otro que representa tuberías horizontales (ver fig. 5.12). Los objetos image que representan nodos, pueden tener 3 tipos de dibujos (propiedades picture) para señalar que el nodo es de tipo nodo conector, pozo o punto de entrega.

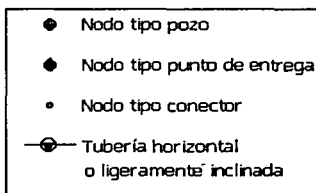


Figura 5.12.- Simbología de objetos gráficos de la red superficial



### V.3.3.2 Creación de objetos

Es importante discutir ahora los lineamientos generales de algoritmos para añadir y eliminar objetos de la red. Estos algoritmos son la base de las subrutinas que realizan estas actividades. Debe siempre tenerse en cuenta las declaraciones de las estructuras vistas en la sección de organización de la información, ya que los algoritmos trabajan con ellas. Es importante mencionar que se invirtió un gran esfuerzo en el diseño de dichos algoritmos, así como su programación y funcionamiento adecuado. Son necesidades no previstas siempre que se desea que la entrada de la información sea gráfica y no con palabras claves o archivos de entrada.

#### Creación de nuevos nodos en la red

La tarea más sencilla es crear nuevos nodos. Lo único que hay que hacer es añadir un nuevo nodo en el arreglo de nodos y en el de imágenes de nodos.

##### Algoritmo 5.1.- Añadir Nodo

- 1.-Redimensionar el arreglo de nodos y de imágenes de nodos de  $N_{\text{nodos}}$  a  $N_{\text{nodos}}+1$
- 2.-Inicializar a 0 el número de conexiones del último nodo
- 3.-Asignar el tipo de nodo que será el nodo recién creado (propiedad  $I_d$ )

#### Creación de nuevas tuberías horizontales en la red

Esta es otra tarea simple, aunque requiere ya algo más que simplemente redimensionar el arreglo de estos elementos. Es necesario crear conexiones en los nodos "origen" y "destino" (recuérdese que un tubo siempre une dos nodos.) El nodo destino no hace diferencia o "no sabe" si él es el destino o el origen. Lo único que registra es un nodo más al cual está conectado y por medio de qué tubo.

## Algoritmo 5.2.- Añadir Tubo Horizontal a la Red

- 1.-Especificar el número o índice del nodo origen y del nodo destino.
- 2.-Redimensionar el arreglo de tubos horizontales de  $N_{tub}$  a  $N_{tub} + 1$
- 3.-Redimensionar el número de conexiones del nodo origen de  $N_{con\_org}$  a  $N_{con\_org} + 1$
- 4.-Redimensionar el número de conexiones del nodo destino de  $N_{con\_dest}$  a  $N_{con\_dest} + 1$
- 5.-Redimensionar el arreglo de imágenes de tubos igual al nuevo número de tubos
- 6.-Asignar la propiedad `destino` de la conexión recién creada del nodo origen igual al índice del nodo destino
- 7.-Asignar la propiedad `No_tubo` la conexión recién creada del nodo origen igual al índice del tubo recién creado ( $N_{tub} + 1$ )
- 8.-Asignar la propiedad `destino` de la conexión recién creada del nodo destino igual al índice del nodo origen
- 9.-Asignar la propiedad `No_tubo` la conexión recién creada del nodo destino igual al índice del tubo recién creado ( $N_{tub} + 1$ )

**Creación de tuberías de producción, estranguladores y terminaciones**

Tanto tuberías de producción, como estranguladores y terminaciones están agrupados en un solo objeto llamado `tubo_vertical`. A diferencia de los tubos horizontales y los nodos, las estructuras `tubo_vertical` no tienen representación gráfica con un objeto `image`. Cuando se cree este objeto se debe de enlazar a una celda del yacimiento y un nodo tipo pozo en superficie.

**Algoritmo 5.3.- Creación de Tuberías de Producción, Estranguladores y Terminaciones**

- 1.-Especificar el índice o número de celda del yacimiento en el objeto yacimiento en el cual existirá la terminación.
- 2.-Asimismo, especificar el índice o número de nodo al cual se unirá el estrangulador
- 3.-Redimensionar el arreglo de tubos verticales (`tubo_vertical`) de `Ntub_vert` a `Ntub_vert+1` (recuérdese que el objeto `tubo_vertical` contiene ya al `tipo_estrangulador`, `Tipo terminación` y un `tipo_tubo`)
- 4.-Asignar la propiedad `i_yac` del `tubo_vertical` igual al índice de la celda a la cual se enlaza el pozo
- 5.-Asignar la propiedad `i_nodo` del `tubo_vertical` igual al índice del nodo al cual se enlaza el pozo.

**V.3.3.3 Eliminación de objetos**

La manera de eliminar objetos en arreglos es copiando la información de cada objeto posterior al elemento eliminado hacia su precedente inmediato. De esta manera se pierde la información del elemento eliminado y el último elemento es redundante, pues él (último) y su predecesor (penúltimo) contienen la misma información. Después se elimina al último elemento y el número de elementos en el arreglo decrece en 1.

Este algoritmo de eliminación presenta complicaciones que no fueron previstas, pues cada objeto tiene referencias a otros objetos, y si uno es eliminado del arreglo donde se encuentra, se debe de renombrar al resto y renombrar también las referencias: por ejemplo, si se tienen 5 nodos, 1, 2, 3, 4 y 5 y se elimina el nodo 3 el nodo 1 y 2 no sufren cambios, pero ahora la información contenida en el nodo 4 estará en el 3, la del 5 en el 4 y el 5 será borrado (se borra siempre el último elemento, no a los de en medio). Si un tubo o un nodo hacía referencia al nodo 4, ahora deberá hacerlo al nodo 3, pues el nodo 4 ahora es el 3.

Otro ejemplo de complicación es que al estar los nodos enlazados a otros nodos por tubos, para eliminar al nodo es necesario eliminar primero los tubos que tenía conectados, correr las referencias cada que se elimina un tubo, después eliminar al nodo y por último correr las referencias de los nodos.

Por esta razón se discuten los algoritmos en un orden especial, pues un algoritmo utiliza o implica a otros.

### **Eliminación de tubos**

Eliminar nodos requiere, casi siempre, primero eliminar tubos horizontales. Estos a su vez requieren eliminar conexiones dentro de los nodos, así que primero se discute la eliminación de conexiones de los nodos. Al eliminar un tubo se deben intervenir los arreglos de conexiones de los nodos origen y destino para eliminar la conexión de cada uno de ellos que lo implicaba.

#### **Algoritmo 5.4.- Eliminación de Conexión**

- 1.-Especificar el nodo del cual se desea eliminar la conexión (Nnodo)
- 2.-Especificar la posición de la conexión que se va a eliminar en el arreglo de conexiones del nodo (Npos).
- 3.- Asignar a la conexión  $i$ -ésima la información de la conexión  $(i+1)$ -ésima, para  $i$  que va desde Npos hasta el penúltimo elemento del arreglo de conexiones (es decir, se corre la información de los elementos del arreglo en posiciones mayores o iguales a Npos.) Se pierde la información que la conexión Npos-ésima contenía.
- 4.-Redimensionar el arreglo de conexiones del nodo de Ncon\_nodo a Ncon\_nodo-1 (se elimina el último elemento)

Al eliminar al tubo se deben correr las referencias a ese tubo y a todos aquéllos con índice mayor dentro del arreglo.

## Algoritmo 5.5.- Eliminación de Tubos Horizontales

- 1.-Especificar el número de tubo Ntub que se eliminará, así como los nodos a los cuales va unido (es decir que está referenciado en una de las conexiones de esos pozos en la propiedad 'No\_tubo' )
- 2.- Eliminar la conexión en el nodo origen con el algoritmo 5.4
- 3- Eliminar la conexión en el nodo destino con el algoritmo 5.4
- 4.-Para todos y cada uno de los nodos, si contienen dentro de sus conexiones referencias a tubos con índice mayor al índice del tubo eliminado, hacer estos índices menores en una unidad (correr las referencias de las conexiones)
- 5.- Asignar al tubo horizontal i-ésimo la información del tubo horizontal (i+1)-ésimo, para i que va desde Ntub hasta el penúltimo elemento del arreglo de tubos horizontales (es decir, se corre la información de los elementos del arreglo en posiciones mayores o iguales a Npos.) Se pierde la información que el tubo Ntub-ésimo contenía.
- 6.- Asignar a la imagen de tubo horizontal i-ésimo la información de la imagen de tubo horizontal (i+1)-ésimo, para i que va desde Ntub hasta el penúltimo elemento del arreglo de imágenes de tubos horizontales (es decir, se corre la información de los elementos del arreglo en posiciones mayores o iguales a Npos.
- 7.- Redimensionar el arreglo de tubos horizontales de Ntubos a Ntubos-1
- 8.-Redimensionar el arreglo de imágenes de tubos horizontales de Ntubos a Ntubos-1

**Eliminación de tuberías de producción, estranguladores y terminaciones**

La eliminación objetos `tubo_vertical` no requiere correr referencias, pues son ellos quienes "saben" a qué celda del yacimiento y qué nodo están conectados, no el yacimiento o los nodos.

**Algoritmo 5.6.- Eliminación de Tuberías de Producción, Estranguladores y Terminaciones**

- 1.-Especificar el índice de la tubería de producción, pozo y estrangulador ("pozo") que se va a eliminar (Ntub\_vert) en el arreglo de tuberías de producción.
- 2.- A partir de Ntub\_vert correr la información de las tuberías: Asignar al tubo vertical i-ésimo la información del tubo vertical (i+1)-ésimo, para i que va desde Ntub\_vert hasta el penúltimo elemento del arreglo de tubos verticales. Recuérdese que el objeto tubo\_vertical contiene ya al tipo\_estrangulador, Tipo terminación y un tipo\_tubo.
- 3.- Redimensionar el arreglo de tubos horizontales de Ntubos a Ntubos-1.

**Eliminación de nodos**

La eliminación de nodos requiere la eliminación de los tubos a los cuales va conectado el nodo, así como correr referencias.

**Algoritmo 5.7.- Eliminación de Nodos**

- 1.-Especificar el índice Nnodo del nodo en el arreglo de nodos que se va a eliminar.
- 2.-Eliminar todos los tubos horizontales que están conectados al nodo con el algoritmo 5.5
- 3.-Eliminar al pozo, si lo hubiere, conectado al nodo (algoritmo 5.6)
- 4.-Correr las referencias de destino de las conexiones de todos nodos: A toda referencia , en cualquier conexión de nodo, a un número de nodo igual o mayor a Nnodo, se le resta una unidad.



5.-Correr la información de los nodos: A partir de Nnodo asignar al nodo  $i$ -ésimo la información del nodo  $(i+1)$ -ésimo, para  $i$  desde Nnodo hasta el penúltimo elemento del arreglo de nodos.

6.-Redimensionar el arreglo de nodos de Nnodos a Nnodos-1.

#### **V.3.3.4. Edición de los valores de la red**

La introducción y modificación de valores (propiedades de las estructuras) en la red se lleva a cabo por medio de ventanas o cuadros de diálogo en los cuales el usuario escribe en cajas de texto valores de diámetro, longitud, etc. , o bien en mallas que muestran las propiedades de las estructuras agrupadas.

#### **Cuadros de diálogo**

Por cada cuadro de diálogo se deben tener 2 subrutinas; una que lee los valores de las cajas de texto y los introduzca a las estructuras y otra que lea las estructuras y muestre los valores en las cajas de texto del cuadro de diálogo . Con esto se asegura la sincronización cuadro de diálogo- estructuras; cuando el usuario modifica los valores en las cajas de texto se actualiza la información en memoria y los cuadros de diálogo muestran valores ya introducidos a las estructuras cuando es necesario.

Para mostrar un cuadro de diálogo al hacer clic en alguno de los objetos gráficos debe haber un modo edición marcado por una bandera. Entonces se desplegará la ventana que muestra la información de la estructura (por ejemplo un tubo horizontal) representado por el objeto gráfico.

#### **Cuadro de diálogo de tuberías horizontales**

Como primer ejemplo tenemos el cuadro de diálogo de las tuberías horizontales (fig.5.15):

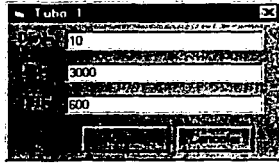


Figura 5.15.- Cuadro de diálogo de tuberías horizontales

Se introducen los valores del diámetro, longitud y temperatura media de flujo para el tubo horizontal es cuestión.

### Cuadro de diálogo de pozos

Ahora, en la edición de pozos (incluyendo tuberías de producción, terminaciones y estranguladores) se tiene el siguiente cuadro de diálogo:

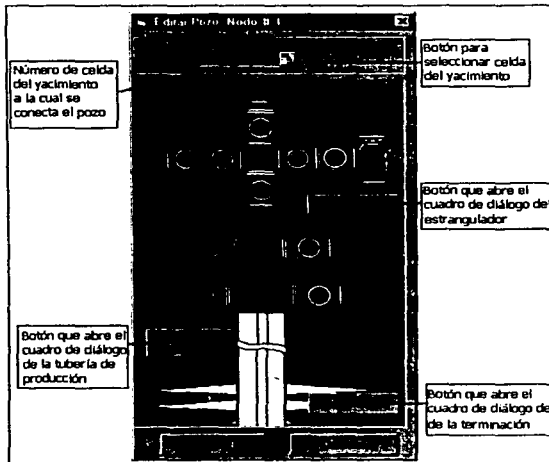


Figura 5.16.- Cuadro de diálogo de edición de los pozos

TESIS CON  
FALLA DE ORIGEN

El botón para seleccionar la celda del yacimiento cambia el modo del cuadro de diálogo (modo selección,) en el cual es posible, la hacer clic en una de las celdas del yacimiento en vista areal, seleccionar la celda interactivamente, más que escribir el número en el espacio correspondiente.

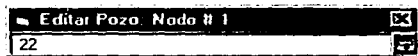


Figura 5.17.- Modo selección del cuadro de diálogo de edición de los pozos

### Cuadro de diálogo de los estranguladores

En este se introduce la temperatura media de flujo del estrangulador, el índice adiabático a esa temperatura, el diámetro del estrangulador en 64avos de pulgada y el coeficiente de descarga (adimensional).

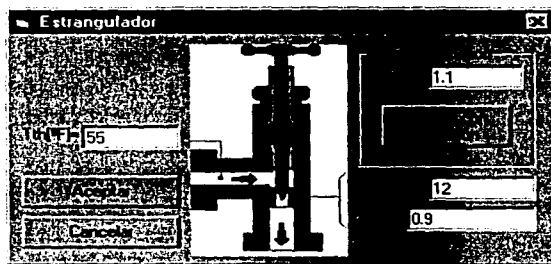


Figura 5.18.- Cuadro de diálogo de los estranguladores.

### Cuadro de diálogo de las tuberías de producción

En este cuadro de diálogo se introduce el diámetro interno de la tubería de producción (en [pie]), su eficiencia de flujo (adimensional) y la rugosidad absoluta por un lado (en [pg]). Por otro el ángulo de inclinación del pozo (en grados) y la longitud de la tubería (en [pie]). Tanto para el ángulo de inclinación como para la longitud de la tubería de

producción, existe la posibilidad de calcularlos de acuerdo a las coordenadas espaciales de la celda del yacimiento y del nodo de la red a los que va conectado el pozo.

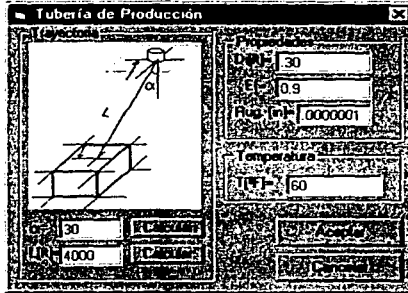


Figura 5.19.- Cuadro de diálogo de la tubería de producción

En la figura 5.19 se observa que hay un botón para calcular el ángulo de inclinación y otro para calcular la longitud. Esto se lleva a efecto para que el usuario pueda decidir si sólo el ángulo o sólo la longitud del pozo, o ambos serán coherentes con la ubicación de las celdas del yacimiento y los nodos. Por ejemplo el usuario podría decidir tomar la longitud que corresponde según la geometría, pero hacer el pozo vertical.

### Cuadro de diálogo de las terminaciones

El objeto terminación, como se vio en la sección de organización de la información, es sólo un conjunto de variables agrupadas convenientemente, ya que en rigor el daño y el coeficiente de flujo no darciano son propiedades del yacimiento. En el cuadro de diálogo se introducen el radio del pozo ( $r_w$  en [pie]), el factor total de daño a la formación y el coeficiente de flujo no darciano (en [d/MPCS], el inverso de millar de pies cúbicos @ C.S. al día)

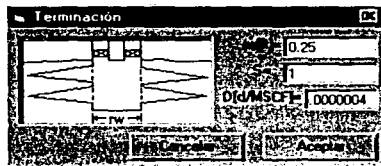
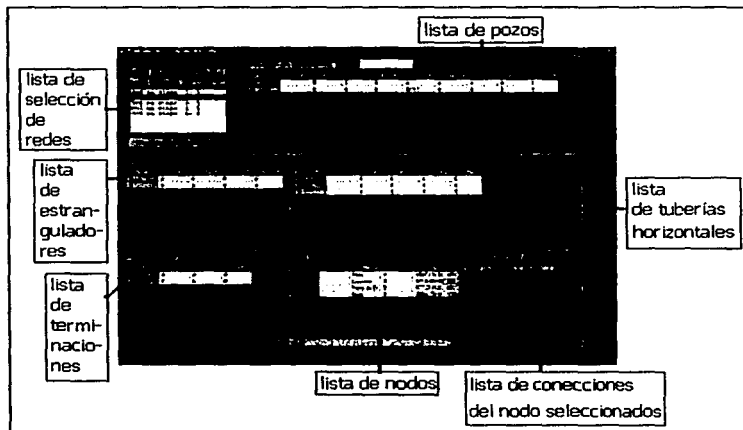


Figura 5.20.- Cuadro de diálogo de las terminaciones

### Cuadro de diálogo general

Además de los cuadros de diálogo ya vistos, existe una pantalla donde se muestra toda la información de las redes y permite la edición de cualquier propiedad de los tubos, nodos, estranguladores, etc... , salvo aquellas propiedades como conexiones de tubos a nodos, que deben ser cambiadas con las herramientas de la figura 5.14.



5.21.- Ventana General para nodos, tubos horizontales, tuberías de producción, terminaciones y estranguladores.

### V.3.4 INTERFAZ GRÁFICA DEL PROGRAMA DE OPERACIONES

En la sección de organización de la información se definieron los objetos programa de operaciones e identificador. En esta sección se discute la manera gráfica que tiene el programa de introducir y modificar la información contenida en dichos objetos, así como un algoritmo que permite obtener al identificador de los nodos de un programa por medio del programa de operaciones.

### Edición de los elementos del programa de operaciones

El programa de operaciones, como se explicó, es básicamente un arreglo de objetos redes (Red\_Superficial) y objetos pozos (arr\_tub\_vert). Cada una de estas redes y arreglos de pozos se editan individualmente con los cuadros de diálogo y algoritmos vistos en la sección de edición de redes. El programa de operaciones se va "poblando" según se le añaden nuevas redes recién creadas. Esto se logra con una variable global del tipo Red\_Superficial y otra de tipo arr\_tub\_vert que estén en sincronía con los objetos gráficos (es decir que cada que se modifiquen los objetos gráficos se modifique acordemente su información y estos últimos sean a su vez un reflejo de aquéllas.) Cuando se desea añadir una nueva red, se redimensionan los arreglos del programa de operaciones y se copia la información de estas variables globales al último elemento de aquéllos. Cuando se desea editar un elemento en específico del programa (red o conjuntos de pozos), se copia su información a las variables globales, se hacen los cambios pertinentes en ellas y luego se copia de vuelta la información ya modificada. La figura 5.22 aclara estos conceptos.

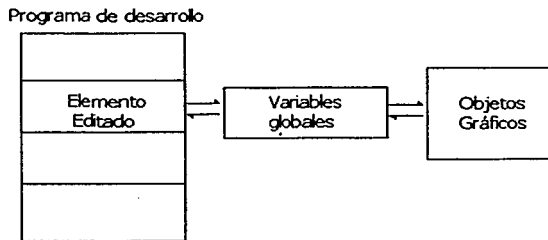


Figura 5.22.- Introducción de redes y pozos al programa de operaciones.

Análogamente a otros diagramas, las flechas en la figura 5.22 representan subrutinas que copian información.

El programa, por medio de botones permite crear una nueva red, "navegar" a través de los elementos del programa de operaciones y registrar los cambios hechos en cualquiera de las redes.

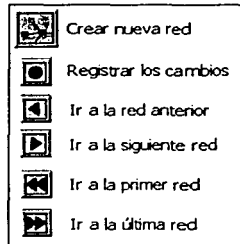


Figura 5.23.- Botones para editar el programa de operaciones

### Obtención del identificador de objetos en el programa de operaciones

El tipo `Identificador_de_Objetos` se definió en la sección de organización de la información. Su objetivo es rastrear la identidad de todos los nodos, tubos horizontales y pozos (`tubo_vertical`.) Puesto que todo elemento de este tipo tiene una propiedad nombre única que lo identifica (respecto a otros elementos de una misma red del programa, pero que se repite en varias redes), es posible "deducir" la información que contiene el identificador de objetos en base al programa de operaciones, o si se quiere ver de otra manera, es un mero reacomodo de la información para que ésta sea más accesible y se requiera menos tiempo para encontrarla (es mejor hacer una sola búsqueda, aunque utilizar espacio para guardarla, que hacer una nueva cada que se requiera)

Algoritmo 5.8.-Obtención del identificador de objetos en base al programa de operaciones

1.-Para  $j=1$  hasta número total de redes

2.- Registra los nombres (abre los registros) de todos los nodos de la red  $i=1$  (no hay contra qué compararlos)

- 3.-Para  $i=2$  hasta el número de nodos en la red  $j$ -ésima
- 4.-Compara el nombre de nodo  $i$  con todos y cada uno de los nombres registrados y abiertos del identificador. Si lo encuentra, especifica que el nombre registrado está representado en la red  $j$  por el nodo  $i$ , de lo contrario registra un nombre nuevo.
- 5.-Siguiente  $i$
- 6.-Cierra los registros abiertos de los nombres que no son nuevos y no estuvieron presentes en la red  $j$
- 7.-Siguiente  $j$
- 8.- Cierra todos los registros de los nombres que al final hayan quedado abiertos.
- 9.-Repetir los pasos 1 a 8 para tubos horizontales y para pozos.

En este algoritmo (5.8) hay algunas cosas que notar: primera es que a lo que se le llama registro corresponde a un tipo `identificador_elemento` (ver definición en la sección de organización de la información.) Cada `identificador_elemento` tiene un arreglo de índices y dos propiedades enteras (`ini` y `fin`). Cuando se dice en 5.8 que se "abre un registro" lo que se hace en realidad es añadir un `identificador_elemento` al arreglo de identificadores de nodos, tubos, o pozos, según lo que se esté tratando y hacer su propiedad `ini` igual a  $j$ . Cuando se dice que se "especifica que el nombre registrado está representado en la red  $j$  por el nodo  $i$ " se está asignando el `Id(j)` del `identificador_elemento` (cuyo nombre sea igual al del nodo  $i$ ) igual a  $i$ . Por último, al decir que se cierra un registro, se hace la propiedad `fin` de ese `identificador_elemento` igual a  $j$  y con ello se indica que no se considerará para nuevas comparaciones en el bucle de  $j$ .

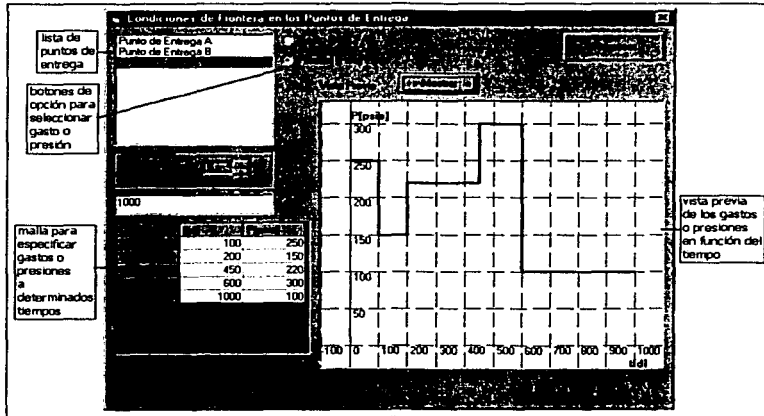
### V.3.5 INTERFAZ GRÁFICA PARA INTRODUCIR LAS CONDICIONES DE FRONTERA

Las condiciones de frontera en los puntos de entrega (sumideros) puede especificarse como presión en función del tiempo o bien gasto en función del tiempo. Cada punto de entrega se identifica en la estructura `Identificador_de_Objetos` por un `identificador_elemento`, sin embargo se requiere de una búsqueda directa en el



programa de operaciones (Programa\_operaciones.) para saber qué nodos son puntos de entrega. Esta búsqueda se realiza una sola vez (con el algoritmo 5.9 que presentaremos aquí) y se guarda en la estructura Condiciones\_de\_Frontera, cuya función es, además de contener la información de las condiciones de frontera es evitar esta búsqueda cada vez que se requiera.

La ventana para introducir la información de las condiciones de frontera da medios de introducir a la presión o el gasto en función del tiempo para cada punto de entrega. Antes de abrir esta pantalla se realiza la búsqueda del algoritmo 5.9. Nótese la vista previa que permite visualizar la gráfica de presión o gasto en función del tiempo.



5.24.- Ventana para introducir las condiciones de frontera

Como es costumbre, la ventana de introducción de datos y las estructuras están sincronizadas con subrutinas que copian información de unas a otras.

El siguiente algoritmo sirve para obtener la dimensión del arreglo `arr_cond` de la estructura `condiciones_de_frontera` en base a la estructura `Identificador_de_Objetos` además de obtener las propiedades `i_in` de cada elemento del arreglo.

Algoritmo 5.9.- Búsqueda de puntos de entrega en la red y dimensionamiento de la estructura de condiciones de frontera.

1.- $i=0$

2.- $i=i+1$

3.-Si el identificador de nodo  $i$  identifica un punto de entrega entonces añade una condición de frontera a la estructura `condiciones_de_frontera`; inicializa su propiedad `Id=i`.

4.-Si  $i$  es menor al número de identificadores de nodos en el identificador regresar al paso 2

### V.3.6 INTERFAZ DE LAS PROPIEDADES PVT DEL GAS

De acuerdo a lo visto en la sección de organización de la información, los datos de viscosidad, factor de compresibilidad y pseudopresión pueden especificarse por tablas o bien por correlación. Si cualquiera de ellos es especificado por correlación, entonces se necesitarán las propiedades pseudocríticas del gas, así como su densidad relativa al aire. Estas propiedades, empero, pueden obtenerse a partir de la composición del gas, si se especifica.

En la siguiente ventana se especifican las propiedades pseudocríticas y valores tabulares se muestra en la figura 5.25.

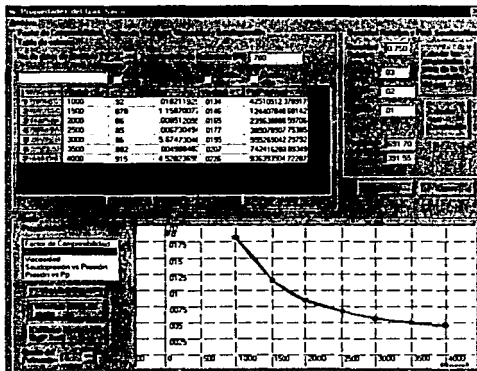


Figura 5.25.- Ventana de Propiedades pseudocríticas y tabulares del gas.

Se explicará a continuación cada una de sus partes.

### Temperatura de referencia y propiedades tabulares

En esta sección se introduce el factor de compresibilidad, factor de volumen, viscosidad y seudopresión en función de la presión (ver figura 5.26.) El número mínimo de presiones a las cuales se reportan propiedades es 2. Para  $Z$ , Viscosidad y seudopresión se tienen además cuadros de opción para especificar si se utilizará correlación para ese dato o bien los valores de las tablas. Si se marca correlación se ignoran los valores en tablas para la propiedad en cuestión.

Por otro lado se tienen botones auxiliares para completar la tabla. Con ellos se pueden calcular propiedades en función de otras como sigue:

- Factor de volumen en función del factor de compresibilidad
- Factor de compresibilidad en función del factor de volumen
- Seudopresión en función de la viscosidad y el factor de compresibilidad

Hay que mencionar que el factor de volumen no se utiliza directamente en los cálculos, pero como es muy común que esté dentro de los datos típicos PVT, el programa permite que se especifiquen y luego en función de ellos se calcula el factor de compresibilidad.

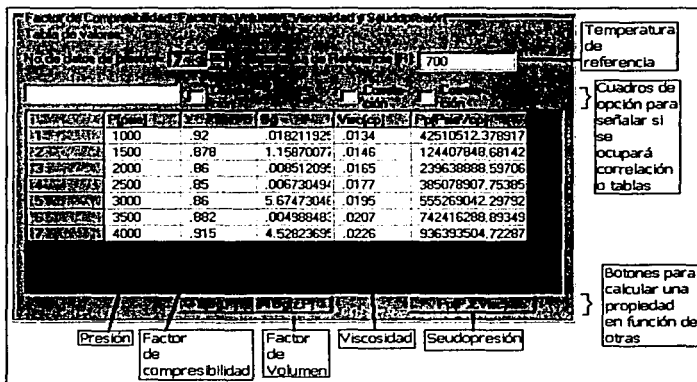


Figura 5.26.- Sección de valores tabulares y temperatura de referencia del gas

En una sección posterior, de este capítulo (sección del Kernel del programa,) se tratarán todas las correlaciones y algoritmos involucrados en los cálculos y obtención de las propiedades del gas.

### Sección de propiedades pseudocríticas, densidad y fracciones molares de componentes no hidrocarburos

En esta sección se introducen todas las propiedades necesarias para alimentar las correlaciones de viscosidad, factor de compresibilidad y pseudopresión.

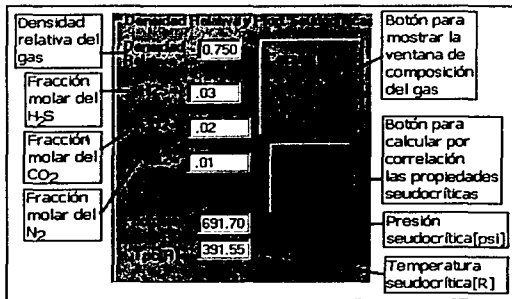


Figura 5.27.- Sección de propiedades pseudocríticas y densidad relativa

Las fracciones molares de los componentes no hidrocarburos son necesarias para hacer correcciones en los valores de viscosidad calculados por correlaciones, lo mismo para calcular las propiedades pseudocríticas si sólo se conoce la densidad relativa del gas. De hecho el botón con el membrete "Calcular con correlación" lleva a efecto dichos cálculos.

El botón para "Calcular las propiedades en función de la composición del gas" abre la ventana donde se especifica la composición del gas (si se conoce) para que en base a ella se calculen los valores de esta sección.

En el apéndice B se presentan en forma de pseudocódigo las correlaciones que se utilizaron en el presente trabajo para la obtención de las propiedades del gas.

### Sección de ajuste de las propiedades con polinomios

El simulador SIMPREG no utiliza búsqueda en tablas (una explicación de esta técnica de interpolación véase en Aziz y Setari(1979)). En vez de esto se ajustan polinomios de las propiedades en función de la presión a la temperatura de referencia. Incluso si se indica que se utilizará una correlación para calcular esa propiedad, se genera internamente una tabla de valores contra la presión y se ajusta un polinomio para esa tabla.

La sección permite al usuario visualizar los datos que ha introducido en las tablas y visualizar la gráfica de un polinomio de grado n para ver la bondad del ajuste. El grado del polinomio lo especifica el usuario.

