

INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA
SECCIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

“COORDINACIÓN DE CAMBIADORES DE DERIVACIÓN
DE TRANSFORMADORES DE SUBESTACIÓN Y
CAPACITORES PARA MINIMIZAR PÉRDIDAS EN REDES
DE DISTRIBUCIÓN”

TESIS

QUE PARA OBTENER EL GRADO DE:

MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA

PRESENTA:

ING. RAÚL GUILLERMO JARA SIXTOS

DR. RICARDO O. MOTA PALOMINO
DIRECTOR DE TESIS

MÉXICO D.F. JUNIO 2012





SIP-14

INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de MEXICO D. F. siendo las 13:00 horas del día 15 del mes de JUNIO del 2012 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación de la E.S.I.M.E. ZAC. para examinar la tesis titulada:

“COORDINACIÓN DE CAMBIADORES DE DERIVACIÓN DE TRANSFORMADORES DE SUBESTACIÓN Y CAPACITORES PARA MINIMIZAR PÉRDIDAS EN REDES DE DISTRIBUCIÓN ”

Presentada por el alumno:

JARA	SIXTOS	RAUL GUILLERMO							
Apellido paterno	Apellido materno	Nombre(s)							
Con registro:									
<table border="1"> <tr> <td style="width: 20px; text-align: center;">A</td> <td style="width: 20px; text-align: center;">1</td> <td style="width: 20px; text-align: center;">0</td> <td style="width: 20px; text-align: center;">0</td> <td style="width: 20px; text-align: center;">4</td> <td style="width: 20px; text-align: center;">5</td> <td style="width: 20px; text-align: center;">2</td> </tr> </table>			A	1	0	0	4	5	2
A	1	0	0	4	5	2			

aspirante de:

MAESTRÍA EN CIENCIAS EN INGENIERÍA ELÉCTRICA

Después de intercambiar opiniones, los miembros de la Comisión manifestaron **APROBAR LA TESIS**, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.


LA COMISIÓN REVISORA

Director(a) de tesis


DR. RICARDO OCTAVIO ARTURO
MOTA PALOMINO


DR. DAVID ROMERO ROMERO
PRESIDENTE

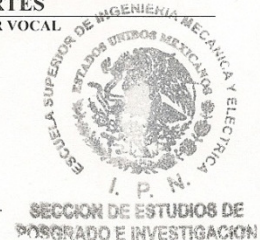

DR. JAIME ROBLES GARCÍA
SECRETARIO

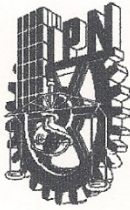

DR. DAVID SEBASTIÁN BALTAZAR
SEGUNDO VOCAL


DR. FERMÍN PASCUAL ESPINO
CORTÉS
TERCER VOCAL

PRESIDENTE DEL COLEGIO DE PROFESORES


DR. MAURO ALBERTO ENCISO AGUILAR





INSTITUTO POLITÉCNICO NACIONAL

SECRETARIA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de México, Distrito Federal, el día **15** del mes de **Junio** del año **2012**, el que suscribe **RAÚL GUILLERMO JARA SIXTOS**, alumno del Programa de **Maestría en Ciencias con especialidad en Ingeniería Eléctrica** con número de registro **A100452**, adscrito a la **Sección de Estudios de Posgrado e Investigación** de la **ESIME Unidad Zacatenco**, manifiesta que es autor intelectual del presente Trabajo de Tesis bajo la dirección del **Dr. Ricardo Octavio Mota Palomino** y cede los derechos del trabajo titulado: **"COORDINACIÓN DE CAMBIADORES DE DERIVACIÓN DE TRANSFORMADORES DE SUBESTACIÓN Y CAPACITORES PARA MINIMIZAR PÉRDIDAS EN REDES DE DISTRIBUCIÓN"**, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, graficas o datos del trabajo sin permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo la siguiente dirección: **rjaras0400@gmail.com** y/o **rmotap@ipn.mx**.

Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Ing. Raúl Guillermo Jara Sixtos



RESUMEN

Debido a la naturaleza de los equipos instalados, la coordinación de fuentes de potencia reactiva en un sistema que se compone de la red de subtransmisión y circuitos de distribución de media tensión resulta en un problema de optimización no-lineal con variables enteras, cuya complejidad se incrementa rápidamente con el número de equipos instalados, con el horizonte de análisis y el número de etapas consideradas.

En este trabajo se propone un método para resolver el problema de control de fuentes de potencia reactiva en redes de distribución usando programación dinámica. El método propuesto está diseñado para obtener un despacho de capacitores en derivación a lo largo del alimentador y se incluye el modelado de los cambiadores de derivación en el transformador de la subestación, de manera tal que se puede buscar que las pérdidas totales se reduzcan, se mejoren los perfiles de voltaje o se reduzca el número de operaciones de los capacitores y de los cambiadores de derivación al cumplir con las restricciones consideradas.

Las restricciones modeladas en el problema analizado son:

- El máximo número permitido de operaciones de conexión o desconexión de los bancos de capacitores.
- El máximo número permitido de operaciones en los cambiadores de derivación del transformador.
- Los límites de voltaje en el alimentador

Para probar la efectividad del método propuesto, se realiza el despacho horario de un sistema de prueba incluyendo diferentes valores de restricciones. De esta manera se encontró que la inclusión al modelo de despacho de capacitores del despacho de los cambiadores de derivación en los transformadores de subestación mejora los perfiles de voltaje y minimiza considerablemente las pérdidas del alimentador.



ABSTRACT

Due to the nature of the installed equipment, the coordination of reactive power sources in a system that is composed by the subtransmission network and distribution feeders in medium voltage turns into a nonlinear optimization problem with integer variables, whose complexity grows fast with respect to the number of installed equipment, the study time and the number of stages considered.

This work proposes a coordination method for solving the reactive sources control problem in a distribution system using dynamic programming. The proposed method is designed to dispatch shunt capacitors and includes the modeling of under load tap changers, such that the total loss is minimized, bus voltage profiles are improved and switching operations of capacitors and tap changers are reduced according to the considered constraints.

