

## Apéndice A. Descripción de la Base de Datos

En este apéndice describimos la base de datos y parámetros utilizados en la realización de nuestros experimentos.

Nombre del archivo	Tipo de red	Tipo de compresor	Número de nodos	Número de ductos	Número de compresores
Srtgee1.txt	En línea	Snarlin-k1	6	3	2
Srtgee2.txt	Tipo árbol	Snarlin-k1	10	6	3
Srtgee5.txt	Con ciclo	Snarlin-k1	8	5	3

Tabla 11. Archivos empleados de la base de datos para la evaluación de las técnicas de pre-procesamiento.

Las Tablas 12 y 13 muestran los parámetros utilizados del gas y de los ductos.

<b>k</b>	1.287
<b>Z</b>	0.95
<b>R</b>	85.2 lbf-ft/lbm-°R
<b>S<sub>g</sub></b>	0.6248

Tabla 12. Parámetros del gas.

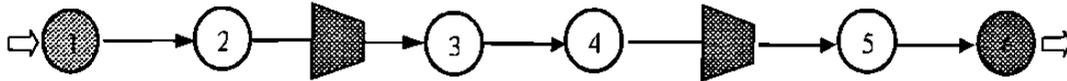
Red	L(millas)	d(ft)	T(°R)	K
En Línea	50	3	519.67	4.1040e07
Tipo árbol	50	3	519.67	4.1040e07
Con ciclo	40	3	519.67	4.1040e07

Tabla 13. Parámetros de los ductos.

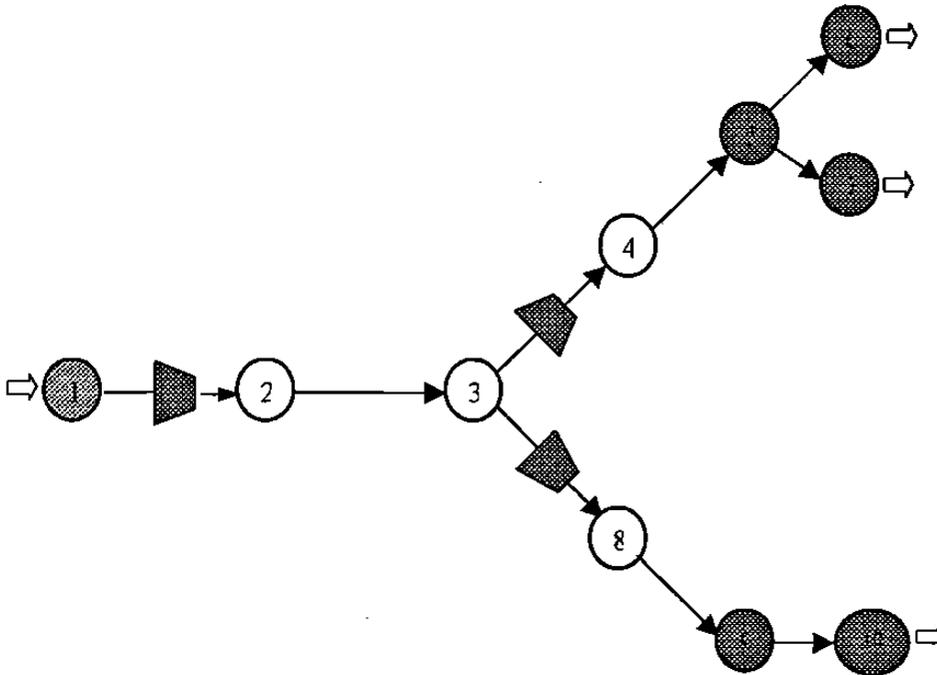
Las Figuras 9, 10 y 11 ilustran el tipo de topología que fue utilizada en las evaluaciones de las técnicas de pre-procesamiento.

Donde:

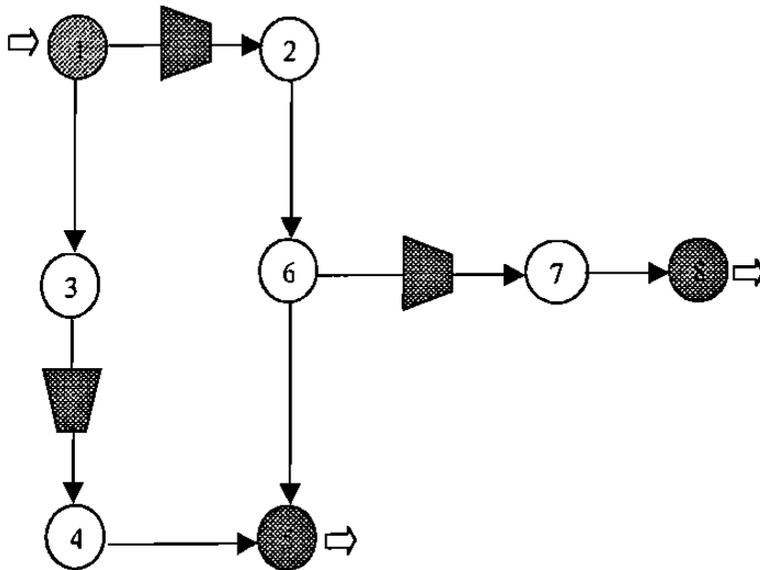
-  Nodo suministro
-  Nodo demanda
-  Nodo de paso
-  Estación compresora



**Figura 9. Red en línea**



**Figura 10. Red tipo árbol**



**Figura 11. Red con ciclo**

**APÉNDICE B**  
**GRÁFICAS DE LOS COMPRESORES**

## Apéndice B. Gráficas de los Compresores

En este apéndice mostramos las gráficas en los dos dominios de operación y eficiencia de los nueve compresores utilizados en las evaluaciones. Éstas fueron obtenidas mediante el uso de los Programas 3, 4 y 5 (en Matlab) los cuales se encuentran disponibles en la página del PISIS (<http://yalma.fime.uanl.mx>).

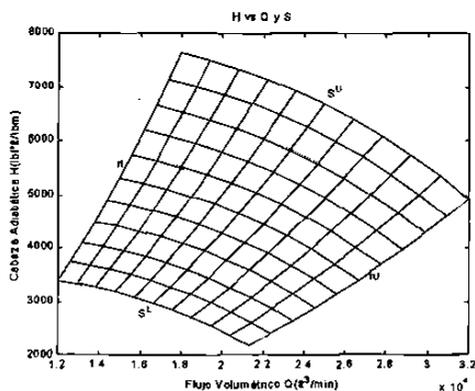


Figura 12. Gráfica del dominio de operación ( $H, Q, S$ ) del compresor Snarlin-k1

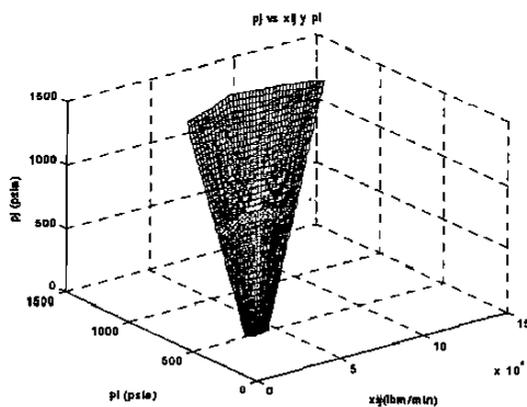


Figura 13. Gráfica del dominio de operación ( $x_{ij}, p_i, p_j$ ) del compresor Snarlin-k1

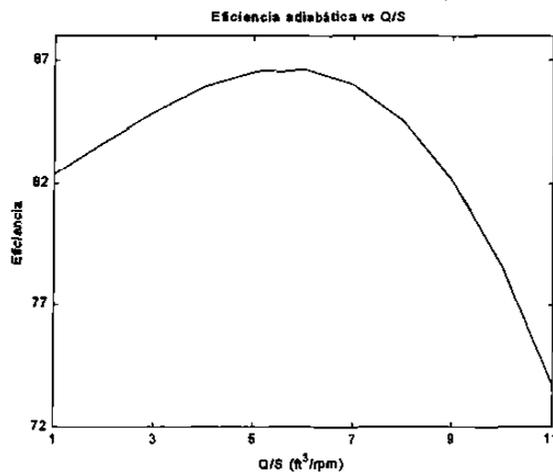


Figura 14. Gráfica de la eficiencia del compresor Snarlin-k1

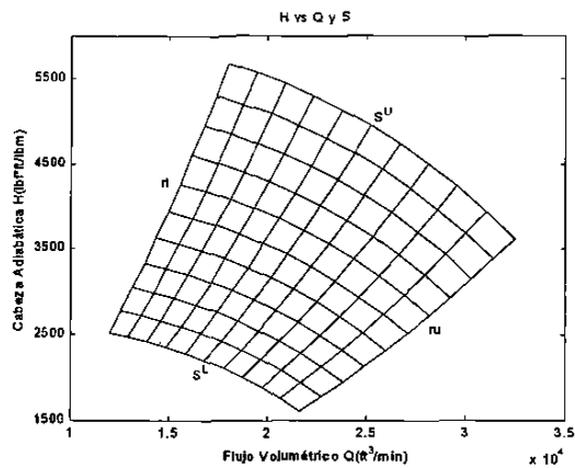


Figura 15. Gráfica del dominio de operación  $(H, Q, S)$  del compresor Rakeey-k1

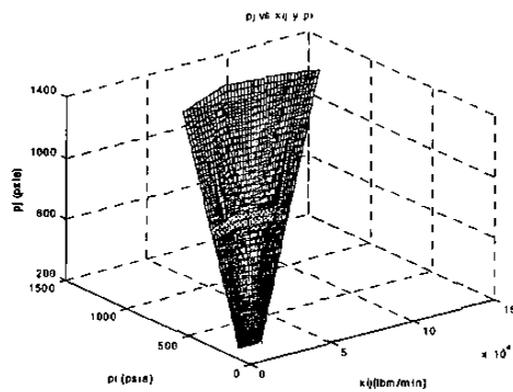


Figura 16. Gráfica del dominio de operación  $(x_{ij}, p_i, p_j)$  del compresor Rakeey-k1

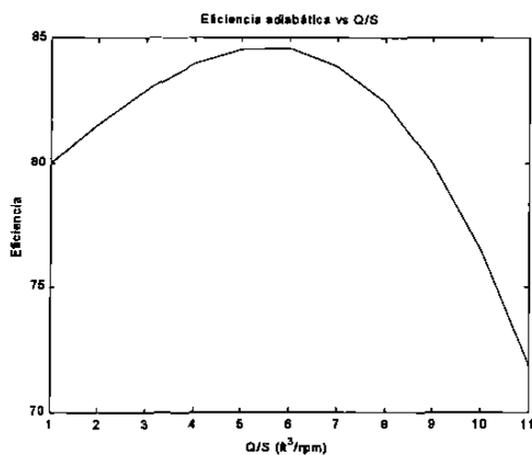


Figura 17. Gráfica de la eficiencia del compresor Rakeey-k1

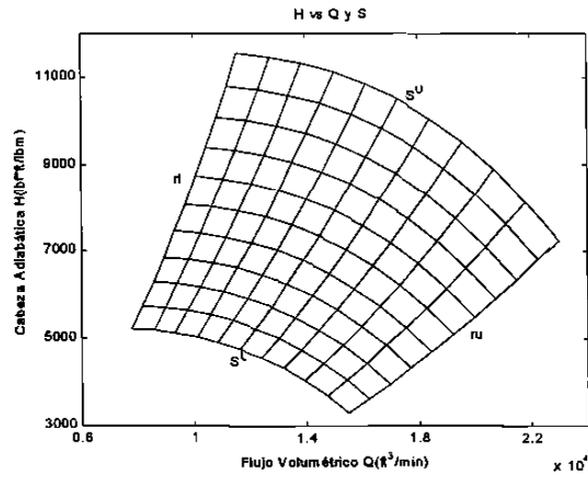


Figura 18. Gráfica del dominio de operación  $(H, Q, S)$  del compresor Rakeey-k2

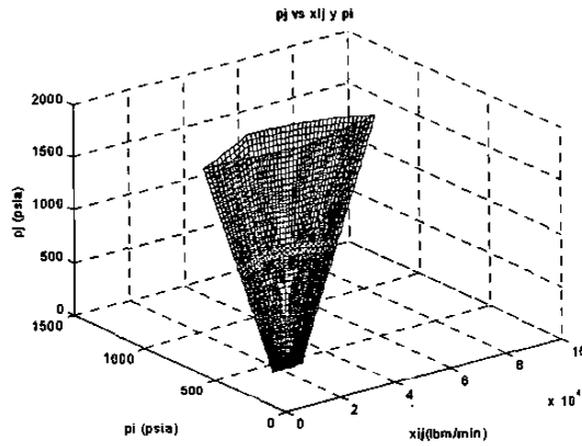


Figura 19. Gráfica del dominio de operación  $(x_i, p_i, p_j)$  del compresor Rakeey-k2

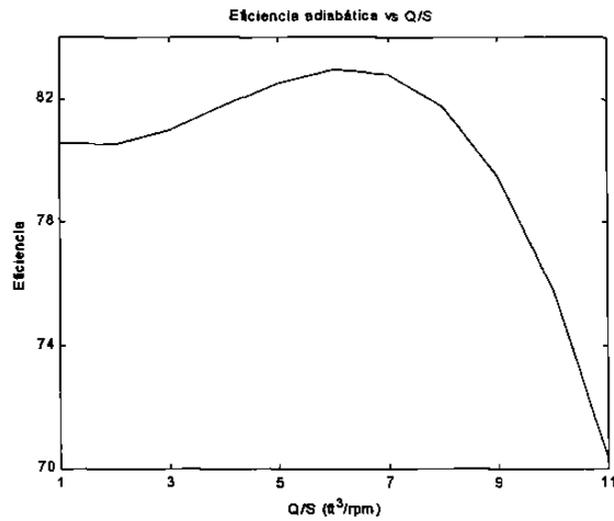


Figura 20. Gráfica de la eficiencia del compresor Rakeey-k2

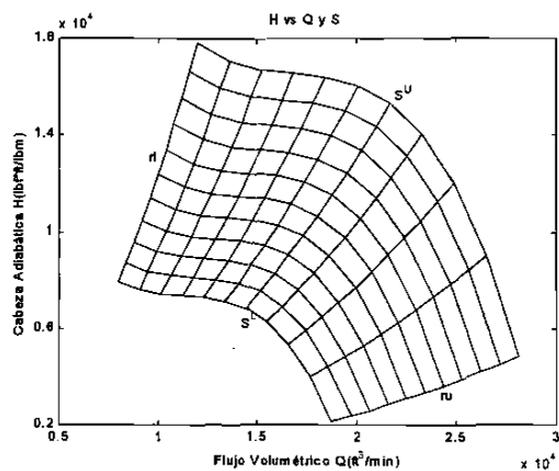


Figura 21. Gráfica del dominio de operación ( $H, Q, S$ ) del compresor Hamper-k1

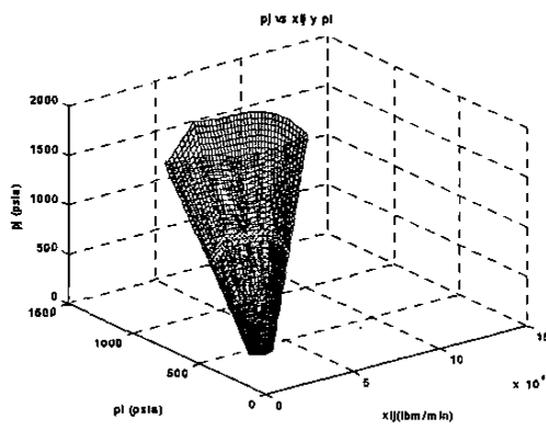


Figura 22. Gráfica del dominio de operación ( $x_i, p_i, p_j$ ) del compresor Hamper-k1

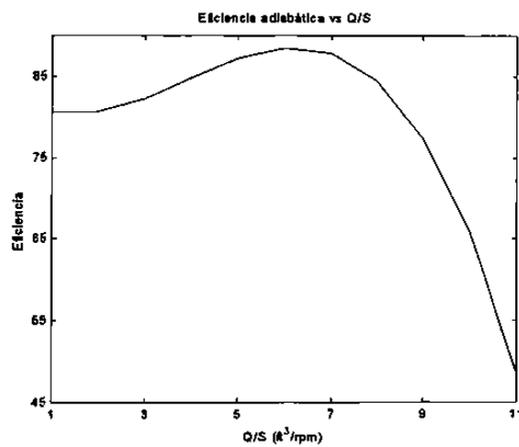


Figura 23. Gráfica de la eficiencia del compresor Hamper-k1

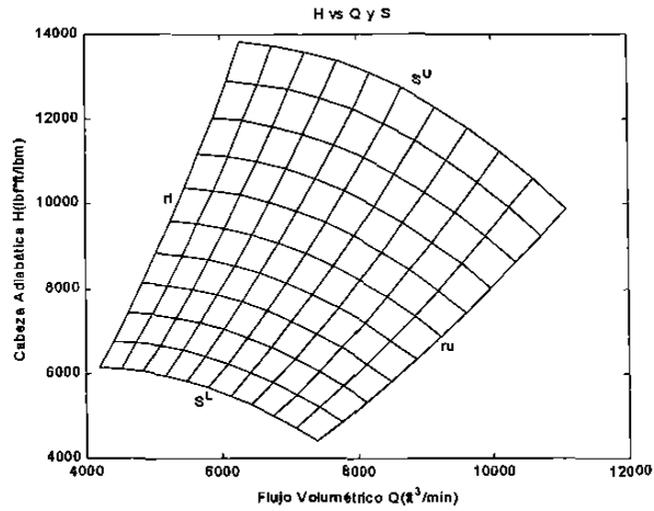


Figura 24. Gráfica del dominio de operación ( $H, Q, S$ ) del compresor Bellvan-k1

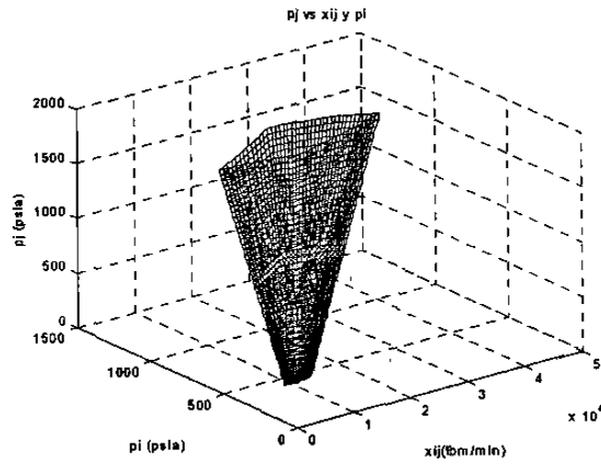


Figura 25. Gráfica del dominio de operación ( $x_{ij}, p_i, p_j$ ) del compresor Bellvan-k1

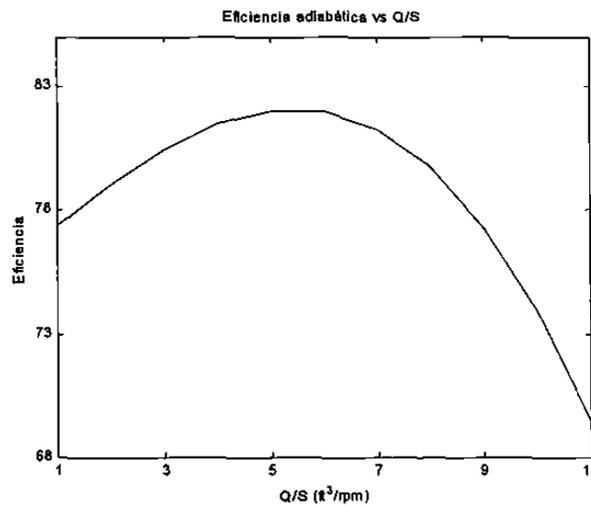


Figura 26. Gráfica de la eficiencia del compresor Bellvan-k1

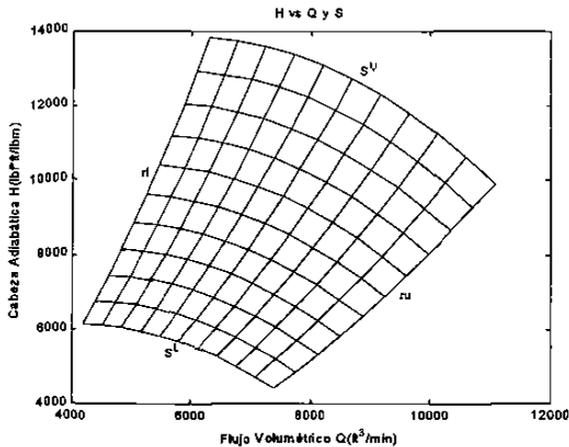


Figura 27. Gráfica del dominio de operación (H, Q, S) del compresor Bellvan-k2

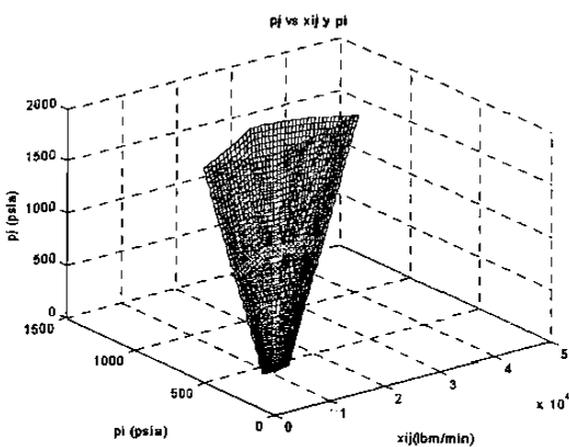


Figura 28. Gráfica del dominio de operación (x<sub>ij</sub>, p<sub>i</sub>, p<sub>j</sub>) del compresor Bellvan-k2