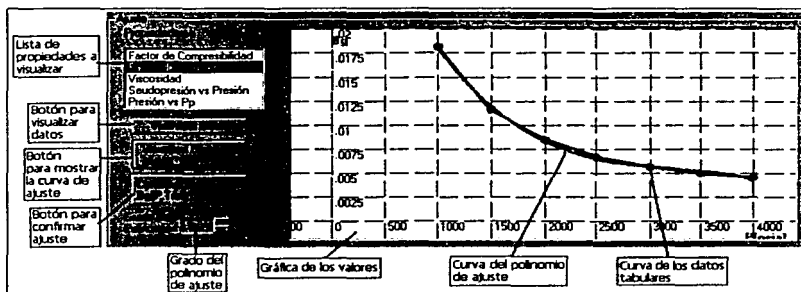


Figura 5.28.- Sección de ajuste de polinomios a valores tabulares o correlaciones

El botón de confirmación del ajuste invoca a subrutinas que realizan un ajuste por mínimos cuadrados de un polinomio a los valores y los almacenan en los arreglos de coeficientes de polinomios de la estructura `Info_PVT`.

La manera como se llevan a cabo los ajustes se tratan en el apéndice B.

### Sección de composición del gas

La composición del gas puede incluir a los compuestos listados en la tabla 5.1, sin embargo, no es necesario que todos estén presentes; de hecho, en la ventana de composición del gas (figura 5.29) se especifica con cuadros de opción cuáles compuestos están presentes y cuáles no.

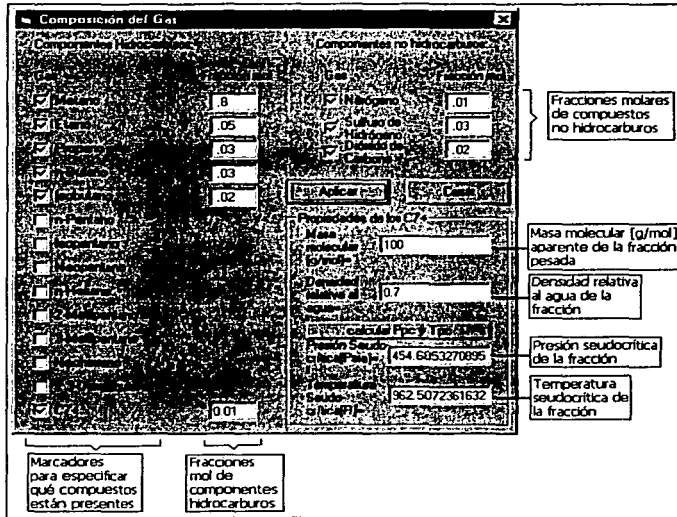


Figura 5.29.- Ventana de composición del gas

En esta misma ventana se especifican las propiedades seudocríticas y masa molecular de la fracción pesada. La densidad relativa al agua de esta fracción es necesaria si se desea calcular por correlación las propiedades seudocríticas de la fracción. Cuando se presiona el botón "Aplicar" se actualizan los cuadros de texto de la sección de propiedades seudorreducidas del gas.

Todas las correlaciones involucradas en estos cálculos se presentan en el apéndice B.

### V.3.7 INTERFAZ DE LOS PARÁMETROS DE LA CORRIDA

La estructura `Parámetros_corrida` es responsable de "decir" a la subrutina que ejecuta el algoritmo 4.1 (algoritmo general) con qué tolerancias, número máximo de iteraciones, cambios máximos en variables y correlaciones se ejecutará la corrida. Estos parámetros se introducen en una ventana (ver figura 5.30) que está sincronizada con la estructura.

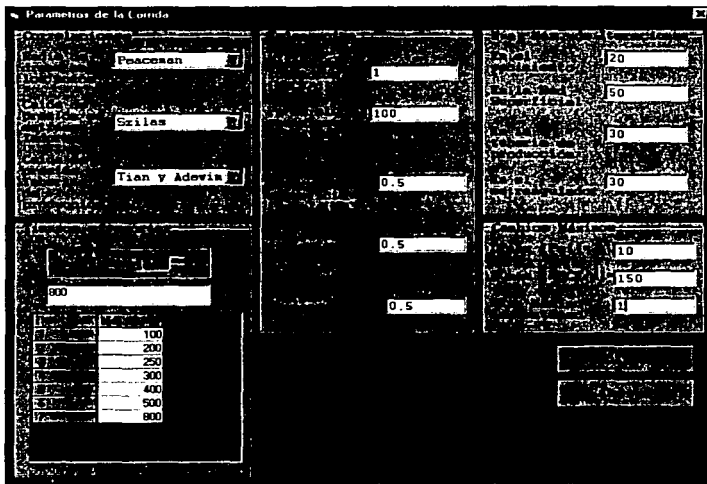


Figura 5.30.- Ventana para introducir los parámetros de la corrida.

Nótese que en la malla de "Tiempos a reportar" se tiene la opción de poner tantos tiempos como uno desee. El estado de la corrida se guardará para esos tiempos y servirá de base para las gráficas del reporte de la corrida.

### V.3.8 INTERFAZ GRÁFICA DE LA CORRIDA DEL SIMULADOR

Una vez que se ha introducido toda la información requerida del diseño de las instalaciones, el yacimiento y el programa de operación, así como las propiedades PVT del gas y los parámetros de la corrida, el usuario puede entrar a la ventana de ejecución de la misma. Esta ventana (figura 5.31) tiene también el propósito de permitir al usuario monitorear el desempeño numérico algunas variables selectas mientras progresa la simulación. De esta manera el usuario está informado de posibles problemas que puedan







- Gasto total de gas del yacimiento ( $q$  [MMPCSD] .)
- Reserva actual ( $G-G_p$  [MMPCSD] .)
- Producción acumulada del yacimiento ( $G_p$  [MMPCSD] .)
- Porcentaje de error de balance de materia acumulado (%MBE Acum.)
- Porcentaje de error de balance de materia instantáneo (%MBE Acum.)
- Presión media del yacimiento ( $P$  [1b/pg2] .)

Los valores, ordenados en tablas, lucen como el siguiente ejemplo:

t	Delt	#It	Qg	G-Gp	Gp	%MBEAcum	%MBEInst	P
0.01	0.02	3	7.66	100,074.2	0.16	-22.88	-2.80	3,159.2
0.03	0.02	3	7.64	100,074.1	0.34	-19.72	-2.41	3,159.2
0.05	0.02	3	7.63	100,073.9	0.52	-13.90	-1.22	3,159.2
0.08	0.02	3	7.62	100,073.7	0.70	-12.04	-1.08	3,159.1
0.10	0.02	3	7.60	100,073.5	0.88	-12.16	-1.08	3,159.1

### V.3.9 GENERADOR DE GRÁFICAS

Los resultados de la corrida, como se ha dicho, son almacenados en la estructura `historia_corrida`. Una manera de interrogar al simulador acerca de los resultados de ésta es por medio de gráficas, por ejemplo presión media del yacimiento contra el tiempo, o la presión de fondo fluyendo de tal o cual pozo contra su gasto. Una vez que se ha decidido mostrar los resultados de la corrida de esa manera, surge la pregunta ineludible de qué variables han de graficarse contra cuáles. La manera como se resolvió la cuestión fue permitir que el usuario eligiera las variables.

Como parte de la interfaz gráfica del simulador SIMPREG se implementó, pues, una ventana que hemos denominado "Generador de Gráficas". En ella el usuario selecciona

de una lista de variables almacenadas en la estructura `historia_corrida` la variable que se graficará en el eje de las abscisas y el eje de las ordenadas.

Puesto que la estructura `historia_corrida` almacena valores puntuales y no curvas, para mostrar estas últimas se unen los puntos con líneas rectas.

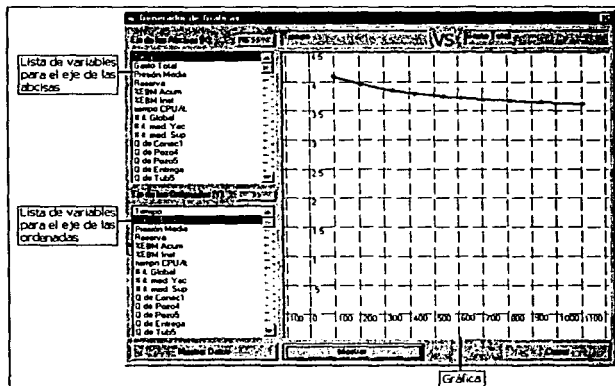
Hay otra cuestión importante que resolver antes de construir una gráfica de esta naturaleza y son los puntos (x,y) inicial y final para graficar, si es que el elemento de las instalaciones de producción del cual se grafica su presión o gasto, no existe durante todos los tiempos almacenados (resultado del programa de operación del yacimiento.) Se debe examinar primero si el tiempo mínimo de corrida (usualmente cero) es mayor al tiempo mínimo en que existe el elemento. De ellos dos se obtiene el valor más tardío. Después se determina el tiempo máximo al cual existe el elemento y se compara con el último tiempo reportado de la corrida. De ellos dos se obtiene el final más prematuro. Estos 2 valores dan un intervalo de tiempo para la variable que serán graficadas. La intersección del intervalo de la variable en X con el de la variable en Y da como resultado el intervalo que se graficará en el generador de funciones.

Se muestra a continuación la lista de variables que es posible graficar:

1. Tiempo [d]
2. Gasto total [MMPCS/d]
3. Presión media [lb/pg<sup>2</sup>]
4. Reserva [MMPCS]
5. %EBM acumulado
6. %EBM instantáneo
7. Tiempo CPU por iteración global[s]
8. Número de iteraciones globales
9. Promedio de iteraciones en el yacimiento
10. Promedio de iteraciones en la red superficial
11. Gasto de los nodos [MPCS/d]
12. Gasto a través de los tubos horizontales [MPCS/d]
13. Gasto de los pozos [MPCS/d]
14. Presiones de los nodos [lb/pg<sup>2</sup>]
15. Presiones en la cabeza de los pozos [lb/pg<sup>2</sup>]

16. Presiones de fondo fluyente [lb/pg<sup>2</sup>]
17. Presiones en las celdas del yacimiento [lb/pg<sup>2</sup>]

En la figura 5.32 se muestra al generador de gráficas



5.32.- Generador de Gráficas

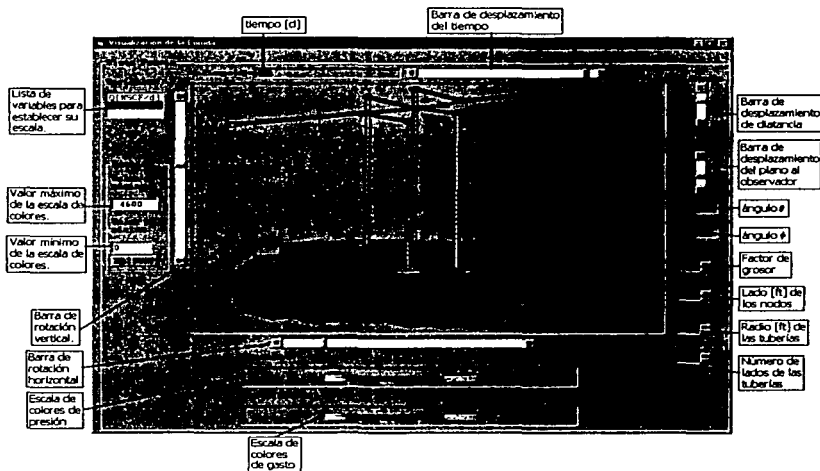
La ventana mostrada en la fig. 5.32 no es la única manera de observar los resultados de la corrida. También se implementó una ventana para mostrar los resultados de la corrida en 3D, semejante a la ventana principal de edición del yacimiento.

### V.3.10 Ventana de visualización en 3D de la corrida

En la ventana para visualizar la corrida en 3D es posible ver una representación gráfica de las instalaciones superficiales acopladas al yacimiento por medio de figuras tridimensionales. Estas figuras tridimensionales pueden rotarse, agrandarse e incluso se les pueden modificar algunas de sus características (como los diámetros de las tuberías, altura de las celdas del yacimiento etc.) para que se distingan mejor o por razones puramente estéticas. En el apéndice A se discute con cierto detalle la derivación de las fórmulas de transformación para proyectar objetos tridimensionales a un plano así como algoritmos para graficar los objetos. En esta sección discutiremos sólo el uso de la ventana

de visualización y cómo mediante una barra de desplazamiento se pueden hacer cambiar los tiempos registrados de la corrida.

Esta ventana (fig. 5.36) posee un cuadro de dibujo donde se grafica, barras espaciadoras para rotar los objetos tridimensionales y barras para aumentar o disminuir su tamaño, según se desee.



5.33.- Ventana de visualización de la corrida en 3D.

Como se aprecia en la figura 5.33, existe una barra de desplazamiento en la parte superior. Ésta tiene como función cambiar el tiempo y mostrar diferentes estados de la presión del yacimiento y de la red según se hayan guardado en la estructura `historia_corrida`. Debe recordarse que los tiempos a los cuales se guardan las presiones y gastos de las celdas del yacimiento y en los elementos de las instalaciones de recolección están determinados por el arreglo `t_rep` de la estructura `Parámetros_corrida` (ver sección V.2.9) Cuando cambia la posición de la "barra de tiempo" (así la denominaremos) cambia también un índice (índice de tiempo actual) que indica qué registro de los arreglos de presiones y gastos de la estructura `historia_corrida` serán utilizados para colorear los objetos gráficos (ver apéndice A.)

Con información combinada de las estructuras `historia_corrida` y `Programa_operaciones` es posible saber qué red del programa de operaciones se debe dibujar acorde al tiempo que se está mostrando (recuérdese que el arreglo `i_red` de la estructura `historia_corrida` da el índice de la red que está funcionando en el momento de guardar los registros de gastos y presiones.)

Las barras de rotación horizontal y vertical hacen cambiar los ángulos desde los cuales el observador vé los objetos ( $\theta$  y  $\phi$ , respectivamente. Ver apéndice A), dando el efecto de rotarlos. La barra de desplazamiento rotulada como "distancia" hace cambiar la distancia del observador al centro, dando el efecto de agrandar o achicar las figuras. La barra rotulada como plano tiene el mismo efecto que la barra "distancia" pero con sentido inverso (es decir, si se sube el cursor de la barra se encogen los objetos mientras que con la barra "distancias" se agrandan.)

El rótulo titulado "grosor" muestra el valor de un factor que afecta la altura de las celdas (esto con el fin de mejorar la visión que uno tiene de ellas, pues su altura no se compara con la altura de los pozos.) El rótulo titulado "DL[pie]" muestra la longitud del lado de los cubos que representan a los nodos de la red superficial. El rótulo titulado "R[pie]" muestra el radio de los cilindros que representan a las tuberías. Finalmente el rótulo titulado "No. lados" muestra el número de lados de los cilindros (el área transversal de los cilindros es un polígono regular de  $n$  caras inscrito en un círculo de radio  $R$ .)

En la lista titulada "propiedades" se puede elegir a la presión o al gasto para mostrar su valor máximo o mínimo en la escala de colores que el usuario puede modificar y/o definir en los cuadros de texto rotulados como "Valor máximo" y "Valor mínimo". Estas escalas de colores se muestran en sus recuadros correspondientes y sirve para colorear los objetos de acuerdo al valor que tengan de presión (en el caso de celdas y nodos) o gasto (en el caso de tuberías.) Para mayor información sobre la graficación en 3D recomendamos al lector que revise el apéndice A.

## V.4 KERNEL DE PROCESAMIENTO DE LA INFORMACIÓN

En la terminología del presente trabajo, entendemos por Kernel (o núcleo) al conjunto de subrutinas que realizan los cálculos más importantes del simulador, es decir que son

necesarios para la ejecución de la corrida. Es importante que el lector revise la sección de organización de la información puesto que las subrutinas utilizan a las estructuras definidas en dicha sección. Las subrutinas están depuradas y reducidas a sus instrucciones primordiales. Dejamos fuera detalles que no fueran relevantes y pudieran complicar u oscurecer innecesariamente la claridad y simplicidad del código.

Se presentan sólo 3 subrutinas, las más importantes del simulador y las declaraciones de muchas otras que son utilizadas por dichas subrutinas. El código es muy estructurado y casi no se pueden identificar instrucciones simples de Visual Basic. La mayoría de las instrucciones son llamadas a subrutinas que realizan alguna tarea con las estructuras. Sin embargo, pensamos que es fácil entender el código si se ha comprendido bien la filosofía de cómo está organizada la información y si se entienden a fondo los algoritmos 2.1, 3.1 y 4.1 (Cálculo de la distribución de presiones en el yacimiento, resolución de la red y algoritmo global.) En muchos casos, ni siquiera se accede directamente a las propiedades de las variables, quedando ocultos los detalles complejos.

#### V.4.1 SUBROUTINA DE RESOLUCIÓN DE LA RED

Recomendamos que el lector revise el algoritmo 3.1. La función básica de esta subrutina es ejecutar dicho algoritmo. Los argumentos que tiene la función son una variable tipo Gas, un vector X de variables de la red, una variable de tipo Red\_Superficial, tolerancia y número máximo de iteraciones. En el vector de variables recibe los valores supuestos de las incógnitas de la red y una vez que las haya encontrado (es decir que haya convergido el método) las devuelve en ese mismo vector. Algunas tareas que le son necesarias a la subrutina son modificar los valores de la red con los valores del vector de variables (X), calcular vectores de funciones del vector de incógnitas (ec. 3.15), calcular la matriz jacobiana (ec. 3.9), encontrar un vector DX (Delta X) resolviendo la ecuación matricial (3.8) y probar que la diferencia de cada variable entre la iteración actual y la pasada quede dentro de la tolerancia (Ec. 3.10) . Las siguientes subrutinas llevan a cabo dichas tareas. Es necesario que las revisemos antes de proceder a la subrutina de resolución de la red.

Subrutina Modifica\_Arr\_Nodos

Declaración:



Sub Modifica\_Arr\_Nodos (X() como Doble, Red como Red\_Superficial)

Descripción:

Modifica los valores de presión o gasto del arreglo de nodos de la red, con los valores del vector x.

#### Subrutina Calcula\_Vector\_F

Declaración:

Sub Calcula\_Vector\_F (g como Gas, X() como Doble, Red como Red\_Superficial, VF() como Doble)

Descripción:

Calcula el vector de funciones del método de Newton (ec. 3.25) en base a las propiedades del Gas g, características de la Red y el vector de incógnitas x. El resultado lo devuelve dentro del vector VF de la misma dimensión que x.

#### Subrutina Calcula\_Jacobiano

Declaración:

Sub Calcula\_Jacobiano (g como Gas, Red como Red\_Superficial, Jacob() como Doble)

Descripción

Calcula la matriz jacobiana en base a las propiedades del Gas g, la Red y lo devuelve en un arreglo bidimensional (Jacob.)

Subrutina Resuelve\_Sistema\_Matricial\_Jacob,VF,DX

Declaración:

Sub Resuelve\_Sistema\_Matricial (Matriz() como doble, Vect\_ind() como doble, Solución() como doble)

Descripción:

Resuelve el sistema matricial  $\text{Matriz} \times \text{Solución} = \text{Vect\_ind}$ . La solución la devuelve el vector solución (El método que utiliza es Decomp y Solve, pero podría ser cualquier otro.)

Subrutina Suma\_Vectores

Declaración:

Sub Suma\_Vectores (A() como doble, B() como doble, factor como doble)

Descripción:

Lleva a cabo la operación vectorial  $A = A + \text{factor} * B$ . El resultado lo devuelve en el vector A.

Función dentro\_tolerancia

Declaración:

Función dentro\_tolerancia(DX() como doble, tol como doble) como booleano

Descripción:

Verifica que el valor absoluto de todos y cada uno de los elementos del vector DX sean menores a tol.

Subrutina principal de resolución de la red.

## Definición:

Sub Resuelve\_red (g como Gas, X() como Doble, Red como Red\_Superficial, tol como Doble, nit como Entero)

Dim i como entero

Dim DX() como Doble

Dim Jacob() como doble

Dim Esc como Doble

Dim ban\_termina como Booleano

Modifica\_Arr\_Nodos X, Red

i = 0

Haz

    i = i + 1

    Si i = 1 O i = 2 Entonces

        Esc = 0.5

    Sino

        Esc = 1

    Fin del Si

Calcula\_Vector\_F(g, X, Red, VF)

Calcula\_Jacobiano (g, Red, Jacob)

Resuelve\_Sistema\_Matricial Jacob, VF, DX

Suma\_Vectores X, DX, Esc

Modifica\_Arr\_Nodos X, Red

ban\_termina = dentro\_tolerancia(DX, tol)

Bucle mientras ban\_termina = Falso Y i < nit

Fin de la Subrutina

**Comentarios:**

La subrutina comienza modificando la red con los valores de las incógnitas proporcionadas por el vector X. Después entra a un ciclo iterativo en donde para cada iteración debe calcular el vector F (ec. 3.15), la matriz jacobiana (ec. 3.9), resolver el sistema matricial de la ecuación (3.8), sumar al vector X el vector DX, modificar la red con los nuevos valores del vector X y finalmente probar si el vector DX entra dentro de la tolerancia. En caso contrario sigue en el bucle hasta que se alcance la convergencia o bien el número de iteraciones sobrepase al número máximo de iteraciones especificado. En el código real hay discriminación para los casos de convergencia o no convergencia, pero no se han presentado aquí por simplicidad. Nótese también que la suma entre X y DX se hace con un factor Esc afectando a DX. Según Stoner(1969) debe ser 0.5 para las primeras 2 iteraciones.

**V.4.2 SUBROUTINA DE RESOLUCIÓN EL YACIMIENTO**

La subrutina de "Cálculo de seudopresiones" (llamada `Resuelve_Yacimiento`) ejecuta el algoritmo 2.1. Procederemos de la misma manera como en la subrutina anterior. Primero presentaremos las declaraciones de las subrutinas secundarias que sean llamadas por la principal y luego presentaremos la definición de esta última.

El algoritmo 2.1 requiere encontrar la distribución de seudopresiones en el yacimiento, los índices de productividad de los pozos (ec. 2.72) y los gastos de cada uno (ec. 2.43).

Subrutina `Calcula_Indices_de_Productividad`**Declaración:**

```
Sub Calcula_Indices_de_Productividad (Yac como Yacimiento, Pozos como  
arr_tub_vert)
```

**Descripción:**

Calcula los índices de productividad de cada celda donde hubiera una terminación ( de acuerdo a Pozos.) Los índices se almacenan en la propiedad J de las celdas del yacimiento Yac.

Subrutina Calcula\_vector\_de\_alfa

Declaración:

Sub Calcula\_vector\_de\_alfa (Yac como Yacimiento, Info\_gas como Info\_PVT, Volumen\_Celdas() como doble, Pp\_tiempo\_anterior() como doble, Pp\_nuevo() como doble, Delt como doble, alfa() como doble)

Descripción:

Tiene como finalidad calcular un vector (arreglo) de constantes  $\alpha$  ("alfa"), dadas por la ecuación (2.44). Para ello es necesario que se pasen a la función como argumentos al Yacimiento (Yac), características del gas (Info\_gas), El volumen de las celdas [(ft<sup>3</sup>) (Volumen\_Celdas), seudopresiones de las celdas en la iteración pasada [(lb/pg<sup>2</sup>)<sup>2</sup>/cp] (Pp\_tiempo\_anterior,) seudopresiones de las celdas en el tiempo actual [(lb/pg<sup>2</sup>)<sup>2</sup>/cp] (Pp\_nuevo) y el intervalo de tiempo [d]. El resultado lo devuelve en el vector alfa.

Subrutina modifica\_Diagonal\_Principal

Declaración:

modifica\_Diagonal\_Principal (Yac como Yacimiento, alfa() como doble, A() como doble, A\_mod() como doble)

Descripción:

En la ecuación :

$$a_{C,i,j}^{n+1} = a_{Ei,j} + a_{Wi,j} + a_{Si,j} + a_{Ni,j} + J_{i,j}^{n+1} + \alpha_{i,j}^{n+1}, \quad (2.50)$$

existen cuatro constantes (llamadas coeficientes de flujo, dados por las ecuaciones (2.46), (2.47), (2.48) y (2.49), respectivamente) y dos variables los índices de productividad y las variables  $\alpha$  ("alfa"). La finalidad de la subrutina modifica\_Diagonal\_Principal es sumar a la adición de esas cuatro constantes el índice de productividad y la variable alfa para cada celda para así obtener el coeficiente de flujo central de cada celda en un arreglo A\_mod. Al argumento A, que debe contener el coeficiente de flujo central, ec.

(2.50), menos el índice de productividad y la variable alfa, le suma el índice de productividad de la celda, extraída de las celdas del yacimiento y la variable alfa de cada celda). El resultado lo devuelve en el vector  $A_{mod}$  sin alterar  $A$ . Se llama "modifica a la diagonal principal" por estar los coeficientes de flujo centrales en la diagonal principal de la matriz del sistema matricial de la ec. (2.52). Ver también la figura (2.3).

#### Subrutina Copia\_de\_un\_arreglo\_a\_otro

Declaración:

Copia\_de\_un\_arreglo\_a\_otro (Pp\_ieración\_anterior () como doble, Pp\_nuevo () como doble)

Descripción:

Copia el contenido de un arreglo a otro arreglo de la misma dimensión. En la subrutina para ejecutar el algoritmo 2.1 la utilizamos específicamente para copiar los valores de pseudopresiones recién calculadas (Pp\_nuevo) a un vector de pseudopresiones de la iteración anterior (Pp\_ieración\_anterior) antes de que Pp\_nuevo.

#### Subrutina Calcula\_Vector\_Términos\_Independientes

Declaración:

Calcula\_Vector\_Términos\_Independientes (Yac como yacimiento, alfa() como doble, Pp\_nuevo() como doble, d\_vect() como doble)

Descripción:

Calcula el vector de términos independientes (ec. 2.51). Requiere al yacimiento (Yac), el vector de "alfas" (alfa), el vector recién calculado de pseudopresiones (Pp\_nuevo). El resultado lo devuelve en el vector d\_vect.

Subrutina Elimina\_LU

## Declaración:

Sub Elimina\_LU (n\_M, A\_it() como doble, B() como doble, C() como doble, E1() como doble, E2() como doble, u\_e1() como doble, u\_e2 como doble, com() como entero, fin como entero, d\_vect() como doble, Pp\_nue() como doble)

## Descripción:

Resuelve el sistema matricial de la ecuación (2.52) para el vector de pseudopresiones (Pp\_nue). Utiliza una serie de vectores (arreglos) que contienen los elementos de la diagonal principal (A\_it), central superior (B), central inferior (C) y los elementos separados a la izquierda y derecha de las diagonales (E1 y E2, respectivamente). Es muy recomendable que el lector revise la sección de desarrollo del modelo numérico del yacimiento y de la solución de matrices por el método LU para familiarizarse con la terminología empleada. Por otro lado, los vectores u\_e1 y u\_e2 así como com y fin son auxiliares en el método y no es recomendable que los expliquemos, pues pueden confundir al lector y no añaden nada esencial a la explicación. El vector de términos independientes está dado por d\_vect. El resultado se devuelve en el vector Pp\_nue.

Subrutina Calcula\_Gastos\_de\_Pozos

## Declaración:

Sub Calcula\_Gastos\_de\_Pozos (Yac como Yacimiento, Pp\_nuevo() como doble, Pozos como arr\_tub\_vert)

## Descripción:

Calcula los gastos de cada pozo empleando información del yacimiento (Yac), las pseudopresiones de las celdas (Pp\_nuevo), y los pozos mismos (Pozos). Los resultados se almacenan en las propiedades q de las celdas que tuvieron una terminación.

Función entra\_en\_tolerancia

Declaración:

Función entra\_en\_tolerancia(Pp\_nueva() como doble,  
Pp\_iteración\_anterior() como doble, tol como doble) como booleano

Descripción:

Determina si cada uno de los elementos de la diferencia de los vectores Pp\_nueva y Pp\_iteración\_anterior entra dentro de la tolerancia tol. Devuelve un valor de verdadero o falso según cumpla o no cumpla con la condición, respectivamente.

Subrutina principal de cálculo de pseudopresiones en el yacimiento

Definición:

Sub Resuelve\_Yacimiento (Yac como Yacimiento, Prop\_PVT como Info\_PVT, A() como Doble, B() como Doble, C() como Doble, E1() como Doble, E2() Como Doble, u\_e1() Como entero, u\_e2() como entero, com() como entero, fin() como Entero, Delt como Doble, tol como Doble, nit como entero, alfa() como Doble, Volumen\_Celdas () como Doble, Pozos() como tubo\_vertical, Pp\_tiempo\_anterior () como Doble, Pp\_nuevo() como Doble )

Dim n\_celdas como Entero  
Dim Contador como Entero

Dim d\_vect() como Doble  
Dim Pp\_iteración\_anterior () como Doble  
Dim ban como Boolean  
Dim A\_it() como Doble

n\_celdas = LSuperior(Yac.cel)

ReDim alfa\_ant(1 To n\_celdas)  
ReDim Pp\_iteración\_anterior(1 To n\_celdas)

ReDim d\_vect(1 To n\_celdas)

Calcula\_Indices\_de\_Productividad Yac, Pozos



```
Calcula_vector_de_alfa Yac, Infogl, Volumen_Celdas, Pp_tiempo_anterior, Pp_nuevo, Prop_PVT
Delt, alfa
```

```
modifica_Diagonal_Principal Yac, alfa, A, A_it
```

```
Contador = 0
```

```
Haz
```

```
Contador = Contador + 1
```

```
Copia_de_un_arreglo_a_otro Pp_iteración_anterior, Pp_nuevo
```

```
Calcula_Vector_Términos_Independientes Yac, alfa, Pp_nuevo, d_vect
```

```
Elimina_LU n_M, A_it, b, C, E1, E2, u_e1, u_e2, com, fin, d_vect, Pp_nue
```

```
Calcula_Gastos_de_Pozos Yac, Pp_nuevo, Pozos
```

```
Calcula_Indices_de_Productividad Yac, Pozos
```

```
calcula_vector_de_alfa Yac, Prop_PVT, Volumen_Celdas, Pp_tiempo_anterior,
Pp_nuevo, Prop_PVT Delt, alfa
```

```
modifica_Diagonal_Principal Yac, alfa, A, A_it
```

```
ban = entra_en_tolerancia(Pp_nuevo, Pp_iteración_anterior, tol)
```

```
Bucle Mientras ban = Falso Y contador <= nit
```

```
Fin de la Subrutina
```

#### Comentarios:

La subrutina `Resuelve_Yacimiento` ejecuta el algoritmo (2.3). Es necesario que el lector tenga cabal comprensión de este algoritmo para entender fácilmente la subrutina. Los argumentos que son "pasados" son: el yacimiento (`Yac`), las propiedades PVT del gas (`Prop_PVT`), los coeficientes de flujo ordenados en las diagonales de la matriz (ec. 2.52), (`A,B,C, E1 y E2`), vectores auxiliares (`u_e1, u_e2, com, fin`), vector de "alfas", (`alfa`), vector de volúmenes de las celdas (`Volumen_Celdas`), arreglo de pozos (`Pozos`), seudopresiones del tiempo anterior (`Pp_tiempo_anterior`) y ,finalmente, intervalo de tiempo (`Delt`), tolerancia (`tol`) y número máximo de iteraciones.