The modeled constraints in the analyzed problem are:

- The maximum allowable number of switching operations in a day for each capacitor.
- The maximum allowable number of switching operations in a day for under load tap changer
- The feeder's section voltage limits.

To demonstrate the effectiveness of the proposed method, the hourly reactive power/voltage control in a distribution test system is performed including different values of constraints. It is found that the inclusion of under load tap changer model in the dispatch of capacitors improves voltage profiles and reduces total feeder loss considerably.



DEDICATORIAS

A mis padres Guadalupe Sixtos Segura y Rene Jara Cortes, por apoyarme en cada paso de mi vida, por esforzarse día a día para que nuestra familia salga adelante, siendo siempre un ejemplo a seguir como personas, pero sobre todo por entregarme su amor incondicional.

A mis hermanos Rene, Jorge y Raquel por estar siempre conmigo, por darme apoyo, por sus consejos, por orientarme, por acompañarme, por su aliento para seguir adelante y por hacer de este lazo de sangre una verdadera unión de fraternidad inquebrantable.

A toda mi familia, por estar al pendiente de mí, por su preocupación y por sus buenos deseos.

A Gladis y a su familia por abrir sus corazones hacia mí y permitirme compartir muy gratos momentos.

A Dios y a la vida por darme la oportunidad de alcanzar mis metas.



AGRADECIMIENTOS

Al Instituto Politécnico Nacional por ser mi alma máter y permitir mi desarrollo académico desde la educación media superior.

Al CONACYT por proveerme el apoyo económico sin el cual no me hubiera sido posible realizar esta empresa.

Al Dr. Ricardo Mota Palomino, por guiarme durante el desarrollo de esta Tesis, por todos sus consejos y sugerencias y sobre todo por estar al pendiente de mis necesidades más allá del ámbito académico.

A la comisión revisora de este trabajo conformada por los doctores David Romero Romero, Jaime Robles García, David Sebastián Baltazar, Fermín Pascual Espino y Germán Rosas Ortiz por todas sus valiosas sugerencias y aportaciones para la buena conclusión de este trabajo.

A mis compañeros de la SEPI por su amistad y colaboración durante todo el posgrado.

A Lilia Cruz Durán por su ardua labor e iniciativa para el funcionamiento del programa de posgrado.



ÍNDICE GENERAL

	Pág.
RESUMEN	IV
ABSTRACT.....	V
DEDICATORIAS.....	VI
AGRADECIMIENTOS.....	VII
ÍNDICE GENERAL.....	VIII
ÍNDICE DE FIGURAS	XII
ÍNDICE DE TABLAS	XIII
GLOSARIO DE TÉRMINOS	XIV
CAPÍTULO 1 INTRODUCCIÓN	1
1.1. Objetivos	1
1.1.1. Objetivo general.....	1
1.1.1. Objetivos particulares.	2
1.2. Justificación	2
1.1. Antecedentes	2
1.2. Limitaciones y alcances	4
1.3. Aportaciones	4
1.4. Estructura de la tesis	5
CAPÍTULO 2 PROGRAMACIÓN DINÁMICA.....	6
2.1. Introducción	6
2.2. El problema del viajero	7
2.3. Elementos de Programación Dinámica	8
2.4. Solución de un problema mediante programación dinámica.....	10
2.5. Despacho económico usando programación dinámica	13
CAPÍTULO 3 DESPACHO DE POTENCIA REACTIVA EN REDES DE DISTRIBUCIÓN.....	15
3.1. Introducción	15



3.2.	Formulación del problema	15
3.2.1.	Control de voltaje.....	20
3.2.2.	Límites de voltaje para la operación.	21
3.3.	Metodología empleada	22
3.4.	Despacho de capacitores	24
3.5.	Despacho de cambiadores de derivación.	29
CAPÍTULO 4 PRUEBAS Y RESULTADOS.....		32
4.1.	Descripción de la red de prueba	32
4.2.	Perfil de carga considerado.....	34
4.3.	Resultados obtenidos.....	35
CAPÍTULO 5 CONCLUSIONES Y TRABAJOS FUTUROS		46
5.1.	Conclusiones.....	46
5.2.	Recomendaciones para trabajos futuros	46
REFERENCIAS.....		48
APÉNDICE A MODELADO Y ANÁLISIS DE LOS ELEMENTOS DE DISTRIBUCIÓN.....		51
A.1.	Introducción	51
A.2.	Modelo π de la línea.....	51
A.2.1.	Impedancia serie de líneas aéreas y subterráneas	51
A.2.2.	Admitancia en derivación de líneas aéreas o subterráneas.....	52
A.3.	Modelo π exacto de la línea	53
A.4.	Modelado de transformadores trifásicos.....	55
A.5.	Modelado de cargas.....	58
A.6.	Modelado de capacitores en derivación.....	59
A.7.	Modelado de reguladores de voltaje	61
A.7.1.	Regulador monofásico Tipo A	61
A.7.2.	Regulador monofásico Tipo B.	61
A.7.3.	Reguladores trifásicos	64
A.8.	Flujos de potencia trifásicos por el método de barrido regresivo/progresivo	66
APÉNDICE B CÓDIGOS		69
B.1.	Principal.f90.....	69



B.2.	Barrido.f90.....	78
B.3.	ConstantesG2.f90.....	85
B.4.	ConstantesG3.f90.....	85
B.5.	ConstantesG14.f90.....	87
B.6.	Estructuras.f90	89
B.7.	Estructuras2.f90	91
B.8.	Estructuras3.f90	91
B.9.	Estructuras4.f90	92
B.10.	Mcons.f90.....	93
B.11.	PrintSalidaDetallada.f90.....	93
B.12.	Rutinas.f90	97
B.13.	Utileriascomplejas1.f90.....	102
B.14.	Utileriascomplejas2.f90.....	103
B.15.	Activar.f90	105
B.16.	Actualiza.f90.....	106
B.17.	Backtrackt.f90	109
B.18.	Data_despacho.f90	110
B.19.	Despachoc.f90.....	111
B.20.	Despachot.f90	115
B.21.	Gauss.f90.....	118
B.22.	Salidatotal.f90	119
B.23.	Sort.f90.....	120
B.24.	Sort2.f90.....	121
B.25.	Sort2may.f90.....	122
B.26.	Sortc.f90	123
B.27.	Tabla_de_escenarios.f90.....	123
APÉNDICE C ARCHIVO DE ENTRADA.....		125
C.1.	Formato de los datos de entrada.....	125
C.2.	Ejemplo de archivo de entrada para la red de Salama.	129
APÉNDICE D ARCHIVOS DE SALIDA.....		135