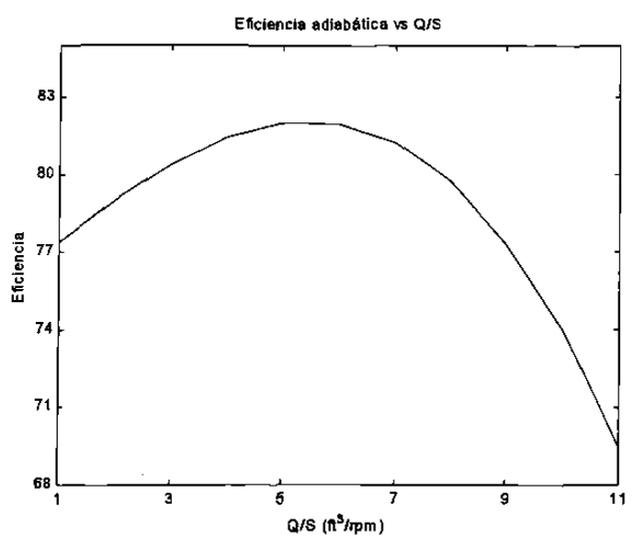


Figura 29. Gráfica de la eficiencia del compresor Bellvan-k2

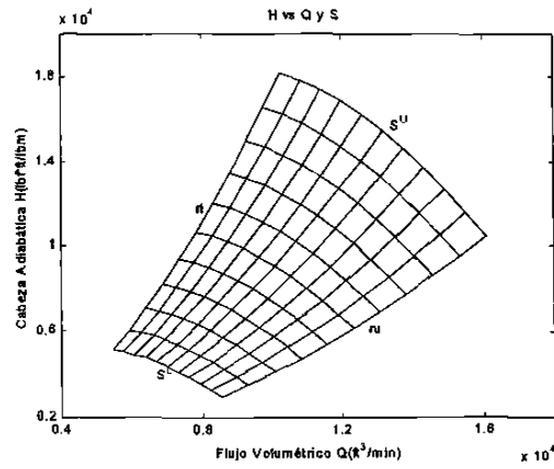


Figura 30. Gráfica del dominio de operación ( $H, Q, S$ ) del compresor Bellvan-k3

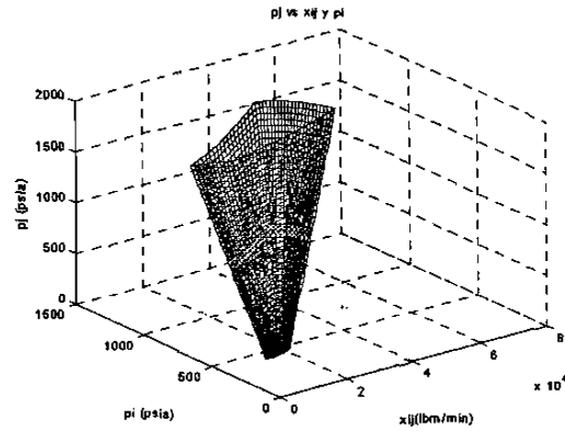


Figura 31. Gráfica del dominio de operación ( $x_{ij}, p_i, p_j$ ) del compresor Bellvan-k3

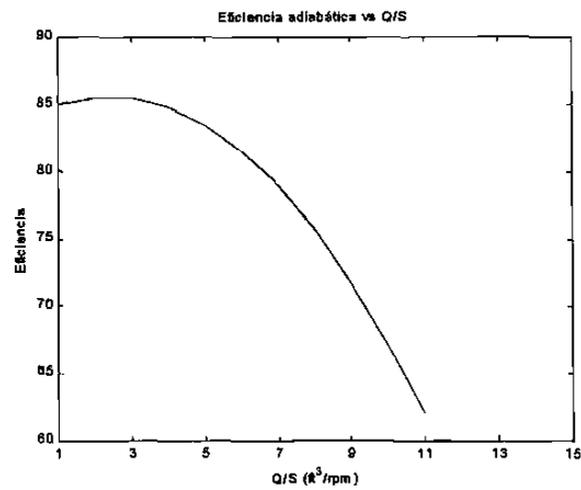


Figura 32. Gráfica de la eficiencia del compresor Bellvan-k3

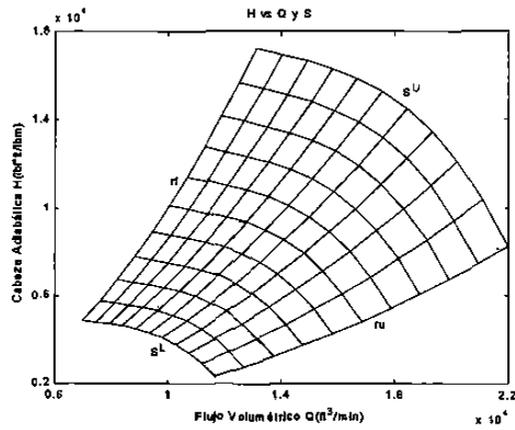


Figura 33. Gráfica del dominio de operación  $(H, Q, S)$  del compresor Bethany-k1

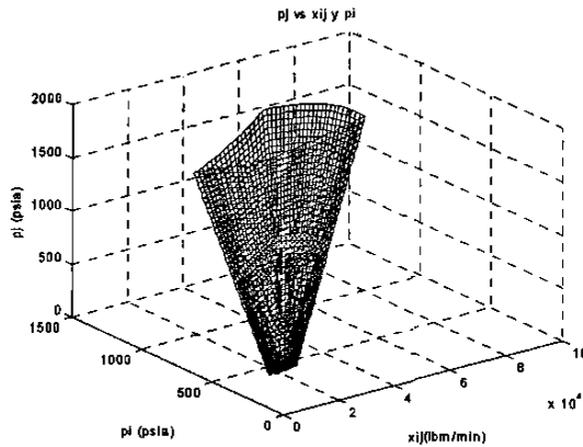


Figura 34. Gráfica del dominio de operación  $(x_{ij}, p_i, p_j)$  del compresor Bethany-k1

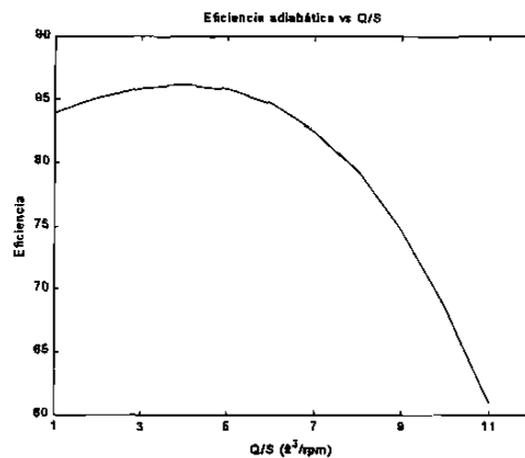


Figura 35. Gráfica de la eficiencia del compresor Bethany-k1

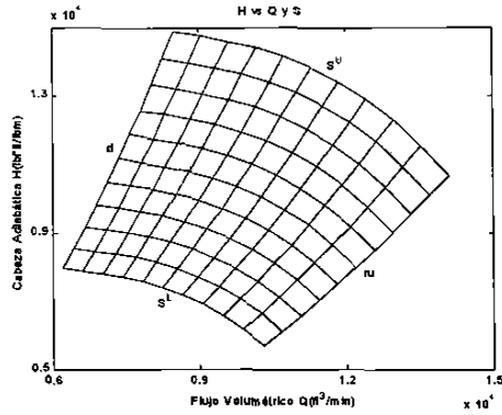


Figura 36. Gráfica del dominio de operación ( $H, Q, S$ ) del compresor Bethany-k2

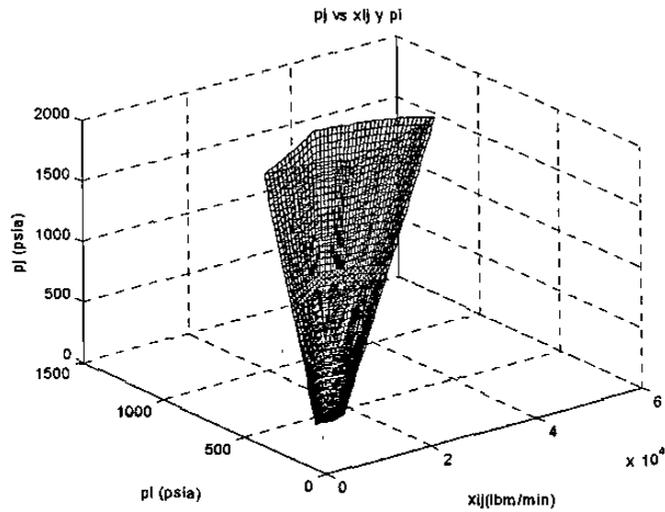


Figura 37. Gráfica del dominio de operación ( $x_{ij}, p_i, p_j$ ) del compresor Bethany-k2

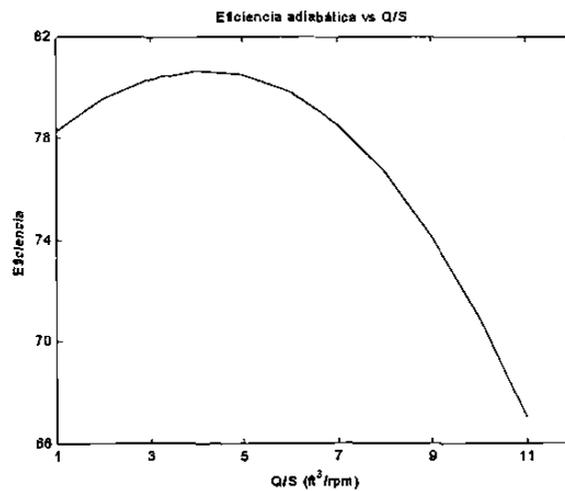


Figura 38. Gráfica de la eficiencia del compresor Bethany-k2

**APÉNDICE C**  
**DESCRIPCIÓN DE GAMS/CONOPT**

## Apéndice C. Descripción de GAMS/CONOPT

Modelos no lineales creados con el modelador algebraico GAMS deben ser resueltos con un algoritmo de optimización no lineal (NLP). Actualmente, existen tres algoritmos estándar disponibles en GAMS estos son: CONOPT, MINOS y SNOPT. El algoritmo CONOPT Drud, (1985 y 1992) que es el que se usa en este estudio de investigación, se encuentra en dos versiones, la versión antigua CONOPT y la versión nueva CONOPT2 que fue la que utilizamos para llevar a cabo las evaluaciones de las técnicas de pre-procesamiento.

Todos los algoritmos para NLP, intentan encontrar un óptimo local. Es casi imposible predecir que tan difícil es resolver un modelo particular con un algoritmo particular, especialmente para modelos NLP. GAMS no puede seleccionar el mejor algoritmo automáticamente. La única forma confiable para encontrar cual algoritmo usar para cada clase de modelo en particular es haciendo experimentos.

CONOPT2 tiene un pre procesador en el cual las variables y ecuaciones recursivas son resueltas y quitadas del modelo. Si se tiene un modelo donde muchas de las ecuaciones pueden ser resueltas una por una, entonces CONOPT2 tomará ventaja de esta propiedad. Similarmente variables intermedias usadas únicamente para definir términos objetivo son eliminadas del modelo y las restricciones son trasladadas dentro de la función objetivo.

CONOPT2 es una herramienta de ayuda que toma los errores fuera del programa de la computadora durante el desarrollo del modelo. CONOPT2 ha sido diseñado para modelos grandes y dispersos. Esto significa que el número de variables y ecuaciones pueden ser grandes. CONOPT2 es diseñado para modelos con funciones suaves, pero también puede ser aplicado a modelos los cuales no tienen funciones diferenciables en GAMS llamados modelos DNLP. Sin embargo, esto no garantiza en absoluto el resultado para esta clase de modelo y algunas veces se obtendrán mensajes con

terminaciones como esta “convergencia demasiado lenta” ó “no hay cambio en la función objetivo aunque el gradiente reducido es mayor que la tolerancia”, lo cual indica que la terminación no fue exitosa. Si es posible, se debería tratar de reformular el modelo DNLP a uno equivalente o aproximadamente de forma equivalente.

## Mensajes en GAMS

GAMS/CONOPT despliega varios tipos de mensajes al terminar. Entre éstos hay cuatro relevantes a la terminación, que son:

- *Solución Óptima: No existen variables super básicas:* Esta solución es una solución extrema óptima localmente, y es determinada por las restricciones únicamente, comúnmente es usualmente muy acertada.
- *Solución Óptima: El gradiente reducido es menor que la tolerancia:* La solución es una solución interior óptima localmente. El componente más grande del gradiente reducido es menor que la tolerancia de optimalidad *rtredg*.
- *Solución Óptima: El error en el valor de la función objetivo óptimo estimado del gradiente reducido y el Hessiano estimado es menor que la tolerancia mínima en el objetivo.* La solución es una solución interior interior localmente óptima. El componente más grande del gradiente reducido es mayor que la tolerancia de optimalidad *rtredg*. Sin embargo, cuando el gradiente reducido es escalado con información del Hessiano estimado de la función objetivo reducida la solución parece óptima. El objetivo debe ser grande o el objetivo reducido debe tener segundas derivadas grandes por eso es aconsejable escalar el modelo.
- *Solución Óptima: Convergencia demasiado lenta. El cambio en el objetivo ha sido menor que xx.xx para xx iteraciones consecutivas:* CONOPT2 se detiene con una solución que parece óptima. El proceso de solución es detenido porque el progreso es

lento. En esta solución el modelo puede tener derivadas grandes por eso es aconsejable escalar este.

- *Solución Factible: La convergencia es demasiado lenta. El cambio en la función objetivo ha sido menor que xx.xx para xx iteraciones consecutivas:* La función no ha mostrado cambios significativos después de que se han ejecutado cierto número de iteraciones.
- *Solución Factible: Las tolerancias son mínimas y no existe un cambio en la función objetivo aunque el gradiente reducido es mayor que la tolerancia.* CONOPT2 para con una solución factible. Los criterios de optimalidad no han sido satisfechos.
- *Solución no acotada. Una variable ha alcanzado "infinito". El valor más grande permitido (rtmaxv) es xx.xx.* El mensaje anterior indica que algunas variable llegan a ser muy grandes antes de que una solución factible a sido encontrada.

Existen algunos otros mensajes enviados por GAMS. Éstos pueden ser vistos en los manuales de GAMS.

### **El Algoritmo CONOPT**

Este método está basado en el algoritmo GRG (Gradiente Reducido Generalizado), sugerido originalmente por Abadie y Carpentier (1969). La implementación actual tiene muchas modificaciones, que hacen a este algoritmo eficiente para modelos grandes y modelos escritos en el lenguaje modelador GAMS. Los detalles sobre el algoritmo pueden ser encontrados en Drud (1985 y 1992). Enseguida se presentan los pasos generales del algoritmo GRG.

### Algoritmo GRG

1. Inicializar y encontrar una solución factible inicial.
2. Calcular el Jacobiano  $J$  de las restricciones.
3. Seleccionar un conjunto de “n” variables básicas,  $x_b$ , tal que  $B$ , la submatriz de la columna básica de  $J$ , sea no singular. Factorizar  $B$ . Las variables restantes,  $x_n$ , son llamadas variables no básicas.
4. Resolver  $B^T u = df/dx_b$  para los multiplicadores  $u$ .
5. Calcular el gradiente reducido,  $r = df/dx - J^T u$ .  $r$  será por definición cero para las variables básicas.
6. Si  $r$  proyectado en las cotas es pequeño, entonces el algoritmo se detiene. El punto actual es cercano al óptimo.
7. Seleccionar un conjunto de variables super básicas,  $x_s$ , como un subconjunto de las variables no básicas que útilmente pueden ser cambiadas, y encontrar una dirección de búsqueda,  $d_s$ , para las variables super básicas basadas en  $r_s$  y posiblemente sobre cierta información de segundo orden.
8. Desarrollar una línea de búsqueda a lo largo de la dirección  $d$ . Para cada paso,  $x_s$  es cambiada en la dirección de  $d_s$  y  $x_b$  es subsecuentemente ajustada a satisfacer  $g(x_b, x_s) = b$  en un proceso pseudo-Newton usando el factor  $B$  del paso 3.
9. Ir al paso 2

El algoritmo termina cuando la dirección de búsqueda ( $d$ ) es igual a cero y la solución ( $\mathbf{x}$ ) cumple con las condiciones de optimalidad de Kuhn-Tucker. Una información más detallada de los fundamentos de GRG y GAMS/CONOPT puede encontrarse en Drud, (1985 y 1992).

**APÉNDICE D**  
**TABLAS DE RESULTADOS**  
**Y**  
**MODELOS EN GAMS**

## Apéndice D. Tablas de Resultados y Modelos en GAMS.

### D.1 Tablas de Resultados

En este apéndice se muestran las tablas completas de los resultados de la evaluación computacional de las técnicas de pre-procesamiento. Las primeras Tablas (13, 14 y 15) muestran los resultados cuando no se le aplica las técnicas de pre-procesamiento al algoritmo, a cada una de las topologías utilizadas. Las demás tablas en cada topología muestran tres escenarios diferentes. En el primer escenario (E1) se evaluaron las siguientes técnicas de pre-procesamiento: acotamiento a las variables de decisión y asignación de un punto inicial (al punto medio de las cotas de cada variable). En el segundo escenario (E2), en adición a proporcionar cotas a las variables y proporcionarle un punto inicial al algoritmo, se le aplicó escalamiento a una restricción del modelo (eficiencia del compresor). Por último, en el tercer escenario (E3), además de aplicar todo lo del escenario (E2) se le aplicó escalamiento a otra restricción del modelo (dominio del compresor). Al final de este Apéndice D se encuentran algunos ejemplos de modelos en GAMS. Al final de cada ejemplo puede verse como fueron aplicadas estas técnicas.

Como se había mencionado anteriormente cada compresor se probó para diferentes valores de flujo. Cada tabla muestra el tipo de solución encontrada por el algoritmo, registrando, en aquellas instancias en donde encuentra soluciones óptimas, el número de iteraciones que utilizó el algoritmo para encontrar esa solución. En cada fila se muestran los resultados para cada compresor

El significado del estado final del algoritmo en este problema se describe a continuación.

- **So: Solución óptima.** El algoritmo puede terminar con dos tipos diferentes de soluciones óptimas locales. Estas pueden ser un punto extremo o un punto interior de la región factible.

- **Sftm:** *Solución factible “tolerancia mínima”*. Aquí el algoritmo termina porque la tolerancia entre la solución anterior y la actual es mínima y no hay un progreso significativo en la solución.
- **Sfcl:** *Solución factible “convergencia demasiado lenta”*. Aquí el algoritmo termina porque el proceso de solución es muy lento, es decir, que la función objetivo no ha mostrado cambios significativos en cada iteración. Después de que se han ejecutado un número de iteraciones que tiene por default el algoritmo y no hay un cambio significativo en la solución el algoritmo termina con este tipo de solución.
- **Snf:** *Solución no factible*. Ésta solución puede aparecer por varias razones. Una de ellas es cuando el algoritmo no puede avanzar hacia la región de factibilidad, es decir, se estanca dentro de una región infactible. Otra posible causa es que alguna de las variables alcance un valor infinito el cual no está permitido ya que el algoritmo tiene un parámetro del cual no deben de pasar las variables.
- **Lia:** *Límite de iteraciones alcanzado*. Este mensaje aparece cuando el algoritmo alcanza el número de iteraciones que tiene por default el algoritmo (10,000 iteraciones) sin encontrar alguna solución.