En la subrutina se calcula el número de celdas del yacimiento ( $L_{\text{Superior}}(\text{Yac.cel})$ ) y redimensiona los arreglos de alfa de la iteración anterior ( $\text{alfa\_ant}$ ), seudopresión de la iteración anterior ( $\text{Pp\_iteración}$ ) y el vector de términos independientes ( $\text{d\_vect}$ ). Posteriormente se calculan los índices de productividad y un vector de alfas iniciales con los valores de seudopresión supuestos (contenidos en  $\text{Pp\_nuevo}$ ) y los del tiempo anterior ( $\text{Pp\_tiempo\_anterior}$ ). Se modifica entonces la diagonal principal de la matriz y se guarda en el vector  $\text{A\_it}$ . En este punto entra el método en un ciclo que se repite hasta que los valores de la seudopresión recién calculada y los de la iteración pasada entren en la tolerancia  $\text{tol}$ . En cada iteración del ciclo se copia el valor de  $\text{Pp\_nuevo}$  a  $\text{Pp\_viejo}$  antes de que se modifique en la nueva iteración, se calcula el vector  $\text{d\_vect}$ , se resuelve el sistema matricial de la ec. (2.52) por el método LU, se calculan los gastos y productividades de los pozos y se modifica de nuevo la diagonal principal de la matriz de coeficientes de flujo. Cuando se haya alcanzado la convergencia los valores de gasto de los pozos habrán quedado registrados en la propiedad  $q$  de las celdas del yacimiento y las seudopresiones en el vector  $\text{Pp\_nue}$ .

### V.4.3 SUBROUTINA DEL SISTEMA YACIMIENTO-POZOS-SISTEMA DE RECOLECCIÓN

Esta subrutina ejecuta el algoritmo 4.1. Hace uso de las dos subrutinas precedentes ( $\text{Resuelve\_Yacimiento}$  y  $\text{Resuelve\_red}$ ) para calcular la distribución de presiones en el yacimiento dadas las presiones de fondo fluyente en los pozos y la distribución anterior así como obtener las presiones en las líneas de flujo de los pozos dados sus gastos y condiciones de frontera en los puntos de entrega. Adicionalmente tendremos que presentar las definiciones de más subrutinas (entre ellas la que calcula la presión en la cabeza de los pozos y la que calcula la presión de fondo fluyente.)

#### Subrutina calcula\_radio\_efectivo

Declaración:

Sub  $\text{calcula\_radio\_efectivo}$  (Yac como Yacimiento, Pozos como Arr\_tub\_vert, correlación como entero)

## Descripción:

Calcula el radio efectivo de los pozos en función de las propiedades de las celdas del yacimiento (Yac), propiedades de la terminación de cada pozo (dentro del arreglo de pozos Pozos, y la correlación utilizada (de acuerdo a un número. 1=Peaceman, en primera instancia.) Los radios efectivos de los pozos se registran en la propiedad ro del estrangulador de cada pozo.

Subrutina obtén\_diagonales\_de\_Matriz

## Declaración:

Sub obtén\_diagonales\_de\_Matriz (Yac como Yacimiento, Infog como Info\_PVT, Delt como doble, A() como doble, B() como doble, C() como doble, E1() como doble, E2() como doble, u\_e1() como entero, u\_e2() como entero, com() como entero, fin() como entero)

## Descripción:

Obtiene a partir de las propiedades de las celdas los coeficientes de flujos ordenados en las diagonales (A, B, C), los vectores de elementos separados de las diagonales (E1 y E2) y otros vectores auxiliares (en ellos está almacenada la posición en la matriz de los elementos separados de las diagonales y el comienzo y fin (es decir columnas inicial y final) en cada renglón de los elementos que deben considerarse como diferentes de la matriz.)

Subrutina Calcula\_Pth\_de\_Red

## Declaración:

Sub Calcula\_Pth\_de\_Red(Red como Red\_superficial, Pozos() como Arr\_tub\_vert, Infog como Info\_PVT, Parm como Parámetros\_Corrída)

Descripción:

Calcula la presión en la cabeza de cada pozos (Pth) en base a la presión en la línea de flujo (extraída de las presiones de los nodos tipos pozo de la red), características de los estranguladores de los pozos, propiedades del gas (Info\_PVT). La correlación con que se calculará la caída de presión a través del estrangulador está determinada por Parm.Corr\_est (1=Skilas.) Los valores calculados se registran en la propiedades Pth de cada pozo.

#### Subrutina Calcula\_Pwf\_de\_Red

Declaración:

```
Sub Calcula_Pwf_de_Red(Red como Red_superficial, Yac como Yacimiento,  
Infog como Info_PVT, Pozos() como Arr_tub_vert, Parm como  
Parámetros_Corrida)
```

Descripción:

Calcula las presiones de fondo fluyente de cada pozo y en base a ellas las seudopresiones de fondo fluyente de las celdas donde hubiere una terminación. Para ello es necesaria la Red (para extraer de ella los gastos de los nodos tipo pozo), las propiedades del gas (Infog), características de las tuberías de producción de los pozos (Pozos) y la correlación que se usará para calcular las caídas de presión (especificada en Parm.Corr\_TP, 1=Tian y Adewumi). Las presiones de fondo fluyente se registran en la propiedad Pth de los pozos, mientras que la seudopresión de fondo fluyente se registra en la propiedad Ppwf de las celdas del yacimiento.

#### Función Compara\_Gastos

Declaración:

```
Función Compara_Gastos(Yac como Yacimiento, Red como Red_Superficial,  
Pozos() como Arr_tub_vert, tol como entero ) como booleano
```

## Descripción:

Compara los gastos contenidos en la propiedad  $a$  de cada celda del yacimiento (Yac) con el gasto de la celda tipo pozo de la Red (gasto de la iteración pasada). El arreglo de Pozos (Pozos) es necesario para extraer de él el índice del nodo y de la celda del yacimiento a los cuales va unido cada pozo. Si la diferencia de los gastos actuales con los de la iteración pasada, para todos los pozos, entran dentro de la tolerancia  $\epsilon_01$ , devuelve un valor verdadero, de lo contrario devuelve falso. Sin importar si se cumple o no la condición, se encarga también de asignar los nuevos valores supuestos de gastos de los pozos a la red. La manera más sencilla es asignando los gastos de las celdas del yacimiento a los de los nodos tipo pozo de la red, pero existen otras maneras de hacerlo (ver el capítulo 4 para el "esquema iterativo de Coats(1968)"). De hecho esta versión de la función no incluye los argumentos de las constantes  $\omega$  ("omega"), pero la versión real de la función sí lo incluye. Para detalles revisar el capítulo 4.

Función Principal del Programa

## Definición:

Sub Ejecución\_Corrida(Yac As Yacimiento, InfogC As Info\_PVT, ProgramaC As Programa\_operaciones, IdenC As Identificador\_de\_Objetos, ConDC As Condiciones\_de\_Frontera, histC As hist\_Corrida, Parm As Parámetros\_corrida)

Dim Volumen\_Celdas () como Doble

Dim A(),b(),C(), E1(),E2() como Doble

Dim u\_e1(), u\_e2(), com(), fin() como Entero

Dim alfa(),alfa\_tiempo\_anterior () como Doble

Dim Pp(), Pp\_tiempo\_anterior () como Doble

Dim arr\_Pp(1 To 2) como elemento\_presiones

Dim tiempo como Doble

Dim n\_Celdas, n\_redes como Entero

Dim act como Entero

Dim ant como Entero

Dim contador como Entero

Dim contador\_guardado como Entero

Dim ban como Booleano

```

Dim Delt como Doble
Dim deltax como Doble
Dim deltaxP como Doble

Dim ban_cambio_red como Booleano

Dim X_red() como Doble

n_celdas = LSuperior(YacC.cel)
n_redes = LSuperior(ProgramaC.Redes)

ReDim Volumen_celdas(1 hasta n_celdas)
ReDim Pp(1 hasta n_celdas)
ReDim Pp_ant(1 hasta n_celdas)

ReDim alfa(1 hasta n_celdas)
ReDim alfa_ant(1 hasta n_celdas)
ReDim P(1 hasta n_celdas)

ReDim arr_Pp(1).P(1 hasta n_celdas)
ReDim arr_Pp(2).P(1 hasta n_celdas)
  ant = 1
  act = 2

'// Código para calcular el volumen de las celdas a Volumen_Celdas, inicializar //
'// el arreglo arr_Pp(act) en base a la distribución de presiones iniciales y proponer//
'// valores supuestos iniciales para todas las incógnitas.//

Para i = 1 To UBound(ProgramaC.arr_pozos)
  calcula_radio_efectivo YacC, ProgramaC.arr_pozos(i), Parm.Corr_Pozo
Siguiente i

obten_diagonales_de_Matriz YacC, InfogC, Delt, A, B, C, E1, E2, u_e1, u_e2, com, fin

  Idenc.i_tiempo_actual = 1 'inicialización de índice de tiempo actual.

Delt = 0.01

Haz
  contador = 0
Haz
  contador = contador + 1

  Resuelve_red(InfogC.GasY, X_red, ProgramaC.Redes(Idenc.i_tiempo_actual), Parm.Tol_RS,
Parm.nit_RS)

```

```
Calcula_Pth_de_Red ProgramaC.Redes(IdenC.i_tiempo_actual),
ProgramaC.arr_pozos(IdenC.i_tiempo_actual), InfogC, Parm)

Calcula_Pwf_de_Red ProgramaC.Redes(IdenC.i_tiempo_actual), Yacc, InfogC,
ProgramaC.arr_pozos(IdenC.i_tiempo_actual), Parm)

Resuelve_Yacimiento (YacC, InfogC, A, B, C, E1, E2, u_e1, u_e2, com, fin, n_M, Delt,
Parm.Tol_Yac, Parm.nit_yac, alfa, Volumen_Celdas, ProgramaC.arr_pozos(IdenC.i_tiempo_actual)
, arr_Pp(ant).P, arr_Pp(act).P)

ban = Compara_Gastos(YacC, ProgramaC.Redes(IdenC.i_tiempo_actual),
ProgramaC.arr_pozos(IdenC.i_tiempo_actual).arr, Parm.Tol_Gasto )

Bucle Mientras ban = Falso Y Contador <= Parm.nit_yac

// Código para establecer el siguiente intervalo de tiempo (delt) //

tiempo = tiempo + delt

Si tiempo >= Parm.t_rep(conta_guarda) Entonces
  // Código para guardar la información del último tiempo calculado //

  conta_guarda = conta_guarda + 1

Fin del Si

// Código para asignar condiciones de frontera y cambiar de red de acuerdo //
// al programa de operación. //
// Si cambia de red IdenC.i_tiempo_actual = IdenC.i_tiempo_actual + 1 //

Copia_de_un_arreglo_a_otro alfa_tiempo_anterior, alfa

If act = 2 Then
  act = 1: ant = 2
Else
  act = 2: ant = 1
End If

Bucle Mientras tiempo <= Parm.t_rep(LSuperior(Parm.t_rep))
```

Fin de la Subrutina

#### Comentarios:

Primeramente es necesario señalar, como se ha venido diciendo, que las versiones de las subrutinas han sido simplificadas. Esta en especial carece de muchos detalles que la función verdadera tiene. Se le ha quitado, por ejemplo, el código para imprimir mensajes durante la corrida (ver sección de interfaz de la corrida), código para inicializar variables (por ejemplo el vector de volúmenes de las celdas) ,código de la asignación de valores iniciales supuestos de las incógnitas, cálculo del intervalo de tiempo de la siguiente iteración (ver apéndice C), del guardado de la historia de la corrida e incluso del cambio de esquema iterativo (Ver capítulo 4.), establecimiento de las condiciones de frontera en los puntos de entrega, etc. . Como tal vez ya habrá notado el lector, en muchos de los casos se sustituyó ese código por comentarios (denotados por lo signos '// //) para indicar que en ese lugar deben ir insertados.

Todo esto se hizo para mantener la presentación simple y dejar fuera los detalles que pudieran interferir o distraer en lo esencial de nuestra discusión.

Habiendo hecho esta advertencia podemos proseguir. En primera instancia se calculan los radios de drenaje de todos y cada uno de los pozos, de todas las redes, se obtienen los vectores de las diagonales principales de la matriz de coeficientes de flujo, se inicializa la propiedad de tiempo actual del identificador de objetos (Idenc) y se inicializa el intervalo de tiempo inicial (De1t). Posteriormente se entra a dos ciclos anidados. El primer ciclo sirve para que se haga el balance de gastos entre la red y el yacimiento (es decir convergencia) y el segundo para hacer avanzar al tiempo actual (tiempo) hasta el tiempo final especificado por los parámetros de la corrida (Parm).

En el primer ciclo se resuelve la red, se calculan las presiones en las cabezas de los pozos, las presiones de fondo fluyente y la distribución de pseudopresiones en el yacimiento, así como el gasto aportado por cada pozo según el yacimiento. Posteriormente se comparan los gastos recién calculados en el yacimiento con los que tenían registrados los nodos tipo pozo de la red en la iteración pasada. Se asignan nuevos gastos a éstos últimos (los nodos) y se prosigue en el ciclo si es que no se ha alcanzado la convergencia.



En el ciclo más grande (segundo ciclo) se calculan (a partir del primer ciclo) los gastos de cada pozo, presiones y gastos en los puntos de entrega y distribución de pseudopresiones en el yacimiento para el tiempo actual, después se elige (calcula) el siguiente intervalo de tiempo y se guarda en la historia de la corrida los valores de las variables para el tiempo actual de ser preciso (de acuerdo a `Param.t_rep`). Posteriormente se revisan los tiempos para asignar condiciones de frontera (se asignan si hay cambio en los valores para el tiempo actual) y se revisa el identificador de objetos para revisar si no hay cambio de red en el tiempo actual (de ser así se añade en 1 a `IdenC.i_tiempo_actual`). Lo que resta es copiar los valores de los vectores `alfa` a `alfa_tiempo_anterior` y revisar que el tiempo actual sea menor o igual al tiempo final de corrida (dado por el tiempo del límite superior de `Param.t_rep`)

Es muy posible que haya causado confusión el empleo del arreglo `arr_pp` y dos índices que alternan entre 1 y 2 (`act` y `ant`). Esto sólo se utiliza para alternar, en un arreglo de arreglos de pseudopresiones definido de 1 a 2, el arreglo de pseudopresiones actuales con el arreglo de pseudopresiones del tiempo anterior para evitar tener que copiar los valores actuales a los de la iteración anterior.

## V.5 ALMACENAMIENTO Y RECUPERACIÓN DE LA INFORMACIÓN

La información de entrada del simulador, así como la de salida es muy numerosa. Todos los simuladores sin excepción, tienen la opción de abrir y guardar archivos, pues es altamente ineficiente el tener que introducir datos cada que se abra el programa y se desee correr. Lo mismo podría decirse del resultado de la corrida; Es completamente ineficiente e indeseable tener que hacer una corrida del simulador cada que se requiera analizar la información que éste genera.

Es tan importante el guardado y recuperación de información, que hay incluso algunos simuladores que leen los datos directamente de archivos y escriben en otros el resultado de las corridas sin que medie entre ellos una interfaz gráfica similar a la nuestra, que es tipo Eclipse™.

Cualquier lector que esté interesado en las formas de guardar archivos desde algún lenguaje de programación puede consultar directamente los manuales de dichos

lenguajes. En nuestro caso, como hemos utilizado Visual Basic 6.0 recomendamos el manual del usuario (1995) así como Entsminger(1969).

Nuestro simulador recupera y guarda archivos de forma secuencial. Para el caso de datos previos a la corrida no se inserta información en archivos guardados previamente, sino que se crean nuevos archivos o se destruyen y crean nuevos con el mismo nombre. La información se lee de variables globales, las cuales han leído la información de los objetos gráficos en las ventanas. La norma básica para la escritura y recuperación secuencial es leer los registros en el mismo orden en que fueron guardados. Como el número de los registros es variable, es forzoso establecer una convención para saber por lo menos el orden de los primeros registros claves que contienen el número de registros variables. De esta manera el archivo contiene en sí mismo la información necesaria a la subrutina para que ésta lea el número correcto de registros y asigne los valores en ellos contenidos correctamente a las variables que interactúan directamente con los objetos gráficos.

Al abrir archivos además es necesario actualizar los objetos gráficos (vía variables que "captan" la información de los archivos) para que en la edición de la información se trabaje con los valores recuperados.

El simulador SIMPREG guarda y escribe la entrada de datos en los siguientes tipos de archivos, con las siguientes extensiones:

Tabla 5.3.- Tipos de Archivos.

Información que guarda	Extensión
Yacimiento	.yac
Programa de operaciones	.pde
Identificador de Objetos	.ide
Propiedades pvt del gas	.pvt
Parámetros de la corrida	.par
Condiciones de Frontera	.cfr
Proyecto	.pry

**Archivos .yac**

Guarda la información de la estructura yacimiento. Su formato es como sigue:

```

Número de celdas del yacimiento (n_cel)
Temperatura
Nombre
Profundidad
Para i= 1 hasta n_cel: I_n de la celda i
Para i= 1 hasta n_cel: I_s de la celda i
Para i= 1 hasta n_cel: I_e de la celda i
Para i= 1 hasta n_cel: I_w de la celda i

Para i= 1 hasta n_cel: Dx de la celda i
Para i= 1 hasta n_cel: Dy de la celda i
Para i= 1 hasta n_cel: h de la celda i
Para i= 1 hasta n_cel: Kx de la celda i
Para i= 1 hasta n_cel: Ky de la celda i
Para i= 1 hasta n_cel: Cr de la celda i
Para i= 1 hasta n_cel: fi de la celda i
Para i= 1 hasta n_cel: P de la celda i
Para i= 1 hasta n_cel: Ban_pozo de la celda i

```

**Archivos .pde**

Guarda la información de la estructura Programa\_operaciones. Su formato es el siguiente:

```

Número de Redes (No_redes)
Número de Arreglos de Pozos (No_pozos)
Número de Intérpretes (No_inte)
Para cada una de las redes:
    Número de nodos (No_nodos)
    Para cada uno de los nodos:
        Altura
        Id
        Nombre

```

Presión

Gasto

Coordenada X

Coordenada Y

Número de conexiones del nodo

**Para cada una de las conexiones:**

No. de nodo destino

No de tubo

Número de tubos (No\_tubos)

**Para cada uno de los tubos:**

Diámetro

Eficiencia de Flujo

Longitud

Nombre

Gasto

Temperatura

**Para cada uno de los arreglos de pozos (de 1 hasta No\_pozos)**

Número de pozos

**Para cada uno de los pozos en el arreglo:**

Ángulo

Rugosidad

No. de nodo conectado

No de Celda conectada

Pth

Pwf

**Para el estrangulador:**

Cd

Diámetro

Índice adiabático

Temperatura

**Para la TP:**

Diámetro

Eficiencia de flujo

Longitud

Nombre

Gasto

Temperatura

Diámetro

**Para la terminación:**

Radio efectivo

Radio del pozo

Daño

**Para cada uno de los intérpretes:**

Número de nodo

Incógnita (P ó Q)

**Archivos .ide**

En ellos se guarda la información de la estructura Identificador\_de\_Objetos.

Número de índices a nodos (n\_inodos)

Número de índices a tubos (n\_itubosh)

Número de índices a pozos (n\_itubosv)

Número de tiempos (n\_tiempos)

**Para cada índice a nodo:**

Inicio (ini)

Fin (fin)

Nombre del nodo

Para i=desde ini hasta fin:

Id(i)

**Para cada índice a tubo horizontal:**

Inicio (ini)

Fin (fin)

Nombre del tubo horizontal

Para i=desde ini hasta fin:

Id(i)

**Para cada índice a tubo vertical(pozo):**

Inicio (ini)

Fin (fin)

Nombre del pozos

Para i=desde ini hasta fin:

Id(i)

**Para cada tiempo:**

Tiempo(i)

**Archivos .pvt**

Contiene la información de la estructura Info\_PVT.

**Para i=1 hasta 17:** Fracción molar del compuesto i  
**Para i=1 hasta 17:** Bandera (existe o no) del compuesto i  
**Para la estructura Gas:**  
Densidad relativa  
Masa molecular  
Presión seudocrítica  
Temperatura seudocrítica  
**Para la estructura Heptanosmás:**  
Densidad relativa  
Masa molecular  
Presión seudocrítica  
Temperatura seudocrítica  
Fracción molar  
Bandera de correlación para z  
Bandera de correlación para viscosidad  
Bandera de correlación para presión seudocrítica (en función de la presión)  
Bandera de correlación para la presión (en función de la seudopresión)  
No. de elementos de presión en la tabla (n\_arrP)  
**Para i=1 hasta n\_arrP:** Presión (i)  
No. de elementos de Z en la tabla (n\_arrZ)  
**Para i=1 hasta n\_arrZ:** Z (i)  
No. de elementos de viscosidad en la tabla (n\_arrVisc)  
**Para i=1 hasta n\_arrVisc:** Visc (i)  
No. de elementos de Factor de Volumen en la tabla (n\_arrBg)  
**Para i=1 hasta n\_arrBg:** Bg (i)  
No. de elementos de Seudopresión en la tabla (n\_Pp)  
**Para i=1 hasta n\_arrPp:** Pp (i)  
Grado del polinomio de Pp (grad\_Pp)  
**Para i=1 hasta grad\_Pp:** coeficiente i-ésimo del polinomio de Pp  
Grado del polinomio de P (grad\_P)  
**Para i=1 hasta grad\_P:** coeficiente i-ésimo del polinomio de P

Grado del polinomio de Z (grad\_Z)  
**Para i=1 hasta grad\_z:** coeficiente i-ésimo del polinomio de Z  
Grado del polinomio de Viscosidad (grad\_Visc)  
**Para i=1 hasta grad\_visc:** coeficiente i-ésimo del polinomio de Viscosidad

### Archivos .par

Guarda los parámetros de la corrida (Parámetros\_corrida.)

Máximo cambio en el cambio de la presión de las celdas (DPLIM)

Máximo cambio en el gasto de los pozos (DQLIM)  
Error Máximo instantáneo en el balance de materia (EBMM)  
Tolerancia en el gasto (Tol\_Gasto)  
Tolerancia en el yacimiento (Tol\_Yac)  
%Tolerancia en la red superficial (Tol\_RS)  
Tolerancia en las tuberías de producción (Tol\_TP)  
Tolerancia en los estranguladores (Tol\_Est)  
Número máximo de iteraciones en el yacimiento (nit\_Yac)  
Número máximo de iteraciones en la red superficial (nit\_RS)  
Número máximo de iteraciones en las tuberías de producción (nit\_TP)  
Número máximo de iteraciones en los estranguladores (nit\_Est)  
Número máximo de iteraciones en el balance de gastos (nit\_yac)  
Clave para la correlación en pozos (nit\_RS)  
Clave para la correlación en estranguladores (nit\_TP)  
Clave para la correlación en la tubería de producción (nit\_Est)  
Número de tiempos a reportar (N\_t\_rep)  
**Para i=1 hasta N\_t\_rep:** tiempo a reportar (i)

### Archivos .cfr

Almacenan la información de la estructura Condiciones\_de\_Frontera.

Número de condiciones de frontera (n\_arr\_cond)

**Para cada una de las condiciones:**

Índice en el indentificador de nodos (I\_in)

Bandera de gasto o presión

Número de valores de gastos o presiones, según sea el caso (n\_val)

**Para i=1 hasta n\_val:** Val(i)

Número de tiempos (n\_tiempo)

**Para i=1 hasta n\_tiempo:** tiempo(i)

### Archivos .pry

Los archivos de extensión .pry sólo guardan una compilación de los nombres de los archivos .yac, .pde, .pvt, .par y .cfr, en ese orden.

### Guardado de la información de la corrida del simulador

Como se ha dicho en secciones anteriores, se almacena en la estructura historia\_corrida. La información de esta estructura se guarda a archivo con la extensión .hst. Su formato es:

Número de tiempos a reportar

**Para cada uno de los tiempos:**

Índice de la red que está funcionando (i)

Número de nodos de la red i

**Para cada uno de los nodos:**

Presión del nodo

**Para cada uno de los nodos:**

Gasto del nodo

Número de nodos de la red i

**Para cada uno de los pozos:**

Presión en la cabeza

Presión de fondo fluyente

Gasto

Número de tubos horizontales de la red i

**Para cada uno de los tubos horizontales:**

Gastos del tubo

Número de Celdas en el Yacimiento



Para cada una de las celdas:

Presión de la celda

¶Error de balance de materia acumulado (EBM)

¶Error de balance de materia instantáneo (EBMI)

Tiempo CPU/iteración global (delt CPU)

Número de iteraciones globales (No\_it)

Promedio de número de iteraciones en el yacimiento

(No\_it\_Yac)

## CAPÍTULO VI

# RESULTADOS Y APLICACIONES

### VI.1 COMPORTAMIENTO DE CONVERGENCIA DEL MÉTODO (Algoritmo 4.1)

A continuación se explican una serie de experimentos numéricos que tienen por objeto determinar, cualitativamente, cuál es el efecto de diferentes parámetros en la convergencia del método (algoritmo 4.1) y en la velocidad de éste. No debemos olvidar que el algoritmo 4.1 es una formulación de acoplamiento explícita con condiciones de frontera de Dirichlet en los pozos y los comportamientos que observaremos, son válidas sólo para esta formulación y condiciones de frontera.

Todos los experimentos están basados en un ejemplo sencillo —que explicaremos a continuación— del cual se varían algunos parámetros solamente. Se concluye de la serie de experimentos que altos índices de productividad de los pozos y resistencia de la red (contrapresión), tienen repercusiones graves en la convergencia.

Configuración de la red y el yacimiento para la serie de experimentos (experimentos 1, 2, 3 y 4):

El sistema está formado por un solo pozo ubicado en el centro de un yacimiento cuadrado (ver fig 6.1), con una corta tubería de descarga.

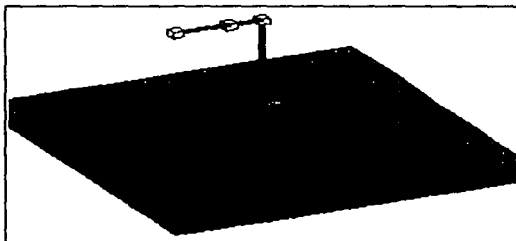


Figura 6.1.- Yacimiento e instalaciones para el Experimento 1.

En las siguientes tablas se resumen las propiedades del yacimiento (es decir iguales en todas las celdas) y los parámetros de la red.

Tabla 6.1.- Propiedades del yacimiento. Experimentos 1, 2, 3 y 4.

$K_x=K_y$	Variable
h	30[pie]
# de celdas	10X10
Dimensiones	1000[pie]X1000[pie]X30[pie]
Presión inicial	Variable
T	700[R]
Profundidad	9000[pie] (a la base)

Tabla 6.2.- Propiedades de la red. Experimentos 1, 2, 3 y 4.

Tubería de Producción (totalmente vertical)			
L[pie]	D[pg]	T[R]	Rug[pg]
9000	3.5	660	$1 \times 10^{-6}$

<b>Estrangulador</b>			
D[pg]	Tth[R]	K	Cd
Variable	600	1.1	0.8
<b>Tuberías horizontales</b>			
	D[pg]	L[pie]	T[R]
Del pozo al nodo	10	1096.97	600
Del nodo al punto de entrega	10	1643.145	600
<b>Terminación</b>			
Daño	Coeficiente de turbulencia		rw[pie]
0	0[d/MPCS]		0.35

## Experimento 1

### Objetivo

Determinar cualitativamente los efectos que tiene la magnitud del índice de productividad  $[(PCS/d)/(lb/pg^2/cp)]$  de los pozos, en el comportamiento de convergencia del método (sin el esquema iterativo de Coats(1968).)

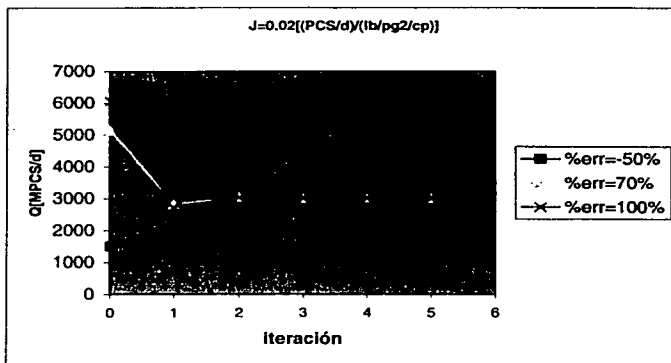
### Descripción del experimento

Se hacen una serie de corridas donde se monitorean los valores de gasto calculado por el yacimiento para un intervalo de tiempo, variando en la configuración del modelo (tablas 6.1 y 6.2) la permeabilidad y los valores iniciales de gasto supuesto para el pozo. Las permeabilidades son manipuladas para dar valores de índice de productividades (Ec. 2.73) controlados. El diámetro del estrangulador para todas las corridas del experimento es 0.5 [pg]. La presión, del yacimiento, es de 2000[lb/pg<sup>2</sup>]. La presión de entrega es de 100[lb/pg<sup>2</sup>].

Tabla 6.3.- Variables del experimento 1.

Serie de corridas	$J[(PCS/d)/(lbpg^2/cp)]$	%Error en gastos iniciales supuestos	Gasto real [MPCS/d]
1	0.02	-50%,70%,100%	3024.97
2	0.05	-50%,50%,70%,100%	5554.88
3	0.1	-50%,50%,70%,100%	8912.79

Como resultados de este primer experimento tenemos las figuras 6.2, 6.3 y 6.4. En ellas se muestran la sucesión de valores de gastos calculados por el simulador del yacimiento a través de las iteraciones.

Figura 6.2.- Experimento 1. Serie de corridas no. 1.  $J=0.02[(PCS/d)/(lbpg^2/cp)]$

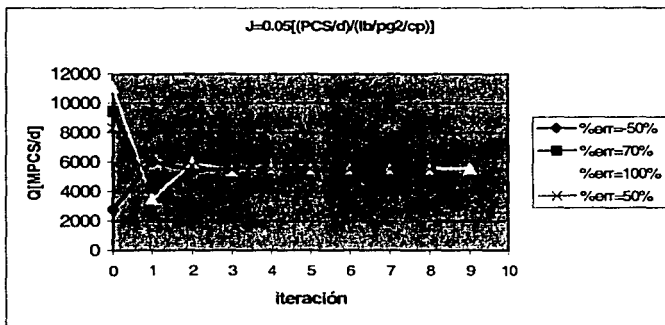


Figura 6.3.- Experimento 1. Serie de corridas no. 2.  $J=0.05[(PCS/d)/(lb/pg^2/cp)]$

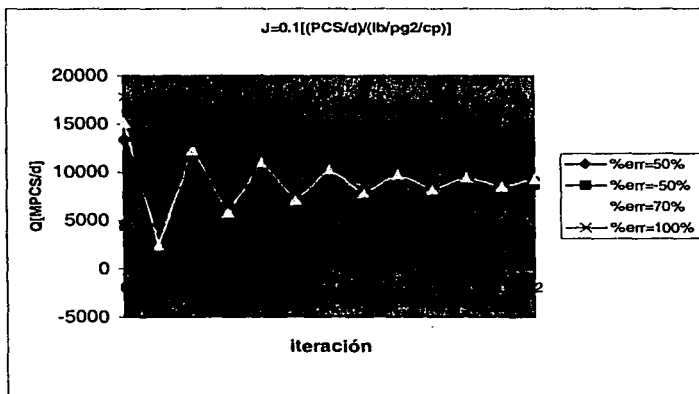


Figura 6.4.- Experimento 1. Serie de corridas no. 3.  $J=0.1[(PCS/d)/(lb/pg^2/cp)]$

Análisis de resultados

El tipo de sucesión que se observa en los gastos es de tipo oscilatoria. Vemos que la amplitud y duración de la oscilación aumenta al aumentar el índice de productividad. Cuando los índices de productividad son bajos, los valores iniciales supuestos para los

gastos no tienen mucho efecto en la velocidad de convergencia: Rápidamente se "amortigua" la oscilación y los gastos entran dentro de la tolerancia especificada. Por otro lado, cuando el índice de productividad de los pozos es grande, las oscilaciones son mucho menos amortiguadas y la amplitud de la oscilación disminuye lentamente. En la gráfica 6.4 se muestra cómo un valor inicial de gasto con +100% de error hace que el método tenga que terminar debido a que en la primera iteración se produce un valor negativo de gasto.