D.1.	Red de Salama con $K_c=1$ (Sin controlar el cambiador de derivaciones)	135
D.2.	Red de Salama con $K_c=2$ (Sin controlar el cambiador de derivaciones)	136
D.3.	Red de Salama con $K_c=3$ (Sin controlar el cambiador de derivaciones)	137
D.4.	Red de Salama con $K_c=4$ (Sin controlar el cambiador de derivaciones)	138
D.5.	Red de Salama con $K_c=5$ (Sin controlar el cambiador de derivaciones)	139
D.6.	Red de Salama con $K_c=6$ (Sin controlar el cambiador de derivaciones)	140
D.7.	Red de Salama con $K_c=1$	141
D.8.	Red de Salama con $K_c=2$	142
D.9.	Red de Salama con $K_c=3$	143
D.10.	Red de Salama con $K_c=4$	144
D.11.	Red de Salama con $K_c=5$	145
D.12.	Red de Salama con $K_c=6$	146



ÍNDICE DE FIGURAS

	Pág.
Figura 2.1 Grafo del problema del viajero.....	7
Figura 2.2 Trayectoria Óptima.....	8
Figura 2.3 Diagrama de una decisión	9
Figura 3.1 Subestación con alimentador de distribución.....	16
Figura 3.2 Diagrama de transición de estados de capacitores.....	18
Figura 3.3 Diagrama de flujo simplificado del programa realizado	24
Figura 3.4 Diagrama de flujo del despacho de capacitores.....	28
Figura 3.5 Posiciones del cambiador de derivaciones en la iteración 1.....	30
Figura 3.6 Posiciones del cambiador de derivaciones en la iteración 2.....	30
Figura 3.7 Diagrama de flujo para el despacho de cambiadores en derivación	31
Figura 3.8 Diagrama de flujo para el despacho de cambiadores de derivación	31
Figura 4.1 Red de prueba de 33 nodos de Salama	32
Figura 4.2 Perfil de carga considerado	34
Figura 4.3 Pérdidas de la red durante el día sin despacho.....	35
Figura 4.4 Pérdidas obtenidas con dos estrategias de control y diferentes valores de K_c . 36	
Figura 4.5 Tiempos de ejecución del despacho coordinado con diferentes valores de K_c . 37	
Figura 4.6 Despacho obtenido con una restricción $K_c=1$	38
Figura 4.7 Despacho obtenido con una restricción $K_c=2$	39
Figura 4.8 Despacho obtenido con una restricción $K_c=3$	40
Figura 4.9 Despacho obtenido con una restricción $K_c=4$	41
Figura 4.10 Despacho obtenido con una restricción $K_c=5$	42
Figura 4.11 Despacho obtenido con una restricción $K_c=6$	43
Figura 4.12 Perfil de voltaje del nodo 9 en diferentes simulaciones	44
Figura 4.13 Perfil de voltaje del nodo 26 en diferentes simulaciones	44
Figura A.1 Modelo π exacto de una línea trifásica.....	53
Figura A.2 Banco de transformación trifásico en general.....	55
Figura A.3 Banco de capacitores en estrella	60
Figura A.4 Banco de capacitores en delta	60
Figura A.5 Regulador monofásico Tipo A en posición elevador.....	61
Figura A.6 Regulador monofásico Tipo B en posición elevador.....	62



ÍNDICE DE TABLAS

	Pág.
Tabla 2.1 Problema del viajero resuelto mediante programación dinámica regresiva	12
Tabla 2.2 Problema del viajero resuelto mediante programación dinámica progresiva.....	12
Tabla 2.3 Costos de generación.....	13
Tabla 2.4 Asignación de las unidades 1 y 2	14
Tabla 2.5 Asignación de la unidad 3	14
Tabla 3.1 Rangos de Voltaje para 120 V de la norma ANSI C84.1.....	21
Tabla 3.2 Voltajes nominales y rango de voltajes de la norma ANSI para media tensión ..	22
Tabla 3.3 Representación de estados para tres capacitores.....	26
Tabla 3.4 Posibles estados sucesores en la etapa k para una red con $CT = 3$ y $KC = 2$...	26
Tabla 4.1 Datos por sección de la red de Salama	33
Tabla 4.2 Datos de los bancos de capacitores usados en la red de Salama.....	34
Tabla A.1 Matrices generalizadas para diferentes conexiones de transformadores.....	56
Tabla A.2 Ecuaciones de corriente de línea para diferentes tipos de carga.	59
Tabla A.3 Ecuaciones de voltaje y corriente para el regulador en posición elevador	62
Tabla A.4 Ecuaciones de voltaje y corriente para el regulador en posición reductor.	63
Tabla A.5 Convención de signos para a_R de la ecuación A.27	64
Tabla A.6 Matrices generalizadas para reguladores	65



GLOSARIO DE TÉRMINOS

k	Etapa de un problema.
$X(k)$	Estado X en la etapa k .
$u(k)$	Decisión en la etapa k .
$f(x(k), u(k), k)$	Ecuación de transición de estado
$l(x(k), u(k), k)$	Función del costo elemental.
$X(0)$	Estado inicial
$\Omega(X, k_0)$	Conjunto de políticas admisibles
P_{loss}	Pérdidas totales del alimentador en un periodo de 24 horas
$P_n[X_1(n), X_2(n), \dots, X_{C_i}(n), TAP_n]$	Pérdidas del alimentador en la hora n cuando el estado del banco de capacitores C_i es $X_i(n)$ y el cambiador de derivaciones está en la posición TAP_n
C_T	Número total de capacitores
TAP_n	Posición del cambiador de derivaciones en la hora n .
K_T	El límite de movimientos del cambiador de derivaciones en un día.
$X_i(n)$	Estado del capacitor i en la hora n .
\oplus	Operación OR exclusiva.
K_C	Límite de operaciones del capacitor en un día.