	Valores de Flujo												
Compresor	100	150	200	250	300	350	400	450	500	550	600	650	700
Snarlin-k1							Snf						
Rakeey-k1								Snf	Snf	Snf	Snf	Snf	Snf
Rakeey-k2			Snf										
Hamper-k1			Snf										
Bellvan-k1			Snf										
Bellvan-k2	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf			
Bellvan-k3	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf			
Bethany-k1		Snf											
Bethany-k2	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf

	Valores de Flujo								
Compresor	750	800	850	900	950	1000	1050	1100	
Snarlin-k1	Snf	Snf	Snf	Snf	Snf				
Rakeey-k1	Snf	Snf	Snf	Snf					
Rakeey-k2	Snf	Snf	Snf	Snf	Snf	Snf			
Hamper-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	
Bellvan-k1									
Bellvan-k2									
Bellvan-k3	Snf	Snf	Snf	Snf	Snf				
Bethany-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf		
Bethany-k2	Snf	Snf	Snf	Snf					

Tabla 14. Resultados sin aplicación de técnicas de pre-procesamiento para la red en línea

	Valores de Flujo											
Compresor	300	400	500	600	700	800	900	1000	1100	1200	1300	1400
Snarlin-k1						Snf	Snf	Snf	Snf	Snf	Snf	Snf
Rakeey-k1								Snf	Snf	Snf	Snf	Snf
Rakeey-k2			Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Hamper-k1				Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Bellvan-k1			Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	
Bellvan-k2			Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	
Bellvan-k3	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Bethany-k1							Snf	Snf	Snf	Snf	Snf	Snf
Bethany-k2		Snf	Snf	Snf	Snf	Snf						

	Valores de Flujo											
Compresor	1500	1600	1700	1800	1900	2000	2100	2200	2300	2400	2500	
Snarlin-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	
Rakeey-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf		
Rakeey-k2	Snf	Snf	Snf	Snf	Snf	Snf						
Hamper-k1	Snf	Snf										
Bellvan-k1												
Bellvan-k2												
Bellvan-k3	Snf	Snf	Snf	Snf	Snf							
Bethany-k1	Snf	Snf	Snf	Snf	Snf	Snf						
Bethany-k2												

Tabla 15. Resultados sin aplicación de técnicas de pre-procesamiento para la red tipo árbol.

Compresor	Valores de Flujo											
	500	600	700	800	900	1000	1100	1200	1300	1400	1500	1600
Snarlin-k1								Snf	Snf	Snf	Snf	Snf
Rakeey-k1												Snf
Rakeey-k2						Snf						
Hamper-k1			Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Bellvan-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Bellvan-k2	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Bellvan-k3		Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Bethany-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Bethany-k2	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf

Compresor	Valores de Flujo											
	1700	1800	1900	2000	2100	2200	2300	2400	2500	2600	2700	
Snarlin-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Rakeey-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Rakeey-k2	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Hamper-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Bellvan-k1	Snf	Snf	Snf									
Bellvan-k2	Snf	Snf	Snf									
Bellvan-k3	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf
Bethany-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf				
Bethany-k2	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf				

Compresor	Valores de Flujo										
	2800	2900	3000	3100	3200	3300	3400	3500	3600	3700	
Snarlin-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf				
Rakeey-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf				
Rakeey-k2	Snf	Snf	Snf	Snf	Snf	Snf	Snf				
Hamper-k1	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	Snf	
Bellvan-k1											
Bellvan-k2											
Bellvan-k3											
Bethany-k1											
Bethany-k2											

Tabla 16. Resultados sin aplicación de técnicas de pre-procesamiento para la red con ciclo

Compresor	Valores de Flujo											
	100	150	200	250	300	350	400	450	500	550	600	650
Snarlin-k1 # iteraciones							So 38	So 35	So 35	So 36	So 37	So 37
Rakeey-k1 # iteraciones								So 36	So 34	So 37	So 38	So 41
Rakeey-k2 # iteraciones			So 84	So 53	So 75	Sftm	So 87	So 1922	So 70	So 162	So 99	So 84
Hamper-k1 # iteraciones			So 37	So 42	So 55	So 61	So 123	So 127	So 136	So 122	So 189	Sftm
Bellvan-k1 # iteraciones	So 21	So 24	So 43	So 34	So 28	So 28	So 51	So 36	So 34	So 32		
Bellvan-k2 # iteraciones	So 21	So 24	So 43	So 34	So 28	So 28	So 51	So 36	So 34	So 32		
Bellvan-k3 # iteraciones		So 33	So 32	So 70	Lia	Sftm	So 65	So 98	Sftm	Snf	Sfcl	So 48
Bethany-k1 # iteraciones			So 26	So 36	So 49	So 38	So 344	So 434	Sftm	So 108	So 167	So 189
Bethany-k2 # iteraciones		So 23	So 35	So 23	So 38	So 33	So 34	So 34	So 38	So 272	Snf	Lia

Compresor	Valores de Flujo									
	700	750	800	850	900	950	1000	1050	1100	
Snarlin-k1 # iteraciones	Snf	So 160	So 48	So 52	So 69	So 36				
Rakeey-k1 # iteraciones	So 40	So 55	So 38	So 46	So 41					
Rakeey-k2 # iteraciones	So 77	So 85	So 166	Sftm	Sftm	So 54	So 62			
Hamper-k1 # iteraciones	Sftm	Sftm	Sftm	Sftm	Lia	Sftm	Sftm	So 52	So 44	
Bellvan-k1 # iteraciones										
Bellvan-k2 # iteraciones										
Bellvan-k3 # iteraciones	So 46	So 46	So 47	So 40	So 41	So 25				
Bethany-k1 # iteraciones	So 351	Sftm	So 261	Sftm	Sftm	So 396	So 49	So 33		
Bethany-k2 # iteraciones	Sftm	Lia	So 40	So 39	So 41					

Tabla 17. Resultados para la red en línea bajo el escenario E1

Donde:

So: Solución óptima

Sftm: Solución factible "tolerancia mínima"

Sfcl: Solución factible "convergencia demasiado lenta"

Snf: Solución no factible

Lia: Límite de iteraciones alcanzado

Resumen de Tabla 17. Total de instancias: 129, total de iteraciones en las instancias con solución óptima: 9568, total de instancias con solución óptima que fueron tomadas para el promedio: 104, total de instancias con solución óptima: 105, promedio de iteraciones: 92

*\*Nota:* Para calcular éste promedio no fue considerada la instancia marcada por la flecha ya que para ésta instancia no se encontró una solución óptima en el escenario E2. Esto se hace con la finalidad de calcular el promedio con el mismo número de instancias en los dos escenarios (E1 y E2) para hacer la comparación de éstos en la prueba de hipótesis.

Compresor	Valores de Flujo											
	100	150	200	250	300	350	400	450	500	550	600	650
Snarlin-k1 # iteraciones							So 36	So 35	So 35	So 36	So 37	So 37
Rakeey-k1 # iteraciones								So 36	So 34	So 37	So 38	So 44
Rakeey-k2 # iteraciones			So 49	So 60	So 53	So 91	So 98	So 73	So 83	So 124	So 64	So 168
Hamper-k1 # iteraciones			So 39	So 42	So 53	So 59	So 162	So 131	So 277	So 122	So 160	So 149
Bellvan-k1 # iteraciones	So 19	So 24	So 40	So 33	So 27	So 26	So 34	So 36	So 34	So 32		
Bellvan-k2 # iteraciones	So 19	So 24	So 40	So 33	So 27	So 26	So 34	So 36	So 34	So 32		
Bellvan-k3 # iteraciones		So 34	So 32	So 70	Sftm	Sftm	So 196	So 86	Sftm	So 229	So 49	So 48
Bethany-k1 # iteraciones			So 46	So 36	So 44	Sftm	So 67	So 539	So 120	So 100	So 163	So 85
Bethany-k2 # iteraciones		So 40	So 35	So 23	So 38	So 33	So 33	So 34	So 38	So 190	Sfcl	So 480

Compresor	Valores de Flujo									
	700	750	800	850	900	950	1000	1050	1100	
Snarlin-k1 # iteraciones	So 77	So 98	So 69	So 50	So 73	So 36				
Rakeey-k1 # iteraciones	So 46	So 47	So 38	So 46	So 41					
Rakeey-k2 # iteraciones	So 74	So 74	So 183	So 72	So 88	So 57	So 60			
Hamper-k1 # iteraciones	So 632	Lia	So 185	So 351	Sftm	So 107	Sftm	So 52	So 44	
Bellvan-k1 # iteraciones										
Bellvan-k2 # iteraciones										
Bellvan-k3 # iteraciones	So 66	So 46	So 47	So 40	So 45	So 25				
Bethany-k1 # iteraciones	So 253	Sfcl	So 114	Sftm	Sftm	So 73	So 49	So 33		
Bethany-k2 # iteraciones	So 50	So 40	So 40	So 39	So 38					

Tabla 18. Resultados para la red en línea bajo el escenario E2

Resumen de Tabla 18. Total de instancias: 129, total de iteraciones en las instancias con solución óptima: 6987, total de instancias con solución óptima que fueron tomadas para el promedio: 104, total de instancias con solución óptima: 117, promedio de iteraciones: 67.2

*\*Nota:* Cuando se realiza la prueba de hipótesis para hacer la comparación de los escenarios (E1 y E2), (E2 y E3) se hace en referencia a las instancias con solución óptima que fueron encontradas en el escenario E1. Para calcular el promedio del escenario E2 sólo se toman en cuenta aquellas instancias con solución óptima que fueron encontradas en el escenario E1 (se excluyen las instancias señaladas en negrita) ya que éstas no fueron consideradas para calcular el promedio en ese escenario. Para la otra comparación (E2 y E3) se excluyen además de las instancias marcadas en negrita las que están señaladas por la flecha ya que éstas no se consideran para calcular el promedio en el escenario E3 porque no son instancias con soluciones óptimas locales.

Para la comparación de los escenarios E2 y E3 los resultados son los siguientes:

Total de iteraciones en las instancias con solución óptima: 6987, total de instancias con solución óptima que fueron tomadas para el promedio: 102, total de instancias con solución óptima: 118 promedio de iteraciones: 68.5

Compresor	Valores de Flujo											
	100	150	200	250	300	350	400	450	500	550	600	650
Snarlin-k1 # iteraciones							So 36	So 35	So 35	So 36	So 37	So 37
Rakeey-k1 # iteraciones								So 36	So 34	So 37	So 38	So 42
Rakeey-k2 # iteraciones			So 22	So 60	So 53	Sftm	So 69	So 105	So 83	So 294	So 231	So 398
Hamper-k1 # iteraciones			So 27	So 38	So 52	So 33	So 88	So 106	So 126	So 74	So 66	So 150
Bellvan-k1 # iteraciones	So 27	So 29	So 42	So 34	So 30	So 21	So 35	So 33	So 33	So 29		
Bellvan-k2 # iteraciones	So 27	So 29	So 42	So 34	So 30	So 21	So 35	So 33	So 33	So 29		
Bellvan-k3 # iteraciones		So 31	So 32	So 70	So 184	So 99	So 173	So 82	So 94	So 138	So 48	So 48
Bethany-k1 # iteraciones			So 37	So 34	So 45	Sftm	So 250	So 399	So 88	So 51	So 56	So 60
Bethany-k2 # iteraciones		So 31	So 25	So 23	So 31	So 33	So 33	So 34	So 38	So 240	Sfcl	So 372

Compresor	Valores de Flujo								
	700	750	800	850	900	950	1000	1050	1100
Snarlin-k1	So	So	So	So	So	So			
# iteraciones	<b>50</b>	39	51	37	37	36			
Rakeey-k1	So	So	So	So	So				
# iteraciones	40	90	37	72	37				
Rakeey-k2	So	So	So	So	So	So	So		
# iteraciones	93	79	113	<b>183</b>	<b>61</b>	69	71		
Hamper-k1	Sftm	Sftm	So	So	So	So	So	So	So
# iteraciones			<b>201</b>	<b>262</b>	<b>599</b>	<b>110</b>	<b>174</b>	<b>52</b>	<b>42</b>
Bellvan-k1									
# iteraciones									
Bellvan-k2									
# iteraciones									
Bellvan-k3	So	So	So	So	So	So			
# iteraciones	46	46	39	38	44	25			
Bethany-k1	Sftm	Sftm	So	Sftm	So	Sftm	So	So	
# iteraciones			104		<b>250</b>		49	33	
Bethany-k2	So	So	So	So	So				
# iteraciones	<b>50</b>	<b>40</b>	36	39	35				

Tabla 19. Resultados para la red en línea bajo el escenario E3.

Resumen de Tabla 19. Total de iteraciones en las instancias con solución óptima: 6489, total de instancias con solución óptima que fueron tomadas para el promedio: 102, total de instancias con solución óptima: 120, promedio de iteraciones: 63.6

*\*Nota:* Para calcular este promedio se excluyen las instancias marcadas en negrita ya que tomamos como referencia para las comparaciones las instancias que fueron encontradas con solución óptima en el escenario E1.

Compresor	Valores de Flujo											
	300	400	500	600	700	800	900	1000	1100	1200	1300	1400
Snarlin-k1						Sftm	So	So	So	So	So	So
# iteraciones							50	54	96	57	70	52
Rakeey-k1								So	So	So	So	So
# iteraciones								57	52	47	45	50
Rakeey-k2			So	So	So	So	So	Sftm	Sftm	Sftm	Sftm	Sftm
# iteraciones			54	133	268	83	91					
Hamper-k1				Sftm	Sftm	Sftm	So	Sftm	Sftm	Sftm	Sftm	Sftm
# iteraciones							140					
Bellvan-k1			So	So	So	So	So	So	So	So	So	
# iteraciones			39	37	37	31	27	51	60	28	52	
Bellvan-k2			So	So	So	So	So	So	So	So	So	
# iteraciones			39	37	37	31	27	51	60	28	52	
Bellvan-k3	So	So	So	So	So	So	So	So	So	Sftm	Sftm	Sftm
# iteraciones	35	49	39	42	45	40	45	36	173			
Bethany-k1							So	So	So	So	So	Sftm
# iteraciones							51	78	282	74	267	
Bethany-k2		So	So	So	So	So						
# iteraciones		38	53	50	42	41						

Compresor	Valores de Flujo										
	1500	1600	1700	1800	1900	2000	2100	2200	2300	2400	2500
Snarlin-k1	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	53	59	49	57	54	56	54	56	74	53	100
Rakeey-k1	So	So	So	So	So	So	So	Sftm	So	So	
# iteraciones	47	51	60	49	65	40	65		58	65	
Rakeey-k2	Sftm	Sftm	Sftm	Sfcl	So	So					
# iteraciones					103	111					
Hamper-k1	Sftm	Sftm									
# iteraciones											
Bellvan-k1											
# iteraciones											
Bellvan-k2											
# iteraciones											
Bellvan-k3	Sftm	Sftm	Sftm	Sftm	Sftm						
# iteraciones											
Bethany-k1	Sftm	Sftm	So	So	So	Sftm					
# iteraciones			365	116	64						
Bethany-k2											
# iteraciones											

Tabla 20. Resultados para la red tipo árbol bajo el escenario E1.

Resumen de Tabla 20. Total de instancias: 112, total de iteraciones en las instancias con solución óptima: 5527, total de instancias con solución óptima que fueron tomadas para el promedio: 78, total de instancias con solución óptima: 79, promedio de iteraciones: 70.9

*\*Nota:* Para calcular éste promedio no fue considerada la instancia marcada por la flecha ya que para ésta instancia no se encontró una solución óptima en el escenario E2. Esto se hace con la finalidad de calcular el promedio con el mismo número de instancias en los dos escenarios (E1 y E2) para hacer la comparación de éstos en la prueba de hipótesis.

Compresor	Valores de Flujo											
	300	400	500	600	700	800	900	1000	1100	1200	1300	1400
Snarlin-k1						Sftm	So	So	So	So	So	So
# iteraciones							50	51	82	57	70	52
Rakeey-k1								So	So	So	So	So
# iteraciones								57	52	46	45	50
Rakeey-k2			So	So	So	So	So	So	Sftm	Sftm	Sftm	Sftm
# iteraciones			54	374	226	85	76	82				
Hamper-k1				Sftm	Sftm	Sftm	So	Sftm	Sftm	Sftm	Sftm	Sftm
# iteraciones							489					
Bellvan-k1			So	So	So	So	So	So	So	So	So	
# iteraciones			39	30	34	31	26	37	39	28	41	
Bellvan-k2			So	So	So	So	So	So	So	So	So	
# iteraciones			39	30	34	31	26	37	39	28	41	
Bellvan-k3	So	So	So	So	So	So	So	So	So	So	Sftm	Sftm
# iteraciones	35	49	39	42	45	40	45	36	196	599		
Bethany-k1							So	So	So	So	So	Sftm
# iteraciones							51	78	335	77	123	
Bethany-k2		So	So	So	So	So						
# iteraciones		45	55	54	42	51						

Compresor	Valores de Flujo										
	1500	1600	1700	1800	1900	2000	2100	2200	2300	2400	2500
Snarlin-k1	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	53	59	52	54	74	56	54	56	47	53	94
Rakeey-k1	So	So	So	So	So	So	So	So	So	So	
# iteraciones	47	51	60	49	59	40	70	<b>60</b>	58	61	
Rakeey-k2	Sftm	So	Sftm	Sfcl	So	Sftm					
# iteraciones		93			100						
Hamper-k1	Sftm	Sftm									
# iteraciones											
Bellvan-k1											
# iteraciones											
Bellvan-k2											
# iteraciones											
Bellvan-k3	Sftm	So	So	So	So						
# iteraciones		<b>119</b>	<b>80</b>	<b>105</b>	<b>563</b>						
Bethany-k1	So	Sftm	So	So	So	Sftm					
# iteraciones	<b>105</b>		665	157	64						
Bethany-k2											
# iteraciones											

Tabla 21. Resultados para la red tipo árbol bajo el escenario E2.

Resumen de Tabla 21. Total de instancias: 112, total de iteraciones en las instancias con solución óptima: 6097, total de instancias con solución óptima que fueron tomadas para el promedio: 78, total de instancias con solución óptima: 87, promedio de iteraciones: 78.2

*\*Nota:* Cuando se realiza la prueba de hipótesis para hacer la comparación de los escenarios (E1 y E2) y (E2 y E3) se hace en referencia a las instancias con solución óptima que fueron encontradas en el escenario E1. Es decir, para calcular el promedio del escenario E2 para la primera comparación sólo se toman en cuenta aquellas instancias con solución óptima que fueron encontradas en el escenario E1 (se excluyen las instancias señaladas en negrita) ya que éstas no fueron consideradas para calcular el promedio en ese escenario. Para la otra comparación (E2 y E3) para calcular el promedio se excluyen las mismas instancias marcadas en negrita ya que para el escenario E3 no hubo otras instancias que no estuvieran en el escenario E1 con solución óptima.