### Conclusiones

Cuando el índice de productividad es grande, las condiciones iniciales desempeñan un papel importante en la convergencia del método (aislando los efectos de diámetros de los estranguladores de los pozos). El método tiene una tendencia oscilatoria que se acentúa a mayores valores de productividad de los pozos.

Una explicación de este fenómeno es que grandes productividades hacen más sensible a un pozo para producir a mayor o menor gasto con pequeñas variaciones de presiones de fondo fluyente. Obviamente las presiones de fondo fluyente (calculadas por las instalaciones) son funciones de los gastos entregados por el yacimiento a las instalaciones en la iteración pasada (iteración pasada del algoritmo 4.1, no confundir con el intervalo de tiempo anterior.) Si las condiciones de frontera en los puntos de entrega son a presión constante, a mayor gasto la presión de fondo fluyendo calculada por las instalaciones aumentará (condiciones de Dirichlet), disminuyendo el gasto aportado por el yacimiento, que su vez causa que disminuya la presión de fondo fluyente calculada y vuelva a aumentar el gasto y así sucesivamente.

## **Experimento 2**

### Objetivo

Determinar cualitativamente los efectos que tiene la magnitud de la presión del yacimiento en el comportamiento de convergencia del método (sin el esquema iterativo de Coats(1968).)

Descripción del experimento

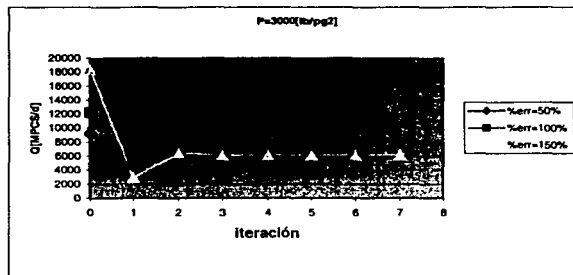
Se hacen una serie de corridas donde se monitorean los valores de los gastos calculado por el yacimiento para cada iteración, variando en la configuración del modelo (tablas 6.1 y 6.2) la presión del yacimiento (uniforme para todas las celdas) y los valores iniciales de gasto supuesto para el pozo. El diámetro del estrangulador para todas las corridas del experimento es 32/64 [pg]. La presión de entrega es 100[lb/pg<sup>2</sup>]. El índice de productividad del pozo es 0.02 [(PCS/d)/(lb/pg<sup>2</sup>/cp)].

Tabla 6.4.- Variables del experimento 2.

# de serie de corridas	P[lb/pg <sup>2</sup> ]	% error en gastos iniciales supuestos	Gasto convergido [MPCS/d]
1	3000	50%,100%,150%	6129.77
2	8000	100%,200%	23684.46
3	15000	100%,200%	45611.98

Resultados del experimento

Los resultados de las series de corridas 1, 2 y 3 se muestran en las figuras 6.5, 6.6 y 6.7.

Figura 6.5.- Experimento 2. Serie de corridas no. 1. P=3000[lb/pg<sup>2</sup>]



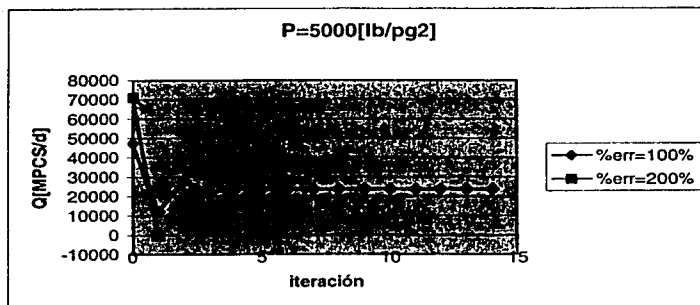


Figura 6.6.- Experimento 2. Serie de corridas no. 2. P=8000[lb/pg<sup>2</sup>]

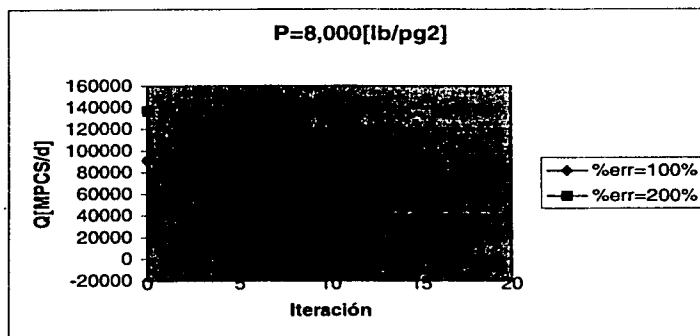


Figura 6.7.- Experimento 2. Serie de corridas no. 3. P=15000[lb/pg<sup>2</sup>]

### Análisis de Resultados

Las gráficas muestran que la tendencia oscilatoria es casi independiente de la presión del yacimiento, aunque hay que señalar que en los casos considerados, una mayor presión de yacimiento equivale a un mayor gasto final o de convergencia y mayores amplitudes de oscilación

(en [MPCS/d]). Este aumento de amplitudes de oscilación resulta ser, no obstante similar en todas las gráficas en cuanto al *porcentaje de error* en cada iteración, que es casi el mismo en las 3 gráficas (para un porcentaje de error inicial dado.)

Vemos que en estos casos para errores de 200% el método debe terminar debido al cálculo de gastos negativos.

Conclusiones

La presión del yacimiento hace mayor la amplitud de oscilación "real" (en [MPCS/d]) del método, pero prácticamente deja intacta la magnitud de oscilación "relativa" (en % de error con respecto al valor convergido.), cosa que no ocurre a mayores índices de productividad.

**Experimento 3**

Objetivo

Determinar cualitativamente los efectos que tiene el diámetro del estrangulador en el comportamiento de convergencia del método (sin el esquema iterativo de Coats(1968), algoritmo 4.2 )

Descripción del experimento

Se hacen una serie de corridas donde se monitorean los valores del gasto calculado por el yacimiento para cada iteración, variando en la configuración del modelo (tablas 6.1 y 6.2) el diámetro del estrangulador y los valores iniciales de gasto supuesto para el pozo. La presión del yacimiento (uniforme en todas las celdas) es 2000[lb/pg<sup>2</sup>]. La presión de entrega es 100[lb/pg<sup>2</sup>]. El índice de productividad del pozo es 0.02 [(PCS/d)/( lb/pg<sup>2</sup> /cp)].

Tabla 6.5.- Variables del experimento 3.

# de serie de corridas	Diámetro de estrangulador [pg/64]	% error en gastos iniciales supuestos	Gasto real [MPCS/d]
------------------------	-----------------------------------	---------------------------------------	---------------------

1	Sin estrangulador	50%,100%,200%	3192.0552
2	64	50%,100%,200%	3189.1093
3	32	50%,100%,200%	3102.2715
4	20	50%,100%,200%	2700.3524
5	16	50%,100%,200%	2311.0291
6	15	50%,100%	2179.8158
7	14.4	50%,100%	2094.0698
8	13.4	50%,100%	1971.6778
9	13.4	5%	1971.6778

Los diámetros del estrangulador se eligieron hasta 13.4/64[pg], porque diámetros más pequeños para este ejemplo presentan el mismo comportamiento que se va acentuado a menor diámetro. Podemos tener cabal idea de dicho comportamiento con la serie de diámetros considerados, sin el inconveniente de determinar valores que sean ilustrativos para diámetros más pequeños.

#### Resultados del experimento

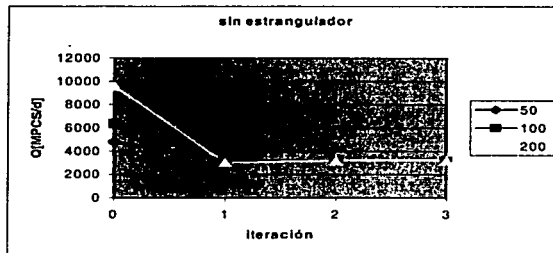


Figura 6.8.- Experimento 3. Serie de experimentos no. 1.  
Comportamiento sin estrangulador

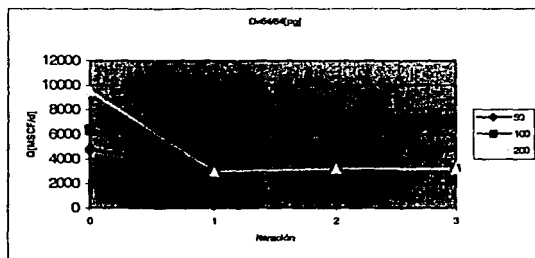


Figura 6.9.- Experimento 3. Serie de experimentos no. 2.  
D=64/64[pg]

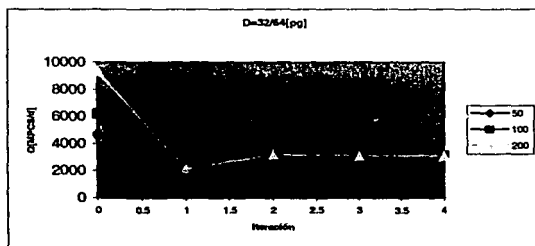


Figura 6.10.- Experimento 3. Serie de experimentos no. 3. D=32/64[pg]

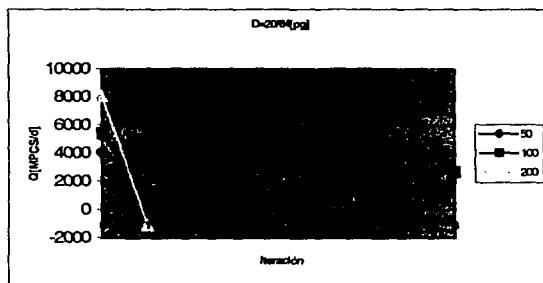
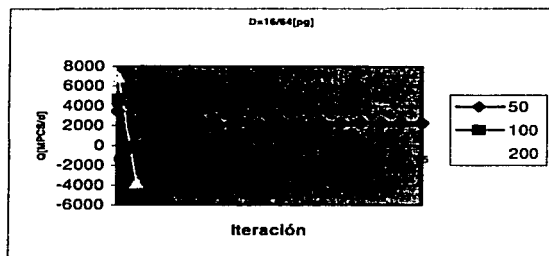
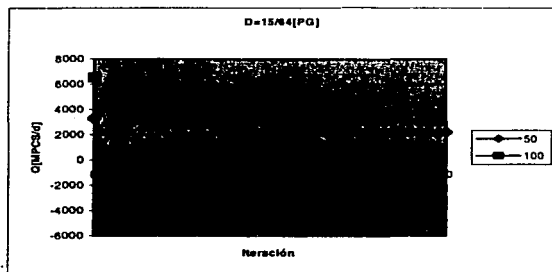
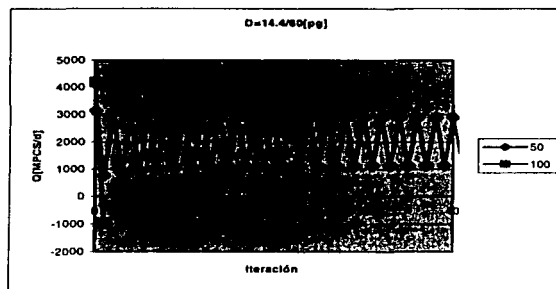


Figura 6.11.- Experimento 3. Serie de experimentos no. 4. D=20/64[pg]

Figura 6.12.- Experimento 3. Serie de experimentos no. 5.  $D=16/64$  [pg]Figura 6.13.- Experimento 3. Serie de experimentos no. 6.  $D=15/64$  [pg]Figura 6.14.- Experimento 3. Serie de experimentos no. 7.  $D=14.4/60$  [pg]

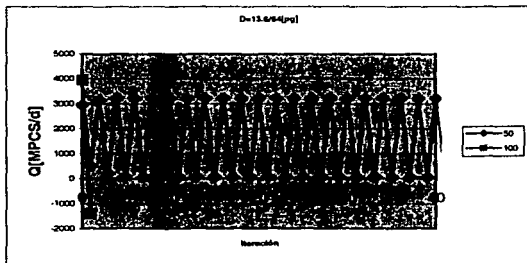


Figura 6.15.- Experimento 3. Serie de experimentos no. 8.  $D=13.6/60$  [pg]

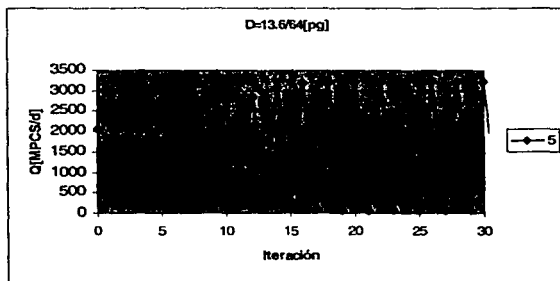


Figura 6.16.- Experimento 3. Serie de experimentos no. 9.  $D=13.6$  [pg], %err inicial=5%

Análisis de resultados

En las figuras 6.8 y 6.9 se observa que el comportamiento del sistema con un estrangulador muy abierto ( $D=1$  [pg]) y sin estrangulador es casi el mismo. Esto es congruente con la realidad, pues diámetros muy grandes causan una caída de presión cada vez menor. De estos dos ejemplos se observa que si no hubiera caídas de presión en la red que hicieran variar la presión de fondo fluyente con el gasto (en este ejemplo las caídas de presión en las tuberías son insignificantes y la presión de entrega fija) el comportamiento tendría la menor oscilación.

Ahora, en la serie de figuras de la 6.10 a 16.13, se observa que a menores diámetros de los estranguladores el comportamiento oscilatorio aumenta, análogamente a lo observado con los índices de productividad, pero un comportamiento no visto hasta este momento comienza a desarrollarse ya en la figura 6.14. La oscilación se hace cada vez menos amortiguada y ya no converge el método o lo hace muy lentamente. En la figura 6.15 se observa que para un error inicial del 50% no hizo sino aumentar para oscilar entre errores absolutos aún más grandes.

Para dilucidar esta situación y observar si el porcentaje de error inicial es el que determina si se queda oscilando permanentemente el método o no, se repitió la corrida con el diámetro de 13.6/60[pg], pero esta vez con un porcentaje de error inicial tan sólo de 5%, figura 6.16. Para nuestra sorpresa, en vez de tender a un valor menor, los gastos oscilaron con amplitud más grande y tendieron asintóticamente hasta los mismos valores de la corrida anterior (con 50% de error). Es aparente que no importa qué tan buena sea la aproximación inicial, el método hará que la oscilación aumente hasta una amplitud constante y oscilará indefinidamente.

### Conclusiones

Los diámetros de los estranguladores desempeñan un papel primordial en la convergencia del método. A menores diámetros el método se vuelve más oscilatorio, hasta divergir sin importar el valor inicial de gastos en los pozos. La mayor tendencia oscilatoria puede explicarse con el hecho de que pequeños diámetros causan grandes caídas de presión, muy sensibles a los cambios de gasto y esto hace que la presión de fondo fluyente varíe considerablemente, causando a su vez disturbios en gastos en los pozos.

### **Conclusión de los experimentos 1, 2 y 3**

Esta serie de experimentos ha mostrado que la velocidad de convergencia y la existencia misma de ésta es dependiente de las características de los estranguladores, productividad de los pozos y hasta en cierta medida las presiones del yacimiento (que más bien dan la "fuerza" a la productividad de los pozos para variar el gasto). Diámetros pequeños de estranguladores y pozos de alta productividad combinados, constituyen

ciertamente una mala combinación para la convergencia del método. Por otro lado, las características observadas por efectos de los estranguladores no se limitan a éstos solamente; Pudimos haber hecho el experimento 3 con un gasoducto lo suficientemente largo y estrecho como para sustituir o acentuar la variación de la presión de fondo fluyente, con los cambios de gasto debida a los estranguladores. Es aparente entonces que entre mayor resistencia oponga la red (contrapresión) por unidad de gasto (cualquier configuración que ésta tuviere), se hará más inestable el método cuando en los puntos de entrega es la presión la que está especificada. Como se explicó en el capítulo 4, existe un método para mejorar la velocidad de convergencia del algoritmo 4.1 y es lo que hemos llamado el "esquema iterativo de Coats(1968)" (algoritmo 4.2) . El experimento siguiente (experimento 4) tiene por objeto mostrar cómo el algoritmo 4.2 aumenta la velocidad de convergencia y evita en muchos casos que diverja el método.

#### **Experimento 4**

##### Objetivo

Determinar cualitativamente los efectos que tiene esquema iterativo de Coats (algoritmo 4.2) en el algoritmo 4.1 y qué comportamiento(s) de convergencia puede(n) esperarse al aplicar dicho esquema.

##### Descripción del experimento

El experimento es exactamente igual al experimento 3 (ver tabla 6.5), sólo que se aplica el esquema iterativo de Coats(1968) (ver capítulo 4). No se muestran todas las gráficas ya que el método no tiene efecto sino hasta la 5ª iteración. Para los casos de no estrangulador,  $D=64/64$ [pg] y  $32/64$ [pg] el método original converge antes de que entre en vigor el esquema. Para el último caso ( $D=13.6/64$ [pg], %error inicial=5%) no es tan sobresaliente la gráfica como para %error inicial=50%. Por estas razones sólo se muestran las gráficas para las corridas con  $D=20/64$ [pg],  $16/64$ [pg],  $15/64$ [pg],  $14.4/64$ [pg] y  $13.6$ [pg] (50% error inicial), figuras 6.17 a 6.21.



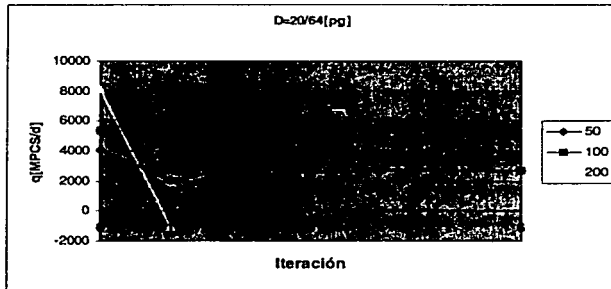
Resultados del experimento

Figura 6.17.-Experimento 4. Serie de corridas no.1.  
D=20/64[pg].

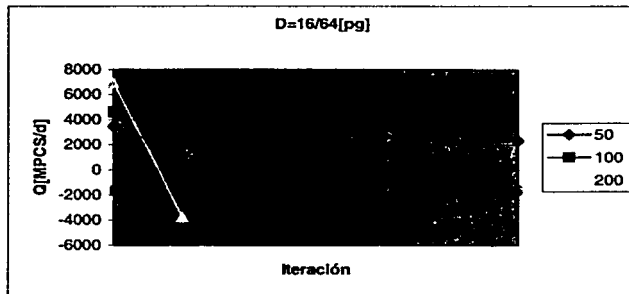


Figura 6.18.- Experimento 4. Serie de corridas no.2.  
D=16/64[pg].

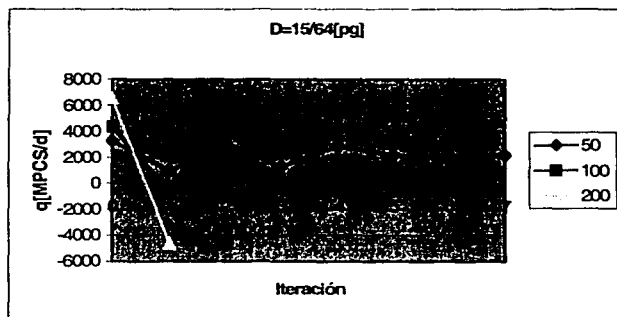


Figura 6.19.- Experimento 4. Serie de corridas no.3.  
D=15/64[pg].

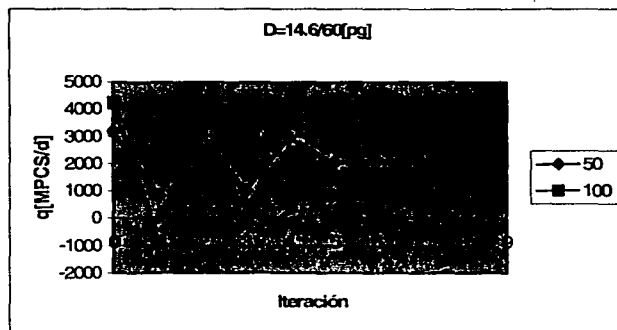


Figura 6.20.- Experimento 4. Serie de corridas no.4.  
D=14.4/64[pg].

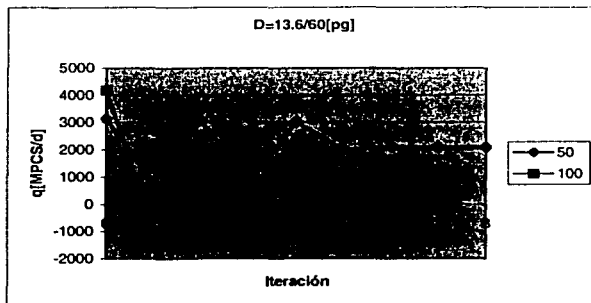


Figura 6.21.- Experimento 4. Serie de corridas no.5.

D=13.6/64 [pg].

#### Análisis de resultados

Antes de discurrir acerca de las gráficas arriba presentadas, hay que tener siempre presente que el esquema iterativo en condiciones normales no entra en vigor sino hasta la cuarta iteración, por requerir el método del valor de las 3 primeras iteraciones (1, 2 y 3). En este experimento numérico en particular, hemos permitido que el esquema entre en vigor una iteración después de la que debería (en la 5ª), para que se aprecie mejor la tendencia inicial antes del método, en contraste con la que se aprecia con el esquema ya en vigor.

En las figuras 6.19 y 6.20 el método original tiene poca tendencia oscilatoria y rápidamente converge. No son muy notorios los beneficios del esquema de Coats, sin embargo, conforme el diámetro de estrangulador disminuye (figura 6.20), se hace más conspicuo cómo a la quinta iteración se fija el gasto a un valor casi intermedio entre los extremos de la oscilación y después converge hasta una tolerancia muy pequeña. El ejemplo más ilustrativo de cómo el método puede hacer que se alcance la convergencia cuando por el método normal era imposible es con  $D=13.6/64$  [pg]. Si contrastamos las figuras 6.15 y 6.16 con la 6.21, salta a la vista la diferencia tajante entre la oscilación natural causada por el estrangulador y la rápida convergencia una vez estimados el valor de convergencia del gasto para el pozo ( ec. 4.20) y el factor  $\omega$  (ec. 4.24).

Es de notarse que por aplicarse el método hasta la 4ª iteración, no es posible evitar que el algoritmo 4.1 pueda divergir antes de esta iteración. En todos los casos sigue habiendo valores iniciales que causan que a las primeras iteraciones se calculen gastos negativos.

### Conclusiones

Ciertamente el algoritmo 4.2 acelera la convergencia del algoritmo 4.1. Incluso para casos donde diverge o converge lentísimamente puede obtener la convergencia en 2 o 3 iteraciones después de aplicado. Sin embargo es necesaria alguna corrección para los casos en que los valores de los gastos se van a valores negativos. Esto podría lograrse fijando arbitrariamente el valor del gasto igual a 1 [MPCS/d] u otro valor pequeño cuando el gasto sea menor a cero; sin embargo, esto causaría disturbios en la observación de los valores "naturales" obtenidos con el algoritmo 4.1 y no se estimaría bien el gasto de convergencia ni el factor  $\omega$ .

### **Comentarios acerca de la inestabilidad del método usado en el presente trabajo**

La inestabilidad del método a causa de los estranguladores es ya bien conocida. Schiozer(1994) señala que ya sea la formulación del acoplamiento yacimiento-instalaciones implícita o explícita, el uso de las condiciones de frontera de Dirichlet es la forma menos idónea para iterar cuando hay estranguladores; analizó los 2 tipos de condiciones de frontera y observó que sin estranguladores las condiciones de Newman tienen oscilaciones y las de Dirichlet no, y cuando hay estranguladores es a la inversa.

Analizó también para los casos en que las condiciones de frontera en los puntos de entrega fueran gastos y no presiones y llegando a la conclusión de que para ese caso la tendencia oscilatoria es muchísimo menos pronunciada y las instalaciones, en vez de causar oscilaciones no hacen sino dividir el gasto según la capacidad de cada pozo.

La aplicación del algoritmo 4.2 resuelve parcialmente el problema de la convergencia. Tal vez la mejor solución sería resolver para la formulación explícita con las condiciones de Newman o Dirichlet alternadamente dependiendo de los diámetros de los estranguladores y productividades de los pozos, aunque podría intentarse algo con

cráteros para fijar los gastos de los pozos cuando el método lleva a valores negativos de éstos.

## VI.2 EJEMPLO DE UN CAMPO HIPOTÉTICO

Presentamos un ejemplo de un caso hipotético para ilustrar la operación del simulador.

Se trata de un yacimiento con forma irregular, relieve estructural y con cierta heterogeneidad y anisotropía (ver figuras 6.22, 6.23 y 6.24). Se perfora una serie de pozos a diferentes tiempos (figuras 6.25, 6.26, 6.27, 6.28, 6.29, 6.30 y 6.31), a la par que se añaden troncales a la red de recolección. En este caso la red de recolección tiene un solo punto de entrega (llamado "Punto de Entrega A", fig. 6.32), el cual tiene especificada una presión de 1000[lb/pg<sup>2</sup>] (presión de entrega. Fig. 6.33). En las figuras 6.34 a 6.38 se muestran las propiedades de los diferentes componentes de la red cuando ya han sido instalado todos. En cada etapa se añade un pozo con el número de esa etapa.

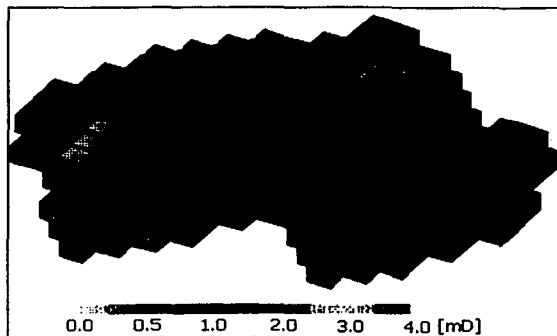


Figura 6.22.- Distribución de permeabilidades en X.

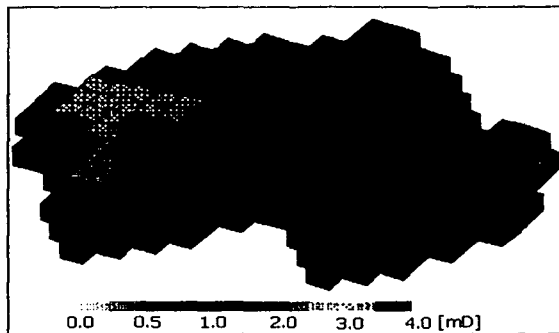


Figura 6.23.- Distribución de permeabilidades en Y.

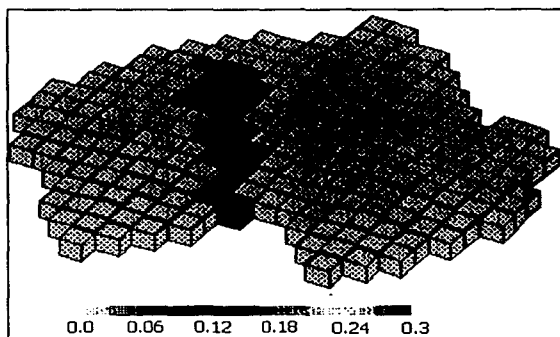


Figura 6.24.- Distribución de porosidades.

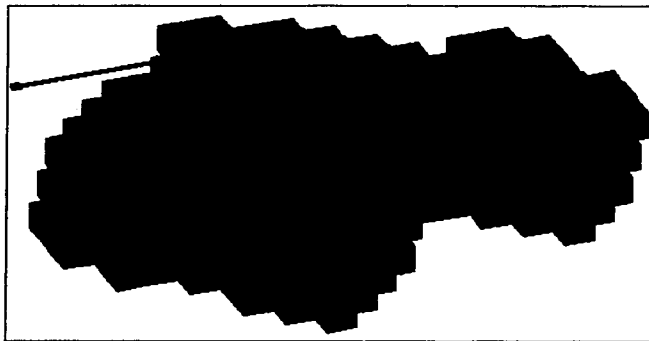


Figura 6.25.-Estado de la red 1/8

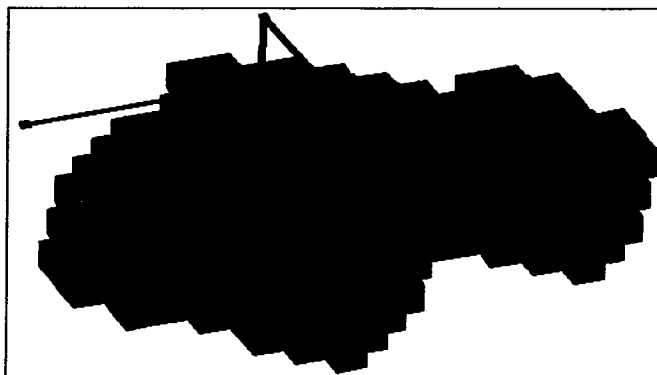


Figura 6.26.- Estado de la red 2/8.

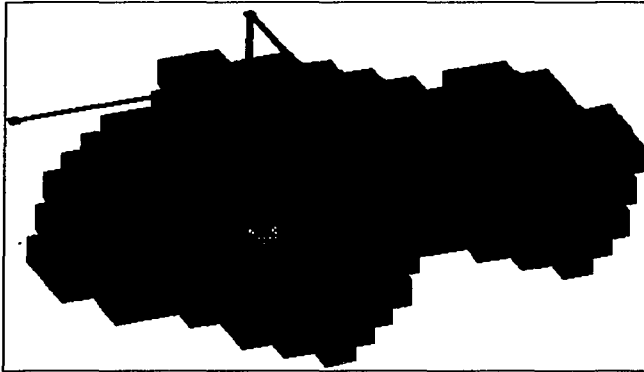


Figura 6.27.- Estado de la red 3/8

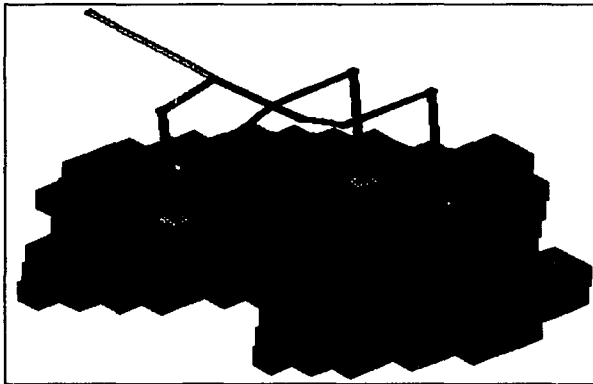


Figura 6.28.- Estado de la red 4/8



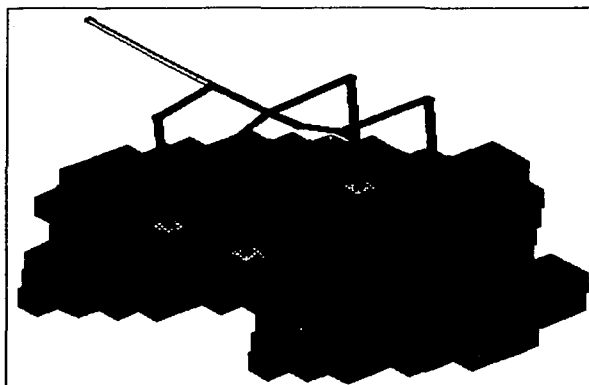


Figura 6.29.- Estado de la red 5/8

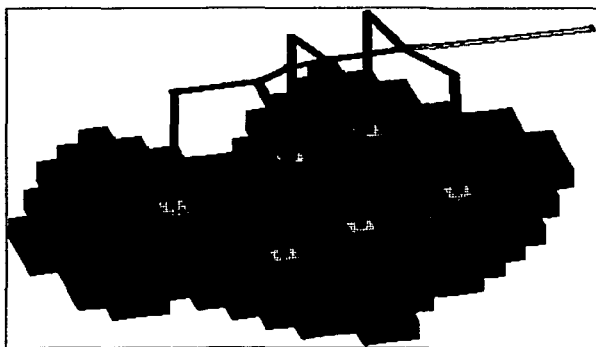


Figura 6.30.- Estado de la red 6/8

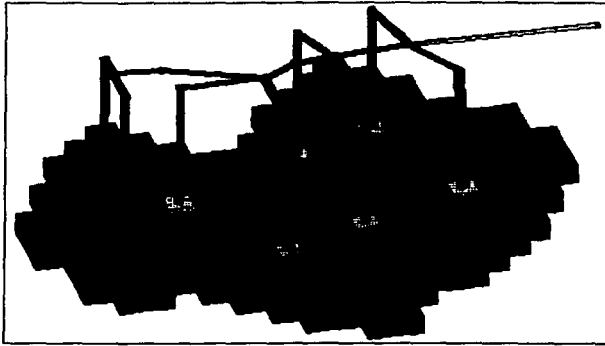


Figura 6.31.- Estado de la red 8/8

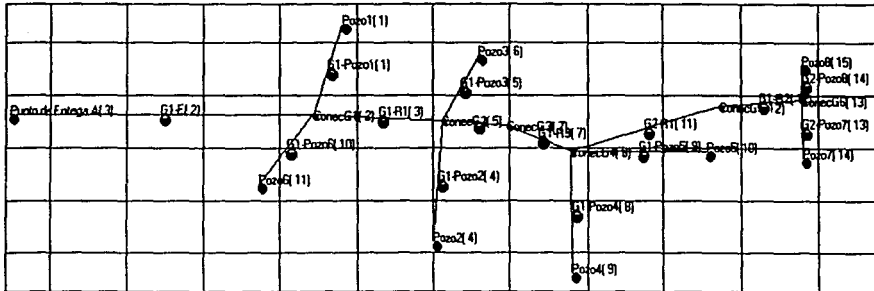


Figura 6.32.- Red (8/8) en vista areal.