V^{min}	Voltaje mínimo permitido de V_j .
V_j	Voltaje del nodo j.
V^{max}	Voltaje máximo permitido de V_j .
$P_n[X_1(n), X_2(n), \dots, X_{C_t}(n)]$	Pérdidas del alimentador en la hora n cuando el estado del banco de capacitores C_i es $X_i(n)$
$P_n[TAP_n]$	Pérdidas del alimentador en la hora n cuando el cambiador de derivaciones está en la posición TAP_n
ANSI	Instituto Nacional Americano de Estándares (American National Standards Institute)
$Nesc$	Escenarios guardados por etapa
TAP	Cambiador de derivaciones del transformador de subestación.
$[z_{abc}]$	Matriz de impedancias de fase trifásica.
$[y_{abc}]$	Matriz de admitancias de fase trifásica.
V_{ag_n}	Voltaje de la fase "a" a Tierra en el nodo "n".
I_{a_n}	Corriente de la fase "a" en el nodo "n".
$[VLG_{abc}]_n$	Vector de voltajes de fase del nodo "n".
$[I_{abc}]_n$	Vector de corrientes del nodo "n".
$[a]$	Matriz generalizada "a".
$[b]$	Matriz generalizada "b".
$[c]$	Matriz generalizada "c".



$[d]$	Matriz generalizada "d".
H1, H2, H3, H4	Terminales de alta tensión de un transformador trifásico.
X1, X2, X3, X4	Terminales de baja tensión de un transformador trifásico.
A_i, B_i y d_i	Matrices constantes dependientes del tipo de conexión de los bancos de transformación.
$[W]$	Matriz constante para transformar voltajes de línea a línea a voltajes de línea a neutro.
n_i	Relación de transformación
PQ	Modelo de carga de potencia constante.
Z	Modelo de carga de impedancia constante.
I	Modelo de carga de corriente constante.
S_a, S_b, S_c	Potencias complejas de las fases a, b y c.
$\delta_a, \delta_b, \delta_c$	Ángulos de la potencia compleja de las fases a, b y c.
$\theta_a, \theta_b, \theta_c$	Ángulos de los voltajes compleja de las fases a, b y c.
$\alpha_a, \alpha_b, \alpha_c$	Ángulos de las corrientes de las cargas de las fases a, b y c.
$ IL_a , IL_b , IL_c $	Magnitud de las corrientes de las cargas de las fases a, b y c.
B_C	Susceptancia de un banco de capacitores.
IC_a, IC_b, IC_c	Corrientes de línea de un banco de capacitores.



a_R

Relación de transformación de un regulador a pasos.

$$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_m^{(k)}$$

Vector de corriente inyectada en el nodo m en la iteración k.

$$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{carga\ m}^{(k)}$$

Vector de corriente de la carga conectada en el nodo m en la iteración k.

$$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{derivación\ m}^{(k)}$$

Vector de corriente de los capacitores en derivación conectados al nodo m en la iteración k.

$$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{nm}^{(k)}$$

Vector de corrientes totales entre los nodos n y m

$$\sum \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{mp}^{(k)}$$

Sumatoria de los vectores de corrientes de todas los elementos conectados al nodo m desde un nivel superior.



CAPÍTULO 1

INTRODUCCIÓN

El control de voltaje se ha vuelto un aspecto primordial en los sistemas eléctricos de potencia, debido a que se debe mantener el perfil de voltaje dentro de límites operativos para no disminuir la vida útil de los equipos y no afectar la estabilidad de voltaje del sistema.

Para las redes de distribución, se define el control de voltaje como la aplicación de diferentes estrategias suficientes para mantener la magnitud de los voltajes nodales dentro de los límites establecidos, para diferentes condiciones operativas en el sistema. [1].

En general, la solución al problema del control de voltaje es planteada a través de la coordinación de fuentes o equipos que manejan la potencia reactiva del sistema: compensadores síncronos, reguladores de voltaje, bancos de capacitores o transformadores con cambiador de derivaciones; sin embargo, las plantas generadoras operan en puntos alejados de los centros de consumo, por lo que los sistemas de distribución de energía están generalmente imposibilitados para disponer de compensación sincrónica, de modo que el control de voltaje en sistemas de distribución es resuelto usualmente con reguladores de voltaje, bancos de capacitores y transformadores con cambiador de derivaciones. [2]

Los capacitores han sido ampliamente empleados en sistemas de distribución para lograr la reducción de pérdidas, ampliar capacidad del sistema y mejorar el perfil de voltaje del alimentador. [3]

1.1. Objetivos

1.1.1. Objetivo general

- Desarrollar una metodología para el despacho óptimo de bancos de capacitores y del cambiador de derivaciones en transformadores de subestación usando programación dinámica que minimice las pérdidas de potencia real en el sistema.



1.1.1. Objetivos particulares.

- Mantener los perfiles de voltaje de la red de distribución dentro de los límites de operación permitidos.
- Minimizar el número de operaciones de control a realizar durante el despacho propuesto.

1.2. Justificación

Debido a la naturaleza de los equipos instalados, la coordinación de fuentes de potencia reactiva en un sistema que se compone de la red de subtransmisión y circuitos de distribución de media tensión resulta en un problema de optimización no-lineal con variables enteras, cuya complejidad se incrementa rápidamente con el número de equipos instalados, con el horizonte de análisis y número de etapas consideradas.

La mayoría de los esfuerzos en la industria eléctrica se han dedicado al estudio, análisis y mejora de los sistemas de transmisión, por lo que es muy poca la atención que reciben las redes de distribución. Los esquemas de control en redes de distribución que existen actualmente no consideran una conexión entre el control de los cambiadores de derivación en subestaciones y la conexión y desconexión de bancos de capacitores a lo largo de la red de distribución, lo que ocasiona que exista descoordinación entre la operación de estos dispositivos y pérdidas en la red que pueden ser minimizadas

Mediante el empleo de técnicas de despacho de potencia reactiva se pueden reducir las pérdidas del alimentador, mejorar los perfiles de voltaje de la red de distribución y reducir el flujo de potencia reactiva en el transformador principal.