Compresor	Valores de Flujo											
	300	400	500	600	700	800	900	1000	1100	1200	1300	1400
Snarlin-k1 # iteraciones						So 69	So 49	So 44	So 44	So 52	So 46	So 43
Rakeey-k1 # iteraciones								So 48	So 49	So 46	So 44	So 51
Rakeey-k2 # iteraciones			So 54	So 58	So 62	So 76	So 78	So 86	Sftm	Sftm	Sftm	Sftm
Hamper-k1 # iteraciones				Sftm	Sftm	So 59	So 521	Sftm	Sftm	Sftm	Sftm	Sftm
Bellvan-k1 # iteraciones			So 43	So 22	So 22	So 33	So 26	So 31	So 37	So 28	So 28	
Bellvan-k2 # iteraciones			So 43	So 22	So 22	So 33	So 26	So 31	So 37	So 28	So 28	
Bellvan-k3 # iteraciones	So 35	So 49	So 39	So 42	So 45	So 40	So 37	So 28	So 134	So 61	Sftm	Sftm
Bethany-k1 # iteraciones							So 51	So 78	So 306	So 74	So 723	Sftm
Bethany-k2 # iteraciones		So 45	So 48	So 51	So 42	So 85						

Compresor	Valores de Flujo											
	1500	1600	1700	1800	1900	2000	2100	2200	2300	2400	2500	
Snarlin-k1 # iteraciones	So 51	So 50	So 55	So 55	So 56	So 56	So 54	So 56	So 47	So 53	So 47	
Rakeey-k1 # iteraciones	So 44	So 45	So 60	So 47	So 59	So 40	So 57	So 44	So 54	So 40		
Rakeey-k2 # iteraciones	Sftm	So 117	Sftm	Sftm	So 91	Sftm						
Hamper-k1 # iteraciones	So 85	Sftm										
Bellvan-k1 # iteraciones												
Bellvan-k2 # iteraciones												
Bellvan-k3 # iteraciones	Sftm	So 84	So 97	So 97	So 114							
Bethany-k1 # iteraciones	Sftm	Sftm	So 137	So 92	So 63	So 80						
Bethany-k2 # iteraciones												

Tabla 22. Resultados para la red tipo árbol bajo el escenario E3.

Resumen de Tabla 22. Total de instancias: 112, total de iteraciones en las instancias con solución óptima: 5566, total de instancias con solución óptima que fueron tomadas para el promedio: 78, total de instancias con solución óptima: 90, promedio de iteraciones: 67.5

*\*Nota:* Para calcular este promedio se excluyen las instancias marcadas en negrita ya que tomamos como referencia para la comparación (E2 y E3) las instancias que fueron encontradas con solución óptima en el escenario E1.

Compresor	Valores de Flujo											
	500	600	700	800	900	1000	1100	1200	1300	1400	1500	1600
Snarlin-k1 # iteraciones								Snf	So 142	So 128	Sftm	So 125
Rakeey-k1 # iteraciones												So 168
Rakeey-k2 # iteraciones						Sftm	So 62	So 89	So 114	So 173	So 203	So 141
Hamper-k1 # iteraciones			So 67	So 52	So 129	Sftm 114	So 85	So 181	So 101	So 86	Sftm	Sftm
Bellvan-k1 # iteraciones	So 137	So 141	So 119	So 155	So 100	So 48	So 48	So 50	So 70	So 77	So 58	So 48
Bellvan-k2 # iteraciones	So 137	So 141	So 119	So 155	So 100	So 48	So 48	So 50	So 70	So 77	So 58	So 48
Bellvan-k3 # iteraciones		So 60	So 89	So 46	So 48	So 90	So 70	So 48	So 143	So 58	So 60	So 65
Bethany-k1 # iteraciones	So 63	So 75	So 81	So 72	So 97	So 68	So 132	Sftm	So 76	So 67	Sftm	So 61
Bethany-k2 # iteraciones	So 52	So 41	So 67	So 56	So 81	So 82	So 83	So 82	So 70	So 157	So 65	So 104

Compresor	Valores de Flujo											
	1700	180	1900	2000	2100	2200	2300	2400	2500	2600	2700	
Snarlin-k1 # iteraciones	So 128	So 41	So 51	So 45	So 50	So 43	So 44	So 51	So 135	So 46	So 47	
Rakeey-k1 # iteraciones	Sftm	So 39	So 50	So 42	So 45	So 52	So 45	So 51	So 72	So 45	So 49	
Rakeey-k2 # iteraciones	Sftm	Sftm	So 75	So 183	So 93	So 281	So 135	So 140	So 105	So 78	So 102	
Hamper-k1 # iteraciones	So 77	Sftm	Sftm	So 81	Sftm	Sftm	Sftm	Sftm	Sftm	Sftm	Sftm	
Bellvan-k1 # iteraciones	So 50	So 46	So 49									
Bellvan-k2 # iteraciones	So 50	So 46	So 49									
Bellvan-k3 # iteraciones	So 93	So 82	So 60	So 66	So 55	So 83	So 94	So 99	So 58	So 51	So 70	
Bethany-k1 # iteraciones	So 127	So 80	So 71	So 111	So 58	So 123	So 46	So 39				
Bethany-k2 # iteraciones	So 107	So 99	So 101	So 96	So 81	So 74	So 98					

Compresor	Valores de Flujo									
	2800	2900	3000	3100	3200	3300	3400	3500	3600	3700
Snarlin-k1 # iteraciones	So 140	So 130	So 53	So 77	So 76	So 54	So 46			
Rakeey-k1 # iteraciones	So 41	So 155	So 49	So 45	So 124	So 46	So 49			
Rakeey-k2 # iteraciones	Sftm	Sftm	So 93	So 151	So 104	So 93	So 109			
Hamper-k1 # iteraciones	Sftm	Sftm	Sftm	Sftm	Sftm	So 71	So 72	So 73	So 90	So 89
Bellvan-k1 # iteraciones										
Bellvan-k2 # iteraciones										
Bellvan-k3 # iteraciones										
Bethany-k1 # iteraciones										
Bethany-k2 # iteraciones										

Tabla 23. Resultados para la red con ciclo bajo el escenario E1.

Resumen de Tabla 23. Total de instancias: 188, total de iteraciones en las instancias con solución óptima: 13241, total de instancias con solución óptima que fueron tomadas para el promedio: 158, total de instancias con solución óptima: 163, promedio de iteraciones: 83.8

*\*Nota:* Para calcular este promedio se excluyen las instancias marcadas en negrita ya que éstas no fueron encontradas en el escenario E2 como soluciones óptimas y por lo tanto no fueron consideradas para calcular el promedio en ese escenario.

Compresor	Valores de Flujo											
	500	600	700	800	900	1000	1100	1200	1300	1400	1500	1600
Snarlin-k1 # iteraciones								So 77	So 142	So 126	So 64	So 57
Rakeey-k1 # iteraciones												So 167
Rakeey-k2 # iteraciones						So 217	So 60	So 90	So 66	So 101	So 146	So 86
Hamper-k1 # iteraciones			So 67	So 52	So 129	Sftm 114	So 74	So 74	So 101	So 86	Sftm	Sftm
Bellvan-k1 # iteraciones	So 133	So 59	So 57	So 50	So 100	So 48	So 53	So 50	So 61	So 75	So 58	So 48
Bellvan-k2 # iteraciones	So 133	So 59	So 57	So 50	So 100	So 48	So 53	So 50	So 61	So 75	So 58	So 48
Bellvan-k3 # iteraciones		So 70	So 84	So 46	So 48	So 91	So 70	So 95	So 76	So 61	So 60	So 106
Bethany-k1 # iteraciones	So 63	So 73	So 90	So 72	Sftm	So 81	So 173	Sftm	So 76	So 147	Sftm	So 60
Bethany-k2 # iteraciones	So 52	So 41	So 68	So 56	So 81	So 82	So 75	So 58	So 70	So 135	So 65	So 75

Compresor	Valores de Flujo										
	1700	180	1900	2000	2100	2200	2300	2400	2500	2600	2700
Snarlin-k1	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	73	48	48	46	49	43	44	161	87	46	47
Rakeey-k1	So	So	So	So	So	So	So	So	So	So	SSo
# iteraciones	76	45	47	42	44	41	40	40	41	46	49
Rakeey-k2	So	So	So	So	So	So	So	Sftm	So	So	So
# iteraciones	139	143	117	103	93	157	107		108	78	105
Hamper-k1	Sftm	Sftm	Sftm	So	Sftm						
# iteraciones				89							
Bellvan-k1	So	So	So								
# iteraciones	50	54	49								
Bellvan-k2	So	So	So								
# iteraciones	50	54	49								
Bellvan-k3	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	89	72	80	66	59	83	94	99	58	50	82
Bethany-k1	So	Sftm	So	So	So	So	So	So			
# iteraciones	110		73	88	58	75	46	39			
Bethany-k2	So	Sftm	So	So	So	So	So				
# iteraciones	102		95	86	70	69	69				

Compresor	Valores de Flujo									
	2800	2900	3000	3100	3200	3300	3400	3500	3600	3700
Snarlin-k1	So	So	So	So	So	So	So			
# iteraciones	163	138	53	77	75	67	46			
Rakeey-k1	So	So	So	So	So	So	So			
# iteraciones	42	67	49	45	115	56	46			
Rakeey-k2	So	Sftm	So	So	So	So	So			
# iteraciones	139		98	122	109	97	110			
Hamper-k1	Sftm	Sftm	Sftm	So						
# iteraciones				75	92	70	72	73	87	89
Bellvan-k1										
# iteraciones										
Bellvan-k2										
# iteraciones										
Bellvan-k3										
# iteraciones										
Bethany-k1										
# iteraciones										
Bethany-k2										
# iteraciones										

Tabla 24. Resultados para la red con ciclo bajo el escenario E2.

Resumen de Tabla 24. Total de instancias: 188, total de iteraciones en las instancias con solución óptima: 11990, total de instancias con solución óptima que fueron tomadas para el promedio: 158, total de instancias con solución óptima: 167, promedio de iteraciones: 75.9

*\*Nota:* Cuando se realiza la prueba de hipótesis para hacer la comparación de los escenarios (E1 y E2) y (E2 y E3) se hace en referencia a las instancias con solución óptima que fueron encontradas en el escenario E1. Es decir, para calcular el promedio

del escenario E2 para la primera comparación sólo se toman en cuenta aquellas instancias con solución óptima que fueron encontradas en el escenario E1 (se excluyen las instancias señaladas en negrita) ya que éstas no fueron consideradas para calcular el promedio en ese escenario. Para la otra comparación (E2 y E3) para calcular el promedio además de excluir las instancias en negrita se excluyen las instancias marcadas por la flecha ya que para el escenario E3 no se encontraron esas instancias con solución óptima y no fueron tomadas para e promedio.

Compresor	Valores de Flujo											
	500	600	700	800	900	1000	1100	1200	1300	1400	1500	1600
Snarlin-k1								So	So	So	So	So
# iteraciones								47	58	124	64	58
Rakeey-k1												So
# iteraciones												118
Rakeey-k2						So						
# iteraciones						200	60	89	73	102	129	86
Hamper-k1			So	So	So	Sftm	So	So	So	So	So	So
# iteraciones			63	50	129	107	79	73	98	78	158	208
Bellvan-k1	So	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	70	42	51	56	50	44	62	46	44	50	40	38
Bellvan-k2	So	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	70	42	51	56	50	44	62	46	44	50	40	38
Bellvan-k3		So	So	So	So	So	So	So	So	So	So	So
# iteraciones		62	59	46	48	74	70	113	87	58	57	104
Bethany-k1	So	So	So	So	Sftm	So	So	So	So	So	Sftm	So
# iteraciones	61	80	80	63		58	145	94	73	143		61
Bethany-k2	So	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	49	41	55	56	64	60	75	59	70	93	61	75

Compresor	Valores de Flujo											
	1700	180	1900	2000	2100	2200	2300	2400	2500	2600	2700	
Snarlin-k1	So	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	54	48	47	46	42	42	43	169	87	46	47	
Rakeey-k1	So	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	63	40	47	42	44	41	42	40	42	45	49	
Rakeey-k2	So	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	135	166	141	237	95	127	106	108	129	71	95	
Hamper-k1	So	Sftm	Sftm	Sftm	Sftm	Sftm	So	Sftm	Sftm	Sftm	So	
# iteraciones	78						185				120	
Bellvan-k1	So	So	So									
# iteraciones	41	37	49									
Bellvan-k2	So	So	So									
# iteraciones	41	37	49									
Bellvan-k3	So	So	So	So	So	So	So	So	So	So	So	So
# iteraciones	80	72	58	66	55	60	63	65	58	47	79	
Bethany-k1	So	Sftm	So	So	So	So	So	So				
# iteraciones	74		73	95	58	60	46	39				
Bethany-k2	So	Sftm	So	So	So	So	So					
# iteraciones	101		77	50	70	69	59					

Compresor	Valores de Flujo									
	2800	2900	3000	3100	3200	3300	3400	3500	3600	3700
Snarlin-k1	So	So	So	So	So	So	So			
# iteraciones	110	92	53	62	73	54	88			
Rakeey-k1	So	So	So	So	So	So	So			
# iteraciones	41	67	49	45	91	46	46			
Rakeey-k2	So	So	So	So	So	So	So			
# iteraciones	<b>103</b>	<b>87</b>	<b>105</b>	101	106	130	110			
Hamper-k1	Sftm	So	So	Sftm	Sftm	So	So	So	So	So
# iteraciones		<b>71</b>	<b>101</b>			70	72	78	128	85
Bellvan-k1		↗	↗							
# iteraciones										
Bellvan-k2										
# iteraciones										
Bellvan-k3										
# iteraciones										
Bethany-k1										
# iteraciones										
Bethany-k2										
# iteraciones										

Tabla 25. Resultados para la red con ciclo bajo el escenario E3.

Resumen de Tabla 25. Total de instancias: 188, total de iteraciones en las instancias con solución óptima: 10972, total de instancias con solución óptima que fueron tomadas para el promedio: 157, total de instancias con solución óptima: 174, promedio de iteraciones: 69.9

**\*Nota:** Para calcular este promedio se hace referencia a las instancias con solución óptima encontradas en el escenario E1. Las instancias marcadas en negrita son las instancias con solución óptima encontradas en este escenario que pertenecen al escenario E1 y que son excluidas para calcular el promedio. Aparte de excluir las instancias en negritase excluyen las instancias marcadas con flecha ya que éstas no fueron encontradas en el escenario E2 como soluciones óptimas y por lo tanto no fueron consideradas para calcular el promedio en ese escenario.

## D.2 Modelos en GAMS

En esta sección mostramos algunos ejemplos de modelos en GAMS.

Todos los archivos de GAMS utilizados se encuentran disponibles en la pagina de internet del PISIS (<http://yalma.fime.uanl.mx/~pisis>)

### 1. Modelo para la red en línea, utilizando datos del compresor Snarlin-k1, aplicando las técnicas de acotamiento a las variables, punto inicial y escalamiento.

```
$Title Optimización inteligente de redes de Transporte de gas natural
$Inlinecom{ }
```

Sets

```
  i      nodos en la red / 1, 2, 3, 4, 5, 6 /
  ir(i)  nodos oferta   /1/
  id(i)  nodos demanda  /6/
```

```
  prop   propiedades del ducto /longitud, diametro, friccion, capacidad/
```

```
  alias(i,j);
```

Sets

```
  A(i,j)  arcos en la red /1.2, 2.3, 3.4, 4.5, 5.6/
  Ac(i,j) arcos compresores en la red /2.3, 4.5/
  Ap(i,j) arcos ductos en la red /1.2, 3.4, 5.6/;
```

Scalar {Estos escalares son usados para calcular las ecuaciones que ligán a las variables originales (H,Q,S) con (x<sub>ij</sub>, p<sub>i</sub>, p<sub>j</sub>)}

{Propiedades del gas}

```
  gas_K /133050/      {constante del ducto}
  gas_e /1.287/       {exponente isoentropico}
  gas_R /85.2/        {constante del gas en (lbf-ft/lbm-R)}
  gas_Z /0.95/        {factor de compresibilidad del gas}
  gas_Sg /0.6248/     {gravedad especifica}
  gas_T /519.67/     {temperatura promedio en la succion (^R)}
  fac /33.1944/      {factor de conversion de MMCS D a (lbm-min)}
```

Table datducto(i,j,prop) {Propiedades del ducto: longitud(mi), diametro(ft), friccion}

	longitud	diámetro	fricción	capacidad
1.2	50	3	0.0085	1200
3.4	50	3	0.0085	1200
5.6	50	3	0.0085	1200;

## Parameter

$qu(i,j)$  {limite superior del flujo volumetrico del compresor (i,j) unidades (ft<sup>3</sup>/min)}  
 $ql(i,j)$  {limite inferior del flujo volumetrico del compresor (i,j) unidades (ft<sup>3</sup>/min)}  
 $su(i,j)$  {limite superior de velocidad del compresor (i,j) unidades (ft/min)}  
 $sl(i,j)$  {limite inferior de velocidad del compresor (i,j) unidades (ft/min)}  
 $AH(i,j)$  {Coeficiente para la ecuacion del Dominio de operacion de la estación compresora (i,j)}  
 $BH(i,j)$  {Coeficiente para la ecuacion del Dominio de operacion de la estación compresora (i,j)}  
 $CH(i,j)$  {Coeficiente para la ecuacion del Dominio de operacion de la estación compresora (i,j)}  
 $DH(i,j)$  {Coeficiente para la ecuacion del Dominio de operacion de la estación compresora (i,j)}  
 $AE(i,j)$  {Coeficiente para la ecuacion de eta del compresor (i,j)}  
 $BE(i,j)$  {Coeficiente para la ecuacion de eta del compresor (i,j)}  
 $CE(i,j)$  {Coeficiente para la ecuacion de eta del compresor (i,j)}  
 $DE(i,j)$  {Coeficiente para la ecuacion de eta del compresor (i,j)}  
 $pu(i)$  {limite superior de presion en el nodo i (psia)}  
 $pl(i)$  {limite inferior de presion en el nodo i (psia)}  
 $rl(i,j)$  {valor minimo del cociente Q/S en la estacion (i,j)}  
 $ru(i,j)$  {valor maximo del cociente Q/S en la estacion (i,j)}  
 $diam(i,j)$  {conversion del diametro del ducto (i,j) de ft a pulgadas}  
 $cf(i,j)$  {conversion de la capacidad del flujo del gas en el ducto (i,j) en MMSCFD --> {lbm/min}}  
 $C$  {constante del gas}  
 $c1(i,j)$  {resistencia del ducto (i,j) con unidades de {psia<sup>2</sup>/MMSCFD<sup>2</sup>}  
 $r(i,j)$  {convercion la resistencia del ducto (i,j) de {psia<sup>2</sup>/MMSCFD<sup>2</sup> --> psia<sup>2</sup>/(lbm/min<sup>2</sup>)}  
 $ncomp(i,j)$  {numero de compresores usados en la estacion (i,j)}  
 $gas\_m$  {proporcion del radio especifico}  
 $gase$  {propiedades del gas}  
 $PHI\_SURGE$  {ecuacion para calcular el hmin}  
 $PHI\_STONE$  {ecuacion para calcular el hmax}  
 $hu(i,j)$  {limite superior de cabeza adiabatica del compresor (i,j) unidades (lbf-ft/lbm)}  
 $hl(i,j)$  {limite inferior de cabeza adiabatica del compresor (i,j) unidades (lbf-ft/lbm)};

$qu(i,j) = 32000;$   
 $ql(i,j) = 12000;$   
 $su(i,j) = 5250;$   
 $sl(i,j) = 3500;$   
 $AH(i,j) = 0.000294;$   
 $BH(i,j) = -0.000000914;$   
 $CH(i,j) = 0.00000112;$   
 $DH(i,j) = -0.000000674;$   
 $AE(i,j) = 126;$   
 $BE(i,j) = -45.7;$   
 $CE(i,j) = 14.2;$   
 $DE(i,j) = -1.33;$   
 $pu(i) = 800;$   
 $pl(i) = 200;$   
 $rl(i,j) = ql(i,j)/sl(i,j);$   
 $ru(i,j) = qu(i,j)/su(i,j);$   
 $diam(i,j) = datducto(i,j,'diametro')*12;$   
 $cf(i,j)\$Ap(i,j) = datducto(i,j,'capacidad')*fac;$   
 $C = gas\_K*gas\_Z*gas\_T*gas\_Sg;$   
 $c1(i,j)\$Ap(i,j) = C*((datducto(i,j,'longitud')*datducto(i,j,'friccion'))/(diam(i,j)**5));$   
 $r(i,j) = c1(i,j)/(fac**2);$   
 $ncomp(i,j)\$Ac(i,j) = 1;$   
 $gas\_m = (gas\_e - 1)/gas\_e;$   
 $gase = (gas\_Z*gas\_R*gas\_T)/gas\_m;$

$$\begin{aligned} \text{PHI\_SURGE}(i,j)\$Ac(i,j) &= \text{AH}(i,j) + \text{BH}(i,j)*\text{rl}(i,j) + \text{CH}(i,j)*(\text{rl}(i,j)**2) + \text{DH}(i,j)*(\text{rl}(i,j)**3); \\ \text{PHI\_STONE}(i,j)\$Ac(i,j) &= \text{AH}(i,j) + \text{BH}(i,j)*\text{ru}(i,j) + \text{CH}(i,j)*(\text{ru}(i,j)**2) + \text{DH}(i,j)*(\text{ru}(i,j)**3); \\ \text{hl}(i,j) &= (\text{sl}(i,j)**2)*\text{PHI\_STONE}(i,j); \\ \text{hu}(i,j) &= (\text{su}(i,j)**2)*\text{PHI\_SURGE}(i,j); \end{aligned}$$