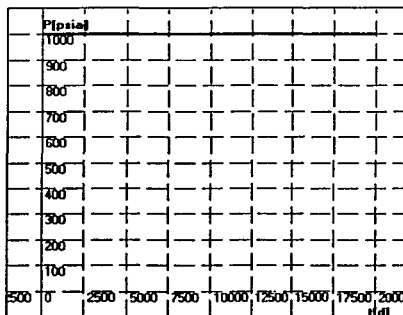


Figura 6.33.- Presión en el punto de entrega A.

Pozo	Diámetro	Profundidad	Producción	Presión	Temperatura	Velocidad	Factor de Compresión	Factor de Expansión
Pozo1	4	9701.49901	1	634.85	88.9927757	.0006	186	1
Pozo2	4	9702.36617	1	634.85	88.7345910	.0006	57	4
Pozo3	4	9703.34077	1	634.85	88.4964676	.0006	171	6
Pozo4	4	9701.44444	1	634.85	89.0112768	.0006	43	9
Pozo5	4	9701.70495	1	634.85	88.9256542	.0006	117	10
Pozo6	4	9701.34234	1	634.85	89.0468568	.0006	88	11
Pozo7	4	9700.29174	1	634.85	89.5556287	.0006	119	14
Pozo8	4	9700.19995	1	634.85	89.6321708	.0006	178	15

Figura 6.34.- Propiedades de los pozos.

Diámetro	Producción	Presión	Temperatura	Factor de Compresión
32	519.67	1.1	.85	
32	519.67	1.1	.85	
32	519.67	1.1	.85	
32	519.67	1.1	.85	
32	519.67	1.1	.85	
32	519.67	1.1	.85	
32	519.67	1.1	.85	
32	519.67	1.1	.85	
32	519.67	1.1	.85	

Figura 6.35.- Propiedades de los estranguladores.

Terminación	D(d/MSCF)	S(m³/d)	rw(ft)
0		-1	.58
0		-1	.58
0		-1	.58
0		-1	.58
0		-1	.58
0		-1	.58
0		-1	.58
0		-1	.58

Figura 6.36.- Propiedades de las terminaciones.

Nodo	WGS	WGS	WGS	WGS	WGS
Pozo1	Pozo	0	3811.3151E	8320.82031	
ConecG1	Conector	0	3440.8666E	6545.30517	
Punto de Er	Punto de Er	0	2954101563	6545.30517	
Pozo2	Pozo	0	4998.0932E	4071.0317E	
ConecG2	Conector	0	5126.8417E	6428.3212E	
Pozo3	Pozo	0	5590.8901E	7679.2949E	
ConecG3	Conector	0	5977.5307E	6332.0927E	
ConecG4	Conector	0	6780.3481E	5835.1416E	
Pozo4	Pozo	0	6795.0454E	3447.7463E	
Pozo5	Pozo	0	8552.8984E	5826.4199E	
Pozo6	Pozo	0	2732.7490E	5213.1977E	
ConecG5	Conector	0	8721.5361E	6712.6049E	
ConecG6	Conector	0	9785.5654E	6853.9707E	
Pozo7	Pozo	0	9798.9169E	5731.4492E	
Pozo8	Pozo	0	9785.3867E	7500.4394E	

Figura 6.37.- Propiedades de los nodos.

Nodo	WGS	D (ft)	WGS	WGS	WGS	
G1-Pozo1	4		1812.8638963E	1	519.67	Tubo
G1-E	24		3981.4716796E	1	519.67	Tubo
G1-R1	24		1687.0202980E	1	519.67	Tubo
G1-Pozo2	4		2363.8727125E	1	519.67	Tubo
G1-Pozo3	4		1967.2964301E	1	519.67	Tubo
G1-R2	24		859.89913831E	1	519.67	Tubo
G1-R3	24		945.44631025E	1	519.67	Tubo
G1-Pozo4	4		2387.0214693E	1	519.67	Tubo
G1-Pozo5	4		1774.2699506E	1	519.67	Tubo
G1-Pozo6	4		1508.2087909E	1	519.67	Tubo
G2-R1	8		2132.5636170E	1	519.67	Tubo
G1-R2	8		1074.7757530E	1	519.67	Tubo
G2-Pozo7	4		1125.0558878E	1	519.67	Tubo
G2-Pozo8	4		646.46982457E	1	519.67	Tubo

Figura 6.38.- Propiedades de las tuberías horizontales.

En la figura 6.39 se muestran los tiempos a los cuales se van añadiendo los pozos a la red.

	t[d]
	150
	300
	450
	600
	800
	1200
	1250
	20000

Figura 6.39 Tiempos a los cuales deja de funcionar  
Cada red.

Debe recordarse que el tiempo asignado a cada red corresponde al tiempo en que deja de funcionar la red. El tiempo al cual comienza a funcionar la red  $i$  es al tiempo  $i-1$ . El tiempo  $i=-1$  se define como 0[d]. La presión inicial del yacimiento es 2.135[lb/pg<sup>2</sup>]. Su profundidad es de 9.700[pies] y su temperatura de 750[R] (290[F]). La compresibilidad de la roca (uniforme en todo el yacimiento es 0.00003[pg<sup>2</sup>/lb]) El volumen inicial de gas a c.s. es de 27,500 millones de pies cúbicos. Cada una de las celdas mostradas en las figuras 6.22 a 6.32 es de 600[pies]X600[pies]. Los espesores varían de 25[pies] a 44[pies]. Como se observa en la tabla 6.34, todas las tuberías de producción tienen un diámetro interno de 4[pg]. El objetivo de la corrida es conocer la distribución de presiones en el yacimiento así como los gastos de cada pozo y el gasto total como función del tiempo.

Los parámetros de la corrida son:

- Tolerancia de gastos de los pozos: 2[MPC/d]
- Tolerancia de pseudopresiones en el yacimiento: 10[lb<sup>2</sup>/pg<sup>4</sup>/cp]
- % de tolerancia en la red: 0.05%
- Tolerancia de presiones de fondo fluyente: 0.01 [lb/pg<sup>2</sup>]
- Tolerancia de presión en la cabeza del pozo: 0.01 [lb/pg<sup>2</sup>]
- Cambio máximo permitido de presión: 5[lb/pg<sup>2</sup>]
- Cambio máximo permitido de gasto de los pozos: 10[MPCS/d]

El ejemplo se corre desde 0[d] hasta 3600[d]. A continuación se muestra la distribución de presiones tomada cada 200[d] (figuras 6.40 a 6.42). Se puede notar en ella además la sucesión de etapas de la red.

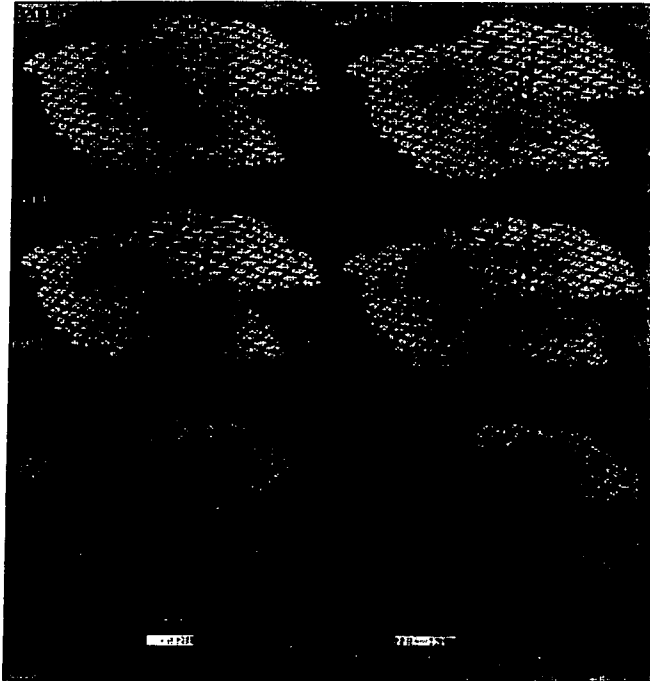


Figura 6.40.- Distribución de presiones para  $t=200, 400,$   
 $600, 800, 1000$  y  $1200[d]$

**TESIS CON  
FALLA DE ORIGEN**

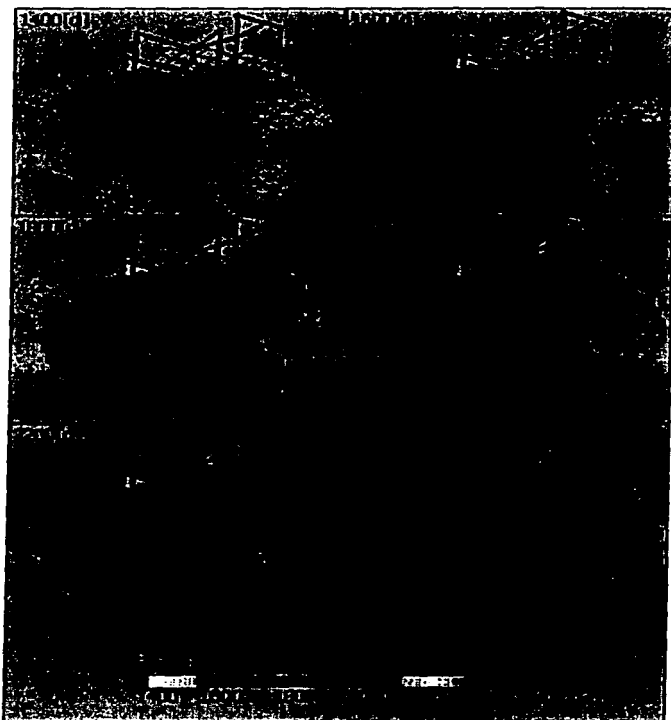


Figura 6.41.- Distribución de presiones para  $t=1400, 1600,$   
 $1800, 2000, 2200$  y  $2400[d]$

FALLA DE ORIGEN  
TESIS COM

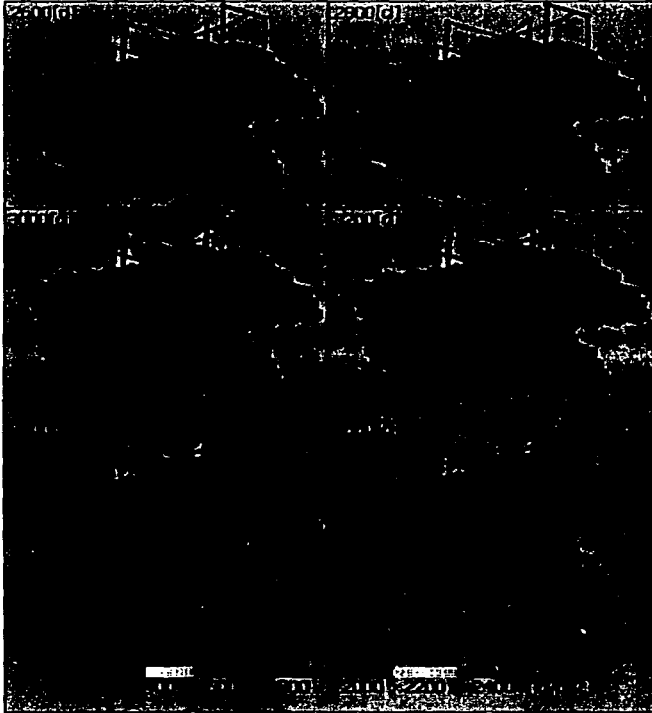


Figura 6.42.- Distribución de presiones para  $t=2600, 2800, 3000, 3200, 3400$  y  $3600[d]$



A continuación se muestran algunas gráficas, (figuras 6.43 a 6.45) como son la reserva contra el tiempo, el gasto total contra el tiempo y presión media contra el tiempo, todas generadas con el simulador SIMPREG.

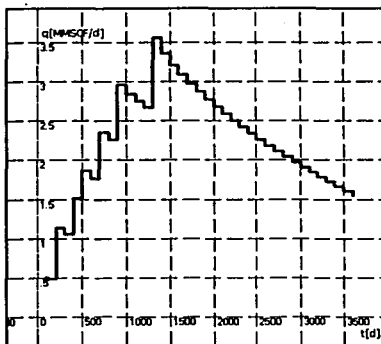


Figura 6.43.- Gasto total contra tiempo.

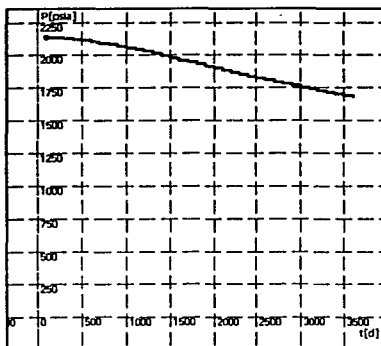


Figura 6.44.- Presión media [lb/pg²]

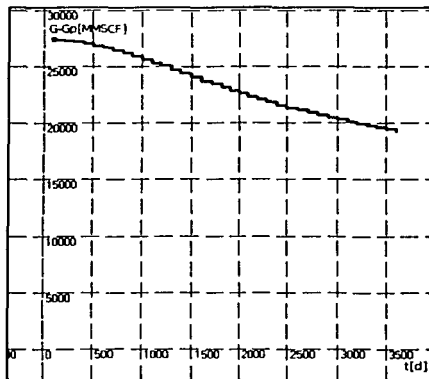


Figura 6.45.- Reserva contra tiempo.

### Comentarios

Por tratarse de un ejemplo ilustrativo, sólo se perforaron 8 pozos para desarrollar un campo de 27,500 millones de pies cúbicos a c.s. . Por esta razón tarda aproximadamente 10 años en caer la presión media de 2,135 [lb/pg<sup>2</sup>] a 1,750 [lb/pg<sup>2</sup>]. En ese tiempo sólo se alcanzan a recuperar 8,500 millones de pies cúbicos a c.s. En la figura 6.43 se observa que la perforarse perfora un nuevo pozo la producción aumenta con un pico súbito y paulatinamente desciende hasta que otro pozo se perfora. Lo más interesante de este ejemplo es ver cómo cambia la distribución de presiones en el yacimiento con el paso del tiempo. En la secuencia de imágenes mostradas en las figuras 6.40 a 6.42 se aprecia que las caídas de presión comienzan en los pozos y se propagan radialmente al resto del yacimiento. Existe interferencia entre pozos y hay áreas del yacimiento "vírgenes", donde la presión cae lentamente debido al drene de los pozos. La heterogeneidad del yacimiento de este modelo se manifiesta en franjas con valores constantes de porosidad o permeabilidad en las direcciones X y Y. No resulta tan sencillo correlacionar la distribución de presiones con la distribución de permeabilidades; sin embargo en otras pruebas que se realizaron se observó que la anisotropía causa un perfil de drene elíptico para los pozos, mientras que zonas de alta permeabilidad donde se perfora un pozo se

depressionan rápidamente, mientras que zonas de baja permeabilidad tienden a depresionarse lentamente.

En este ejemplo no se muestra la distribución de presiones en los nodos de la red ni de gastos en los tubos. Esto se debe a la corta distancia de cada tubo (ver figura 6.31) aunada a diámetros grandes y gastos relativamente pequeños para estas distancias y diámetros, prácticamente no hay cuellos de botella en la red y las presiones después de los estranguladores en cada pozo son sólo unas cuantas libras/pg<sup>2</sup> mayores a la presión de entrega. El gasto que transporta la línea principal de recolección es la suma de los gastos de cada pozo.

### VI.3 EJEMPLO DE OPTIMIZACIÓN

El siguiente ejemplo de optimización pretende ilustrar cómo puede utilizarse la simulación conjunta del yacimiento con instalaciones para resolver problemas de optimización.

En este ejemplo el objetivo es encontrar el valor presente neto (VPN) máximo, a partir de la explotación de un yacimiento con un número fijo de pozos en un estado avanzado de agotamiento natural. La red superficial tiene un solo punto de entrega y un compresor al final de este último. Los costos de compresión varían según la potencia del compresor y la capacidad (potencia máxima) del mismo. Por otro lado las ganancias están en función de los gastos, pero mayores gastos requieren mayores costos de compresión, pues se requiere que el compresor entregue el gas a una presión fija para su transmisión y para aumentar los gastos del yacimiento, es necesario reducir la presión de succión del compresor, lo que supone mayor potencia y por ende mayores costos de compresión. En el balance entre costos de compresión y mayores ganancias debe encontrarse la curva óptima de presión de entrega que maximice el VPN dentro del horizonte de planeación considerado.

Para este ejemplo se retoman los datos del ejemplo anterior (mismo yacimiento, misma red, etc.). Véanse las figuras 6.21 a 6.32 y 6.34 a 6.39. La distribución de presiones iniciales del yacimiento se muestra en la figura 6.46.

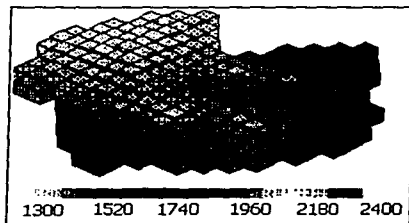


Figura 6.46.- Distribución de presiones iniciales para el ejemplo de optimización.

Para calcular el valor presente neto se toman en cuenta las siguientes consideraciones:

- La producción no estará sujeta a una cantidad diaria de venta por contrato, pero la presión de salida del compresor debe quedar constante en todo momento.
- El precio del gas permanece constante durante todo el período considerado.
- Los costos de compresión por operación del compresor son únicamente los del combustible.
- Los costos de compra e instalación del compresor son proporcionales a la potencia máxima del compresor.
- La potencia máxima del compresor es igual a la potencia máxima que se vaya a utilizar en el periodo de tiempo considerado.
- El volumen de gas a c.s. producido en un intervalo de tiempo  $\Delta t$  se capitaliza inmediatamente al final de dicho período.
- Los costos de funcionamiento del compresor por combustible se descuentan al final de cada intervalo de tiempo  $\Delta t$ .
- El costo de compra e instalación del compresor se descuenta al tiempo en que por primera vez la potencia del compresor sea  $>0$ [Hp].
- El valor de salvamento del compresor es 0[dls].
- No se consideran inflación ni impuestos.
- El valor del dinero en el tiempo se calcula a una TMAR (tasa mínima atractiva de retorno) anual constante y con capitalización continua.

Para el ejemplo, empleamos los valores siguientes:

Costo de compra e instalación del compresor por Hp: 480[dls/Hp].

Consumo específico de combustible del compresor: 240[MPCS/d/Hp].

Tasa mínima atractiva de retorno: 15% anual.

Presión a la salida del compresor: 1.000[lb/pg<sup>2</sup>].

Precio del gas: 3[dls/MPCS].

Período de tiempo considerado para el análisis: 3600[d]

Inicialmente se tenía contemplado en el experimento obtener la curva óptima de presión de succión del compresor contra el tiempo; sin embargo, en la práctica dicha curva debe discretizarse en periodos donde la presión de succión permanezca constante. La elección del número de dichos periodos de tiempo implica determinar el número máximo de corridas que deberán hacerse, pues la presión de cada intervalo de tiempo se comporta como una variable independiente y el valor presente neto es entonces función de varias variables. Si se elige dividir al intervalo de tiempo 0 - 3600[d] en 10 subintervalos, y variar la presión en cada uno de ellos de acuerdo a, digamos, 10 valores fijos que pueda tomar dicha presión del intervalo, entonces se necesitan  $10^{10} = 10,000,000,000$  corridas para poder encontrar la combinación que haga máximo el valor presente neto. Evidentemente hay que restringir tanto el número de intervalos como el número de valores que pueda tomar la presión en cada intervalo para llegar a un número razonable de corridas.

En nuestro caso, para facilitar el análisis y reducir el tiempo de cálculo para encontrar el VPN máximo sólo consideramos dos intervalos de tiempo y 10 valores posibles de presión por cada intervalo para hacer 100 corridas del simulador. En rigor no estamos obteniendo la curva que maximiza el VPN, pero es una aproximación que podría refinarse hasta obtener el VPN máximo. No se intentó establecer o utilizar alguna metodología para hacer más eficiente la optimización, pues escapa a los alcances del presente trabajo.

## Metodología

Se dividió al intervalo de tiempo en dos subintervalos. El primero va de 0[d] a 1.800[d] y el segundo de 1.800[d] hasta 3.600[d]. Se realizaron 2 series de corridas. En la primera se varía la presión del primer intervalo de tiempo desde 1000[lb/pg<sup>2</sup>] hasta 100[lb/pg<sup>2</sup>] (ver la

tabla 6.6). En la segunda se varía la presión del segundo intervalo desde 180[lb/pg<sup>2</sup>] hasta 360[lb/pg<sup>2</sup>]. En las dos series de corridas se varió la presión del segundo intervalo de acuerdo a la fórmula:

$$P_{seg.int.} = \lambda P_{prim.int.} \quad (6.1)$$

Los valores que  $\lambda$  puede tomar son: 1, 0.5, 0.4 y 0.2.

Tabla 6.6.- Valores de presión para la primera serie de corridas

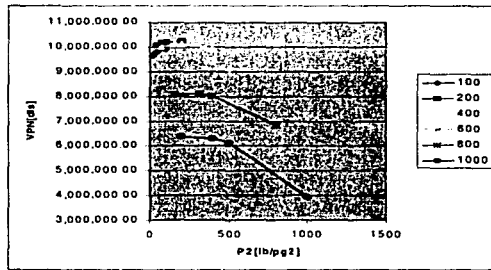
Presión de 0-1,800[d]	Presión de 1,800-3,600[d]
100	20, 40, 50, 100
200	40,80,100,200
400	80,160,200,400
600	120,240,300,600
800	160,320,400,800
1000	200,400,500,1000

Tabla 6.7.- Valores de presión para la segunda serie de corridas

Presión de 0-1,800[d]	Presión de 1,800-3,600[d]
360	72, 144, 180, 360
340	68,136,170,340
300	60,120,150,300
260	52,104,130,260
220	44,88,110,220
180	36,72,90,180

### Resultados de la primera serie de corridas

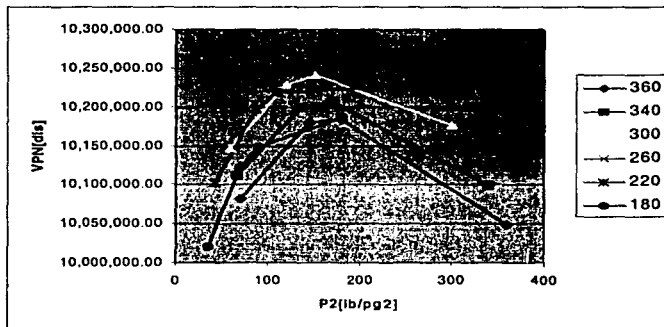
En la figura 6.47 se muestran curvas del VPN en función de la presión del segundo intervalo de tiempo ( $P_2$ ), a una presión del primer intervalo constante ( $P_1$ ).



6.47.- Resultados de la primera serie de corridas

Como puede apreciarse en la figura, la curva que posee el máximo valor presente neto es la de  $P_1=200$  [lb/pg<sup>2</sup>] (con  $P_2=200$  [lb/pg<sup>2</sup>]). En la segunda serie de corridas se investiga más a detalle los valores de  $P_1$  cercanos a 200 [lb/pg<sup>2</sup>], pues no se puede saber con certeza de la primera serie de corridas si la presión  $P_1=200$  [lb/pg<sup>2</sup>] es la óptima "global" o sólo de esta serie de corridas. Como resultado de la segunda serie de corridas se obtiene la gráfica de la figura 6.48.

**Resultados de la segunda serie de corridas**



6.48.- Resultados de la segunda serie de corridas.

La gráfica de la segunda serie de corridas revela que el valor presente neto máximo no se presente en  $P_1=200$  [lb/pg<sup>2</sup>] sino cerca de 260 [lb/pg<sup>2</sup>]. Para este caso la presión  $P_2$  a la cual se obtiene el máximo (siguiendo la curva  $P_1=260$  [lb/pg<sup>2</sup>]) es  $P_2=130$  [lb/pg<sup>2</sup>]. Es posible hacer más corridas con valores más y más cercanos al óptimo hasta encontrarlo; sin embargo la diferencia entre ellos es sutil y el número de corridas se incrementa. Con el ejemplo de estas dos series de corridas se ha indicado el camino que puede seguirse hasta encontrar el VPN máximo.

Podemos concluir de este ejemplo que no se cuenta con una metodología adecuada para variar los parámetros; el número de corridas puede incrementarse de manera exagerada. En este ejemplo se supuso al inicio que se encontraría la curva que maximizara al VPN, y en su lugar sólo hemos hecho una aproximación burda con valores constantes de presión en dos intervalos. ¡Para ello tuvimos que realizar 48 corridas!



## CAPÍTULO VII

# CONCLUSIONES Y RECOMENDACIONES

### VII.1 CONCLUSIONES

1. Se construyó un simulador de un yacimiento de gas seco con pozos e instalaciones de recolección en superficie, llevando a cabo el acoplamiento con una formulación explícita usando las condiciones de frontera de Dirichlet. Se prestó atención a la interfaz gráfica con el usuario y se implementó un algoritmo sencillo para graficar en tercera dimensión tanto al yacimiento como a tuberías. Utilizando este simulador, se investigó el comportamiento de convergencia del método empleado y se ejemplificó cómo puede ser empleado el simulador para resolver problemas de optimización.
2. El lenguaje Visual Basic facilita la tarea de crear las ventanas y objetos gráficos con una programación de alto nivel. Sin embargo, a nivel subrutinas esta misma característica se convierte en desventaja, pues la gran cantidad de cálculos se realiza lentamente en comparación a como se llevaría a cabo en Visual C o C++.
3. La programación no pudo optimizarse por el tiempo y alcances de la tesis. Hay procesos como el del yacimiento que tienen un alto grado de ineficiencia debido a que se repiten cálculos innecesariamente, se utiliza tiempo en copiar arreglos completos dentro de los ciclos etc. Por otro lado, se tuvo poco cuidado en las operaciones de punto flotante: Hay operaciones que deben adecuarse para llevarse

- a cabo con un número mínimo de sumas y multiplicaciones o para reducir errores por truncamiento.
4. Se analizó el comportamiento numérico de un gran número de correlaciones y métodos; sin embargo, no se abundó lo suficiente en ellos debido a la extensión de la tesis y la limitante de tiempo.
  5. Una suposición fuerte que se hizo fue considerar valores constantes y arbitrarios de temperatura en tuberías horizontales, verticales e inclinadas, así como en estranguladores.
  6. El uso de estructuras hizo posible un manejo más ordenado de la información; sin embargo, la relación entre elementos de una estructura y otra se volvió compleja (se tuvo que utilizar un sistema de índices a elementos de arreglos), además que se debe redimensionar constantemente y revisar el tamaño de los arreglos. La programación orientada a objetos es más adecuada en este caso, pues para relacionar un elemento de una colección con otra (ya no arreglos de estructuras contenidos en estructuras), basta declarar propiedades de los objetos como otros objetos y así, se asignan apuntadores a los objetos mismos, en vez de índices a posiciones en arreglos. Por otro lado, el encapsulamiento también es recomendable, pues muchas veces se tienen que "pasar" las estructuras como argumentos a funciones, con tareas relacionadas sólo con esas estructuras. No hay duda de que para tratar problemas de redes la programación orientada a objetos es mejor opción que la programación estructurada.
  7. La experimentación numérica mostró cualitativamente los efectos que diferentes componentes del sistema tienen en la convergencia del método, usando una formulación explícita con condiciones de frontera de Dirichlet entre el yacimiento y la red. La resistencia que tiene la red al flujo (representada principalmente por los estranguladores) y la productividad de los pozos, dan tendencias oscilatorias al método, haciendo al número de iteraciones necesario para alcanzar la convergencia muy dependiente de los valores iniciales supuestos. La misma dependencia de la velocidad de convergencia con los valores iniciales supuestos, tanto de la red como de gastos de los pozos, causa que cuando haya cambio de red o de condición de frontera en los puntos de entrega el método tarde en converger.

## VII.2 RECOMENDACIONES

1. Un modelo más riguroso deberá considerar flujo multifásico en el yacimiento y en las tuberías. Se deben implementar más aditamentos en la red y calcular la distribución de temperaturas en la misma. Es necesario que se realice una investigación sobre los efectos que tienen los cambios de temperatura en las tuberías en la simulación.
2. El comportamiento de convergencia puede mejorarse encontrando un factor de peso para calcular los valores supuestos de gasto de cada iteración, pero queda por diseñarse un algoritmo que corrija la situación de contraflujo (gastos negativos), sin repercutir en la correcta y rápida determinación este factor.
3. En el simulador del presente trabajo no se implementó la opción de reanudar corridas pasadas. Esta deficiencia se hizo crítica al llevar a cabo el ejemplo de optimización. En muchas ocasiones la diferencia entre una corrida y otra consistió sólo en el valor de un parámetro en un intervalo de tiempo determinado, quedando el resto de la corrida idéntica antes de dicho intervalo. Entonces es necesario que ya sea que se haga un análisis de sensibilidad o se resuelva un problema de optimización, tener siempre la opción de reiniciar corridas pasadas a partir de ciertos tiempos.
4. Al realizar las corridas se observó que el tener un programa de operaciones rígido (perforación de nuevos pozos, adición de tuberías y nodos a la red, etc.) es una debilidad al tratar problemas de optimización. Es deseable que el programa sea capaz de seleccionar valores (como diámetro de los estranguladores o presiones de entrega, por ejemplo), de manera automática. Esto es posible cambiando la manera como se implemente el programa de operaciones. En el presente trabajo el programa de operaciones es un conjunto de redes que entran en funcionamiento una después de otra, pero podría implementarse también como una sola red a la cual se le hacen modificaciones. Estas modificaciones, que implicarían la creación de nuevos elementos a la red o la variación de parámetros de los ya existentes, pueden llevarse a efecto por medio de comandos o palabras claves. Los comandos pueden hacerse entonces condicionales y ejecutarse automáticamente cuando se cumpla su condición de ejecución (por ejemplo, "perforar" un pozo nuevo cuando la producción caiga debajo de cierto valor límite). Resulta ahora evidente que en el presente trabajo

se utilizó sólo una de esas posibles condiciones, en la cual se requiere que el tiempo adquiera algún valor determinado para llevar a cabo la modificación.