1.1. Antecedentes

En trabajos previos [4, 5, 6] la localización de capacitores se ha tratado como un problema de planeación a nivel distribución, sin embargo el control en tiempo real y el despacho de capacitores en sistemas de distribución ha sido limitado. De acuerdo a los trabajos [7] y [8], usualmente se considera que las cargas a lo largo del alimentador cambian proporcionalmente durante el día. Esto se debe a la falta de información en tiempo real de los sistemas de distribución. Muchos sistemas usan un horario fijo como estrategia para la conexión y desconexión de bancos de capacitores, por ejemplo, conectar todos los



capacitores durante las horas pico de 8 am a 6 pm en la operación diaria. Estos métodos distan mucho de ser óptimos, dado que las cargas del alimentador varían con la hora del día, el día de la semana y temporada del año. Sin embargo este método es aceptable cuando la información de la carga no está disponible a los operadores del sistema [3].

En [3] se hace una aproximación basada en programación dinámica para determinar el despacho de los capacitores para las próximas 24 horas, esta aproximación puede considerar cualquier patrón de carga diaria suponiendo que se tenga pronosticado. El método empleado analiza todas las combinaciones posibles para cada etapa y determina el estado predecesor para cada estado y al final realiza un barrido regresivo para determinar el itinerario óptimo para el despacho de capacitores.

En [8] se usa un procedimiento basado en algoritmos genéticos para determinar los niveles de carga a usar durante el día y el itinerario del despacho. El modelo de carga que se emplea incluye el 50% de carga como potencia constante y el otro 50% de carga como impedancia constante. El modelo se basa en una descomposición de la curva de carga por niveles, donde cada nivel tiene una duración determinada de tiempo de acuerdo al pronóstico de carga realizado. Se incluyen diferentes simulaciones donde se permite operar desde 2 hasta 10 veces los bancos de capacitores por día.

En [9] se realiza el despacho de capacitores y cambiadores en derivación bajo carga basándose en el pronóstico de carga del transformador principal y su voltaje en el bus primario usando una aproximación basada en programación dinámica. El algoritmo guarda las tres posiciones del cambiador de derivación que hacen que el voltaje secundario de la subestación se acerque lo más posible al valor objetivo, con estos valores se analizan las posibles combinaciones de estado del banco de capacitores junto con las posiciones del cambiador de derivaciones y se restringe el número de estados almacenados a solo 36 desde la etapa 2 hasta la 24.

En [10] se plantea el desarrollo de una red neuronal de tres capas para generalizar el problema. Como valores de entrada se usa P , Q y V tomados de seis puntos de medición en el circuito de distribución y después se agregan las variables de salida en forma de retroalimentación. El número de variables de salida depende del número de elementos de control contemplados, ya que a cada elemento de control se le asigna una salida de la red neuronal. Para probar el control neuronal se usa una simulación de Montecarlo con 12000 unidades de medición para entrenamiento. Se simulan cinco patrones de carga que van desde una curva sinusoidal suavizada hasta una secuencia aleatoria, mientras que las cargas se modelan como potencia compleja constante.

En [11] se descompone el problema de control de fuentes de potencia reactiva en dos niveles, uno a nivel subestación y otro a nivel alimentador. Se usa una aproximación de programación dinámica y un algoritmo de control de lógica difusa para atender cada



problema respectivamente. Se guardan tres posiciones para el cambiador de derivaciones para cada etapa del primer problema. Al resolver el segundo problema se usa una estrategia de "fuzzificación" con el método de Mamdani.

En [12] se realiza el despacho de los cambiadores de derivación del transformador en la subestación, bancos de capacitores en subestación y bancos de capacitores a lo largo del alimentador basándose en el pronóstico de cargas horarias de cada sección del alimentador y del voltaje del bus primario. Primero se obtiene el estado de cada bancos de capacitores en los alimentadores para cada hora usando programación dinámica, y se guarda un número determinado de itinerarios con el menor costo en cada etapa, posteriormente se hace el despacho de los cambiadores de derivación bajo carga.

En [13] se propone una formulación multi objetivo llamada método de ponderación normalizada que incluye cuatro objetivos, potencia reactiva, pérdidas en el alimentador, desviación de voltaje (minimiza la máxima desviación y la desviación total de los voltajes en el alimentador) y perfil de voltaje. Este trabajo incluye cambiadores de derivación y bancos de capacitores a lo largo del alimentador y usa algoritmos genéticos para optimización.

1.2. Limitaciones y alcances

El método que este trabajo propone es una herramienta para la operación de redes de distribución. La metodología propuesta se aplica en una red de prueba, donde se supone que la carga varía proporcionalmente durante el día de acuerdo a un porcentaje de la carga máxima para cada hora del día. Sin embargo, la metodología es compatible con el uso de rutinas de pronóstico de carga.

1.3. Aportaciones

Se desarrolló un programa en lenguaje FOTRAN 90 basado en un algoritmo de programación dinámica para obtener la coordinación de los bancos de capacitores en alimentadores de media tensión con la posición de los cambiadores de derivación del transformador de subestación en una red de distribución; lo anterior para cada etapa de un periodo de estudio de un día dividido en 24 etapas de 1 hora cada uno, a fin de minimizar las pérdidas de potencia real en el alimentador a lo largo del día con un mínimo de operaciones a realizar.



1.4. Estructura de la tesis

En esta sección se describe de forma general cada uno de los capítulos que componen este trabajo de tesis.

Capítulo 1: Introducción. Se plantea el problema de control de voltaje en redes de distribución, se delimita el trabajo a realizar, se menciona su objetivo, su justificación y su aporte, además de dar una descripción de la estructura de la tesis.

Capítulo 2: Programación dinámica. Se fundamenta los conceptos y el desarrollo de la programación dinámica, que es la base del algoritmo desarrollado.

Capítulo 3: Despacho de potencia reactiva en redes de distribución. Se formula de forma matemática el problema del despacho de potencia reactiva planteado y se explica la metodología empleada para resolver el problema.