## Parameter

$o(ir)$  {Cantidad de flujo de los nodos oferta (MMSCFD)}  
/1 400/

$d(id)$  {Cantidad de flujo de los nodos demanda (MMSCFD)}  
/6 400/;

{Conversion de unidades: MMSCFD --> lbm/min}

$o(ir) = o(ir)*\text{fac};$   
 $d(id) = d(id)*\text{fac};$

## Parameter

$b(i)$  {rhs de las ecuaciones de balance del flujo de masa (lbm/min)};  
 $b(i) = 0;$  {Inicializamos  $b(i)$ }  
 $b(ir) = o(ir);$  {Conjunto de  $b(i)$  para los nodos oferta}  
 $b(id) = -d(id);$  {Conjunto de  $b(i)$  para los nodos demanda}

display rl,ru,gas\_m,diam,C,gase,cf,c1,r,b,hl,hu;

## Variables

$Y$  {Funcion Objetivo Min la sum de los costos del combustible en todas las estaciones}  
 $x(i,j)$  {Flujo de masa en cada arco (i,j) en la red unidades(lbm/min)}  
 $p(i)$  {Presion en el nodo i unidades (lbf/in^2)}  
 $s(i,j)$  {Velocidad del flujo en la estacion (i,j) unidades (rpm)}  
 $q(i,j)$  {Flujo volumetrico en la estacion (i,j) unidades (ft^3/min)}  
 $h(i,j)$  {Cabeza adiabatica en la estacion (i,j) unidades (lbf\*ft/lbm)}  
 $g(i,j)$  {Costo de combustible en la estacion (i,j) unidades (lbf\*ft/min)}  
 $eta(i,j)$  {Eta eficiencia en la estacion (i,j)}  
 $ra(i,j)$  {razon del cociente Q/S};

Positive Variable x,p,s,q,h,g,eta,ra;

## Equations

costtotal {Function Objetivo}

cost(i,j) {Costo en la estacion (i,j)}

balance(i)        {Balance de flujo en cada nodo i}  
 flujogas(i,j)    {Ecuacion de flujo de gas en cada ducto (i,j)}  
 cabeza(i,j)      {Cabeza adiabatica H en la estacion (i,j)}  
 flvolum(i,j)     {Flujo volumetrico Q en cada estacion (i,j)}  
 dominio(i,j)     {Dominio de la estacion compresora (i,j)}  
 eficiencia(i,j)   {Eficiencia de la estacion compresora (i,j)}  
 razon(i,j)        {Razon del cociente Q/S}  
 presion(i,j)      {restriccion de la presion de succion y la presion de descarga};

costtotal..        Y =e= sum((i,j)\$Ac(i,j), g(i,j));  
 cost(i,j)\$Ac(i,j).. g(i,j)\*eta(i,j) =e= 1 \* (x(i,j)\*h(i,j));  
 balance(i)..        sum(j\$A(i,j), x(i,j)) - sum(j\$A(j,i), x(j,i)) =e= b(i);  
 flujogas(i,j)\$Ap(i,j).. (p(i)\*\*2 - p(j)\*\*2) =e= r(i,j) \* ((x(i,j))\*\*2);  
 cabeza(i,j)\$Ac(i,j).. p(j)\*\*gas\_m =e= p(i)\*\*gas\_m\*((1/gase)\*(h(i,j))+1);  
 flvolum(i,j)\$Ac(i,j).. ncomp(i,j)\*144\*p(i)\*q(i,j) =e= (gas\_Z\*gas\_R\*gas\_T)\*x(i,j);  
 dominio(i,j)\$Ac(i,j).. h(i,j) =e=  
 (AH(i,j) + BH(i,j)\*ra(i,j) + CH(i,j)\*(ra(i,j)\*\*2) + DH(i,j)\*(ra(i,j)\*\*3)) \* (s(i,j)\*\*2);  
 eficiencia(i,j)\$Ac(i,j).. eta(i,j) =e=  
 AE(i,j) + BE(i,j)\*ra(i,j) + CE(i,j)\*(ra(i,j)\*\*2) + DE(i,j)\*(ra(i,j)\*\*3);  
 razon(i,j)\$Ac(i,j).. s(i,j)\*ra(i,j) =e= q(i,j);  
 presión(i,j)\$Ac(i,j).. p(i) =l= p(j);

Model mincosto /all /;

{ Los limites de presion dados en psia (lbf/in^2) }

p.up(i) = pu(i);  
 p.lo(i) = pl(i);  
 p.l(i) = (p.up(i) + p.lo(i)) / 2.0 ;

{ Limites de operacion en cada compresor (ft^3/min) }

q.up(i,j) = qu(i,j);  
 q.lo(i,j) = ql(i,j);  
 q.l(i,j) = (q.up(i,j) + q.lo(i,j)) / 2.0 ;

{ Limites de operacion en cada compresor (ft^3/min) }

h.up(i,j) = hu(i,j);  
 h.lo(i,j) = hl(i,j);  
 h.l(i,j) = (h.up(i,j) + h.lo(i,j)) / 2.0 ;

```

{ Limites de la razon del flujo volumetrico y velocidad}
ra.up(i,j) = ru(i,j);
ra.lo(i,j) = rl(i,j);
ra.l(i,j) = (ra.lo(i,j) + ra.up(i,j)) / 2.0 ;

{ Capacidad de flujo de gas en el ducto(i,j) y en la estacion (i,j) (lbm/min) }
x.up(i,j)$Ap(i,j) = cf(i,j);
x.l(i,j)$Ap(i,j) = (x.lo(i,j) + x.up(i,j)) / 2.0 ;
x.up(i,j)$Ac(i,j) = (ncomp(i,j)*qu(i,j) * p.up(i)* 144)/(gas_Z * gas_R * gas_T);
x.lo(i,j)$Ac(i,j) = (ql(i,j) * p.lo(i)* 144)/(gas_Z * gas_R * gas_T);
x.l(i,j)$Ac(i,j) = (x.lo(i,j) + x.up(i,j)) / 2.0;

{ Limites de velocidad en cada compresor (rpm)}
s.up(i,j) = su(i,j);
s.lo(i,j) = sl(i,j);
s.l(i,j) = (s.lo(i,j) + s.up(i,j)) / 2.0 ;

{ Eficiencia del compresor (0-100 %)}
eta.up(i,j) = 100;
eta.lo(i,j) = 50;
eta.l(i,j) = (eta.lo(i,j) + eta.up(i,j)) / 2.0 ;

{ Costo de combustible del compresor(i,j)}
g.up(i,j)$Ac(i,j) = (x.up(i,j)*h.up(i,j))/eta.lo(i,j);
g.lo(i,j)$Ac(i,j) = (x.lo(i,j)*h.lo(i,j))/eta.up(i,j);
g.l(i,j) = (g.lo(i,j) + g.up(i,j))/2.0;

{ CONOPT, SNOPT, MINOS, MINOS5 }

Option nlp = CONOPT2;

mincosto.scaleopt = 1;

eficiencia.scale(i,j)$Ac(i,j) = 1.0e+01;

dominio.scale(i,j)$Ac(i,j) = 1.0e01;

Solve mincosto using nlp minimizing Y;

Display x.l, x.m, p.l, s.l, q.l, h.l, g.l, eta.l, ra.l;

```

## 2. Modelo para la red tipo árbol, utilizando datos del compresor Rakeey-k2, aplicando las técnicas de acotamiento a las variables y punto inicial.

```

$title Optimizacion inteligente de redes de Transporte de gas natural
$inlinecom{ }

```

Sets

```

i      nodos en la red /1*10/
ir(i)  nodos oferta /1/
id(i)  nodos demanda /5, 6, 7, 9, 10/
prop   propiedades del ducto /longitud, diametro, friccion, capacidad/
alias(i,j);

```

## Sets

A(i,j) arcos en la red /1.2, 2.3, 3.4, 3.8, 4.5, 5.6, 5.7, 8.9, 9.10/  
 Ac(i,j) arcos compresores en la red /1.2, 3.4, 3.8/  
 Ap(i,j) arcos ductos en la red /2.3, 4.5, 5.6, 5.7, 8.9, 9.10/;

Scalar {Estos escalares son usados para calcular las ecuaciones que ligan a las variables originales (H,Q,S) con (xij, pi, pj)}

## {Propiedades del gas}

gas\_K /133050/ {constante del ducto}  
 gas\_e /1.287/ {exponente isoentropico}  
 gas\_R /85.2/ {constante del gas en (lbf-ft/lbm-R)}  
 gas\_Z /0.95/ {factor de compresibilidad del gas}  
 gas\_Sg /0.6248/ {gravedad especifica}  
 gas\_T /519.67/ {temperatura promedio en la succion (^R)}  
 fac /33.1944/ {factor de conversion de MMSCFD a (lbm-min)}

Table datducto(i,j,prop) {Propiedades del ducto: longitud(mi), diametro(ft), friccion, capacidad(MMSCFD)}

	longitud	diámetro	fricción	capacidad
2.3	50	3	0.0085	3000
4.5	50	3	0.0085	3000
5.6	50	3	0.0085	3000
5.7	50	3	0.0085	3000
8.9	50	3	0.0085	3000
9.10	50	3	0.0085	3000;

## Parameter

qu(i,j) {limite superior del flujo volumetrico del compresor (i,j) unidades (ft<sup>3</sup>/min)}  
 ql(i,j) {limite inferior del flujo volumetrico del compresor (i,j) unidades (ft<sup>3</sup>/min)}  
 su(i,j) {limite superior de velocidad del compresor (i,j) unidades (ft/min)}  
 sl(i,j) {limite inferior de velocidad del compresor (i,j) unidades (ft/min)}  
 AH(i,j) {Coeficiente para la ecuacion del Dominio de operacion de la estacion compresora (i,j)}  
 BH(i,j) {Coeficiente para la ecuacion del Dominio de operacion de la estacion compresora (i,j)}  
 CH(i,j) {Coeficiente para la ecuacion del Dominio de operacion de la estacion compresora (i,j)}  
 DH(i,j) {Coeficiente para la ecuacion del Dominio de operacion de la estacion compresora (i,j)}  
 AE(i,j) {Coeficiente para la ecuacion de eta del compresor (i,j)}  
 BE(i,j) {Coeficiente para la ecuacion de eta del compresor (i,j)}  
 CE(i,j) {Coeficiente para la ecuacion de eta del compresor (i,j)}  
 DE(i,j) {Coeficiente para la ecuacion de eta del compresor (i,j)}  
 rl(i,j) {valor minimo del cociente Q/S en la estacion (i,j)}  
 ru(i,j) {valor maximo del cociente Q/S en la estacion (i,j)}  
 diam(i,j) {conversion del diametro del ducto (i,j) de ft a pulgadas}  
 cf(i,j) {conversion de la capacidad del flujo del gas en el ducto (i,j) en MMSCFD --> {lbm/min}}  
 C {constante del gas}  
 cl(i,j) {resistencia del ducto (i,j) con unidades de {psia<sup>2</sup>/MMSCFD<sup>2</sup>}}  
 r(i,j) {convercion la resistencia del ducto (i,j) de {psia<sup>2</sup>/MMSCFD<sup>2</sup> --> psia<sup>2</sup>/(lbm/min<sup>2</sup>)}}  
 ncomp(i,j) {numero de compresores usados en la estacion (i,j)}  
 gas\_m {proporcion del radio especifico}  
 gase {propiedades del gas}  
 PHI\_SURGE {ecuacion para calcular el hmin}  
 PHI\_STONE {ecuacion para calcular el hmax}  
 hu(i,j) {limite superior de cabeza adiabatica del compresor (i,j) unidades (lbf-ft/lbm)}  
 hl(i,j) {limite inferior de cabeza adiabatica del compresor (i,j) unidades (lbf-ft/lbm)};

```

qu(i,j) = 23000;
ql(i,j) = 7800;
su(i,j) = 5200;
sl(i,j) = 3500;
AH(i,j) = 0.00043364;
BH(i,j) = -0.000021123;
CH(i,j) = 0.000020026;
DH(i,j) = -0.0000053597;
AE(i,j) = 180.59;
BE(i,j) = -110.53;
CE(i,j) = 39.717;
DE(i,j) = -4.6045;
rl(i,j) = ql(i,j)/sl(i,j);
ru(i,j) = qu(i,j)/su(i,j);
diam(i,j) = datducto(i,j,'diametro')*12;
cf(i,j)$Ap(i,j) = datducto(i,j,'capacidad')*fac;
C = gas_K*gas_Z*gas_T*gas_Sg;
c1(i,j)$Ap(i,j) = C*((datducto(i,j,'longitud')*datducto(i,j,'friccion'))/(diam(i,j)**5));
r(i,j) = c1(i,j)/(fac**2);
ncomp(i,j)$Ac(i,j) = 1;
gas_m = (gas_e - 1)/gas_e;
gase = (gas_Z*gas_R*gas_T)/gas_m;
PHI_SURGE(i,j)$Ac(i,j) = AH(i,j) + BH(i,j)*rl(i,j) + CH(i,j)*(rl(i,j)**2) + DH(i,j)*(rl(i,j)**3);
PHI_STONE(i,j)$Ac(i,j) = AH(i,j) + BH(i,j)*ru(i,j) + CH(i,j)*(ru(i,j)**2) + DH(i,j)*(ru(i,j)**3);
hl(i,j) = (sl(i,j)**2)*PHI_STONE(i,j);
hu(i,j) = (su(i,j)**2)*PHI_SURGE(i,j);

```

## Parameter

o(ir) {Cantidad de flujo de los nodos oferta (MMSCFD)}  
/1 1500/

d(id) {Cantidad de flujo de los nodos demanda (MMSCFD)}  
/5 300  
6 300  
7 300  
9 300  
10 300/;

## Parameter

b(i) {rhs de las ecuaciones de balance del flujo de masa (lbm/min)};  
b(i) = 0; {Inicializamos b(i)}  
b(ir) = o(ir); {Conjunto de b(i) para los nodos oferta}  
b(id) = -d(id); {Conjunto de b(i) para los nodos demanda}

## Parameter

pu(i) {limite superior de presion en el nodo i (psia)}  
pl(i) {limite inferior de presion en el nodo i (psia)};

pu(i) = 1500;  
pl('1') = 200;  
pl('2') = 200;  
pl('3') = 250;  
pl('4') = 200;  
pl('5') = 250;  
pl('6') = 250;

$p1('7')$  = 250;  
 $p1('8')$  = 250;  
 $p1('9')$  = 250;  
 $p1('10')$  = 200;

display rl,ru,gas\_m,diam,C,gase,cf,c1,r,b,hl,hu;

#### Variables

$Y$  {Funcion Objetivo Mın la sum de los costos del combustible en todas las estaciones}  
 $x(i,j)$  {Flujo de masa en cada arco (i,j) en la red unidades(lbm/min)}  
 $p(i)$  {Presion en el nodo i unidades (lbf/in<sup>2</sup>)}  
 $s(i,j)$  {Velocidad del flujo en la estacion (i,j) unidades (rpm)}  
 $q(i,j)$  {Flujo volumetrico en la estacion (i,j) unidades (ft<sup>3</sup>/min)}  
 $h(i,j)$  {Cabeza adiabatica en la estacion (i,j) unidades (lbf\*ft/lbm)}  
 $g(i,j)$  {Costo de combustible en la estacion (i,j) unidades (lbf\*ft/min)}  
 $\eta(i,j)$  {Eta eficiencia en la estacion (i,j)}  
 $ra(i,j)$  {razon del cociente Q/S};

Positive Variable x,p,s,q,h,g,eta,ra;

#### Equations

$costtotal$  {Function Objetivo}  
 $cost(i,j)$  {Costo en la estacion (i,j)}  
 $balance(i)$  {Balance de flujo en cada nodo i}  
 $flujogas(i,j)$  {Ecuacion de flujo de gas en cada ducto (i,j)}  
 $cabeza(i,j)$  {Cabeza adiabatica H en la estacion (i,j)}  
 $flvolum(i,j)$  {Flujo volumetrico Q en cada estacion (i,j)}  
 $dominio(i,j)$  {Dominio de la estacion compresora (i,j)}  
 $eficiencia(i,j)$  {Eficiencia de la estacion compresora (i,j)}  
 $razon(i,j)$  {Razon del cociente Q/S};

$costtotal..$   $Y = e = \sum((i,j) \$Ac(i,j), g(i,j));$   
 $cost(i,j) \$Ac(i,j)..$   $g(i,j) * \eta(i,j) = e = 1 * (x(i,j) * h(i,j));$

```

balance(i)..      sum(j$A(i,j), x(i,j)) - sum(j$A(j,i), x(j,i)) =e= b(i);

flujogas(i,j)$Ap(i,j).. (p(i)**2 - p(j)**2) =e= r(i,j) * ((x(i,j))**2);

cabeza(i,j)$Ac(i,j)..  p(i)**gas_m =e= p(i)**gas_m*((1/gase)*(h(i,j))+1);

flvolum(i,j)$Ac(i,j)..  ncomp(i,j)*144*p(i)*q(i,j) =e= (gas_Z*gas_R*gas_T)*x(i,j);

dominio(i,j)$Ac(i,j)..  h(i,j) =e=
(AH(i,j) + BH(i,j)*ra(i,j) + CH(i,j)*(ra(i,j)**2) + DH(i,j)*(ra(i,j)**3)) * (s(i,j)**2);

eficiencia(i,j)$Ac(i,j).. eta(i,j) =e=
AE(i,j) + BE(i,j)*ra(i,j) + CE(i,j)*(ra(i,j)**2) + DE(i,j)*(ra(i,j)**3);

razon(i,j)$Ac(i,j)..    s(i,j)*ra(i,j) =e= q(i,j);

```

Model mincosto /all /;

{ Los limites de presion dados en psia (lbf/in<sup>2</sup>) }

```

p.up(i) = pu(i);
p.lo(i) = pl(i);
p.l(i) = (p.up(i) + p.lo(i)) / 2.0 ;

```

{ Limites de operacion en cada compresor (ft<sup>3</sup>/min) }

```

q.up(i,j) = qu(i,j);
q.lo(i,j) = ql(i,j);
q.l(i,j) = (q.up(i,j) + q.lo(i,j)) / 2.0 ;

```

{ Limites de operacion en cada compresor (ft<sup>3</sup>/min) }

```

h.up(i,j) = hu(i,j);
h.lo(i,j) = hl(i,j);
h.l(i,j) = (h.up(i,j) + h.lo(i,j)) / 2.0 ;

```

{ Limites de la razon del flujo volumetrico y velocidad }

```

ra.up(i,j) = ru(i,j);
ra.lo(i,j) = rl(i,j);
ra.l(i,j) = (ra.lo(i,j) + ra.up(i,j)) / 2.0 ;

```

{ Capacidad de flujo de gas en en ducto (lbm/min) }

```

x.up(i,j)$Ap(i,j) = cf(i,j);
x.l(i,j)$Ap(i,j) = (x.lo(i,j) + x.up(i,j)) / 2.0 ;
x.up(i,j)$Ac(i,j) = (ncomp(i,j)*qu(i,j) * p.up(i)* 144)/(gas_Z * gas_R * gas_T);
x.lo(i,j)$Ac(i,j) = (ql(i,j) * p.lo(i)* 144)/(gas_Z * gas_R * gas_T);
x.l(i,j)$Ac(i,j) = (x.lo(i,j) + x.up(i,j)) / 2.0;

```

{ Limites de velocidad en cada compresor (rpm) }

```

s.up(i,j) = su(i,j);
s.lo(i,j) = sl(i,j);
s.l(i,j) = (s.lo(i,j) + s.up(i,j)) / 2.0 ;

```

{ Eficiencia del compresor (0-100 %) }

```

eta.up(i,j) = 100 ;
eta.lo(i,j) = 50;
eta.l(i,j) = (eta.lo(i,j) + eta.up(i,j)) / 2.0 ;