5. Se tiene también el problema de la gran cantidad de información, tanto en memoria RAM, como en archivos, que requiere un programa formado por varias redes. En muchas ocasiones la información es repetitiva y el usuario debe tomarse la molestia en escribirla una y otra vez en redes diferentes al editar el programa. Debe además revisar que ésta esté completa en todas ellas antes de intentar ejecutar el simulador. Por otra parte, desde el punto de vista del esfuerzo de programación, se presenta el caso contrario, pues es más fácil hacer una interfaz gráfica con una sucesión de redes rígidas que se editan, que una sola red a la cual se le hacen modificaciones con comandos. Al hacerse con comandos, para editar el estado de la red a diferentes tiempos se tendrían que generar y guardar los primeros, según el usuario haga la edición. El problema no es sencillo, pero en definitiva en la ejecución del simulador es mejor el enfoque de comandos que el de redes rígidas, aunque este último implique más esfuerzo de programación.
  
6. Cuando se utilice un simulador para resolver problemas de optimización o análisis de sensibilidad, se deben automatizar las corridas variando uno o varios parámetros en cada una de acuerdo a patrones preestablecidos y escribir código para que se guarden los resultados en archivos para su análisis posterior. Es aconsejable investigar métodos adaptativos para encontrar más rápido los valores de los parámetros que optimicen la función objetivo. Por la experiencia que tuvo el autor del presente trabajo, el tiempo para realizar las corridas puede ser excesivo si no se varían los parámetros con algún método que restrinja el número de éstas, desechando las que de acuerdo con algún criterio y con base en información de los resultados de las corridas pasadas se prevea que no obtendrán el valor óptimo de la función objetivo. Es recomendable también que durante las corridas se procese la información lo más posible, para reducir el tiempo de análisis posterior de la información. Si es posible guardar en archivos tablas que resuman valores de variables selectas (o valores calculados con base en ellas) de un grupo de corridas, tanto mejor.



# NOMENCLATURA

$A$	Matriz de coeficientes de flujo
$a$	coeficiente de flujo
$\alpha$	Factor para linealizar la ecuación del yacimiento en diferencias finitas, para cada celda
$\alpha$	Vector que contiene a los factores $\alpha$ de las celdas
$B$	Factor de volumen
$c_f$	Compresibilidad de la roca
$\bar{C}$	Vector posición del centroide de una figura plana en el espacio
$C_1, C_2$	Constantes
$C7+$	fracción pesada del gas
$C_d$	Coefficiente de descarga
$D$	Factor de flujo turbulento
$D_{64}$	Diámetro del estrangulador en 64avos de pulgada
$d$	Diámetro de la tubería
$d$	Término del vector independiente
$dist$	Distancia del observador al origen
$\bar{d}$	Vector independiente
$\bar{d}$	Vector de distancias del observador al centroide de un polígono
$dV$	Diferencial de volumen
$dx, dy$	Diferencial de X y de Y, respectivamente
$\Delta t$	Intervalo de tiempo
$\Delta x, \Delta y$	Diferencia en X y Y, respectivamente
$\delta$	Densidad relativa del gas (aire=1)
$E$	Eficiencia de flujo
$e_t, e_n, e_b$	Vectores unitarios tangencial, normal y binormal.
$\varepsilon$	Rugosidad de la tubería
$\eta_1, \eta_2, \eta_3, \eta_4, \eta_5, \eta_6, \eta_7$	Variables para resolver la ecuación de Tian y Adewumi
$\bar{F}$	Vector de funciones en el método de Newton para resolver la red
$f$	Factor de fricción de tuberías
$\varphi$	Porosidad
$\varphi$	Angulo que forma el vector $\bar{V}$ con el eje z
$g$	Aceleración de la gravedad
$\bar{g}$	Vector secundario del método LU
$h$	Espesor de la celda
$l$	Vector posición del punto de intersección
$l$	Punto de intersección del segmento de recta que une un punto con el observador con el plano de visualización

## NOMENCLATURA

$I_{m\acute{a}x}$	Número máximo de celdas en i
$i, j, k$	Vectores unitarios que definen el sistema de referencia absoluto
$J$	Matriz Jacobiana
$J'$	Índice de productividad para pozos de gas
$J_{m\acute{a}x}$	Número máximo de celdas en j
$J'$	Vector de índices de productividad
$K$	Tensor de permeabilidades
$k$	Índice adiabático
$k_x$	Permeabilidad en x
$k_y$	Permeabilidad en y
$k_z$	Permeabilidad en z
$L$	Matriz triangular superior
$\lambda$	Distancia del observador al plano de visualización
$M$	Ancho de banda de una matriz pentadiagonal
$M_m$	Masa molar
$m$	Gasto másico
$\mu$	Viscosidad
$N$	Dimensión de una matriz cuadrada
$N_R$	Número de Reynolds
$P.E.$	Conjunto de puntos de entrega en la red
$P_e$	Presión en la línea de flujo
<i>Pozos</i>	Conjunto de pozos
$P_p$	Seudopresión de gas real
$\bar{P}_p$	Vector de pseudopresiones
$P_{ih}$	Presión en la cabeza del pozo
$Q$	Gasto volumétrico a condiciones estándar que se sale de un nodo
$q$	Gasto volumétrico a condiciones atmosféricas
$R_u$	Constante universal de los gases
$R_x$	Error de truncamiento en la dirección X.
$R_y$	Error de truncamiento en la dirección Y.
$r_c$	Radio de drene del pozo
$r_o$	Radio efectivo del pozo
$r_w$	radio del pozo
$r$	Vector posición de un punto en el espacio respectivamente
$\bar{r}_p$	Vector que une al origen del sistema relativo con el punto de intersección
$\rho$	Densidad
$s$	Factor de daño a la formación
$T$	Temperatura

## NOMENCLATURA

$T$	Transmisibilidad del medio poroso
<i>Term</i>	Conjunto de terminaciones
<i>tol</i>	Tolerancia
<i>Tubos _ hor</i>	Conjunto de tubos horizontales de la red
$\theta$	Angulo que forma la proyección horizontal de $\vec{V}$ con el eje x
$U$	Matriz triangular inferior
$V$	Volumen
$V_p$	Volumen poroso
$\vec{V}$	Vector posición del observador
$\vec{v}$	Vector velocidad
$W$	Trabajo requerido en términos de operaciones de punto flotante
$X$	Vector de incógnitas en el método de Newton para resolver la red
$X_i$	Variable i-ésima de la red
$x, y, z$	Distancias en la direcciones X, Y y Z, respectivamente
$\xi$	Propiedad o variable cualquiera
<i>Yac</i>	Conjunto de celdas de un yacimiento
$z$	Factor de compresibilidad
$z$	Vector de factor de compresibilidad
$\frac{\partial}{\partial \xi}$	Derivada parcial con respecto a $\xi$
$\nabla$	Gradiente
$\nabla \cdot$	Divergencia

## Subíndices

<i>atm</i>	A condiciones atmosféricas
<i>c</i>	Centroide
<i>c</i>	Crítico
<i>C</i>	Central
<i>CONV</i>	Convergiendo
<i>E</i>	En dirección Este
<i>f</i>	De la roca
<i>g</i>	Del gas
<i>i</i>	Celda i
<i>j</i>	Celda j
$i \pm \frac{1}{2}$	Promedio entre la celda i y la celda i+1
$j \pm \frac{1}{2}$	Promedio entre la celda j y la celda j+1
<i>It _ Global</i>	De la iteración global
<i>k</i>	Iteración k-ésima
<i>m</i>	Molar
<i>med</i>	Medio
<i>N</i>	En dirección Norte
<i>p</i>	Poroso
<i>Red _ sup</i>	De la red superficial



## NOMENCLATURA

<i>S</i>	En dirección Sur
sup	Supuesto
<i>th</i>	En la cabeza del pozo
<i>u</i>	Universal
<i>W</i>	En dirección Oeste
<i>wf</i>	De fondo fluyente
$\xi - \zeta$	Del nodo $\xi$ al nodo $\zeta$
<i>yac</i>	Del yacimiento

## Superíndices

<i>f</i>	Hacia delante
<i>n</i>	Nivel de tiempo actual
<i>n + 1</i>	Siguiente nivel de tiempo
( <i>k</i> )	Iteración k-ésima

## BIBLIOGRAFÍA

- [1] Al-Hussainy, R., Ramey, H.J. Jr. y Crawford, P.B.: "The Flow of Real Gases Through Porous Media", Trans.,SPE of AIME, 18, Mayo 1966, pp. 624-36 (JPT)
- [2] Aziz, K. y Settari A.: Petroleum Reservoir Simulation. Applied Science Publishers, Essex, Inglaterra, 1979.
- [3] Bánzer S. C.: Correlaciones Numéricas P.V.T. Edición Preliminar, Maracaibo, 1996
- [4] Bohannon, J.M.: "A linear Programming Model for Optimum Development of Multi-Reservoir Pipeline Systems," JPT pi429-i436, Nov. 1970.
- [5] Brown, K.E. y Lee, F.J.: "Nodal Systems Analysis of Oil and Gas Wells" Artículo SPE 14214, Journal of Petroleum Technology, Octubre de 1985, pp. 1751-1763.
- [6] Bui-Tran, V., Pajon, J-L y Joseph, P.: "Three-Dimensional Reservoir Visualization" Artículo SPE 22301. Presentado en la Sixth SPE Petroleum Computer Conference, Dallas, Tejas, Junio 17-20, 1991.
- [7] Burden, R.L. y Faires J. D.: Numerical Analysis, PWS-KENT Publishing Company, Boston 1989.
- [8] Byer, T.J. : Preconditioned Newton Methods for Simulation of Reservoirs with Surface Facilities. Tesis Doctoral. Universidad de Stanford, Mayo 2000.
- [9] Byer, T.J.,Edwads, M.G. y Aziz,K.: "A Preconditioned Newton Method for Reservoirs with Surface Facilities" Artículo SPE 51895,Presentado en el 1999 SPE Reservoir Simulation Symposium, Houston, Tejas, 14-17 Febrero 1999.
- [10] Byer, T.J.,Edwads, M.G. y Aziz,K.: "Preconditioned Newton Methods for Fully Coupled Reservoir and Surface Models" Artículo SPE 49001, presentado en la 1998 SPE

## BIBLIOGRAFÍA

- Annual Technical Conference and Exhibition. Nueva Orleans, Luisiana, 27-30 de Septiembre 1998.
- [11] Cain, G. M., y Shehata, M.T.: " A Computerized System to Optimize Daily Oil and Gas Production in Kuwait," Artículo SPE 9622, presentado en la SPE Middle East Oil Technical Conference, Manama, Bahrain, Mar.9-12, 1981
- [12] Coats, K.H. : "A note on an elementary Iterative Problem", Soc. Pet. Eng. J. Sept.(Artículo SPE 2133),1968, 205-206
- [13] Demetre, G.P. y Farouq Ali, S.M.: "A Review of Analytical well Models Used in Reservoir Simulation" Artículo SPE 28179. Junio de 1994.
- [14] Dempsey, J.R., Patterson, J.K., Coats, K.H. y Brill, J.P.: "An Efficient Model for Evaluating Gas Field Gathering System Design". Artículo SPE 3161, presentado en el SPE Central Plains Regional Meeting, Noviembre 1970.
- [15] Ding, Y. y Renard,G.: "Representation of Wells in Numerical Reservoir Simulation", Artículo SPE 29123, presentado en el 13<sup>th</sup> SPE Symposium on Reservoir Simulation, San Antonio, Tejas. Febrero, 1995.
- [16] Economides, M.J. , Hill, A.D. y Ehlig-Economides C.: Petroleum Production Systems, Prentice Hall, 1994,Nueva Jersey, EE.UU.
- [17] Emmanuel, A.S., y Ranney, J.C.: "Studies of Offshore Reservoir with an Interface Reservoir/Piping Network Simulator" JPT, Marzo de 1981, pp. 399-406.
- [18] Entsminger, G.: Secrets of the Visual Basic for Windows Masters, primera edición. Sams Publishing. Carmel Indiana. 1992
- [19] Foley, J.D., Feiner, S.K. y Hughes, J.F.: Computer Graphics: Principles and Practice. Segunda edición en C. Addison-Wesley Publishing Company. 1990 EE.UU.AA.
- [20] Garaicochea, F. ,Bernal, C. y López, O.: Transporte de Hidrocarburos por Ductos. Colegio de Ingenieros Petroleros de México. 1991.
- [21] Hegguler, G., Barua, S. y Bard,W.: "Integration of a field Surface and Production Network with a Reservoir Simulator" Artículo SPE 38937, SPE Computer Applications, Junio de 1997.
- [22] Huang, A.Y. y Ziauddin, Z.: "Use of Computer Graphics in Large-Scale Reservoir Simulation" Artículo SPE 20343, Presentado en el 5th SPE Petroleum Computer Conference, Denver Colorado, 1990.
- [23] Ikoku, Chi U.: Natural Gas Engineering, Penn Well Books. 1980.
- [24] Kumar, S.: Gas Production Engineering, Gulf Publishing Company. 1987.
- [25] Lee, W.J. y Wattenbarger, R.A.: Gas Reservoir Engineering, SPE Textbooks Series, SPE 1996.

## BIBLIOGRAFÍA

- [26] McCain W.D. Jr.: *The Properties of Petroleum Fluids*, Segunda Edición. Penn Well Books, Tulsa 1989.
- [27] McCain W.D. Jr.: "Reservoir-Fluid Property Correlations – State of the Art" Artículo SPE 18571. *SPE Reservoir Engineering*. Mayo 1991. Pp. 266-272.
- [28] Mendenhall, W.; Scheaffer, R.L. y Dennis, D.W., *Estadística Matemática con Aplicaciones*. Grupo Editorial Iberoamérica. México D.F., 1986.
- [29] Microsoft Visual Basic Language Reference Version 4. Microsoft Corporation. 1995
- [30] Muskat, M: *Physical Principles of Oil Production*, Mc-Graw-Hill Book Co.,Inc., Nueva York 1949.
- [31] Nogaret, C.: *Implementation of a Network-Based Approach in an Object Oriented Reservoir Simulator*, Tesis de Maestría, Universidad de Stanford, Stanford, Mayo de 1996.
- [32] Odeh, A.S.: "Reservoir Simulation ...What is it". Artículo SPE 2790. *JPT*, Noviembre 1969. pp. 1383-1388.
- [33] Peaceman, D.W.: "Interpretation of Well-block pressures in Numerical Reservoir Simulation With Square Grid Blocks and Anisotropic Permeability", Artículo SPE 10528, presentado en el Sixth Symposium of Reservoir Simulation, Nueva Orleans Febrero, 1982.
- [34] Peaceman, D.W.: "Interpretation of Well-block pressures in Numerical Reservoir Simulation" Artículo SPE 6893, *Soc. Pet. Eng. J.* Abril, 1978. pp. 183-194.
- [35] Russell, D.G., Goodrich, J.H., Perry, G.E. y Bruskotter, J.F.: "Methods for Predicting Gas Well Performance" *J. Pet. Tech.*, Enero de 1966. Pp. 99-108; *Trans., AIME*, 237.
- [36] Santamaría Vite, I.: *Optimación de Redes de Gas*, Tesis de Licenciatura. Facultad de Ingeniería, U.N.A.M., México D.F., 2001.
- [37] Schiozer, D. J., y Aziz, K.: "Effect of Chokes on Simultaneous Simulation of Reservoir and Surface Facilities" Artículo SPE 26308, presentado en la 3ª SPE Latin American & Caribbean Petroleum Conference, Buenos Aires, Abr. 27-29, 1994.
- [38] Schiozer, D. J., y Aziz, K.: "Use of Domain Decomposition for Simultaneous Simulation of Reservoir and Surface Facilities" Artículo SPE 27876, presentado en la Western Regional Meeting, Long Beach, California, Marzo 22-25 de 1994.
- [39] Stackel, A. W., y Brown, H.M.: "An Example Approach to Predictive Well Management in Reservoir Simulation," *SPE paper* 7698, 1981.
- [40] Standing, M.B. y Katz, D.L.: "Density of Natural Gases," *Trans. AIME*, 146, 140-149, 1942.

## BIBLIOGRAFÍA

- [41] Starley, G. P., Masino Jr., W. H., Weiss, J.L., y Boiling, J.D.: "Application of Full-Field Simulation to Development Planning and Reservoir Management at the Ku-paruk River Field," Artículo SPE 20045 presentado en el 60th California Regional Meeting, Ventura, CA, Abril. 4-6, 1990.
- [42] Startzman, R.A., Brummett, W.M., Ranney, J.C., Emmanuel, A.S. y Toronyi, R.M.: "Computer Combines Offshore Facilities and Reservoir Forecasts", Petroleum Engineer, Mayo de 1977.
- [43] Stoitsits, R. F.: "Dynamic production system nodal analysis," Artículo SPE 24791, presentado en la 67th Annual Technical Conference and Exhibition of SPE, Washington, DC, EE.UU., oct. 4-7, 1992.
- [44] Stoitsits, R.F. et al.: "Application of Nonlinear Adaptive Model for Rigorous Representation of Production Facilities in Reservoir Simulation," Artículo SPE 24898, presentado en la 67th Annual Technical Conference and Exhibition of SPE, Washington, DC, EE.UU., Oct. 4-7, 1992.
- [45] Stoner, A.S.: "Steady State Analysis of Gas Production, Transmission and Distribution Systems" Artículo SPE 2554, presentado en la 44th Annual Fall Meeting of SPE-AIME, Denver Colorado, Sept. 28-Oct 1, 1969.
- [46] Szilas, A.P.: Production and Transport of Oil and Gas, Elsevier, Amsterdam, 1975.
- [47] Takacs, G. y Turzo, Z.: "Nodal System Analysis Using Object-Oriented Programming Techniques". Artículo SPE 26250, presentado en la SPE Petroleum Computer Conference Luisiana, EE.UU.AA., Julio 1993.
- [48] Tian, S. y Adewumi, M. A.: "Development of Analytical Design Equation for Gas Pipelines". Artículo SPE 24861. SPE Production & Facilities, Mayo 1994. Pp. 100-106.
- [49] van Poolen, H.K., Breitenbach, E.A. y Thurnau, D.H.: "Treatment of Individual Wells and Grids in Reservoir Modeling", Soc. Pet. Eng. J., Diciembre de 1968. Pp. 341-346.
- [50] Wallace, D. J., y van Spronsen, E.: "A Reservoir Simulation Model with Platform Production / Injection Constraints for Development Planning of Volatile Oil Reservoirs," Artículo SPE 12261, presentado en el Reservoir Simulation Symposium, San Francisco, Nov. 15-18, 1983.
- [51] Walpole, R.E. y Myers, R.H.: Probabilidad y Estadística. Tercera Edición en Español. Mc Graw Hill/ Interamericana de México, México D.F., 1991
- [52] Weiss, M.A.: Data Structures and Algorithm Analysis in C, Benjamin Cummings Publishing Company, Redwood California, 1993.

## APÉNDICE A

# VISUALIZACIÓN EN 3D DEL YACIMIENTO Y LA RED SUPERFICIAL

### A.1 INTRODUCCIÓN

La mayoría de los simuladores (sean comerciales, simuladores propios de compañías operadoras o académicos,) tienen una interfaz que permite al usuario visualizar las propiedades y geometría de las celdas del yacimiento en bloques tridimensionales. El simulador presentado en esta tesis, por consiguiente, no deberá ser la excepción.

Muchos de los datos de entrada y de salida de un simulador están asociados a puntos con coordenadas tridimensionales en el espacio, como pueden ser propiedades estáticas como la geometría misma de las celdas, propiedades petrofísicas de cada celda (permeabilidad, porosidad, compresibilidad de la roca), valores que calcula el simulador para cada celda (presiones, saturaciones, etc.) y que cambian con el tiempo. Sin herramientas de visualización gráfica de estos números se debe hacer un esfuerzo para imaginar la estructura tridimensional del yacimiento o entender el flujo de fluidos en el yacimiento por medio de resultados presentados en forma tabular. En nuestro caso individual es de nuestro interés mostrar tanto la geometría y propiedades de las celdas del yacimiento como la distribución de presiones (no calculamos saturaciones) y mostrar de alguna manera los gastos y presiones asociados a cada tubería de producción, estrangulador, tubería horizontal y nodo de las instalaciones de producción.

La bibliografía para el tema es variada, y se encuentra en su mayoría en libros que competen a la graficación por computadora. Un libro que abarca todos los aspectos de graficación por computadora es el de Foley y cols(1990). Para el tema en específico de la visualización en 3D de yacimientos tenemos a Bui-Tran y cols(1991) y Huang y cols(1990).

Antes de entrar de lleno en la graficación del yacimiento e instalaciones, empero, es necesario que presentemos la "transformación" de coordenadas en 3D a coordenadas en 2D, pues es el punto de partida de los algoritmos de graficación de cuerpos tridimensionales. Discutiremos después algo acerca de los algoritmos para resolver el problema de la visibilidad y entonces presentaremos el algoritmo que utiliza la interfaz del simulador para mostrar al yacimiento por una lado y la red por otro.

## A.2 TRANSFORMACIÓN DE COORDENADAS TRIDIMENSIONALES A BIDIMENSIONALES

Cuando percibimos la realidad por medio de nuestros ojos, lo que observamos dependerá del punto determinado donde nos encontremos (posición) y la dirección hacia donde dirijamos nuestra mirada (orientación del observador.) Cuando se desea representar un conjunto de objetos tridimensionales (mundo observable) a través de una pantalla que funciona como una ventana que nos permite ver al mundo observable se procede a definir a un observador virtual, llamado también en la literatura (Foley y cols. 1990) como "cámara sintética" , cuya posición en el espacio está dada por un vector posición  $P$  y su orientación por un sistema coordenado, es decir, un conjunto de vectores ortonormales; tangencial, normal y binormal.

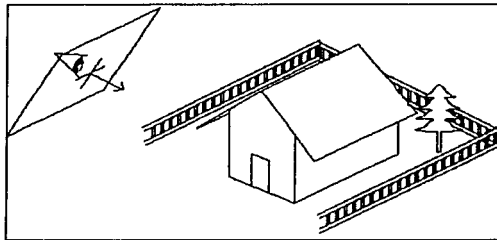


Figura A.1.- Observador de objetos en 3D

Para que este observador pueda "ver" puntos con coordenadas en 3 dimensiones  $(x,y,z)$  en un plano de dos  $(x,y)$  es necesario hacer una proyección de estos puntos a un plano de visualización. Este plano está definido por un vector perpendicular que es paralelo al vector tangencial del sistema coordenado relativo del observador (ver figura A.2.). Hay una distancia  $\lambda$  entre el plano y el observador. La proyección de un punto en el plano es, entonces la intersección de un segmento de recta del punto al observador con el plano de visualización.

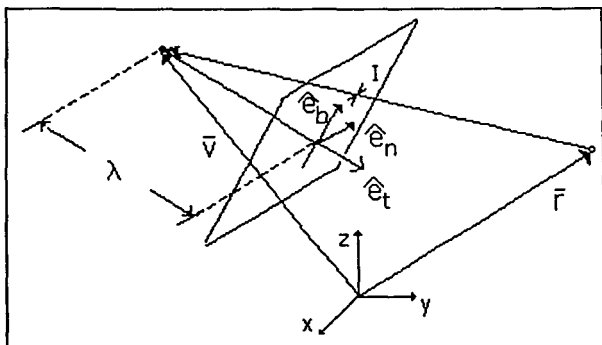


Figura A.2.- Proyección de un punto en el plano de visualización.

En la figura A.2 se muestra al vector  $\vec{V}$  (posición del observador), al vector  $\vec{r}$  (vector posición del punto proyectado), el plano de visualización perpendicular al vector tangencial  $(\hat{e}_t)$ . El punto  $I$  inscrito en el plano es la intersección del segmento de recta  $rV$  y el plano. La dirección  $X$  relativa en el plano coincide con el vector unitario  $\hat{e}_n$  y la de  $Y$  con  $\hat{e}_b$ . Las coordenadas  $X$  y  $Y$  del punto  $I$  son las componentes escalares en las direcciones  $\hat{e}_n$  y de  $\hat{e}_b$  del vector  $\vec{I} - \vec{r}_p$  ( $\vec{r}_p$  es un vector con magnitud  $\lambda$  y dirección  $\hat{e}_t$ , tal que  $\vec{V} + \vec{r}_p$  es el origen en las coordenadas  $X,Y$  en el plano de visualización)



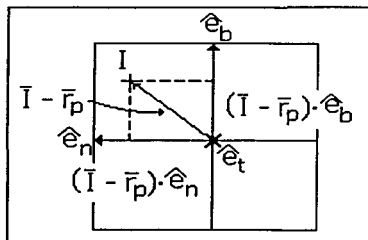


Figura A.3.- Coordenadas X y Y del punto de intersección, según las coordenadas relativas del plano de visualización

De acuerdo a esta nomenclatura, puede demostrarse que el vector  $I$  está dado por:

$$I = \frac{[(V - r) \cdot e_t + \lambda]}{(V - r) \cdot e_t} (V - r) \tag{A.1}$$

y las coordenadas X y Y del punto  $I$  en el sistema coordenado relativo de la figura A.3 son:

$$x = \frac{[(V - r) \cdot e_t + \lambda]}{(V - r) \cdot e_t} (V - r) \cdot e_n \tag{A.2}$$

$$y = \frac{[(V - r) \cdot e_t + \lambda]}{(V - r) \cdot e_t} (V - r) \cdot e_b \tag{A.3}$$

Con las ecuaciones (A.1), (A.2) y (A.3) se pueden obtener las coordenada en 2D de un punto en 3D. Si deseamos obtener la proyección de un segmento de recta no es necesario proyectar todos sus puntos, sino sólo proyectar sus extremos, y luego unirlos en el plano de visualización. Todo punto de este segmento de recta será la proyección de un punto del segmento original. De manera análoga, si deseamos obtener la proyección de un polígono (y de hecho así lo haremos) se deben proyectar sus vértices y luego unirlos con segmentos de rectas.

### Caso especial del observador mirando hacia atrás de sí y proyecciones ortogonales

Se pueden simplificar bastante las ecuaciones (A.1) a (A.3) en el caso especial en que el observador siempre esté mirando "hacia atrás", es decir:

$$\bar{z}_i = -\frac{V}{|V|}. \quad (\text{A.4})$$

Este es un caso conveniente, pues necesitaremos observar una serie de objetos tridimensionales para inspeccionarlos desde diferentes ángulos y a diferentes distancias. Una alternativa sería hacerlos rotar, con el observador fijo, pero se tienen que realizar transformaciones lineales de rotación que pueden ser costosas en términos de tiempo. Otra opción al parecer más conveniente, es hacer rotar al observador en torno al origen del sistema coordenado y orientarlo siempre hacia el centro. De esta manera observará a los objetos desde diferentes ángulos si éstos se encuentran más cercanos al centro que él mismo.

Las dimensiones de los objetos y los efectos de perspectiva serán función la distancia del observador al origen (absoluto) y por la distancia que lo separa del plano de visualización.

Es ventajoso para este caso definir al vector posición  $V$  en coordenadas esféricas con los ángulos  $\varphi$  (ángulo que forma el vector con el eje  $z$ ) y  $\theta$  (ángulo que forma la componente vectorial horizontal del vector con el eje  $x$ .) y la distancia del observador al origen, . Así es posible cambiar la posición del observador manteniendo su distancia al origen constante (dando el efecto de rotar los objetos) o bien aumentar o disminuir esta distancia sin cambiar los ángulos de perspectiva (dando el efecto de hacer más grandes o más pequeñas las figuras.)

El vector  $V$  es entonces:

$$V = dist(\text{sen } \varphi \cos \theta \bar{i} + \text{sen } \varphi \text{sen } \theta \bar{j} + \cos \varphi \bar{k}), \quad (\text{A.5})$$

donde  $dist$  es la distancia del observador al origen.

Con operaciones algebraicas escalares y vectoriales se puede simplificar a (A.3) y (A.2) hasta llegar a:

$$x = \frac{-\lambda}{dist - \bar{r} \cdot \bar{e}_a} \bar{r} \cdot \bar{e}_a \quad (A.6)$$

$$y = \frac{-\lambda}{dist - \bar{r} \cdot \bar{e}_b} \bar{r} \cdot \bar{e}_b \quad (A.7)$$

Entre más alejado se encuentre el observador del punto, el efecto de perspectiva se va perdiendo más y más, pues los segmentos de recta que unen al observador con los puntos van tendiendo a ser normales al plano de visualización, es decir que la proyección se acerca a una proyección ortogonal. Analizando las ecuaciones (A.6) y (A.7) se observa que en el término

$$\frac{-\lambda}{dist - \bar{r} \cdot \bar{e}_i}$$

la cantidad  $\bar{r} \cdot \bar{e}_i$ , se va haciendo insignificante con respecto a  $dist$  cuando éste último es muy grande. Si suprimimos de las ecuaciones (A.6) y (A.7) el término  $\bar{r} \cdot \bar{e}_i$  (aun cuando la suposición  $dist \gg \bar{r} \cdot \bar{e}_i$  no sea cierta) lo que obtendremos es una proyección ortogonal perfecta. Finalmente (A.6) y (A.7) quedan escritas:

$$x = \frac{-\lambda}{dist} \bar{r} \cdot \bar{e}_a \quad (A.8)$$

$$y = \frac{-\lambda}{dist} \bar{r} \cdot \bar{e}_b \quad (A.9)$$

Las ecuaciones (A.8) y (A.9) muestran que para una proyección ortogonal es indiferente hacer la distancia del observador al origen grande o la del observador al plano de visualización pequeña (o viceversa.)

### **A.3 PROBLEMA DE LA VISIBILIDAD**

Para mostrar objetos opacos en 3D (que no sean transparentes) no es suficiente con proyectar sus vértices y luego unir estos últimos con líneas. Ya sea que los objetos tridimensionales estén representados por un conjunto de figuras planas o por superficies alabeadas, es necesario colorear las áreas las superficies de objetos para hacerlos aparecer como sólidos. Para hacer esto correctamente, es necesario determinar qué caras o superficies ocultan a otras. A este problema se le conoce como el problema de la visibilidad.

El problema puede enunciarse de la siguiente manera:

Dados un conjunto de puntos en el espacio (que pueden pertenecer a figuras planas, superficies o sólidos,) determinar qué puntos ocultan a cuáles en una proyección plana y pintar siempre los puntos visibles y ocultar los no visibles.

En la literatura (ver Foley y cols(1990) ) existe una vasta gama de algoritmos para realizar esta tarea y su elección depende del problema a tratar. Como regla general los puntos que se encuentren más cercanos al observador obscurecerán a los que estén más alejados si se encuentran en la misma línea del observador al punto. Ahora bien, el problema que resolveremos no considera puntos individuales como tales, sino paralelepípedos que representan celdas del yacimiento, o bien cilindros que representan tuberías, etc. , es decir que obtendremos la visibilidad entre objetos tridimensionales de una clase tal que estén representados por un conjunto de figuras planas (lados de cubos, lados o tapas de cilindros, etc.) Entonces el algoritmo adecuado será el que trate de manera eficiente este tipo de objetos y éste es el de ordenamiento por distancias.

#### **Algoritmo de ordenamiento por distancias**

Este es un algoritmo para pintar de una manera correcta polígonos (más que punto por mundo) en el espacio. El algoritmo es:

Algoritmo A.1

- 1.-Ordenar todos los polígonos en orden descendente de distancias al observador

2.-Resolver las ambigüedades causadas por el traslape entre polígonos, dividiendo a los polígonos de ser necesario.

3.-Pintar a los polígonos en el orden antes mencionado.