Capítulo 4: Pruebas y resultados. Se realizan simulaciones en un sistema de prueba con el programa desarrollado.

Capítulo 5: Conclusiones. Se analizan los resultados obtenidos en el capítulo 4 y en base a esto se indican las conclusiones a las que se llegó.



CAPÍTULO 2

PROGRAMACIÓN DINÁMICA.

2.1. Introducción

Para resolver un problema en el mundo real, no siempre se puede trabajar directamente en el, sino que se hace a través de un modelo. En ese modelo, se contemplan las características fundamentales del mismo, y se eliminan todas las que no tienen tanta importancia en el desarrollo del problema. La solución al problema real será mejor dependiendo de la exactitud del modelo.

En algunos casos, modelar el problema supone, de alguna forma, simplificarlo; incluso en casos en el que el problema esté formulado computacionalmente de forma completa, hace falta hacer aproximaciones para hacer su resolución más simple usando un método determinado. Siempre hay que tener en cuenta, sin embargo, que modelos diferentes del problema darán soluciones diferentes del mismo. Pero generalmente encontrar una solución aproximada a un modelo exacto será mejor que encontrar una solución exacta a un modelo aproximado.

Un problema que se presenta al tratar de resolver problemas de optimización combinatoria, es la explosión combinatoria, la cual consiste en que hay un número muy grande de alternativas que son candidatas a solución para dicho problema. Para lo cual, si se tienen que examinar todas o la mayoría de éstas, tomaría mucho tiempo el poder hacerlo.

En algunos problemas, donde el espacio de búsqueda es suficientemente pequeño y numerable, se pueden usar métodos de búsqueda exhaustiva, en donde se examinan todas las soluciones posibles, y se toma como buena aquella que tenga el valor óptimo de la función objetivo.

El problema con la búsqueda exhaustiva no sólo es que a menudo no es factible, sino que tiene un mal comportamiento con respecto al aumento del tamaño del espacio de búsqueda; y el hecho de que necesite una cantidad de memoria de la computadora proporcional al tamaño del problema, hace que no se pueda aplicar en la mayor parte de los casos



La programación dinámica se aplica a problemas de optimización, en donde existen muchas soluciones posibles, donde cada solución tiene un valor y se desea encontrar una solución con el valor óptimo (ya sea máximo o mínimo). El algoritmo de programación dinámica divide el problema original en subproblemas, y resuelve un subproblema una vez, y entonces se guarda la respuesta en una tabla, evitando el trabajo de volver a calcular la respuesta cada vez que el subproblema es encontrado. [14]

2.2. El problema del viajero

Uno de los problemas más comunes para ejemplificar el uso de programación dinámica es el problema del viajero [15]. Un motociclista hace un viaje desde la ciudad A hasta la ciudad J, pero no existe una carretera que las una directamente, por lo que tiene que pasar por otras ciudades. El objetivo del problema es encontrar la ruta desde la ciudad A hasta la ciudad J con el menor costo. Estas ciudades y los costos asociados a cada tramo del viaje se muestran en la figura 2.1.

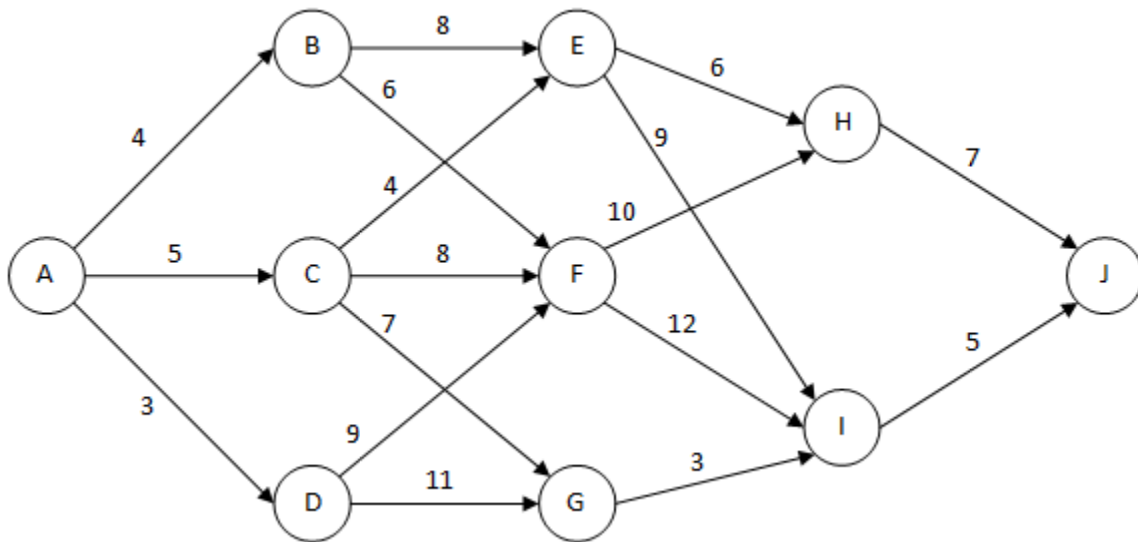


Figura 2.1 Grafo del problema del viajero

Inicialmente el viajero considera las rutas con menor costo en cada etapa de la ruta, por lo que la trayectoria que selecciona es A-D-F-H-J, que suma un costo total de 29 unidades ($=3+9+10+7$), sin embargo, observa que para llegar de la ciudad A hasta la ciudad F es más económico pasar por la ciudad B en lugar de la D, por lo que se decide analizar todas las posibles trayectorias entre A y J calculando los costos totales de cada una y se decide por la trayectoria A-C-G-I-J, cuyo costo total sería de 20 unidades.

Analizar cada una de las trayectorias no es una técnica eficiente. Si se intentara resolver un problema similar con cinco ciudades en cada etapa se generarían 125 trayectorias.