```

```
{ Costo de combustible del compresor(i,j)
g.up(i,j)$Ac(i,j) = (x.up(i,j)*h.up(i,j))/eta.lo(i,j);
g.lo(i,j)$Ac(i,j) = (x.lo(i,j)*h.lo(i,j))/eta.up(i,j);
g.l(i,j) = (g.lo(i,j) + g.up(i,j))/2.0;
```

```
{ CONOPT, SNOPT, MINOS, MINOS5 }
```

```
Option nlp = CONOPT2;
```

```
mincosto.scaleopt = 1;
```

```
Solve mincosto using nlp minimizing Y;
```

```
Display x.l, x.m, p.l, s.l, q.l, h.l, g.l, eta.l, ra.l;
```

### 3. Modelo para la red con ciclo, utilizando datos del compresor Bellvan-k1, aplicando las técnicas de acotamiento a las variables, punto inicial, escalamiento de las ecuaciones de la eficiencia y del dominio.

```
$Title Optimizacion inteligente de redes de Transporte de gas natural
```

```
$Inlinecom{ }
```

```
Sets
```

```
  i      nodos en la red /1*8/
```

```
  ir(i)  nodos oferta /1/
```

```
  id(i)  nodos demanda /5, 8/
```

```
  prop   propiedades del ducto /longitud, diametro, friccion, capacidad/
```

```
  alias(i,j);
```

```
Sets
```

```
  A(i,j)  arcos en la red /1.2, 1.3, 2.6, 3.4, 4.5, 6.5, 6.7, 7.8/
```

```
  Ac(i,j) arcos compresores en la red /1.2, 3.4, 6.7/
```

```
  Ap(i,j) arcos ductos en la red /1.3, 2.6, 4.5, 6.5, 7.8/;
```

```
Scalar {Estos escalares son usados para calcular las ecuaciones que ligan a las variables
  originales (H,Q,S) con (x_ij, p_i, p_j)}
```

```
{Propiedades del gas}
```

```
gas_K /133050/ {constante del ducto}
```

```
gas_e /1.287/ {exponente isoentropico}
```

```
gas_R /85.2/ {constante del gas en (lbf-ft/lbm-°R)}
```

```
gas_Z /0.95/ {factor de compresibilidad del gas}
```

```
gas_Sg /0.6248/ {gravedad especifica}
```

```
gas_T /519.67/ {temperatura promedio en la succion (°R)}
```

```
fac /33.1944/ {factor de conversion de MMCS D a (lbm-min)}
```

Table datducto(i,j,prop) {Propiedades del ducto: longitud(mi), diametro(ft), friccion, capacidad (MMSCFD)}

	longitud	diametro	friccion	capacidad
1.3	40	3	0.0085	2500
2.6	40	3	0.0085	2500
4.5	40	3	0.0085	2500
6.5	40	3	0.0085	2500
7.8	40	3	0.0085	2500;

Parameter

qu(i,j)	{limite superior del flujo volumetrico del compresor (i,j) unidades (ft <sup>3</sup> /min)}
ql(i,j)	{limite inferior del flujo volumetrico del compresor (i,j) unidades (ft <sup>3</sup> /min)}
su(i,j)	{limite superior de velocidad del compresor (i,j) unidades (ft/min)}
sl(i,j)	{limite inferior de velocidad del compresor (i,j) unidades (ft/min)}
AH(i,j)	{Coeficiente para la ecuacion del Dominio de operacion de la estacion compresora (i,j)}
BH(i,j)	{Coeficiente para la ecuacion del Dominio de operacion de la estacion compresora (i,j)}
CH(i,j)	{Coeficiente para la ecuacion del Dominio de operacion de la estacion compresora (i,j)}
DH(i,j)	{Coeficiente para la ecuacion del Dominio de operacion de la estacion compresora (i,j)}
AE(i,j)	{Coeficiente para la ecuacion de eta del compresor (i,j)}
BE(i,j)	{Coeficiente para la ecuacion de eta del compresor (i,j)}
CE(i,j)	{Coeficiente para la ecuacion de eta del compresor (i,j)}
DE(i,j)	{Coeficiente para la ecuacion de eta del compresor (i,j)}
pu(i)	{limite superior de presion en el nodo i (psia)}
pl(i)	{limite inferior de presion en el nodo i (psia)}
rl(i,j)	{valor minimo del cociente Q/S en la estacion (i,j)}
ru(i,j)	{valor maximo del cociente Q/S en la estacion (i,j)}
diam(i,j)	{conversion del diametro del ducto (i,j) de ft a pulgadas}
cf(i,j)	{conversion de la capacidad del flujo del gas en el ducto (i,j) en MMSCFD --> {lbm/min}
C	{constante del gas}
c1(i,j)	{resistencia del ducto (i,j) con unidades de {psia <sup>2</sup> /MMSCFD <sup>2</sup> }
r(i,j)	{conversion la resistencia del ducto (i,j) de {psia <sup>2</sup> /MMSCFD <sup>2</sup> --> psia <sup>2</sup> /(lbm/min <sup>2</sup> )}
ncomp(i,j)	{numero de compresores usados en la estacion (i,j)}
gas_m	{proporcion del radio especifico}
gase	{propiedades del gas}
PHI_SURGE	{ecuacion para calcular el hmin}
PHI_STONE	{ecuacion para calcular el hmax}
hu(i,j)	{limite superior de cabeza adiabatica del compresor (i,j) unidades (lbf-ft/lbm)}
hl(i,j)	{limite inferior de cabeza adiabatica del compresor (i,j) unidades (lbf-ft/lbm)};

qu(i,j)	= 11100;
ql(i,j)	= 4200;
su(i,j)	= 6300;
sl(i,j)	= 4200;
AH(i,j)	= 0.00022289;
BH(i,j)	= -0.00026112;
CH(i,j)	= 0.00013082;
DH(i,j)	= -0.000044737;
AE(i,j)	= 81.094;
BE(i,j)	= -70.282;
CE(i,j)	= 106.5;
DE(i,j)	= -39.94;
pu(i)	= 1200;
pl(i)	= 300;

```

rl(i,j) = ql(i,j)/sl(i,j);
ru(i,j) = qu(i,j)/su(i,j);
diam(i,j) = datducto(i,j,'diametro')*12;
cf(i,j)$Ap(i,j) = datducto(i,j,'capacidad')*fac;
C = gas_K*gas_Z*gas_T*gas_Sg;
c1(i,j)$Ap(i,j) = C*((datducto(i,j,'longitud')*datducto(i,j,'friccion'))/(diam(i,j)**5));
r(i,j) = c1(i,j)/(fac**2);
ncomp(i,j)$Ac(i,j) = 1;
gas_m = (gas_e - 1)/gas_e;
gase = (gas_Z*gas_R*gas_T)/gas_m;
PHI_SURGE(i,j)$Ac(i,j) = AH(i,j) + BH(i,j)*rl(i,j) + CH(i,j)*(rl(i,j)**2) + DH(i,j)*(rl(i,j)**3);
PHI_STONE(i,j)$Ac(i,j) = AH(i,j) + BH(i,j)*ru(i,j) + CH(i,j)*(ru(i,j)**2) + DH(i,j)*(ru(i,j)**3);
hl(i,j) = (sl(i,j)**2)*PHI_STONE(i,j);
hu(i,j) = (su(i,j)**2)*PHI_SURGE(i,j);

```

## Parameter

```

o(ir)      {Cantidad de flujo de los nodos oferta (MMSCFD)}
/1 1400/

```

```

d(id)      {Cantidad de flujo de los nodos demanda (MMSCFD)}
/5 600
8 800/;

```

```
{Conversion de unidades: MMSCFD --> lbm/min}
```

```

o(ir) = o(ir)*fac;
d(id) = d(id)*fac;

```

## Parameter

```

b(i)      {rhs de las ecuaciones de balance del flujo de masa (lbm/min)};
b(i) = 0;  {Inicializamos b(i)}
b(ir) = o(ir);  {Conjunto de b(i) para los nodos oferta}
b(id) = -d(id); {Conjunto de b(i) para los nodos demanda}

```

```
display rl,ru,gas_m,diam,C,gase,cf,c1,r,b,hl,hu;
Variables
```

```

Y      {Funcion Objetivo Min la sum de los costos del combustible en todas las estaciones}
x(i,j) {Flujo de masa en cada arco (i,j) en la red unidades(lbm/min)}
p(i)   {Presion en el nodo i unidades (lbf/in^2)}
s(i,j) {Velocidad del flujo en la estacion (i,j) unidades (rpm)}
q(i,j) {Flujo volumetrico en la estacion (i,j) unidades (ft^3/min)}
h(i,j) {Cabeza adiabatica en la estacion (i,j) unidades (lbf*ft/lbm)}
g(i,j) {Costo de combustible en la estacion (i,j) unidades (lbf*ft/min)}
eta(i,j) {Eta eficiencia en la estacion (i,j)}
ra(i,j) {razon del cociente Q/S};

```

```
Positive Variable x,p,s,q,h,g,eta,ra;
```

## Equations

```

costtotal {Function Objetivo}
cost(i,j) {Costo en la estacion (i,j)}
balance(i) {Balance de flujo en cada nodo i}
flujogas(i,j) {Ecuacion de flujo de gas en cada ducto (i,j)}

```

cabeza(i,j)      {Cabeza adiabatica H en la estacion (i,j)}  
 flvolum(i,j)    {Flujo volumetrico Q en cada estacion (i,j)}  
 dominio(i,j)    {Dominio de la estacion compresora (i,j)}  
 eficiencia(i,j) {Eficiencia de la estacion compresora (i,j)}  
 razon(i,j)      {Razon del cociente Q/S};

costtotal..       $Y = e = \text{sum}((i,j)\$Ac(i,j), g(i,j));$   
 cost(i,j)\\$Ac(i,j)..  $g(i,j)*\text{eta}(i,j) = e = 1 * (x(i,j)*h(i,j));$   
 balance(i)..       $\text{sum}(j\$A(i,j), x(i,j)) - \text{sum}(j\$A(j,i), x(j,i)) = e = b(i);$   
 flujogas(i,j)\\$Ap(i,j)..  $(p(i)**2 - p(j)**2) = e = r(i,j) * ((x(i,j))**2);$   
 cabeza(i,j)\\$Ac(i,j)..  $p(j)**\text{gas}_m = e = p(i)**\text{gas}_m*((1/\text{gase})*(h(i,j))+1);$   
 flvolum(i,j)\\$Ac(i,j)..  $\text{ncomp}(i,j)*144*p(i)*q(i,j) = e = (\text{gas}_Z*\text{gas}_R*\text{gas}_T)*x(i,j);$   
 dominio(i,j)\\$Ac(i,j)..  $h(i,j) = e =$   
 $(AH(i,j) + BH(i,j)*ra(i,j) + CH(i,j)*(ra(i,j)**2) + DH(i,j)*(ra(i,j)**3)) * (s(i,j)**2);$   
 eficiencia(i,j)\\$Ac(i,j)..  $\text{eta}(i,j) = e =$   
 $AE(i,j) + BE(i,j)*ra(i,j) + CE(i,j)*(ra(i,j)**2) + DE(i,j)*(ra(i,j)**3);$   
 razon(i,j)\\$Ac(i,j)..  $s(i,j)*ra(i,j) = e = q(i,j);$

Model mincosto /all /;

{ Los limites de presion dados en psia (lbf/in<sup>2</sup>) }

p.up(i) = pu(i);  
 p.lo(i) = pl(i);  
 p.l(i) = (p.up(i) + p.lo(i)) / 2.0 ;

{ Limites de operacion en cada compresor (ft<sup>3</sup>/min) }

q.up(i,j) = qu(i,j);  
 q.lo(i,j) = ql(i,j);  
 q.l(i,j) = (q.up(i,j) + q.lo(i,j)) / 2.0 ;

{ Limites de operacion en cada compresor (ft<sup>3</sup>/min) }

h.up(i,j) = hu(i,j);  
 h.lo(i,j) = hl(i,j);  
 h.l(i,j) = (h.up(i,j) + h.lo(i,j)) / 2.0 ;

{ Limites de la razon del flujo volumetrico y velocidad }

ra.up(i,j) = ru(i,j);  
 ra.lo(i,j) = rl(i,j);  
 ra.l(i,j) = (ra.lo(i,j) + ra.up(i,j)) / 2.0 ;

{ Capacidad de flujo de gas en en ducto (lbm/min) }

x.up(i,j)\\$Ap(i,j) = cf(i,j);  
 x.l(i,j)\\$Ap(i,j) = (x.lo(i,j) + x.up(i,j)) / 2.0 ;  
 x.up(i,j)\\$Ac(i,j) = (ncomp(i,j)\*qu(i,j) \* p.up(i)\* 144)/(gas\_Z \* gas\_R \* gas\_T);  
 x.lo(i,j)\\$Ac(i,j) = (ql(i,j) \* p.lo(i)\* 144)/(gas\_Z \* gas\_R \* gas\_T);  
 x.l(i,j)\\$Ac(i,j) = (x.lo(i,j) + x.up(i,j)) / 2.0;

```

{ Limites de velocidad en cada compresor (rpm)}
s.up(i,j) = su(i,j);
s.lo(i,j) = sl(i,j);
s.l(i,j) = (s.lo(i,j) + s.up(i,j)) / 2.0 ;

{ Eficiencia del compresor (0-100 %)}
eta.up(i,j) = 100;
eta.lo(i,j) = 50;
eta.l(i,j) = (eta.lo(i,j) + eta.up(i,j)) / 2.0 ;

{ Costo de combustible del compresor(i,j)}
g.up(i,j)$Ac(i,j) = (x.up(i,j)*h.up(i,j))/eta.lo(i,j);
g.lo(i,j)$Ac(i,j) = (x.lo(i,j)*h.lo(i,j))/eta.up(i,j);
g.l(i,j) = (g.lo(i,j) + g.up(i,j))/2.0;

{ CONOPT, SNOPT, MINOS, MINOS5 }

Option nlp = CONOPT2;

mincosto.scaleopt = 1;

eficiencia.scale(i,j)$Ac(i,j) = 1.0e+01 ;

dominio.scale(i,j)$Ac(i,j) = 1.0e02;

Solve mincosto using nlp minimizing Y;

Display x.l, x.m, p.l, s.l, q.l, h.l, g.l, eta.l, ra.l;

```

**APÉNDICE E**  
**PROGRAMAS UTILIZADOS EN LAS EVALUACIONES**  
**(C y MATLAB)**

## Apéndice E. Programas Utilizados en las Evaluaciones (C y MATLAB)

En este apéndice pueden encontrarse algunos de los programas más importantes realizados durante las evaluaciones. En cada programa se explica la función para la cual fue elaborado y cuales son los pasos necesarios para ejecutarlo.

### Programa 1

Este programa se ejecuta para la lectura de datos de H, Q, S y  $\eta$  de un compresor, con la finalidad de que estos datos queden en un archivo de matlab el cual se utiliza para realizar todos los calculos necesarios para evaluar el compresor. Este programa generará dos archivos a partir de un archivo .txt los cuales corresponden a uno de sólo lectura y otro de escritura. Para que el programa pueda ser ejecutado el archivo .txt debe tener el siguiente formato:

1) El inicio del archivo comienza con líneas de comentarios, las cuales deben comenzar con una C (mayúscula). 2) Después de los comentarios debe tener dos líneas para especificaciones. Una de ellas es para la velocidad mínima y velocidad máxima separadas por un espacio y la segunda para flujo volumétrico mínimo y flujo máximo también separados por un espacio. 3) Por último después de estas dos líneas deberán seguir los datos numéricos correspondientes a las características de cada una de las muestras (una fila por muestra) tomadas de dicho compresor. Dichas características se darán en el siguiente orden 1)Cabeza Adiabática, 2)Flujo Volumétrico, 3)Velocidad y 4)Eficiencia adiabática, todas separadas por un espacio. Para ejecutar este programa se debe de teclear a.out y el nombre del archivo sin extensión separados por un espacio y pulsar la tecla enter para leer el archivo de solo lectura. Para observar el archivo de escritura debera teclear la letra "m" y el nombre del archivo con extensión .mlb separados por un espacio y pulsar la tecla enter.

```
#include<stdio.h>
#include<stdlib.h>

main(int argc, char **argv) {

int i=0;
char c;
double s_min, s_max;
double f_min, f_max;
double head[50], flow[50], speed[50], effad[50];
int npuntos;
FILE *fpt;
FILE *fnum;
char name[50];
if (argc !=2) {
printf("sólo dame el nombre del archivo sin extensión\n");
```

```

    printf("error en el número de parámetros...\n");
    exit(1);
} /*checa si el número de argumentos es el correcto*/

strcpy(name,argv[1]);
strcat(name, ".txt");
fpt = fopen(name, "r");

if (fpt==NULL)
{
    printf("no se puede abrir el archivo indicado\n");
    exit(1);
}

while ((c = getc(fpt))!=EOF)
{
    if(c=='C')
    {
        while(getc(fpt)!='\n');
        continue;
    }

    ungetc(c,fpt);
    break;
}
/* Lectura de datos del archivo */

fscanf(fpt, "%lf %lf", &s_min, &s_max);
fscanf(fpt, "%lf %lf", &f_min, &f_max);
while(fscanf(fpt, "%lf %lf %lf %lf\n", &head[i], &flow[i], &speed[i], &effad[i])!=EOF)
/* printf(" %6.2lf %6.2lf %6.2lf %6.2lf\n", head[i], flow[i], speed[i], effad[i]);
i++;
}
npuntos = i;
printf("Numero de puntos=%d\n", npuntos);
printf("Velocidad Minima(ft/min)= %.2lf, Velocidad Maxima(ft/min)= %.2lf\n", s_min, s_max);
printf("Flujo Minimo(ft^3/min)= %.2lf, Flujo Maximo(ft^3/min)= %.2lf\n", f_min, f_max);
printf(" head(lbf*ft/min) flow(ft^3/min) speed(ft/min) effad(%) \n");
for (i = 0; i<=npuntos-1; i++)
printf(" %6.2lf %6.2lf %6.2lf %6.2lf\n", head[i], flow[i], speed[i], effad[i]);

fclose(fpt);
/*concatena el nombre del archivo con la extensión.mlb*/

strcpy(name2,argv[1]);
strcat(name2, ".mlb");
fnum = fopen(name2, "w");

```

```

if(fnum==NULL)
{
printf("No se puede abrir el archivo indicado\n");
exit(1);
}
}
fprintf(fnum,"%6.2lf %6.2lf\n",s_min,s_max);
fprintf(fnum,"%6.2lf %6.2lf\n",f_min,f_max);
fprintf(fnum,"%d\n", npuntos);
for (i = 0; i<=npuntos-1; i++)
    fprintf(fnum,"%6.2lf %6.2lf %6.2lf %6.2lf\n", head[i], flow[i],
speed[i], effad[i]);

fclose(fnum);

}

```

## Programa 2

Este programa calcula los coeficientes  $A_H$ ,  $B_H$ ,  $C_H$ ,  $D_H$ ,  $A_E$ ,  $B_E$ ,  $C_E$  y  $D_E$ , de las ecuaciones que describen el dominio de operación y eficiencia en un compresor. Para encontrarlos se utiliza el Método de Mínimos Cuadrados (MMC). Este archivo abre el archivo que se genera en C con extensión .mlb para leerlo y realizar las operaciones del Método de Mínimos Cuadrados (MMC).

donde:

$H$  Cabeza adiabática ( $\text{lb}^*\text{ft}/\text{lbm}$ )

$Q$  Flujo volumetrico ( $\text{ft}^3/\text{min}$ )

$S$  Velocidad ( $\text{ft}/\text{min}$ )

$E$  Eficiencia Adiabática (%)

$A_H$ ,  $B_H$ ,  $C_H$  y  $D_H$  son los valores de los coeficientes para la aproximación del ajuste de la curva que da como resultado el MMC, para la ecuación que describe el dominio de operación de un compresor.  $A_E$ ,  $B_E$ ,  $C_E$  y  $D_E$  son los valores de los coeficientes que da como resultado el MMC para la ecuación que describe la eficiencia en un compresor. Para ejecutar este programa se le tiene que proporcionar el nombre del archivo que contiene los datos del compresor.

```

file = input('Dame el nombre del archivo sin extensión ','s')
file1 = [file, '.mlb']
fid=fopen(file1, 'r');
v=fscanf(fid, '%f', 4);
SL = v(1);
SU = v(2);
QL = v(3);
QU = v(4);
n=fscanf(fid, '%f', 1);
M=fscanf(fid, '%f', [4, n]);