Con este algoritmo el polígono más cercano se superpone a todo o parte de lo que haya sido pintado antes de él (polígonos más lejanos, que fueron pintados antes). El resultado final ver las caras visibles y ocultar todo o parte de las no visibles.

El punto 2 del algoritmo es necesario si existen traslapes entre polígonos. Para el caso del yacimiento los rectángulos que forman las caras de las celdas coinciden en un área completa, más que intersectarse. Las pruebas obtenidas con este algoritmo muestran que se obtienen resultados correctos omitiendo el punto 2.

En el punto 1 del algoritmo se obtienen las "distancias de los polígonos al observador". Hablando propiamente, sólo es posible obtener distancias entre dos puntos, no entre una figura plana y un punto (una figura tiene un número infinito de puntos.) Lo que se entiende por distancia del polígono es la distancia de su centroide al punto del observador.

Nótese que este algoritmo no utiliza técnica alguna de agrupar elementos de objetos, y todos los polígonos que se vayan a mostrar, aunque se pertenezcan a diferentes objetos (celdas tubos, etc ..) deben ser ordenados y pintarse en el orden que dicta el algoritmo para que se asegure la correcta visibilidad.

## **A.4 IMPLEMENTACIÓN DE LA GRAFICACIÓN EN 3D PARA EL YACIMIENTO**

La información geométrica del yacimiento está completa prácticamente con las propiedades  $x, y, z, D_x, D_y$  y  $h$  (altura) de los tipos Celda (ver sección de la organización de la información.) Cada celda tiene 6 caras o lados a saber:

- Lados  $xy$  y  $x'y'$
- Lados  $xz$  y  $x'z'$

- Lados yz y y'z'

Donde xy, xz, yz representan al plano al cual es paralela esa cara y las comillas denotan caras opuestas.

Los centroides de esas caras son los centros geométricos de los rectángulos que forman, y pueden ser obtenidos de la información de las celdas con el siguiente algoritmo en forma de pseudocódigo:

Suponiendo que Centroides es un arreglo de vectores (con coordenadas x,y,z según la definición aquí abajo) y (xc,yc,zc) son las coordenadas del centroide (centro geométrico) de la celda:

Tipo Vector 'Definición del tipo vector

X como doble

Y como doble

Z como doble

Fin del tipo

Para la celda i = 1 hasta el número total de celdas en el yacimiento

xc = .celda(i).x

yc = .celda(i).y

Zc = .celda(i).h / 2

K = K + 1 'cara Y'Z'

Centroides(K).x = xc - .cel(i).Dx / 2

Centroides(K).y = yc

Centroides(K).z = Zc

K = K + 1 'cara YZ

Centroides(K).x = xc + .cel(i).Dx / 2

Centroides(K).y = yc

Centroides(K).z = Zc

K = K + 1 'cara X'Z'

Centroides(K).x = xc

Centroides(K).y = yc - .cel(i).Dy / 2

Centroides(K).z = Zc

K = K + 1 'cara XZ

Centroides(K).x = xc

```

Centroides(K).y = yc + .cel(i).Dy / 2
Centroides(K).z = Zc
K = K + 1 'cara X'Y'
Centroides(K).x = xc
Centroides(K).y = yc
Centroides(K).z = Zc - .cel(i).h / 2
K = K + 1 'cara XY
Centroides(K).x = xc
Centroides(K).y = yc
Centroides(K).z = Zc + .cel(i).h / 2
Siguiete i

```

Como se vé, el arreglo de centroides es lineal y entran los centroides de todas las caras del yacimiento.

De una manera parecida se extrae de la información de las celdas del yacimiento las coordenadas (en vectores con componentes x,y,z) de los cuatro vértices de las caras de todas las celdas del yacimiento. La estructura que contendrá los vértices de cada cara, así como el número de celda se llamará Cuadrilátero:

#### Tipo Cuadrilátero

Caras (1 a 4) como Vector

I\_celda como entero 'Índice de la celda a la cual pertenece el cuadrilátero.

Fin del tipo

Se hace un arreglo de tipo Cuadrilátero que corra paralelo al de centroides, tal que el centroide en la posición i-ésima es el centroide del cuadrilátero i-ésimo. El arreglo de centroides es la base para obtener el arreglo de distancias de los centroides al vector posición del observador. Una vez obtenido este arreglo de distancias se ordena de mayor a menor, y luego se pintan los cuadriláteros en ese orden. Nos es imposible, no obstante, conocer este orden sin llevar un índice que identifique las distancias rearmadas con su respectivo cuadrilátero. Este arreglo de índices debe ordenarse paralelo al de distancias, y de hecho, es lo único que nos interesa obtener del ordenamiento; un vector de índices a cuadriláteros reacomodado. Si los cuadriláteros se ordenan en el orden xy, x'y', xz, x'z', yz, y'z' de la celda 1 a la n, entonces el vector de índices (que llamaremos *In\_cuad*) queda definido como:

$$In\_cuad = [a_i], a_i = i, \quad (A.10)$$

es decir, sus elementos son 1, 2, 3, ... hasta el número de cuadriláteros (que es número de celdas X 6).

Cuando haya sido ordenado el vector  $In\_cuad$ , nos dará los índices de los cuadriláteros a dibujar en orden de mayor a menor distancia, que es lo que requiere el algoritmo A.1.

Las distancias entre los centroides y el observador son los módulos de la diferencia de sus vectores posición respectivos. Es decir si el centroide C es

$$\vec{C}_i = x_{c_i}\hat{i} + y_{c_i}\hat{j} + z_{c_i}\hat{k}. \quad (A.11)$$

La distancia del cuadrilátero i-ésimo al observado es:

$$d_i = |\vec{C}_i - \vec{V}|. \quad (A.12)$$

Y el vector de estas distancias es:

$$\vec{d} = [d_i]. \quad (A.13)$$

La ecuación (A.12) requiere 3 restas, elevar 3 números al cuadrado, 3 sumas y una raíz cuadrada por cada cuadrilátero. Podemos reducir el número de operaciones (y por consiguiente el tiempo que toma realizarlas) utilizando una definición de distancia alternativa, que es válida si la distancia  $dist$  del observador al origen es grande. La componente escalar del vector C en la dirección  $\hat{e}_i$  (esto es, la distancia del punto C al plano de visualización.) es:

$$d_i = \vec{C}_i \cdot \hat{e}_i. \quad (A.14)$$

que requiere sólo 3 multiplicaciones y 3 sumas.



## Algoritmos de ordenamiento

Nada se ha dicho acerca de los algoritmos para ordenar las distancias. Una revisión excelente de algoritmos de ordenamientos puede encontrarse en el libro de Weiss(1993). Los algoritmos son "Bubblesort", "Shellsort", "Heapsort", "Mergesort" y "Quicksort". El algoritmo más rápido es "Quicksort", que es recursivo. Dicho algoritmo no se presenta aquí, pero puede verse en Weiss(1993) para una presentación y análisis completo.

Puesto que este ordenamiento debe hacerse cada vez que se quiera mostrar en pantalla al yacimiento (y en general también a la red,) no está por demás preocuparse un poco por el tiempo que le lleva al algoritmo ordenar determinado número de cuadriláteros. En Weiss(1993) se deduce que el tiempo para ordenar un vector de  $n$  elementos es del orden  $O(n^2)$  para el peor caso y  $O(n \cdot \ln(n))$  para el mejor. Algunas pruebas realizadas (con datos aleatorios) por el autor de esta tesis parecen mostrar la siguiente relación entre tiempo de ordenamiento y tamaño del vector a ordenar:

$$t_n = C_1 \cdot n. \quad (\text{A.15})$$

es decir lineal con el número de caras a ordenar ( $C_1$  para la computadora en particular fue de  $2 \times 10^{-5} [s]$ ). Un algoritmo modificado de Quicksort para que ordene paralelamente a los vectores  $\vec{d}$  y  $\ln\_cuad$  requiere un tiempo:

$$t_n = C_2 \cdot n \cdot \ln(n). \quad (\text{A.16})$$

Las constantes  $C_1$  ni  $C_2$  dependen del algoritmo, sino de factores de la máquina específica en que se corra ( para la máquina en que se realizaron las pruebas,  $7 \times 10^{-6} [s]$ ). De acuerdo a (A.16) para un yacimiento de 1000 celdas le tomará al algoritmo "Quicksort" ordenar  $\vec{d}$  e  $\ln\_cuad$  :

$$t_n = C \cdot 6000 \cdot \ln(6000) = 52197C. \quad (\text{A.17})$$

Para  $C = 7 \times 10^{-6} [s]$  se tardará 0.36[s], lo cual es un tiempo muy bueno. Incluso si la computadora fuera muy lenta, cien o doscientas celdas podrían ser ordenadas con una rapidez muy grande.

### Visualización de propiedades físicas y escala de colores

La visualización tridimensional del yacimiento tiene la función, además de mostrar la geometría de las celdas, representar valores de propiedades físicas (campos escalares) que son función de la posición e incluso del tiempo (para el caso de las presiones en el yacimiento, por ejemplo.) Estas propiedades pueden ser representadas por un color asignado a cada valor de la propiedad, es decir que se utilizará una escala de colores. Coloreando las celdas con este color, en función del valor que la propiedad tenga para esa celda, es posible tener una visión gráfica del campo escalar en el yacimiento.

Una manera de asignar una escala de colores a una propiedad escalar  $\xi$  (porosidad, permeabilidad, compresibilidad, presión, etc.) es hacer mezclas de colores en proporciones que sean una función de  $\xi$ . En Visual Basic existe la función RGB (de las iniciales de Red, Green y Blue) que permite hacer mezclas de colores con proporciones de rojo, verde y azul. Entonces podemos hacer la proporción de rojo función de  $\xi$ , la de verde función de  $\xi$  y la de azul función de  $\xi$ .

Otra buena idea es establecer una escala "normalizada" de colores. Si tenemos acotados los valores máximo y mínimo esperados del valor de  $\xi$  ( $\xi_{max}$  y  $\xi_{min}$ , respectivamente,) entonces las proporciones pueden quedar en función de  $\xi_{norm}$  (propiedad normalizada), que se definiremos como:

$$\xi_{norm} = \frac{\xi - \xi_{min}}{\xi_{max} - \xi_{min}} \quad (A.18)$$

La idea principal de  $\xi_{norm}$  es que sea un valor entre 0 y 1, así podremos usar la misma función que asigna colores para propiedades que tienen diferentes rangos de valores (rangos establecidos por el usuario para mostrar los colores, no que tengan que ver con valores máximos o mínimos que use el simulador en sus corridas.)

Las proporciones de verde, rojo y azul en función de  $\xi_{norm}$  que utilizamos en la escala normalizada de nuestro simulador son:

$$Verde = \begin{cases} \frac{255\xi_{norm}}{3} \text{ si } 0 \leq \xi_{norm} \leq \frac{1}{3} \\ 255 \text{ si } \frac{1}{3} < \xi_{norm} \leq \frac{2}{3} \\ 255(3\xi_{norm} - 2) \text{ si } \frac{2}{3} < \xi_{norm} \leq 1 \end{cases} \quad (A.19)$$

$$Rojo = \begin{cases} 255\xi_{norm} \text{ si } 0 \leq \xi_{norm} \leq \frac{1}{3} \\ 255(2 - 3\xi_{norm}) \text{ si } \frac{1}{3} < \xi_{norm} \leq \frac{2}{3} \\ 0 \text{ si } \frac{2}{3} < \xi_{norm} \leq 1 \end{cases} \quad (A.20)$$

$$Azul = \begin{cases} 0 \text{ si } 0 \leq \xi_{norm} \leq \frac{2}{3} \\ 255(3\xi_{norm} - 2) \text{ si } \frac{2}{3} < \xi_{norm} \leq 1 \end{cases} \quad (A.21)$$

La siguiente gráfica muestra graficadas a (A.19), (A.20) y (A.21), así como la escala de colores resultante.

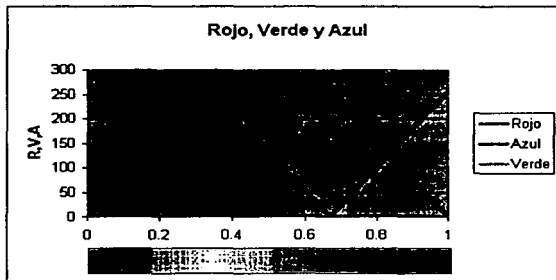


Figura A.4.- Escala de Colores

Integrando todos los conceptos vistos en esta sección, las imágenes que produce nuestro simulador de un yacimiento son como las de la figura A.5.

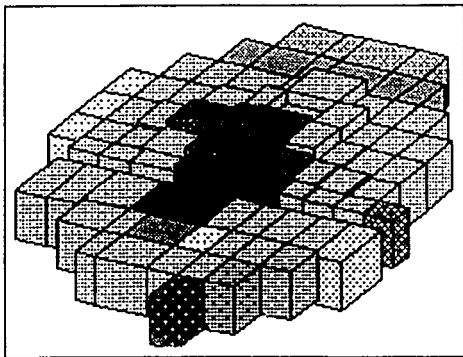


Figura A.5.- Ejemplo de visualización de en yacimiento en 2D con el simulador SIMPREG 1.1

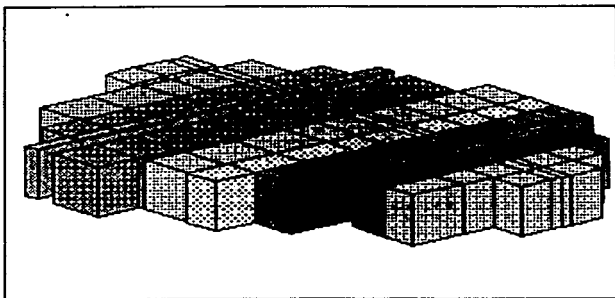


Figura A.6.- Ejemplo de visualización de en yacimiento en 2D con el simulador SIMPREG 1.1

### **código para la graficación en 3D**

En esta sección presentamos y discutimos las subrutinas que utiliza el simulador para graficar cuadriláteros en tercera dimensión. Primero es necesario que presentemos las definiciones de algunas estructuras que utilizan las subrutinas. La primera de ellas ya se había presentado, es la estructura Cuadrilátero:

Tipo Cuadrilátero

Caras (1 a 4) como Vector

I\_celda como entero 'Índice de la celda a la cual pertenece el cuadrilátero.

Fin del tipo

Como se dijo anteriormente, en esta estructura se almacenan los vectores posición de cada una de los vértices un cuadrilátero en el espacio. I\_celda es el índice a la celda de la malla a la cual pertenece el cuadrilátero.

La segunda estructura necesaria para las subrutinas es cuadriláteros, un arreglo de estructuras tipo cuadrilátero.

```
Type cuadriláteros
  arr() As cuadrilátero
End Type
```

Finalmente definimos una estructura que contiene un arreglo bidimensional de tipos Single (precisión sencilla). En esta estructura se almacena el vector "transformado" de 3D a 2D.

```
Type tipo_vectorS
  arr(1 To 2) As Single
End Type
```

### Subrutina proy2D

Definición:

Sub proy2D (Rp() As Double, dist As Double, ele As Double, cost As Double, sent As Double, cosf As Double, senf As Double, V2D() As Single)

```
  Dim RpRv As Double
  Dim RpN As Double
  Dim RpB As Double
  Dim coc As Double
```

```
  RpN = Rp(1) * sent + Rp(2) * cost
  RpB = Rp(1) * cost * cosf - Rp(2) * sent * cosf + Rp(3) * senf
```

```
  coc = ele / dist
  V2D(1) = coc * RpN
  V2D(2) = coc * RpB
```

End Sub

Comentarios:

La subrutina Proj2D "convierte" de coordenadas en 3D a coordenadas en 2D realizando una proyección al plano de visualización de acuerdo a (A.8) y (A.9). Los argumentos de la función son Rp (vector posición  $\vec{r}$  del punto en 3D), la distancia Dist. de acuerdo a (A.5), la distancia  $\lambda$  del observador al plano, denotada por  $e_{1e}$  y los cosenos y senos de los ángulos  $\theta$  y  $\varphi$  ( $\cos\theta$ ,  $\sin\theta$  para  $\theta$  y  $\cos\varphi$ ,  $\sin\varphi$  para  $\varphi$ ). El resultado es almacenado en v2D que es un vector bidimensional de tipo single.

En las operaciones,  $R_{pB}$  y  $R_{pN}$  son las componentes escalares de  $\vec{r}$  en las direcciones  $e_n$  y  $e_b$ , respectivamente, obtenidas con el producto escalar (producto punto) de  $\vec{r}$  con ellos.

Coc es el cociente  $\frac{\lambda}{Dist}$ .

Subrutina colorRGB

Definición:

Function colorRGB(X As Double, min As Double, max As Double) As Long

T = (X - min) / (max - min)

If T <= 1 / 3 Then

Ro = 255

Ve = 3 \* T \* 255

ElseIf T > 1 / 3 And T <= 2 / 3 Then

Ve = 255

Ro = (1 - 3 \* (T - 1 / 3)) \* 255

ElseIf T > 2 / 3 And T <= 1 Then

Ve = (1 - 3 \* (T - 2 / 3)) \* 255

A = (3 \* (T - 2 / 3)) \* 255

End If

colorRGB1 = RGB(Ro, Ve, A)

End Function

Comentarios:

Devuelve el número de color RGB (entero largo) de acuerdo con (A.19), (A.20) y (A.21) para un valor X de una propiedad  $\xi$  cualquiera, acotada entre  $\xi_{\text{mín}}$  y  $\xi_{\text{máx}}$  (argumentos max y min, respectivamente) . La variable T es  $\xi_{\text{norm}}$ , de acuerdo con (A.18).

Subrutina Obtén\_arreglo\_distanciasDefinición:

Sub Obtén\_arreglo\_distancias (Centroides As arreglo\_vector, fiv As Double, tetav As Double, dist As Double, arr\_dist() As arreglo\_double)

Dim n As Integer

Dim Ex As Double

Dim Ey As Double

Dim Ez As Double

Dim total As Double

n = UBound(Centroides.Vect)

ReDim arr\_dist(1 To n)

Ex = -Sin(-fiv \* Pi / 180) \* Cos(-tetav \* Pi / 180) \* dist

Ey = -Sin(-fiv \* Pi / 180) \* Sin(-tetav \* Pi / 180) \* dist

Ez = -Cos(-fiv \* Pi / 180) \* dist

For i = 1 To n

With Centroides.Vect(i)

arr\_dist(i) = ((.arr(1) + Ex) ^ 2 + (.arr(2) + Ey) ^ 2 +  
(.arr(3) + Ez) ^ 2) ^ 0.5

End With

Next i

End Sub

Comentarios:

La subrutina Obtén\_arreglo\_distancias devuelve en un arreglo arr\_dist las distancias del observador a cada uno de los centroides de los cuadriláteros, ordenados según el orden que tiene el arreglo Centroides. La operaciones que realiza son de acuerdo a las ecuaciones (A.11), (A.12) y (A.13), que se explicaron anteriormente. Los argumentos de la subrutina son el arreglo de centroides de los cuadriláteros (Centroides), el vector posición

del observador  $\vec{V}$ , definido por el ángulo  $\varphi$  (fiv),  $\theta$  (tetav) y Dist (Dist), véase ecuación (A.5).

Las variables Ex, Ey y Ez, almacenan los valores negativos de las componentes escalares del vector  $\vec{V}$ . Posteriormente se obtiene la distancia del observador al centroide, para cada uno de los centroides, y se almacena en el arreglo arr\_dist.

#### Subrutina Obtén\_arreglo\_índices

##### Definición:

```
Sub Obtén_arreglo_índices (arr_cuad1 As cuadriláteros, arr_i() As
Integer)
  Dim n_it As Integer
  n_it = UBound(arr_cuad1.arr)
  ReDim arr_i(1 To n_it)
  For i = 1 To n_it
    arr_i(i) = i
  Next i
End Sub
```

##### Comentarios:

La subrutina Obtén\_arreglo\_índices obtiene el vector de índices In\_cuad necesarios para llevar un registro del orden en que queden ordenadas las distancias del observador a los centroides después de ejecutar el algoritmo "Quicksort". La asignación de los índices es por demás sencilla, de acuerdo con (A.10). Los argumentos de la subrutina son el arreglo de cuadriláteros arr\_cuad1, del cual se extrae su dimensión con Ubound. El arreglo In\_cuad se devuelve en arr\_i.

#### Subrutina Obtén\_arreglo\_píncel

##### Definición:

```
Sub Obtén_arreglo_píncel (V() As Double, Vmin As Double, Vmax As Double,
arr_píncel() As Long)
  Dim n_ant As Integer
  Dim n_it As Integer
  Dim n_act As Integer
```



```

Dim res As Long
Dim Val As Double
Dim i As Integer

    n_ant = UBound(arr_pincel)
    n_it = UBound(V)
    n_act = n_ant + n_it

    ReDim Preserve arr_pincel(1 To n_act)
cont_Obtén_arreglo_pincel_vect_corr:

    i = 0
    For K = n_ant + 1 To n_act
        i = i + 1
        arr_pincel(K) = colorRGB (V(i), Vmin, Vmax)
    Next K

End Sub

```

Comentarios:

La subrutina `Obtén_arreglo_pincel` hace un llamado sucesivo a la subrutina `colorRGB` con el objeto de obtener y almacenar en un arreglo `arr_pincel` los valores RGB correspondientes a los valores de una cantidad física cualquiera acotada entre  $V_{min}$  y  $V_{max}$ . El vector  $V$  contiene los valores de dicha cantidad física y para el caso en que se utiliza son las presiones de las celdas del yacimiento, permeabilidades, porosidades o compresibilidades de la roca. Los argumentos de la subrutina son el arreglo de valores de la propiedad ( $V$ ) y los valores mínimo y máximo ( $V_{min}$  y  $V_{max}$ , respectivamente) de la propiedad física. Devuelve como resultado el arreglo `arr_pincel`. La subrutina ha sido adecuada para que el arreglo `arr_pincel` sea redimensionado si es que ya contenía valores, pues la subrutina sirve también para extraer los colores de valores de gastos en tuberías. Como se verá en la definición de la subrutina `dibuja_cuadriláteros`, sólo es necesario obtener una vez el arreglo de colores para graficar varias veces los cuadriláteros (variando  $\varphi$  ( $fiv$ ),  $\theta$  ( $tetav$ ) y  $Dist$  ( $Dist$ )). Cuando se cambie la propiedad que se graficará, o bien cambien los valores de las propiedades de la malla, entonces debe llamarse de nuevo a la subrutina `Obtén_arreglo_pincel`.

Subrutina dibuja\_cuadriláterosDefinición

Sub dibuja\_malla\_prop2(caja As PictureBox, arr\_cuad As cuadriláteros, control As arr\_vector,  
 pinceles1() As Long, escX As Double, escY As Double, escZ As Double, dist As Double, ele As  
 Double, tetav As Double, fiv As Double)

Dim pincelv As Long  
 Dim arr\_dist1() As Double  
 Dim arr\_índices() As Integer  
 Dim n\_cuad As Integer  
 Dim i As Integer  
 Dim K As Integer

Dim cost As Double  
 Dim sent As Double  
 Dim senf As Double  
 Dim cosf As Double  
 Dim VS\_malla As tipo\_vectorS  
 Dim Fact\_x As Double  
 Dim Fact\_y As Double  
 Dim izq As Double  
 Dim tope As Double

Dim res As Long  
 Dim resl As Long  
 Dim cuadrAPI\_P As Arr\_VAPI

cost = Cos(Pi / 180 \* tetav)  
 sent = Sin(Pi / 180 \* tetav)  
 cosf = Cos(Pi / 180 \* fiv)  
 senf = Sin(Pi / 180 \* fiv)

Obtén\_arreglo\_distancias1 control, fiv, tetav, dist \* 300, arr\_dist1  
 Obtén\_arreglo\_índices arr\_cuad, arr\_índices

n\_cuad = UBound(arr\_cuad.arr)  
 QSort arr\_dist1, arr\_índices, 1, n\_cuad

Fact\_x = caja.Width / caja.ScaleWidth / 15.1  
 Fact\_y = caja.Height / caja.ScaleHeight / 15.1  
 izq = -caja.ScaleLeft  
 tope = -caja.ScaleTop

caja.Cls

```

For i = n_cuad To 1 Step 1
  K = arr_indices(i)

  proy2D arr_cuad.arr(K).Esq(1).arr, dist, ele, cost, sent, cosf, senf, VS_malla.arr
  cuadrAPI_P.arr(1).X = (izq + VS_malla.arr(1)) * Fact_x
  cuadrAPI_P.arr(1).Y = (tope + VS_malla.arr(2)) * Fact_y

  proy2D arr_cuad.arr(K).Esq(2).arr, dist, ele, cost, sent, cosf, senf, VS_malla.arr
  cuadrAPI_P.arr(2).X = (izq + VS_malla.arr(1)) * Fact_x
  cuadrAPI_P.arr(2).Y = (tope + VS_malla.arr(2)) * Fact_y

  proy2D arr_cuad.arr(K).Esq(3).arr, dist, ele, cost, sent, cosf, senf, VS_malla.arr
  cuadrAPI_P.arr(3).X = (izq + VS_malla.arr(1)) * Fact_x
  cuadrAPI_P.arr(3).Y = (tope + VS_malla.arr(2)) * Fact_y

  proy2D arr_cuad.arr(K).Esq(4).arr, dist, ele, cost, sent, cosf, senf, VS_malla.arr
  cuadrAPI_P.arr(4).X = (izq + VS_malla.arr(1)) * Fact_x
  cuadrAPI_P.arr(4).Y = (tope + VS_malla.arr(2)) * Fact_y

  pincelv = CreateSolidBrush(pinceles1(arr_cuad.arr(K).I_cel))

  res1 = SelectObject(caja.hdc, pincelv)
  res = Polygon(caja.hdc, cuadrAPI_P.arr(1), 4)
  res = DeleteObject(res)
  res = DeleteObject(res1)
  res = DeleteObject(pincelv)

Next i
  caja.Refresh
End Sub

```

Comentarios:

La subrutina `dibuja_cuadriláteros` pinta en una caja de dibujo `caja` los cuadriláteros del arreglo de cuadriláteros `arr_cuad` coloreando cada uno según los colores almacenados en el arreglo `pinceles1` (vea en la subrutina la expresión `pinceles1(arr_cuad.arr(K).I_cel)`). Antes de proseguir hay que notar algunas cosas: Visual Basic no posee subrutinas propias como el lenguaje C para dibujar cuadriláteros. Por esta razón se tuvieron que utilizar funciones de las librerías API (Application User Interface) de archivos .dll. Las funciones que se utilizaron son:

- `Polygon`: Dibuja polígonos sobre un cuadro de dibujo o una ventana.

- `CreateSolidBrush`: Crea un objeto pincel (Brush) de acuerdo a un código RGB de color, mismo que será usado para rellenar los objetos gráficos que se dibujen
- `SelectObject`: Selecciona un objeto previamente creado, por ejemplo un objeto pincel para después utilizarse como color de fondo de los polígonos.
- `DeleteObject`: Elimina un objeto creado con anterioridad, como puede ser un polígono o un pincel.

Para utilizar estas funciones es necesario declararlas en algún módulo del programa. Las declaraciones son las siguientes:

```
Declare Function Polygon Lib "gdi32" (ByVal hdc As Long, lpPoint As POINTAPI, ByVal nCount As Long) As Long
```

```
Declare Function CreateSolidBrush Lib "gdi32" (ByVal crColor As Long) As Long
```

```
Declare Function SelectObject Lib "gdi32" (ByVal hdc As Long, ByVal hObject As Long) As Long
```

```
Declare Function DeleteObject Lib "gdi32" (ByVal hObject As Long) As Long
```

La documentación de dichas funciones puede encontrarse en los manuales de lenguaje C para Windows. Para el caso que nos atañe, hay una estructura que aún no hemos definido y es necesaria para la utilización de la función `Polygon`. Como se observa en la declaración de la función, se requiere una variable de tipo `POINTAPI`. Este tipo está definido como:

```
Type POINTAPI
    X As Long
    Y As Long
End Type
```

Cada `POINTAPI` es un vértice de un polígono que se vaya a dibujar por medio de `Polygon`. No obstante estar declarado el argumento en Visual Basic `lpPoint` como una variable simple tipo `POINTAPI`, en la definición de C la variable es un arreglo (apuntador

a POINTAPI). La manera de pasar arreglos de Visual Basic a funciones de C es pasando el primer elemento del arreglo, como en el ejemplo siguiente:

```
res = Polygon(caja.hdc, cuadrAPI_P.arr(1), 4)
```

Como se observa, el arreglo `cuadrAPI_P.arr` se pasa a la función de C haciendo referencia a su primer elemento. Recomendamos que el lector lea la bibliografía (Enstlinger (1992), Microsoft Visual Basic Language Reference (1995)) para obtener detalles acerca del uso de funciones de .dll's.

Ahora comentaremos las operaciones que realiza la subrutina `dibuja_cuadriláteros`. Comienza calculando los senos y cosenos de los ángulos  $\varphi$  y  $\theta$ , posteriormente obtiene el arreglo de distancias de los vértices de los cuadriláteros al vector posición. Acto seguido, obtiene el arreglo de índices y hace una llamada a la subrutina `qsort` para que ordene los arreglos de distancias y de índices en orden decreciente. Debido a que la función `Polygon` no respeta las escalas que tenga el cuadro de texto o ventana donde se dibujen los polígonos, es necesario calcular factores de escalas en X y Y (`Fact_X` y `Fact_Y`, respectivamente) para graficar según la escala los polígonos. De igual manera es necesario encontrar el extremo superior izquierdo (`izq` y `tope`) del cuadro para calcular las coordenadas X y Y reales para unas coordenadas X y Y relativas acordes al sistema coordenado de la caja de dibujo. A continuación se entra a un ciclo para "convertir" a 2D las cuatro esquinas de cada cuadrilátero de acuerdo al orden del `arr_índices` (que fue modificado ya por `qsort`). Cada vector bidimensional de los vértices del cuadrilátero en cuestión es almacenado en un elemento del arreglo `cuadrAPI.arr`, posteriormente se crea un objeto pincel de acuerdo a los códigos RGB almacenados en `pincales1`, selecciona el objeto pincel recién creado y dibuja el polígono K-ésimo. Puesto que los objetos gráficos dibujados en el cuadro de dibujo no aparecen automáticamente, al final del ciclo se llama al método `Refresh` de la caja de dibujo para actualizarla con respecto a los polígonos dibujados.

## APÉNDICE B

# CORRELACIONES PVT PARA GASES SECOS

En la sección de organización se presentaron las estructuras para almacenar la información relacionada con la mezcla gaseosa del yacimiento (estructuras Gas, Heptanos más e Info\_PVT.). En la sección de interfaz gráfica con el usuario, en el capítulo 5, sección V.3.6 o, se presentaron las ventanas en las cuales se especifican las propiedades pseudocríticas y composicionales del gas y se llevan a cabo los ajustes del factor de compresibilidad, viscosidad y pseudopresión por medio de polinomios. Sin embargo, no se presentaron las correlaciones por medio de las cuales se calculan propiedades, en función de otras si es que no se conocen. En este apéndice presentaremos dichas correlaciones, aunque no pretendemos ser exhaustivos; De hecho sólo presentaremos aquéllas que utilizamos en nuestro simulador. Existen muchas obras (incluyendo una tesis de la Facultad de Ingeniería reciente, ver Santamaría(2001)) donde se revisan todas las correlaciones (Bánzer(1996), McCain(1991)) o parte de ellas (Garaicochea y cols(1991), Mc Cain(1989), Economides y cols(1994)), además de que en la mayoría de los libros de ingeniería de yacimientos se dedica alguna sección o apéndice para tratar dichas correlaciones.