2.3. Elementos de Programación Dinámica

Las técnicas de programación dinámica están basadas en el principio de optimalidad de Bellman. Ese principio hace que sea posible construir códigos computacionales capaces de retener las características relevantes del procedimiento de enumeración completa, con un esfuerzo computacional menor. El principio de optimalidad puede ser enunciado como:

“Dada una trayectoria óptima del punto A al punto C, cualquier parte de la trayectoria a partir de un punto intermedio B hasta C es la trayectoria óptima de B hasta C” [16].

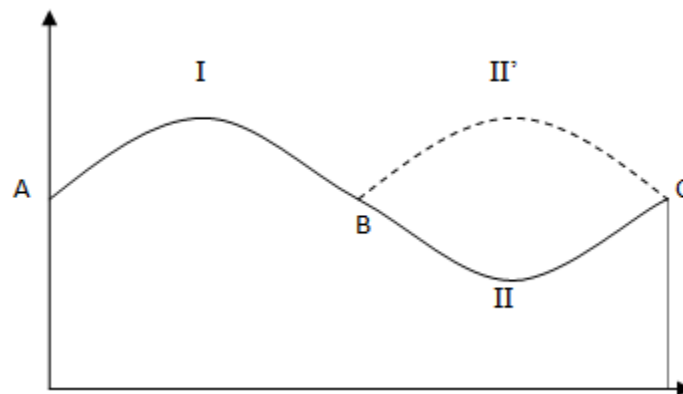


Figura 2.2 Trayectoria Óptima

En la figura 2.2, si la trayectoria I-II es el camino óptimo desde A hasta C, por el principio de optimalidad la trayectoria II es el camino óptimo desde B hasta C. La prueba por contradicción es inmediata en este caso: Asuma que el camino II' sea el camino óptimo entre B y C. Entonces la trayectoria I-II', que contradice el hecho de que I-II es la trayectoria óptima desde A hasta C. Luego II es el camino de menor costo entre B y C.

Antes de desarrollar la programación dinámica es importante definir algunos conceptos [16].

El punto en el que se inicia o concluye un desplazamiento se le llama etapa. Formalmente la *etapa* es una variable discreta k que determina el orden en que ocurren modificaciones en el sistema.

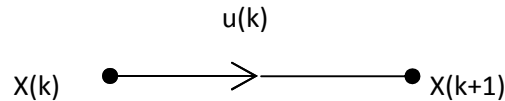


$$k = 0, 1, 2, \dots, N \quad (2.1)$$

$$K = \{0, 1, 2, \dots, N\} \quad (2.2)$$

En cada etapa del problema puede haber uno o más estados que se deriven del desarrollo del problema. El *estado* es una variable X que describe completamente el sistema en una etapa determinada. Puede suceder que un mismo estado pueda ocurrir en diferentes etapas, por lo que se denota como $X(k)$ para asociar el estado X con la etapa k .

Para cambiar de un estado a otro en la etapa siguiente, se debe tomar una decisión. La *decisión* es una variable u que se aplica en un estado $X(k)$ para llevarlo al estado $X(k+1)$ y se denota como $u(k)$.



Al pasar de una etapa a otra es probable que no se puedan alcanzar ciertos estados, el *conjunto de estados factibles* son los estados que X puede alcanzar en la etapa k :

$$X(k) = \{x_1, x_2, \dots\} \quad (2.3)$$

De manera análoga, dado un determinado estado del sistema $X(k)$, existe un *conjunto de decisiones admisibles* que pueden ser tomadas a partir de $X(k)$.

Para relacionar el estado en una etapa con la decisión aplicada y el nuevo estado resultante se utiliza una ecuación de transición de estado, que es la ecuación recursiva del sistema.

$$f: (x(k), u(k), k) \rightarrow f(x(k), u(k), k) = x(k + 1) \quad (2.4)$$

Al aplicar una decisión en la etapa k desde el estado $x(k)$ para llegar al estado $x(k+1)$ se genera un costo elemental resultante de esa decisión, cuya función está dada por:

$$l: (x(k), u(k), k) \rightarrow l(x(k), u(k), k) \quad (2.5)$$

El *estado inicial* del sistema es la condición en que el sistema se encuentra en la etapa inicial. Ese es un estado único, o sea:



$$X(0) = \{x(0)\} \quad (2.6)$$

Una *política admisible* aplicada a partir de $X(K_0)$ es una secuencia de decisiones:

$$[u(k)]_{k_0}^{N-1} = \{u(k_0), u(k_0 + 1), \dots, u(N - 1)\} \quad (2.7)$$

De tal modo que si se define:

$$x(k + 1) = f(x(k), u(k), k) \quad \forall \quad k = k_0, \dots, N - 1 \quad (2.8)$$

entonces:

$$u(k) \in U(x(k), k) \quad \forall \quad k = k_0, \dots, N - 1 \quad (2.9)$$

$$x(k) \in X(k) \quad \forall \quad k = k_0, \dots, N - 1 \quad (2.10)$$

El conjunto de políticas admisibles es representado por $\Omega(X, k_0)$.

Una política admisible produce una trayectoria que es el conjunto de estados alcanzados por la aplicación de la política.

La función objetivo o criterio de optimización esta dado por:

$$J: (x(k_0), [u(k)]_{k_0}^{N-1}, k_0) \rightarrow \sum_{k=k_0}^{N-1} (x(k), u(k), k) \quad (2.11)$$

Con base en todos estos elementos, se puede ya formular el problema de programación dinámica como: “Encontrar, si existe, una política admisible que aplicada al estado inicial, lleve el sistema a la etapa final optimizando la función objetivo” [17].

2.4. Solución de un problema mediante programación dinámica

Para resolver un problema de programación dinámica se debe:

- Identificar etapas, estados y variables de decisión:
 - Cada etapa debe tener asociada una o más decisiones (problema de optimización), cuya dependencia de las decisiones anteriores está dada exclusivamente por las variables de estado.



- Cada estado debe contener toda la información relevante para la toma de decisión asociada al período.
- Las variables de decisión son aquellas sobre las cuales debemos definir su valor a modo de optimizar el beneficio acumulado y modificar el estado de la próxima etapa.
- Describir las ecuaciones de recurrencia: Nos deben indicar como se acumula la función objetivo y como varían las funciones de estado de una etapa a otra.
- Resolución: Debemos optimizar cada subproblema por etapas en función de los resultados de la resolución del subproblema siguiente. Notar que las ecuaciones de recurrencias estén bien definidas.