```

```

fclose(fid)
b=M';
H=b(:,1);
Q=b(:,2);
S=b(:,3);
E=b(:,4);
y=H./(S.^2);
x=Q./S;
x2=x.*x;
x3=x2.*x;
X=[ones(n,1) x x2 x3];
a=X*X;
b=X*y;
c=X*E;
bsol=a\b;
AH = bsol(1)
BH = bsol(2)
CH = bsol(3)
DH = bsol(4)
efic=a\c;
AE = efic(1)
BE = efic(2)
CE = efic(3)
DE = efic(4)
% Los coeficientes generados y los límites de velocidad y flujo volumetrico son
guardados % en un archivo .txt de escritura, ya que después son utilizados en trabajos
posteriores.
file2 = [file, '.txt']
fid=fopen(file2,'w');
fprintf(fid,'Smin Smax Qmin Qmax\n');
fprintf(fid,'%6.2f %6.2f %6.2f %6.2f\n', v(1),v(2),v(3),v(4));
fprintf(fid,'AH BH CH DH\n');
fprintf(fid,'%2.8f %2.8f %2.8f %2.8f\n', bsol(1),bsol(2),bsol(3),bsol(4));
fprintf(fid,'AE BE CE DE\n');
fprintf(fid,'%2.8f %2.8f %2.8f %2.8f\n', efic(1),efic(2),efic(3), efic(4));
fclose(fid)

```

### Programa 3

Este programa gráfica el dominio del compresor en el dominio  $(Q,H,S)$ . Para ejecutar este programa, primero se debe ejecutar el programa del método de mínimos cuadrados (MMC) (metmc.m), ya que este guarda en otro archivo con el mismo nombre del compresor los datos de  $S^L$ ,  $S^U$ ,  $Q^L$ ,  $Q^U$  y los valores encontrados de  $A_H$ ,  $B_H$ ,  $C_H$ ,  $D_H$ , los cuales se necesitan para ejecutar este programa.

```

file = input('Dame el nombre del archivo sin extensión ', 's')
file1 = [file, '.txt']
fid = fopen(file1, 'r');
datos = fscanf(fid, '%4s ', 4);
v = fscanf(fid, '%f', 4);
datos2 = fscanf(fid, '%2s', 4);
bsol = fscanf(fid, '%f', 4);
fclose(fid)
SL = v(1);
SU = v(2);
QL = v(3);
QU = v(4);
AH = bsol(1);
BH = bsol(2);
CH = bsol(3);
DH = bsol(4);
surge = QL / SL;
stonewall = QU / SU;
ranS = SU - SL;
numint = 10;
global ps w pd
propS = surge:(stonewall-surge)/numint:stonewall;
speed = SL:ranS/numint:SU;
for k = 1:length(speed)
    Q(k,:) = speed(k)*propS;
    QS = propS;
    H(k,:) = (AH*ones(size(QS)) + BH*QS + CH*QS.^2 + DH*QS.^3)*speed(k).^2;
    S(k,:) = speed(k)*ones(size(QS));
end
figure(1)
for k = 1:length(speed)
    plot(Q(k,:), H(k,:), 'b-')
    hold on
end
for k = 1:length(Q)
    plot(Q(:,k), H(:,k), 'r-')
    %hold on
end
hold off
xlabel('Flujo Volumétrico Q(ft3/min)')
ylabel('Cabeza Adiabática H(lbf*ft/lbm)')
title('H vs Q y S')
gtext('SL')
gtext('SU')
gtext('rl')
gtext('ru')
figure(2)

```

```

for k = 1:length(speed)
    plot3(Q(k,:), H(k,:), S(k,:), 'b-')
    hold on
end
for k = 1:length(Q)
    plot3(Q(:,k), H(:,k), S(:,k), 'r-')
    hold on
end
xlabel('Flujo volumétrico Q(ft^3/min)')
ylabel('Cabeza Adiabática H(lbf*ft/lbm)')
zlabel('S(ft/min)')
hold off

```

#### Programa 4

Este programa gráfica el dominio del compresor en el dominio  $(x_{ij}, p_i, p_j)$  que para los programas en matlab lo manejamos como  $(w, ps, pd)$ , donde  $w$  representa el flujo de masa,  $ps$  la presión de succión y  $pd$  la presión de descarga. Para ejecutar este programa, primero se debe ejecutar el programa del método de mínimos cuadrados (MMC) (`metmc.m`), ya que este guarda en otro archivo con el mismo nombre del compresor los datos de  $S^L$ ,  $S^U$ ,  $Q^L$ ,  $Q^U$  y los valores encontrados de  $A_H$ ,  $B_H$ ,  $C_H$ ,  $D_H$ ,  $A_E$ ,  $B_E$ ,  $C_E$  y  $D_E$ , los cuales se necesitan para ejecutar este programa.

```

file = input('Dame el nombre del archivo sin extension ', 's')
file1 = [file, '.txt']
fid = fopen(file1, 'r');
datos = fscanf(fid, '%4s', 4);
v = fscanf(fid, '%f', 4);
datos2 = fscanf(fid, '%2s', 4);
bsol = fscanf(fid, '%f', 4);
datos3 = fscanf(fid, '%2s', 4);
efic = fscanf(fid, '%f', 4);
fclose(fid)
SL = v(1);
SU = v(2);
QL = v(3);
QU = v(4);
AH = bsol(1);
BH = bsol(2);
CH = bsol(3);
DH = bsol(4);
AE = efic(1);
BE = efic(2);
CE = efic(3);
DE = efic(4);
surge = QL/SL;

```

```

stone = QU/SU;
% Compute the data for W, ps, pd.
k=1.287;      Z=0.95;          R=85.2;
Ts=519.67;   m=(k-1)/k;
psmin=200;   psmax=1200;
nn=40;
for i=1:nn+1
for j=1:nn+1
    ps(i,j)=psmin + (j-1)*(psmax-psmin)/nn;
    rhos= 144.0*ps(i,j)/(Z*R*Ts);
    w(i,j)= rhos*( QL +(i-1)*( QU - QL)/nn);

    qq=w(i,j)/rhos;
    rr=qq/ SL;
    hSmin= SL ^2*([1 rr rr^2 rr^3]*[AH BH CH DH]);
    rr=stone;
    hstone=qq^2*([1 rr rr^2 rr^3]*[ AH BH CH DH])/rr^2;
    rr=qq/ SU;
    hSmax= SU ^2*([1 rr rr^2 rr^3]*[ AH BH CH DH]);
    rr=surge;
    hsurge=qq^2*([1 rr rr^2 rr^3]*[ AH BH CH DH])/rr^2;
    hlow(i,j) = max([hSmin hstone]);
    hup(i,j) = min([hSmax hsurge]);
    pdlow(i,j)= ps(i,j)*(1 + (m/(Z*R*Ts))*hlow(i,j))^(1/m);
    pdup(i,j) = ps(i,j)*(1 + (m/(Z*R*Ts))*hup(i,j))^(1/m);
end
end
for k = 1:nn+1
    pd = pdlow(i,j) + (k-1)*(pdup(i,j) - pdlow(i,j))/nn;
end

figure(1)
plot(w(:,nn/2)', pdlow(:,nn/2)', 'g')
hold on;
plot(w(:,nn/2)', pdup(:,nn/2)', 'm')
gtext('A')
gtext('B')
gtext('C')
gtext('D')
xlabel('tasa de flujo de masa xij(lbm/min)')
ylabel('presión de descarga pj (psia)')
title(['pj vs xij como pi=', num2str(ps(1,nn/2)), ' (psia)'])
print s6_wpspd_fl -dpsc;
figure(2)
surfl(w(:,nn/2-1:nn/2+1), ps(:,nn/2-1:nn/2+1), pdlow(:,nn/2-1:nn/2+1))
hold on;
plot3(w', ps', pdlow', 'y:')

```

```

plot3(w, ps, pdlow,'c:')
plot3(w, ps, pdup,'m-')
plot3(w', ps', pdup,'b-')
surfl(w(:,nn/2-1:nn/2+1), ps(:,nn/2-1:nn/2+1), pdup(:,nn/2-1:nn/2+1))
%surfl(w(:,nn-3:nn+1), ps(:,nn-3:nn+1), pdllow(:,nn-3:nn+1))
%surfl(w(:,nn-3:nn+1), ps(:,nn-3:nn+1), pduup(:,nn-3:nn+1))
xlabel('xij(lbm/min)')
ylabel('pi (psia)')
zlabel('pj (psia)')
whitebg
title('pj vs xij y pi')

```

### Programa 5

Este programa gráfica la eficiencia del compresor. Para ejecutar este programa, primero se debe ejecutar el programa del método de mínimos cuadrados (MMC) (metmc.m), ya que este guarda en otro archivo con el mismo nombre del compresor los datos de  $S^L$ ,  $S^U$ ,  $Q^L$ ,  $Q^U$  y los valores encontrados de  $A_H$ ,  $B_H$ ,  $C_H$ ,  $D_H$ ,  $A_E$ ,  $B_E$ ,  $C_E$  y  $D_E$ , los cuales se necesitan para ejecutar este programa.

```

file = input('Dame el nombre del archivo sin extensión ','s')
file1 = [file, '.txt']
fid=fopen(file1,'r');
datos = fscanf(fid,'%4s',4);
v=fscanf(fid,'%f',4);
datos2 = fscanf(fid,'%2s',4);
bsol = fscanf(fid,'%f',4);
datos3 = fscanf(fid,'%2s',4);
efic = fscanf(fid,'%f',4);
fclose(fid)
SL = v(1);
SU = v(2);
QL = v(3);
QU = v(4);
AH = bsol(1);
BH = bsol(2);
CH = bsol(3);
DH = bsol(4);
AE = efic(1);
BE = efic(2);
CE = efic(3);
DE = efic(4);
surge = QL/SL;
stonewall = QU/SU;
numint = 10;
propQS = surge:(stonewall-surge)/numint:stonewall;

```

```

    QS = propQS;
    n = A_E*ones(size(QS)) + B_E*QS + C_E*QS.^2 + D_E*QS.^3;
    for k = 1:numint
        plot(n)
        hold on
    end
    xlabel('Q/S (ft^3/rpm)')
    ylabel('Eficiencia')
    title('Eficiencia adiabática vs Q/S')

```

### Programa 6

Este programa gráfica la figura de la función objetivo fijando la presión de succión ( $p_s$ ). Para ejecutar este programa, primero se debe ejecutar el programa del método de mínimos cuadrados (MMC) (metmc.m), ya que este guarda en otro archivo con el mismo nombre del compresor los datos de  $S^L$ ,  $S^U$ ,  $Q^L$ ,  $Q^U$  y los valores encontrados de  $A_H$ ,  $B_H$ ,  $C_H$ ,  $D_H$ ,  $A_E$ ,  $B_E$ ,  $C_E$  y  $D_E$ , los cuales se necesitan para ejecutar este programa.

```

global Zr R Ts m al K SL SU QL QU
global AH BH CH DH AE BE CE DE a b
% surge se representará por la letra "a"
% stonewall se representará por la letra "b"
file = input('Dame el nombre del archivo sin extension ', 's')
file1 = [file, '.txt']
fid = fopen(file1, 'r');
datos = fscanf(fid, '%4s', 4);
v = fscanf(fid, '%f', 4);
datos2 = fscanf(fid, '%2s', 4);
bsol = fscanf(fid, '%f', 4);
datos3 = fscanf(fid, '%2s', 4);
efic = fscanf(fid, '%f', 4);
fclose(fid)
SL = v(1);
SU = v(2);
QL = v(3);
QU = v(4);
AH = bsol(1);
BH = bsol(2);
CH = bsol(3);
DH = bsol(4);
AE = efic(1);
BE = efic(2);
CE = efic(3);
DE = efic(4);
a = QL/SL;

```

```

b = QU/SU;
K = 1.287;
Zr = 0.95;
R = 85.2;
Ts = 519.67;
al = 1;
m = (K-1)/K;
psmin=200;
psmax=1200;
ps = (psmin+psmax)/2;
nw=50;
npd=50;
ranS = SU-SL;
ranQ = QU-QL;
wmin = (QL*ps*144)/(Zr*R*Ts);
wmax = (QU*ps*144)/(Zr*R*Ts);
ranw = wmax-wmin;
wc = (ps*144*SL*b)/(Zr*R*Ts);
wd = (ps*144*SU*a)/(Zr*R*Ts);
w = wmin + (i-1)*(wmax-wmin)/nw;
q = (Zr*R*Ts*wc)/(ps*144);
qq = (Zr*R*Ts*wd)/(ps*144);
Ha = (AH + BH*(q/SL) + CH*(q/SL)2 + DH*(q/SL)3)*SL2;
H1 = (AH + BH*a + CH*a2 + DH*a3)*(qq/a)2;
%H2 = (AH + BH*b + CH*b2 + DH*b3)*(q/b)2;
%Hb = (AH + BH*(qq/SU) + CH*(qq/SU)2 + DH*(qq/SU)3)*SU2;
pdlow = (((Ha*m)/(Zr*R*Ts) + 1)(1/m))*ps;
%pdlow = (((H2*m)/(Zr*R*Ts) + 1)(1/m))*ps;
pdup = (((H1*m)/(Zr*R*Ts) + 1)(1/m))*ps;
%pdup = (((Hb*m)/(Zr*R*Ts) + 1)(1/m))*ps;

ranpd=pdup-pdlow;
x=wmin:ranw/nw:wmax;
y=pdlow:ranpd/npd:pdup;
Z = zeros(length(x),length(y));
[X,Y] = meshgrid(x,y);
for i=1:nw+1
    for j=1:npd+1
        if factible(ps,x(j),y(i)) == 1
            Z(i,j) = g_unit(ps,x(j),y(i));
        end
    end
end
end
end
mesh(X,Y,Z)
xlabel('xij(lbm/min)')
ylabel('pj(psia)')
zlabel('g(lbf-ft/min)')

```

title('Función de consumo de combustible (g)')

### Programa 7

Este programa encuentra la diferencia que existe al comparar la función objetivo dándole un punto (w,ps,pd) con cada una de las 6 funciones polinomiales que se utilizan para aproximar la función de costo de combustible. Para fines prácticos: surge se representará por la letra "a" stonewall se representará por la letra "b". Para ejecutar este programa, primero se debe ejecutar el programa de MMC (metmc.m) ya que este guarda en otro archivo con el mismo nombre del compresor los datos de  $S^L$ ,  $S^U$ ,  $Q^L$ ,  $Q^U$  y los valores encontrados de  $A_H$ ,  $B_H$ ,  $C_H$ ,  $D_H$ ,  $A_E$ ,  $B_E$ ,  $C_E$  y  $D_E$ , los cuales se necesitan para ejecutar este programa.

```

global Zr R Ts m al k SL SU
global AH BH CH DH AE BE CE DE
global A1 B1 C1 D1 A2 B2 C2 D2 E2 F2 G2 H2 I2 J2 A3 B3 C3
global A4 B4 C4 D4 E4 F4 A5 B5 C5 A6 B6 C6 D6 E6 F6
file = input('Dame el archivo sin extensión ', 's')
file1 = [file, 'pm.txt']
fid = fopen(file1, 'r');
datos = fscanf(fid, '%5s', 3);
psmed = fscanf(fid, '%f', 1);
wmed = fscanf(fid, '%f', 1);
pdmed = fscanf(fid, '%f', 1);
fclose(fid)
file = input('Dame el nombre del archivo sin extension ', 's')
file1 = [file, '.txt']
fid = fopen(file1, 'r');
datos = fscanf(fid, '%4s', 4);
v = fscanf(fid, '%f', 4);
datos2 = fscanf(fid, '%2s', 4);
bsol = fscanf(fid, '%f', 4);
datos3 = fscanf(fid, '%2s', 4);
efic = fscanf(fid, '%f', 4);
fclose(fid)
SL = v(1);
SU = v(2);
QL = v(3);
QU = v(4);
AH = bsol(1);
BH = bsol(2);
CH = bsol(3);
DH = bsol(4);
AE = efic(1);
BE = efic(2);
CE = efic(3);

```

```

DE = efic(4);
a = QL/SL;
b = QU/SU;
%K = 1.287;
%Zr = 0.95;
%R = 85.2;
%Ts = 519.67;
%al = 1;
%m = (K-1)/K;
psmin=200;
psmax=1200;
nps=10;
nw=10;
npd=10;
sc=1;
% Compute the data for w, ps, pd
for i=1:nps+1
    ps = psmin + (i-1)*(psmax-psmin)/nps;
    wmin = (QL*ps*144)/(Zr*R*Ts);
    wmax = (QU*ps*144)/(Zr*R*Ts);
    wc = (ps*144*SL*b)/(Zr*R*Ts);
    wd = (ps*144*SU*a)/(Zr*R*Ts);
    for i=1:nw+1
        w = wmin + (i-1)*(wmax-wmin)/nw;
        q = (Zr*R*Ts*w)/(ps*144);
        Ha = (AH + BH*(q/SL) + CH*(q/SL)^2 + DH*(q/SL)^3)*SL^2;
        H1 = (AH + BH*a + CH*a^2 + DH*a^3)*(q/a)^2;
        H1a = (AH + BH*b + CH*b^2 + DH*b^3)*(q/b)^2;
        Hb = (AH + BH*(q/SU) + CH*(q/SU)^2 + DH*(q/SU)^3)*SU^2;
        %Obteniendo el límite inferior de Pd
        if w < wc
            pdlow = (((Ha*m)/(Zr*R*Ts) + 1)^(1/m))*ps;
        else
            pdlow = (((H2*m)/(Zr*R*Ts) + 1)^(1/m))*ps;
        end
        %Obteniendo el límite superior de Pd
        if w < wd
            pdup = (((H1*m)/(Zr*R*Ts) + 1)^(1/m))*ps;
        else
            pdup = (((Hb*m)/(Zr*R*Ts) + 1)^(1/m))*ps;
        end
        if pdup > pdlow + 0.00001
            for k=1:npd+1
                pd = pdlow + (k-1)*(pdup-pdlow)/npd;
                P = [ps w pd];
                P1 = [P1; P];
                obj(sc)=g_unit(ps,w,pd);
            end
        end
    end
end

```

```

        sc=sc+1;
    end
else
    pd = pdlow;
    P = [ps w pd];
    P1 = [P1; P];
    obj(sc)=g_unit(ps,w,pd)
    sc=sc+1;
end
end
end
end

%P1 es una matriz que se va formando cada vez que se entra al for%
[fil,col] = size(P1);
sum1 = 0; sum2 = 0; sum3 = 0; sum4 = 0; sum5 = 0; sum6 = 0;
dif11 = 0; dif22 = 0; dif33 = 0; dif44 = 0; dif55 = 0; dif66 = 0;
fil
for i=1:fil
    ps = P1(i,1);
    w = P1(i,2);
    pd = P1(i,3);
    f11=f1(ps,w,pd);
    dif1=abs(obj(i)-f11)/abs(obj(i)+1);
    f22=f2(ps,w,pd);
    dif2=abs(obj(i)-f22)/abs(obj(i)+1);
    f33=f3(ps,w,pd);
    dif3=abs(obj(i)-f33)/abs(obj(i)+1);
    f44=f4(ps,w,pd);
    dif4=abs(obj(i)-f44)/abs(obj(i)+1);
    f55=f5(ps,w,pd);
    dif5=abs(obj(i)-f55)/abs(obj(i)+1);
    f66=f6(ps,w,pd);
    dif6=abs(obj(i)-f66)/abs(obj(i)+1);
    sum1 = sum1 + dif1;
    sum2 = sum2 + dif2;
    sum3 = sum3 + dif3;
    sum4 = sum4 + dif4;
    sum5 = sum5 + dif5;
    sum6 = sum6 + dif6;
    if dif1 > dif11
        dif11 = dif1;
    end
    if dif2 > dif22
        dif22 = dif2;
    end
    if dif3 > dif33
        dif33 = dif3;
    end
end
end

```

```

        if dif4 > dif44
            dif44 = dif4;
        end
        if dif5 > dif55
            dif55 = dif5;
        end
        if dif6 > dif66
            dif66 = dif6;
        end
        errores = [dif1 dif2 dif3 dif4 dif5 dif6];
        diferrores = [diferrores; errores];
    end
    prom1 = sum1/fil
    prom2 = sum2/fil
    prom3 = sum3/fil
    prom4 = sum4/fil
    prom5 = sum5/fil
    prom6 = sum6/fil
    dif1;
    dif2;
    dif3;
    dif4;
    dif5;
    dif6;