Puesto que han sido presentadas ya muchas veces en muchos libros, y es fácil que el lector las consulte en la literatura, quisimos hacer una pequeña aportación diferente a lo publicado hasta ahora y proporcionar en forma de pseudocódigo de Visual Basic las correlaciones. Con ello se puede tener mejor idea de su implementación en nuestro

simulador y pueden ser adaptadas para ser incluidas en cualquier otro programa, no necesariamente de Visual Basic.

## B.1 OBTENCIÓN DE LAS PROPIEDADES SEUDOCRÍTICAS DEL GAS (SIN CONSIDERAR COMPUESTOS NO HIDROCARBUROS)

En nuestro programa se pueden obtener la presión pseudocrítica y temperatura pseudocrítica por dos medios:

- Contando con la densidad relativa del gas solamente
- Contando con las fracciones molares de componentes hidrocarburos, las propiedades críticas de éstos, así como fracción molar de la fracción pesada (C7+) y las propiedades de ésta.

En el primer caso utilizamos la correlación de Sutton (1985) y en el segundo la correlación de W.F. Stewart, S.F. Burkhardt y D. Voo (1959).

### Correlación de Sutton

```
' Correlación de Sutton para Ppc
Función Ppc_Sutton (dg como Doble) como Doble
    Ppc_Sutton = 756.8 - 131 * dg - 3.6 * dg ^ 2
Fin de la Función
```

```
'Correlación de Sutton para Tpc
Función Tpc_Sutton(dg como Doble) como Doble
    Tpc_f = 169.2 + 349.5 * dg - 74 * dg ^ 2
Fin de la Función
```

### Correlación de Stewart, Burkhardt y Voo

```
'Correlación de W.F. Stewart, S.F. Burkhardt y D. Voo (1959)
'seudocríticas de una mezcla de gases.
'Deben tenerse en cuenta las siguientes estructuras:
```

## 'Tipo Gas

```
' Mm como Doble 'Masa molecular del gas
' Pc como Doble 'Presión crítica del gas [psi]
' Tc como Doble 'Temperatura crítica del gas [R]
' y como Doble 'fracción molar del gas en una mezcla
' dr como Doble 'densidad relativa del gas
' Fin del Tipo
```

## ' Tipo Heptanosmás

```
' Mm como Doble 'Masa molecular de la fracción pesada
' Pc como Doble 'Presión crítica de la fracción pesada [psi]
' Tc como Doble 'Temperatura crítica de la fracción pesada [R]
' y como Doble 'fracción molar de la fracción en una mezcla
' dr como Doble 'densidad relativa de la fracción
' Fin del Tipo
```

## ' Tipo MezclaGas

```
' g() como Gas 'Arreglo de gases en una mezcla gaseosa
' n como entero 'Número de gases en la mezcla
' Hm como Heptanosmás 'Fracción pesada
' End Type
```

## Función Propiedades\_Seudocríticas\_Composición (g1 As MezclaGas) como Gas

```
Dim Mmv como Doble 'Masa molecular
Dim J como Doble
Dim Jp como Doble
Dim K como Doble
Dim Kp como Doble
Dim Ej como Doble
Dim EK como Doble
Dim Fj como Doble
Dim S1, S2, S3 como Doble 'Sumatorias
```

Si  $g1.Hm.Y > 0$  Entonces ' Si contiene la fracción pesada calcula factores de corrección

$$Fj = g1.Hm.Tc / g1.Hm.Pc * (1 / 3 * g1.Hm.y + 2 / 3 * g1.Hm.y ^ 2)$$



$E_j = 0.6081 * F_j + 1.1325 * F_j^2 - 14.004 * F_j * g_{1.Hm.y} + 64.434 * F_j * g_{1.Hm.y}^2$

$EK = (0.3129 * g_{1.Hm.y} - 4.8156 * g_{1.Hm.y}^2 + 24.3751 * g_{1.Hm.y}^3) * (g_{1.Hm.Tc} / g_{1.Hm.Pc})^{.5}$

Fin del Si

$M_{mv} = g_{1.Hm.y} * g_{1.Hm.Mm}$  `Inicializa la masa molecular aparente con la contribución de la fracción pesada

Para  $i = 1$  hasta  $g_{1.n}$  `Realiza una serie de sumatorias

$M_{mv} = M_{mv} + g_{1.g(i).y} * g_{1.g(i).Mm}$

$S_1 = S_1 + g_{1.g(i).Y} * g_{1.g(i).Tc} / g_{1.g(i).Pc}$

$S_2 = S_2 + g_{1.g(i).Y} * (g_{1.g(i).Tc} / g_{1.g(i).Pc})^{.5}$

$S_3 = S_3 + g_{1.g(i).Y} * g_{1.g(i).Tc} / (g_{1.g(i).Pc}^{.5})$

Siguiente  $i$

$J = 1 / 3 * S_1 + 2 / 3 * S_2^2$

$K = S_3$

$J_p = J - E_j$

$K_p = K - EK$

Propiedades\_Seudocríticas\_Composición.Tc =  $K_p^2 / J_p$

Propiedades\_Seudocríticas\_Composición.Pc =  $PropPc.Tc / J_p$

Propiedades\_Seudocríticas\_Composición.Mm =  $M_{mv}$

Propiedades\_Seudocríticas\_Composición.dr =  $M_{mv} / M_{ma}$  `Mma=Masa molecular del aire= 28.964[g/mol]

Fin de la Función

## B.2 CORRECCIONES DE LAS PROPIEDADES SEUDOCRÍTICAS POR COMPONENTES NO HIDROCARBUROS

Ya sea que las propiedades seudocríticas se obtengan de la correlación de Sutton o de la correlación de Stewart, Burkhardt y Voo, se deben de corregir si es que contienen gases amargos. En nuestro simulador esto lo hacemos con la corrección de Wichert y Aziz (1972.)

### Correlación de Wichert y Aziz

'Corrección de Ppc y Tpc para H2S y CO2

'Wichert, E. y Aziz, K.

'Ppc=Presión pseudocrítica calculada con Sutton o Stewart, Burkhardt y Voo [psia]

'Tpc=Temperatura pseudocrítica calculada con Sutton o Stewart, Burkhardt y Voo [R]

'yS=fracción molar del sulfuro de hidrógeno

'yC=Fracción molar del dióxido de carbono

Subrutina corrección\_WichAziz(Ppc como Doble, Tpc como Doble, yS como Doble, yC como Doble)

Dim E As Double

$$E = 120 * ((yS + yC) ^ 0.9 - (yS + yC) ^ 1.6) + 15 * (yS ^ 0.5 - yS ^ 4)$$

$$Ppc = Ppc * (Tpc - E) / (Tpc + yS * (1 - yS) * E)$$

$$Tpc = Tpc - E$$

Fin de la Subrutina

### B.3 PROPIEDADES SEUDOCRÍTICAS DE LA FRACCIÓN PESADA (C7+)

En la correlación de Stewart, Burkhardt y Voo es necesario conocer las propiedades pseudocríticas de la fracción pesada (llamada C7+, es decir que incluye a todos los heptanos e hidrocarburos de mayor masa molecular que éstos.) Las propiedades pseudocríticas de esta fracción pueden obtenerse con la correlación de Sutton y Whitson a partir de la temperatura de líquido saturado de la fracción a condiciones estándar y la densidad relativa al agua de la fracción a condiciones estándar.

### Correlación de Sutton y Whitson

'Método de Sutton y Whitson para Ppc y Tpc de los C7+

'Tls =Temperatura de líquido saturado de la fracción pesada [R]

'Hm1=fracción pesada C7+

'Deben tenerse en cuenta las siguientes estructuras:

'Tipo Gas

' Mm como Doble 'Masa molecular del gas  
 ' Pc como Doble 'Presión crítica del gas [psi]  
 ' Tc como Doble 'Temperatura crítica del gas [R]  
 ' y como Doble 'fracción molar del gas en una mezcla  
 ' dr como Doble 'densidad relativa del gas  
 ' Fin del Tipo

' Tipo Heptanosmás

' Mm como Doble 'Masa molecular de la fracción pesada  
 ' Pc como Doble 'Presión crítica de la fracción pesada [psi]  
 ' Tc como Doble 'Temperatura crítica de la fracción pesada [R]  
 ' y como Doble 'fracción molar de la fracción en una mezcla  
 ' dr como Doble 'densidad relativa de la fracción  
 ' Fin del Tipo

Función Propiedades\_Seudocríticas\_Hm (Hm1 como Heptanosmás, Tls como Doble) como Gas

Dim Ppc como Doble 'presión seudocrítica de la fracción pesada[psi]  
 Dim Tpc como Doble 'Temperatura seudocrítica de la fracción pesada [R]  
 Dim A como Doble 'Coeficiente involucrado en los cálculos de la  
 'presión seudocrítica

$$Tpc = 341.7 + 811 * Hm1.dr + (0.4244 + 0.1174 * Hm1.dr) * Tls + (0.4669 - 3.2623 * Hm1.dr) * 10 ^ 5 / Tls$$

$$A = 8.3634 - 0.0566 / Hm1.dr$$

$$A = A - (0.24244 + 2.2898 / Hm1.dr + 0.11857 * Hm1.dr ^ -2) * (10 ^ -3) * Tls$$

$$A = A + (1.4685 + 3.648 / Hm1.dr + 0.47227 * Hm1.dr ^ -2) * 10 ^ -7 * Tls ^ 2$$

$$A = A - (0.42019 + 1.6977 * Hm1.dr ^ 2) * (10 ^ -10) * Tls ^ 3$$

$$Ppc = \text{Exp}(A)$$

Propiedades\_Seudocríticas\_Hm.Pc = Ppc

Propiedades\_Seudocríticas\_Hm.Tc = Tpc

Fin de la Función

### Correlación de Whitson

La temperatura de líquido saturado se obtiene con la correlación de Whitson.

'Ecuación de Witson para Tliqsat(Patm) de C7+

Función T\_liqu\_Sat\_Hm (Hm1 como Heptanosmás) como Doble

$$T\_liqu\_Sat\_Hm = (4.5579 * Hm1.Mm ^ 0.15178 * Hm1.dr ^ 0.15427) ^ 3$$

Fin de la Función

## B.4 CÁLCULO DEL FACTOR DE COMPRESIBILIDAD

En el presente trabajo ocupamos dos correlaciones. La de Gopal (1977) y la de Dranchuk y Abu-Kassem (1975). Ambas se basan en la ley de estados correspondientes (que se consiste en suponer que el factor de compresibilidad es sólo función de la presión seudorreducida y temperatura seudorreducida del gas). Ambas intentan reproducir las curvas de la correlación gráfica de Standing y Katz. La primera es un conjunto de polinomios de  $Z$  en función de la presión seudorreducida para diferentes temperaturas seudorreducidas. En la segunda se encuentra implícito el factor de compresibilidad y se debe obtener por algún método numérico. La correlación de Dranchuk y Abu-Kassem es recomendable, por que es la que mejor se ajusta a las curvas de la correlación de Standing y Katz. Por otro lado, la de Gopal tiene la ventaja de no ser iterativa y puede usarse para obtener una estimación inicial de  $z$  para ser usada en la de Dranchuk-Abu-Kassem.

En el presente trabajo resolvimos la ecuación de Dranchuk y Abu-Kassem por el método de Newton Raphson. Puesto que la velocidad de convergencia (y la convergencia misma) de este método, dependen del valor inicial supuesto de  $z$ ; se utiliza la ecuación de Gopal para encontrar un valor aproximado inicial.

**Correlación de Gopal**

'Pr=Presión seudorreducida, Tr=Temperatura seudorreducida

Función  $z_{Gopal}$ ( Pr como Doble, Tr como Doble) como Doble

Si  $Pr \geq 0$  y  $Pr < 1.2$  Entonces

Si  $Tr \geq 1.05$  y  $Tr < 1.2$  Entonces

$$z_{Gopal} = Pr * (1.6643 * Tr - 2.2114) - 0.3647 * Tr + 1.4385$$

Si no, si  $Tr \geq 1.2$  y  $Tr < 1.4$  Entonces

$$z_{Gopal} = Pr * (0.0522 * Tr - 0.8511) - 0.0364 * Tr + 1.049$$

Si no, si  $Tr \geq 1.4$  y  $Tr < 2$  Entonces

$$z_{Gopal} = Pr * (0.1391 * Tr - 0.2988) + 0.0007 * Tr + 0.9969$$

Si no, si  $Tr \geq 2$  y  $Tr \leq 3$  Entonces

$$z_{Gopal} = Pr * (0.0295 * Tr - 0.0825) + 0.0009 * Tr + 0.9967$$

Fin del Si

Si no, si  $Pr \geq 1.2$  y  $Pr < 2.8$  Entonces

Si  $Tr \geq 1.05$  y  $Tr < 1.2$  Entonces

$$z_{Gopal} = Pr * (-1.357 * Tr + 1.4942) + 4.6315 * Tr - 4.7009$$

Si no, si  $Tr \geq 1.2$  y  $Tr < 1.4$  Entonces

$$z_{Gopal} = Pr * (0.1717 * Tr - 0.3232) + 0.5869 * Tr + 0.1229$$

Si no, si  $Tr \geq 1.4$  y  $Tr < 2$  Entonces

$$z_{Gopal} = Pr * (0.0984 * Tr - 0.2053) + 0.0621 * Tr + 0.858$$

Si no, si  $Tr \geq 2$  y  $Tr \leq 3$  Entonces

$$z_{Gopal} = Pr * (0.0211 * Tr - 0.0527) + 0.0127 * Tr + 0.9549$$

Fin del Si

Si no, si  $Pr \geq 2.8$  y  $Pr < 5.4$  Entonces

Si  $Tr \geq 1.05$  y  $Tr < 1.2$  Entonces

$$z_{Gopal} = Pr * (-0.3278 * Tr + 0.4752) + 1.8223 * Tr - 1.9036$$

Si no, si  $Tr \geq 1.2$  y  $Tr < 1.4$  Entonces

$$z_{Gopal} = Pr * (-0.2521 * Tr + 0.3871) + 1.6087 * Tr - 1.6635$$

Si no, si  $Tr \geq 1.4$  y  $Tr < 2$  Entonces

$$z_{Gopal} = Pr * (-0.0284 * Tr + 0.0625) + 0.4714 * Tr - 0.0011$$

Si no, si  $Tr \geq 2$  y  $Tr \leq 3$  Entonces

$$z_{Gopal} = Pr * (0.0041 * Tr + 0.0039) + 0.0607 * Tr + 0.7927$$

Fin del Si

Si no, si  $Pr \geq 5.4$  y  $Pr \leq 15$  Entonces

$$z_{Gopal} = Pr * (0.711 + 3.66 * Tr)^{-1.4667 - 1.637 / (0.319 * Tr + 0.522)} + 2.071$$

Fin del Si

### Correlación de Dranchuk-Abou-Kassem

El factor de compresibilidad queda implícito en la correlación. Al resolver la ecuación por el método de Newton-Raphson, queda definida una función  $Fz$  que depende de  $z$ , Presión pseudorreducida y temperatura pseudorreducida.

Función  $z_{\text{DranAbouKassem}}$  ( $Pr$  como Doble,  $Tr$  como Doble,  $tol$  como Doble,  $nit$  como entero) como Doble

Dim  $z_i$  como Doble

Dim  $Fz_i$  como Doble

Dim  $DFz_i$  como Doble

Dim  $A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}, A_{11}$  como Doble

Dim  $R_o$  como Doble

Dim  $R_{o2}$  como Doble

Dim  $R_{o5}$  como Doble

Dim  $Exp\_R_o$  como Doble

Dim  $A, B, C, D, E, F$  como Doble

Dim  $F_2, C_2, D_5$  como Doble

Dim  $A_{11\_2}$  como Doble

$A_1 = 0.3265; A_2 = -1.07; A_3 = -0.5339; A_4 = 0.01569; A_5 = -0.05165$

$A_6 = 0.5475; A_7 = -0.7361; A_8 = 0.1844; A_9 = 0.1056; A_{10} = 0.6134$

$A_{11} = 0.721$

$A_{11\_2} = 0.519841$

$A = 0.27 * Pr / Tr$

$B = (A_1 + A_2 / Tr + A_3 / (Tr ^ 3) + A_4 / (Tr ^ 4) + A_5 / (Tr ^ 5))$

$C = (A_6 + A_7 / Tr + A_8 / (Tr ^ 2))$

$D = A_9 * (A_7 / Tr + A_8 / (Tr ^ 2))$

$E = A_{10} * A_{11} / (Tr ^ 3)$

$F = A_{10} / (Tr ^ 3)$

$C_2 = C * 2; F_2 = F * 2; D_5 = D * 5$

$z_i = z_{\text{Gopal}}(Pr, Tr)$  `Calcula la primera aproximación con la  
`correlación de Gopal.

```

Para i = 1 hasta nit 'comienza las iteraciones del método
  zant = zi
  Ro=A/zi: Ro2=Ro^2: Ro5=Ro^5: Exp_Ro=Ro2*Exp(-A11*Ro2)
  Fzi=-zi+1+ B * Ro + C * Ro2 - D * Ro5 + (F + E * Ro2) * Exp_Ro
  DFzi=-1+(-B*Ro-C2*Ro2+D5*Ro5-F2*Exp_Ro*(1+A11*Ro2-A11_2*Ro2^2))/zi

  zi = zi - Fzi / Dfzi 'Ecuación de iteración de Newton-Raphson

  Si Abs(zant - zi) <= tol Entonces
    z_DranAbouKassem = zi
    Salir de la Función
  Fin del Si
  Siguiente i
  z_DranAbouKassem = -1 'mensaje de que no convergió
Fin de la Función

```

## B.5 CÁLCULO DE LA VISCOSIDAD DE GAS SECO SIN COMPUESTOS NO HIDROCARBUROS

### Correlación de Lee, González y Eakin

En este trabajo utilizamos la correlación de Lee, González y Eakin para calcular la viscosidad del gas. Se requiere la densidad del gas a las condiciones de presión y temperatura a las cuales se quiere evaluar la viscosidad del gas, así como la masa molar del mismo. La función de densidad que implementamos entra como argumento al factor de compresibilidad, con el cual se calcula la densidad.

La masa molar del gas se calcula en base a la densidad relativa, la cual entra también como argumento a la función.

```

'Calcula la viscosidad de un gas natural.
'P [psia], T[R],Viscosidad[cp]
'Correlación de Lee,A.L., González, M.H. y Eakin, B.E.
Función Viscosidad_Lee (P como Doble, T como Doble,z como double, dr como
Doble) como Doble
  Dim A como Doble

```

Dim B como Doble

Dim C como Doble

Dim Mm como Doble

Dim Ro como Doble 'densidad del gas a P,T

Dim vol como Doble ' Volumen específico del gas a P,T

Mm=dr/28.964 'Calcula la masa molar

vol = 10.73149765\*z \* T / (P \* g.Mm) 'Volumen específico del gas a  
'P,T [lbm/ft^3]

Ro = 1 / vol \* 0.01671933757 'densidad del gas a P,T [g/cm^3]

A = (9.379 + 0.01607 \* Mm) \* (T ^ 1.5) / (209.2 + 19.26 \* Mm + T)

b = 3.448 + 986.4 / T + 0.01009 \* Mm

C = 2.447 - 0.2224 \* b

Viscosidad\_Lee = A \* (10 ^ -4) \* Exp(b \* Ro ^ C)

Fin de la Función

### Corrección de Vázquez y Beggs

Utilizamos la corrección de Vázquez y Beggs para corregir la viscosidad calculada con la correlación de Lee y cols. cuando hay concentraciones apreciables de nitrógeno, dióxido de carbono o sulfuro de hidrógeno en la composición del gas.

'Corrección de la visc. del gas para N2,CO2, H2S

Función Corrección\_Visc\_Vázquez (Mugv como Doble, dg como Doble, yN como Doble, yC como Doble, yS como Doble) como Doble

Dim CN como Doble 'Término de corrección por concentración de N2

Dim CS como Doble 'Término de corrección por concentración de H2S

Dim CC como Doble 'Término de corrección por concentración de C2

CN = yN \* (0.003682817 \* Log(dg) + 0.00959)

CC = yC \* (0.0039434 \* Log(dg) + 0.00624)

CS = yS \* (0.0036871 \* Log(dg) + 0.00373)

Mug\_corr = Mugv + CN + CC + CS

Fin de la Función



## B.6 CÁLCULO DEL FACTOR DE VOLUMEN

En la interfaz del usuario, se introduce en tablas ya sea al factor de volumen o al factor de compresibilidad cuando se desee utilizar valores medidos de análisis PVT y no correlaciones. Puesto que el simulador trabaja internamente con factores de compresibilidad, es necesario "convertir" de Bg a Z cuando se introduzcan los primeros como dato PVT. La subrutina que lleva a efecto la "conversión" se basa en la definición de factor de volumen:

### Cálculo de factor de volumen en función de Z

Función Bg\_z (p como Doble, T como Doble, z como Doble) como Doble

$$Bg\_z = 0.02827938769 * z * T / p$$

Fin de la Función

### Cálculo de Z en función del factor de volumen

Es posible llevar a cabo la "transformación inversa", con la misma definición:

Función z\_Bg (p como Doble, T como Doble, Bg como Doble) como doble

$$z\_Bg = 35.36144456 * Bg * p / T$$

Fin de la Función

## B.7 CÁLCULO DE LA SEUDOPRESIÓN DEL GAS REAL

El concepto de seudopresión de gas real fue introducido por Al-Hussainy, Ramey y Crawford (1966). Su definición es:

$$Pp(P,T) = 2 \int_0^P \frac{PdP}{z(P,T)\mu(P,T)} \quad (2.7)$$

La integral del lado derecho de la ecuación (2.7) no se presta fácilmente a ser integrada por métodos analíticos y es por tanto necesario utilizar métodos numéricos.

Si definimos la función  $P_{zVisc}$  como:

$$P_{zVisc}(P, T) = \frac{P}{z(P, T)\mu(P, T)} \quad (A.1)$$

Tendremos que integrar (B.1) desde un valor base de presión (que se considera 0 [lb/pg<sup>2</sup>]) hasta un valor de P variable. La manera como hacemos esto es obteniendo una serie de valores de presión, factores de compresibilidad y viscosidad en función de la presión (ya sea de tablas o a partir de las correlaciones recién vistas.) Posteriormente se evalúa (B.1) para cada valor de presión y se almacenan los valores en un arreglo, junto con las presiones a las cuales se calcularon. Con estos dos arreglos (de valores de (B.1) y presión) se obtienen los coeficientes de polinomios de ajuste Splines cúbicos. Con los polinomios Splines se puede integrar (B.1) de la manera siguiente: Se obtiene la integral de cada polinomio Splines (todos tienen la misma forma), con constante de integración igual a cero. Al evaluarse la seudopresión para algún valor de presión se determina el intervalo de partición de los polinomios donde "cae" el valor de P, después se suman todas las integrales de los intervalos anteriores (integral del polinomio valuada en el límite derecho de su respectivo intervalo, es decir, el área bajo el polinomio de cada intervalo.) Dicha suma es la integral desde 0 hasta el límite izquierdo del intervalo donde cae P. Lo que resta es evaluar en el valor de P la integral del polinomio Splines del intervalo donde "cayó" dicha presión. Finalmente se suma ese valor a la sumatoria de los demás intervalos y el resultado final es la integral de (B.1) desde 0 hasta P.

Cuando se desee evaluar a la seudopresión en otro valor (siempre y cuando esté dentro del intervalo donde se hizo el ajuste de los polinomios), no es necesario volver a obtener los coeficientes de los polinomios Splines, basta con repetir el proceso de búsqueda y suma de las integrales de los polinomios.

## B.8 AJUSTE DE PROPIEDADES POR MEDIO DE POLINOMIOS

Como se dijo en la sección de organización de la información, el simulador, para evaluar la viscosidad, el factor de compresibilidad o la seudopresión a una presión determinada, evalúa polinomios que fueron previamente ajustados a las propiedades en función de la presión. Así mismo se obtiene un polinomio de la presión en función de la seudopresión ("transformación inversa") necesario para evaluar el lado izquierdo de la ecuación (2.44).

El grado del polinomio es seleccionado por el usuario. De acuerdo a las gráficas del ajuste (ver sección V.3) se puede juzgar la bondad del ajuste.

Los polinomios son los siguientes:

- Un polinomio de la viscosidad en función de la presión (arreglo `pol_Visc_P` en la estructura `Info_PVT`.)
- Un polinomio del factor de compresibilidad ( $z$ ) en función de la presión (arreglo `pol_z_P` en la estructura `Info_PVT`.)
- Un polinomio de la seudopresión en función de la presión (arreglo `pol_m_P` en la estructura `Info_PVT`.)
- Un polinomio de la presión en función del *logaritmo natural de la seudopresión* (arreglo `pol_P_m` en la estructura `Info_PVT`.)

Se ajusta la seudopresión en función del *logaritmo natural de la presión*, ya que el ajuste en coordenadas cartesianas no obtienen buenos resultados (debido a la gran magnitud de las seudopresiones en  $[\text{lb}^2/\text{pg}^4/\text{Cp}]$  en comparación con la presión en  $[\text{lb}/\text{pg}^2]$ .) Se usa *logaritmo natural* por que en Visual Basic no existe la función *logaritmo base 10*, y su cálculo requiere de cualquier manera obtener primero el *logaritmo natural*.

El método con el que se ajustan polinomios a curvas en nuestro simulador es mínimos cuadrados generalizado para regresión lineal múltiple. Se puede revisar este método en cualquier libro de probabilidad y/o estadística (Ver por ejemplo Mendenhall y cols(1986) o Walpole y Myers(1991).) En Santamaría(2001) existe, un apéndice dedicado al ajuste de polinomios de grado  $n$  a un conjunto de puntos por este método.

## APÉNDICE C

# CONTROL AUTOMÁTICO DEL INTERVALO DE TIEMPO

El procedimiento que seguimos para el control automático del intervalo de tiempo es el de Aziz y Setari(1979). Estos autores señalan que la selección apropiada del intervalo de tiempo debe garantizar dos aspectos:

- a) La estabilidad de la solución.
- b) Que los errores de truncamiento sean aceptables.

Siguiendo a Aziz y Setari(1979), ambos casos, pueden relacionarse, por lo menos cualitativamente, a la razón de cambio de la presión y la saturación en el yacimiento en cada incremento de tiempo. Entre más rápido sea el cambio de éstos, deben usarse intervalos de tiempo más pequeños.

Para nuestro caso las variables importantes a monitorear son los cambios de presión en el yacimiento (DPLIM) y los cambios de gastos en los pozos (DQLIM, si usamos las condiciones de frontera de Dirichlet), o bien las presiones de fondo fluyente (si usamos las condiciones de frontera de Neuman.) Es cosa aparente que el porcentaje de error instantáneo de balance de materia es una función de la tolerancia en el balance de gastos (condiciones de Dirichlet) y de la tolerancia en el yacimiento, pero queda por investigar hasta qué punto es dependiente del intervalo de tiempo usado.

El método es el siguiente:

Algoritmo C.1.-Control automático del incremento de tiempo.

1.-Calcular:

$$DQMAX^n = \max_i \{ \Delta_i, Q_i^n \}, \text{ donde } i \text{ representa la numeración de los pozos} \quad (C.1)$$

$$DPMAX^n = \max_j \{ \Delta_j, P_j^n \}, \text{ donde } j \text{ representa la numeración de las celdas} \quad (C.2)$$

2.-Calcular  $\Delta t^{n+1}$  como:

$$\Delta t^{n+1} = \min \{ \Delta t_Q, \Delta t_P \}, \quad (C.3)$$

donde

$$\Delta t_Q = \Delta t^n \frac{DQLIM}{DQMAX^n} \quad (C.4)$$

$$\Delta t_P = \Delta t^n \frac{DPLIM}{DPMAX^n} \quad (C.5)$$

3.-Después de que se hayan completado los cálculos para el tiempo  $n+1$ , calcular  $DQMAX^{n+1}$  y  $DPMAX^{n+1}$ . El intervalo de tiempo  $\Delta t^{n+1}$  resultó ser aceptable si:

$$DQMAX^{n+1} \leq C_1 DQLIM \quad (C.6)$$

$$DPMAX^{n+1} \leq C_2 DPLIM \quad (C.7)$$

De lo contrario, calcular  $\Delta t^* = \min \{ \Delta t_Q, \Delta t_P \}$ , donde:

$$\Delta t_Q = \Delta t^{n+1} \frac{DQLIM}{DQMAX^{n+1}} \quad (C.8)$$

$$\Delta t_P = \Delta t^{n+1} \frac{DPLIM}{DPMAX^{n+1}} \quad (C.9)$$

Se hace  $\Delta t^{n+1} = \Delta t^*$  y se itera de nuevo para  $n+1$  (es decir que se repiten los cálculos pero con una  $\Delta t^{n+1}$  más pequeña.) Regresar al paso 3.

Las constantes adimensionales  $C1$  y  $C2$  de las ecuaciones (C.6) y (C.7) son mayores o iguales a 1. Si se seleccionan iguales a 1, el cambio máximo esperado en las variables jamás será mayor que los límites impuestos, pero es posible que haga que el valor de  $\Delta t^{n+1}$  calculado sea rechazado varias veces, lo que incrementaría considerablemente el tiempo de cálculo para una sola iteración. Por otro lado con  $C1$  y  $C2$  mayores a 1, se podría esperar que para el siguiente tiempo  $n$  (es decir  $n+2$ ) las ecuaciones (C.4) y (C.5), por medio de  $\Delta t$ , hagan regresar a  $DQMAX$  y  $DPMAX$  a los límites establecidos.

A continuación se muestra con un ejemplo sencillo cómo opera el algoritmo C.1.

Se tiene una sola variable  $Y$  en función de  $x$ . Se sustituye el algoritmo de iteración del simulador por una simple función,  $f(x)=x^2$ .  $DYLIM=1$ ,  $C1=1$ .

Los puntos que llega a calcular en 50 iteraciones se resumen en la figura C.1:

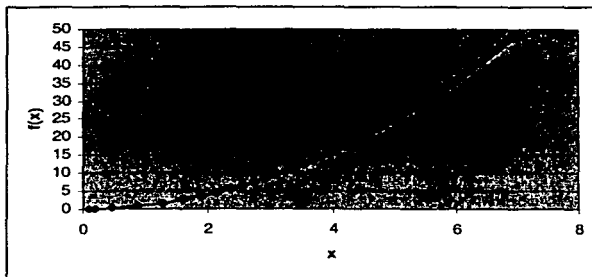


Figura C.1.- Ejemplo del algoritmo C.1: puntos calculados

Nótese cómo los puntos más a la izquierda están más separados que los de la derecha y después se juntan por el incremento más acentuado en la pendiente de la curva.

En la figura C.2 se muestran los  $\Delta x^n$  y los  $\Delta y^n$  (que representan a  $\Delta t^n$  y  $DPMAX^n$ , respectivamente.)

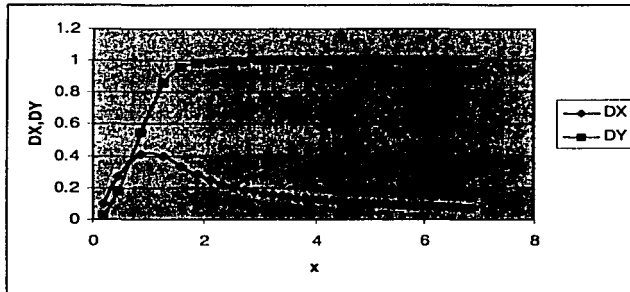


Figura C.2.- Ejemplo del algoritmo C.1, incrementos en X y Y

El límite impuesto al cambio en Y es  $DYLIM=1$ . Nótese cómo los incrementos en x aumentan mientras los incrementos en Y son pequeños en comparación a  $DYLIM$  y luego disminuyen conforme Y se acerca a  $DYLIM$ . Los incrementos en x tienen que disminuir debido a que la pendiente de la curva aumenta. El incremento en Y acaba manteniéndose cerca del límite, pero sin sobrepasarlo.

TESIS CON  
FALLA DE ORIGEN