```

### Programa 8

Este programa encuentra los coeficientes para aproximar las 6 funciones polinomiales que sirven para aproximar la función de consumo de combustible del compresor. Se utiliza el método de mínimos cuadrados (MMC) para encontrar los coeficientes de cada función, para los cuales se hizo una rutina por cada función en la cual se encuentran estos coeficientes. Para ejecutar este programa, primero se debe ejecutar el programa de MMC (metmc.m) ya que este guarda en otro archivo con el mismo nombre del compresor los datos de  $S^L$ ,  $S^U$ ,  $Q^L$ ,  $Q^U$  y los valores encontrados de  $A_H$ ,  $B_H$ ,  $C_H$ ,  $D_H$ ,  $A_E$ ,  $B_E$ ,  $C_E$  y  $D_E$ . Para fines prácticos del programa: surge se representará por la letra "a" y stonewall se representará por la letra "b".

```

global Zr R Ts m al K SL SU
global AH BH CH DH AE BE CE DE
global A1 B1 C1 D1 A2 B2 C2 D2 E2 F2 G2 H2 I2 J2 A3 B3 C3
global A4 B4 C4 D4 E4 F4 A5 B5 C5 A6 B6 C6 D6 E6 F6
file = input('Dame el nombre del archivo sin extension ', 's')
file1 = [file, '.txt']
fid = fopen(file1, 'r');
datos = fscanf(fid, '%4s', 4);

```

```

v = fscanf(fid,'%f',4);
datos2 = fscanf(fid,'%2s',4);
bsol = fscanf(fid,'%f',4);
datos3 = fscanf(fid,'%2s',4);
efic = fscanf(fid,'%f',4);
fclose(fid)
SL = v(1);
SU = v(2);
QL = v(3);
QU = v(4);
AH = bsol(1);
BH = bsol(2);
CH = bsol(3);
DH = bsol(4);
AE = efic(1);
BE = efic(2);
CE = efic(3);
DE = efic(4);
a = QL/SL;
b = QU/SU;
%K = 1.287;
%Zr = 0.95;
%R = 85.2;
%Ts = 519.67;
%al = 1;
%m = (K-1)/K;
psmin=200;
psmax=1200;
nps=10;
nw=10;
npd=10;
% Compute the data for w, ps, pd
for i=1:nps+1
    ps = psmin + (i-1)*(psmax-psmin)/nps;
    wmin = (QL*ps*144)/(Zr*R*Ts);
    wmax = (QU*ps*144)/(Zr*R*Ts);
    wc = (ps*144*SL*b)/(Zr*R*Ts);
    wd = (ps*144*SU*a)/(Zr*R*Ts);
    for i=1:nw+1
        w = wmin + (i-1)*(wmax-wmin)/nw;
        q = (Zr*R*Ts*w)/(ps*144);
        Ha = (AH + BH*(q/SL) + CH*(q/SL)2 + DH*(q/SL)3)*SL2;
        H1 = (AH + BH*a + CH*a2 + DH*a3)*(q/a)2;
        H1a = (AH + BH*b + CH*b2 + DH*b3)*(q/b)2;
        Hb = (AH + BH*(q/SU) + CH*(q/SU)2 + DH*(q/SU)3)*SU2;
        %Obteniendo el límite inferior de Pd
        if w < wc

```

```

pdlow = (((Ha*m)/(Zr*R*Ts) + 1)^(1/m))*ps;
else
pdlow = (((H1a*m)/(Zr*R*Ts) + 1)^(1/m))*ps;
end
%Obteniendo el límite superior de Pd
if w < wd
pdup = (((H1*m)/(Zr*R*Ts) + 1)^(1/m))*ps;
else
pdup = (((Hb*m)/(Zr*R*Ts) + 1)^(1/m))*ps;
end
if pdup>pdlow+0.00001
    for k=1:npd+1
        pd = pdlow + (k-1)*(pdup-pdlow)/npd;
        g=g_unit(ps,w,pd);
        d = [ps w pd g];
        D = [D; d];
    end
else
    pd = pdlow;
    g=g_unit(ps,w,pd);
    d = [ps w pd g];
    D = [D; d];
end
end
end
end

cg1=g1(D)
cg2=g2(D)
cg3=g3(D)
cg4=g4(D)
cg5=g5(D)
cg6=g6(D)

```

**APÉNDICE F**  
**PRUEBA DE HIPÓTESIS UTILIZANDO ESTADÍSTICA**  
**NO PARAMÉTRICA**

## **Apéndice F. Prueba de Hipótesis Utilizando Estadística No Paramétrica**

En este apéndice se describe la prueba de hipótesis la cual se hizo utilizando estadística no paramétrica. La importancia de realizar esta prueba de hipótesis es con la finalidad de mostrar científicamente utilizando las herramientas de la estadística el impacto positivo que tiene la aplicación de las técnicas de pre-procesamiento al algoritmo de optimización utilizado en la evaluación computacional de la fase de pre-procesamiento.

Para realizar la prueba de hipótesis haremos uso de la estadística no paramétrica en donde se desconoce que tipo de distribución tiene la población.

Los procedimientos no paramétricos o de distribución libre se utilizan con frecuencia en el análisis de datos. Las desventajas asociadas con las pruebas no paramétricas son que no se utiliza toda la información proporcionada por la muestra, por lo que una prueba no paramétrica será menos eficiente que el procedimiento paramétrico cuando ambos métodos son aplicables. En consecuencia para alcanzar la misma potencia, una prueba no paramétrica requerirá un tamaño de muestra más grande que la correspondiente prueba paramétrica.

En resumen, si son aplicables tanto la prueba paramétrica como la no paramétrica al mismo conjunto de datos, se debe aplicar la prueba paramétrica más eficiente. Sin embargo, se debe reconocer que las suposiciones de normalidad con frecuencia no se pueden justificar y que no siempre se tienen las mediciones cuantitativas. Es una fortuna que los estadísticos proporcionen un número de procedimientos no paramétricos útiles.

## Metodología

Para el problema en estudio, queremos probar la hipótesis de que las diversas técnicas de pre-procesamiento tienen un impacto positivo en el desempeño del algoritmo, o dicho estadísticamente que el número de iteraciones empleado por el algoritmo de optimización se reduce significativamente. Para cada topología utilizada en la evaluación se realizarán dos pruebas de hipótesis. En la primera prueba, queremos probar la hipótesis de que el número de iteraciones que tarda el optimizador en encontrar un óptimo local en cada una de las instancias para cada compresor en el escenario E1 es mayor que lo que se tarda en encontrar esa solución en el escenario E2. La otra prueba consiste en probar que en el escenario E3 el número de iteraciones que tarda el algoritmo en encontrar la solución es menor que en el escenario E2.

Recordemos que en cada uno de los escenarios se aplican las siguientes técnicas:

E1	E2	E3
*Punto inicial	*Punto inicial	*Punto inicial
*Cotas a las variables	*Cotas a las variables	*Cotas a las variables
	*Escalamiento de la ecuación de la eficiencia	*Escalamiento de la ecuación de la eficiencia y ecuación del dominio

El método no paramétrico que utilizaremos para la prueba de hipótesis que deseamos probar se conoce como *Prueba de los signos* (Walpole y Myers, 1992). Ésta prueba se utiliza para probar la hipótesis de una mediana poblacional. En el caso de muchos procedimientos no paramétricos, la media se reemplaza por la mediana como el parámetro de ubicación pertinente bajo prueba. La contraparte poblacional, representada por  $\mu$  tiene una definición análoga. Dada una variable aleatoria  $X$ ,  $\mu$  se define de tal forma que:

$$P(X > \mu) = P(X < \mu) = 0.5$$

Por supuesto, si la distribución es simétrica, la media y la mediana poblacionales son iguales. Al probar la hipótesis nula  $H_0$  de que  $\mu = \mu_0$  contra una alternativa apropiada, sobre la base de una muestra aleatoria de tamaño  $n$  datos, se reemplaza cada valor muestral que excede a  $\mu_0$  con un signo *más* (+) y cada valor muestral menor que  $\mu_0$  con un signo *menos* (-). Si la hipótesis nula es verdadera y la población es simétrica, la suma de los signos más debe ser aproximadamente igual a la suma de los signos menos. Cuando un signo aparece con mayor frecuencia de lo que debería, con base en el azar únicamente, se rechaza la hipótesis de que la mediana poblacional  $\mu$  es igual que  $\mu_0$ .

El estadístico de prueba apropiado para la prueba de los signos es la variable aleatoria binomial  $X$ , que representa *el número de signos más (+) en la muestra aleatoria*. Si la hipótesis nula de que  $\mu = \mu_0$  es verdadera, la probabilidad de que un valor muestral resulte en un signo *más* o un signo *menos* es igual a  $1/2$ . Por lo tanto, para probar la hipótesis nula  $\mu = \mu_0$ , en realidad se está probando la hipótesis nula de que el número de signos *más* (+) es un valor de una variable aleatoria que tiene una distribución binomial con el parámetro  $p = 1/2$ . Los valores  $p$  para ambas alternativas unilateral y bilateral puede entonces calcularse utilizando esta distribución binomial.

Por ejemplo para nuestra prueba de hipótesis utilizamos la alternativa unilateral, que se representa de la siguiente forma:

$$H_0 : \mu = \mu_0$$

$$H_1 : \mu > \mu_0$$

Aquí se rechaza  $H_0$  en favor de  $H_1$  sólo si la proporción de signos *más* es bastante más grande que  $1/2$ , esto es, cuando  $x$  es grande. De aquí que, si el valor calculado de  $P$ :

$$P = P(X \geq x \text{ cuando } p = 1/2)$$

es menor que  $\alpha$ , se rechaza  $H_0$  en favor de  $H_1$ , donde  $\alpha$  es el nivel de significancia igual a 0.05.

## Resultados

Para llevar a cabo la prueba, ésta puede ser resumida en los siguientes seis pasos. Enseguida se muestra la prueba de hipótesis para las tres topologías utilizadas en las evaluaciones.

- **Prueba de hipótesis para la red en línea (E2 mejor que E1)**

Para probar la hipótesis de que el promedio de iteraciones en el escenario E1 es mayor que en el escenario E2, establecemos la siguiente prueba.

$$H_0 : \mu - \mu_0 = 0 \text{ vs } H_1 : \mu - \mu_0 > 0 \text{ con un } \alpha = 0.05$$

**Estadístico de prueba:** Variable binomial  $X$  con  $p = 1/2$

**Cálculos:** Después de reemplazar cada diferencia positiva con un símbolo “+” y cada diferencia negativa por un símbolo “-“ y descartar las diferencias cero en su compresor correspondiente de las Tablas 17 y 18 del Apéndice D, tenemos que:

Dado que  $n$  es el total de signos positivos y negativos y  $x$  el total de signos positivos:

$$n = 62$$

$$x = 40$$

Utilizando la aproximación de la curva normal, sabiendo que la aproximación es excelente cuando  $n$  es grande:

$$Z = \frac{x - \mu}{\sigma}$$

donde:

$$\mu = np$$

$$\sigma = \sqrt{npq}$$

Sustituyendo en fórmulas se tiene:

$$\mu = 31$$

$$\sigma = 3.93$$

$$Z = 2.16$$

Por lo tanto calculando el valor de la probabilidad  $P$ :

$$P = P(X \geq 40) \cong P(Z > 2.16) = 1 - 0.9846 = \mathbf{0.0154}$$

**Conclusión:** Como  $P < \alpha$  se rechaza  $H_0$ , concluyendo que hay evidencia suficiente para rechazar  $H_0$ , aceptando la hipótesis alternativa que indica que el promedio de iteraciones en E1 es mayor que en E2. Esto puede confirmarse viendo las Tablas 17 y 18 en el Apéndice D, comparando el número de iteraciones obtenido en la Tabla 17 y el número de iteraciones obtenido en la Tabla 18 para cada compresor de la red en línea.

- **Prueba de hipótesis para la red en línea (E3 mejor que E2)**

Para probar la hipótesis de que el promedio de iteraciones en el escenario E2 es mayor que en el escenario E3, establecemos una prueba similar a la anterior. Después de reemplazar cada diferencia positiva con un símbolo “+” y cada diferencia negativa por un símbolo “-” y descartar las diferencias cero en su compresor correspondiente de las Tablas 18 y 19 del Apéndice D, tenemos lo siguiente.

Dado que  $n$  es el total de signos positivos y negativos y  $x$  el total de signos positivos:

$$n = 74$$

$$x = 47$$

Utilizando la aproximación de la curva normal:

$$\mu = 37$$

$$\sigma = 4.41$$

$$Z = 2.15$$

Por lo tanto calculando el valor de la probabilidad  $P$ :

$$P = P(X \geq 47) \cong P(Z > 2.15) = 1 - 0.9842 = \mathbf{0.0158}$$

**Conclusión:** Como  $P < \alpha$  se rechaza  $H_0$ , concluyendo que hay evidencia suficiente para rechazar  $H_0$ , aceptando la hipótesis alternativa que indica que el promedio de iteraciones en E2 es mayor que en E3. Esto puede confirmarse viendo las Tablas 18 y 19 en el Apéndice D, comparando el número de iteraciones obtenido en la Tabla 18 y el número de iteraciones obtenido en la Tabla 19 para cada compresor de la red en línea.

- **Prueba de hipótesis para la red tipo árbol (E2 mejor que E1)**

Para probar la hipótesis de que el número de iteraciones en el escenario E1 es mayor que en el escenario E2, se establece la prueba de hipótesis para la topología en línea. Después de reemplazar cada diferencia positiva con un símbolo “+” y cada diferencia negativa por un símbolo “-“ y descartar las diferencias cero en su compresor correspondiente de las Tablas 20 y 21 del Apéndice D, tenemos lo siguiente.

Dado que  $n$  es el total de signos positivos y negativos y  $x$  el total de signos positivos:

$$n = 39$$

$$x = 24$$

Utilizando la aproximación de la curva normal:

$$\mu = 19.5$$

$$\sigma = 3.12$$

$$Z = 1.28$$

Por lo tanto calculando el valor de la probabilidad  $P$ :

$$P = P(X \geq 24) \cong P(Z > 1.28) = 1 - 0.8997 = 0.1003$$

**Conclusión:** Como  $P > \alpha$ , no se rechaza  $H_0$  inicialmente. Esto puede confirmarse viendo las Tablas 20 y 21 en el Apéndice D, comparando el número de iteraciones obtenido en la Tabla 20 y el número de iteraciones obtenido en la Tabla 21 para cada compresor de la red tipo árbol.

Si se utilizan ocho de los nueve compresores, quitando el último compresor, establecemos de nuevo la prueba.

Realizando otra vez la prueba tenemos que:

$$n = 35$$

$$x = 24$$

Utilizando la aproximación de la curva normal:

$$\mu = 17.5$$

$$\sigma = 2.95$$

$$Z = 2.03$$

Por lo tanto calculando el valor de la probabilidad  $P$ :

$$P = P(X \geq 24) \cong P(Z > 2.03) = 1 - 0.9788 = 0.0212$$

**Conclusión:** Como  $P < \alpha$ , se rechaza  $H_0$ , aceptando la hipótesis alternativa donde el número de iteraciones en el escenario E1 es mayor que en el escenario E2 utilizando los primeros ocho compresores de la evaluación.

• **Prueba de hipótesis para la red tipo árbol (E3 mejor que E1)**

Para probar la hipótesis de que el número de iteraciones en el escenario E2 es mayor que en el escenario E3, la prueba se define con los mismos pasos de las pruebas anteriores. Después de reemplazar cada diferencia positiva con un símbolo “+” y cada diferencia negativa por un símbolo “-” y descartar las diferencias cero en su compresor correspondiente de las Tablas 21 y 22 del Apéndice D, tenemos lo siguiente.

Dado que  $n$  es el total de signos positivos y negativos y  $x$  el total de signos positivos:

$$n = 54$$

$$x = 43$$

Utilizando la aproximación de la curva normal:

$$\mu = 27$$

$$\sigma = 3.67$$

$$Z = 4.21$$

Por lo tanto calculando el valor de la probabilidad  $P$ :

$$P = P(X \geq 43) \cong P(Z > 4.21) \cong 0$$

6) **Conclusión:** Como  $P < \alpha$  se rechaza  $H_0$ , aceptando la hipótesis alternativa que indica que el número de iteraciones en E2 es mayor que en E3. Esto puede confirmarse viendo las Tablas 21 y 22 en el Apéndice D, comparando el número de iteraciones obtenido en la Tabla 21 y el número de iteraciones obtenido en la Tabla 22 para cada compresor de la red tipo árbol.

- **Prueba de hipótesis para la red con ciclo (E2 mejor que E1)**

Para probar la hipótesis de que el número de iteraciones en el escenario E1 es mayor que en el escenario E2, se define de la misma forma que en las topologías anteriores. Después de reemplazar cada diferencia positiva con un símbolo "+" y cada diferencia negativa por un símbolo "-" y descartar las diferencias cero en su compresor correspondiente de las Tablas 23 y 24 del Apéndice D, tenemos lo siguiente.

Dado que  $n$  es el total de signos positivos y negativos y  $x$  el total de signos positivos:

$$n = 99$$

$$x = 62$$

Utilizando la aproximación de la curva normal:

$$\mu = 49.5$$

$$\sigma = 4.97$$

$$Z = 2.41$$

Por lo tanto calculando el valor de la probabilidad  $P$ :

$$P = P(X \geq 62) \cong P(Z > 2.41) = 1 - 0.9920 = 0.008$$

**Conclusión:** Como  $P < \alpha$ , se rechaza  $H_0$ , aceptando la hipótesis alternativa donde el número de iteraciones en E1 es mayor que en E2. Esto puede confirmarse viendo las tablas 23 y 24 en el Apéndice D, comparando el número de iteraciones obtenido en la Tabla 23 y el número de iteraciones obtenido en la Tabla 24 para cada compresor de la red con ciclo.

- **Prueba de hipótesis para la red con ciclo (E3 mejor que E2)**

Para probar la hipótesis de que el número de iteraciones en el escenario E2 es mayor que en el escenario E3, se puede definir de la misma manera que el inciso anterior. Después de reemplazar cada diferencia positiva con un símbolo “+” y cada diferencia negativa por un símbolo “-“ y descartar las diferencias cero en su compresor correspondiente de las Tablas 25 y 26 del Apéndice D, tenemos lo siguiente.

Dado que  $n$  es el total de signos positivos y negativos y  $x$  el total de signos positivos:

$$n = 115$$

$$x = 89$$

Utilizando la aproximación de la curva normal:

$$\mu = 57.5$$

$$\sigma = 5.36$$

$$Z = 5.78$$

Por lo tanto calculando el valor de la probabilidad  $P$ :

$$P = P(X \geq 89) \cong P(Z > 5.78) \cong 0$$

**Conclusión:** Como  $P < \alpha$ , se rechaza  $H_0$ , aceptando la hipótesis alternativa que indica que el número de iteraciones en E2 es mayor que en E3. Esto puede confirmarse viendo las Tablas 25 y 26 del Apéndice D de la topología con ciclo.

## **Conclusión**

Según los resultados obtenidos en las pruebas de hipótesis realizadas para cada topología, queda comprobado científicamente haciendo uso de las herramientas estadísticas, que los resultados obtenidos en la evaluación computacional de las técnicas de pre-procesamiento avalan categóricamente la gran importancia de utilizarlas en cualquier algoritmo de optimización que presente problemas en su proceso de búsqueda, ya que tienen un impacto positivo en la disminución del número de iteraciones que emplea el algoritmo para encontrar una solución óptima local.

## **FICHA AUTOBIOGRÁFICA**

**Yanet Villalobos Morales**

**Candidato para el grado de Maestro en Ciencias en Ingeniería de Sistemas  
Universidad Autónoma de Nuevo León  
Facultad de Ingeniería Mecánica y Eléctrica**

**Tesis**

**PRE-PROCESAMIENTO DE UN PROBLEMA DE OPTIMIZACIÓN DE REDES  
DE GAS NATURAL**

Nació en Mapimí, Dgo, México. Es hija del Sr. Francisco Villalobos y de la Sra. Ventura Morales. Recibió el título de Ing. Mecánico Administrador de la Facultad de Ingeniería Mecánica y Eléctrica (FIME) en la Universidad Autónoma de Nuevo León (1995-2000), con Mención Honorífica, por haber obtenido uno de los mejores promedios de la generación. Recibió una beca de la FIME para estudiar la Maestría en Ciencias en Ingeniería de Sistemas. Dentro de la Maestría participó como Asistente de Investigación en el proyecto de CONACyT J33187-A del cual recibió beca de manutención.