La programación dinámica puede ser aplicada de dos formas, regresiva o progresiva. En la programación dinámica regresiva se analiza el problema desde la última etapa y se busca para cada estado la mejor decisión aplicable en la etapa anterior que conduzca a ese estado, mientras que en la programación dinámica progresiva, se busca desde la primera etapa cual es la mejor decisión aplicable para llegar a la siguiente etapa.

Aunque el problema del viajero mostrado es un problema simple, el algoritmo de programación dinámica se adapta perfectamente e incluso puede hacerlo para problemas mucho más complicados.

Aplicando la programación dinámica al problema del viajero, se usa la siguiente nomenclatura en las tablas 2.1 y 2.2

- k = etapa analizada
- x = estado factible en k
- u = decisión en la etapa k y en el estado x
- x' = estado alcanzado por la aplicación de la decisión u al estado x
- l = costo elemental de la decisión u aplicada al estado x en la etapa k
- J' = costo adicional mínimo a partir de x'
- J = costo adicional mínimo a partir de x
- u* = mejor decisión aplicable en x



Tabla 2.1 Problema del viajero resuelto mediante programación dinámica regresiva

x	u	x'	l	\bar{J}'	$J = l + \bar{J}'$	\bar{J}	u^*
k=3							
H	ir hacia J	J	7	0	7	7	ir hacia J
I	ir hacia J	J	5	0	5	5	ir hacia J
k=2							
E	ir hacia H	H	6	7	13	13	ir hacia H
	ir hacia I	I	9	5	14		
F	ir hacia H	H	10	7	17	17	ir hacia H
	ir hacia I	I	12	5	17		
G	ir hacia I	I	3	5	8	8	ir hacia I
k=1							
B	ir hacia E	E	8	13	21	21	ir hacia E
	ir hacia F	F	6	17	23		
C	ir hacia E	E	4	13	17	15	ir hacia G
	ir hacia F	F	8	17	25		
	ir hacia G	G	7	8	15		
D	ir hacia F	F	9	17	26	19	ir hacia G
	ir hacia G	G	11	8	19		
k=0							
A	ir hacia B	B	4	21	25	20	ir hacia C
	ir hacia C	C	5	15	20		
	ir hacia D	D	3	19	22		
La trayectoria óptima es A-C-G-I-J con un costo total de 20 unidades.							

Tabla 2.2 Problema del viajero resuelto mediante programación dinámica progresiva

x'	u	x	l	\bar{J}'	$J = l + \bar{J}'$	\bar{J}	u^*
k=0							
B	venir de A	A	4	0	4	4	venir de A
C	venir de A	A	5	0	5	5	venir de A
D	venir de A	A	3	0	3	3	venir de A
k=1							
E	venir de B	B	8	4	12	9	venir de C
	venir de C	C	4	5	9		
F	venir de B	B	6	4	10	10	venir de B
	venir de C	C	8	5	13		
	venir de D	D	9	3	12		
G	venir de C	C	7	5	12	12	venir de C
	venir de D	D	11	3	14		
k=2							
H	venir de E	E	6	9	15	15	venir de E
	venir de F	F	10	9	19		
I	venir de E	E	9	9	18	15	venir de G
	venir de F	F	12	9	21		
	venir de G	G	3	12	15		



k=3							
J	venir de H	H	7	15	22		
	venir de I	I	5	15	20	20	venir de I
La trayectoria óptima es A-C-G-I-J con un costo total de 20 unidades.							

En la programación dinámica cada problema tiene sus características y requiere un planteamiento particular para solucionarlo. Sin embargo, una vez que se identifica que un problema puede ser solucionado por programación dinámica y se identifican sus elementos, la preparación del algoritmo no difiere del examinado para el problema del viajero.

2.5. Despacho económico usando programación dinámica

Para ejemplificar un poco más el uso de la programación dinámica en los sistemas eléctricos de potencia, se presenta el siguiente ejemplo [18].

Se consideran tres unidades en el sistema; todas están en servicio. Sus características de entrada – salida no son suaves o convexas Sus datos se muestran en la tabla 2.3

Tabla 2.3 Costos de generación

Niveles de potencia (MW) P1 = P2 = P3	Costos (MBtu/h)		
	F1	F2	F3
0	∞	∞	∞
50	810	750	806
75	1355	1155	1108.5
100	1460	1360	1411
125	1772.5	1655	1704.5
150	2085	1950	1998
175	2427.5	∞	2358
200	2760	∞	∞
225	∞	∞	∞

La demanda total es $D=310$ MW. Esta demanda no coincide con los datos de manera exacta, así que se necesita interpolar entre los valores más cercanos que están disponibles entre estos datos, 300 y 325 MW.

Asignando las unidades 1 y 2, se debe encontrar el mínimo costo para la función 2.12 en el rango permitido de generación de P_2 y para $100 \leq D \leq 350$ MW.



$$f_2 = F_1(D - P_2) + F_2(P_2) \quad (2.12)$$

La búsqueda de datos se da en la tabla 2.4. Se necesita guardar el costo de satisfacer cada valor de D que sea mínimo y el nivel de carga en la unidad 2 para cada nivel de demanda.

Tabla 2.4 Asignación de las unidades 1 y 2

D	$F_1(D)$	$P_2 = 0$	50	75	100	125	150	f_2	P_2^*
(MW)	(MBtu / h)	$F_2(P_2) = \infty$	750	1155	1360	1655	1950	(MBtu / h)	(MW)
0	∞	∞	∞	∞	∞	∞	∞	∞	
50	810	∞	∞	∞	∞	∞	∞	∞	
75	1355	∞	∞	∞	∞	∞	∞	∞	
100	1460	∞	<u>1560</u>	∞	∞	∞	∞	1560	50
125	1772.5	∞	2105	<u>1965</u>	∞	∞	∞	1965	75
150	2085	∞	2210	2510	<u>2170</u>	∞	∞	2170	100
175	2427.5	∞	3177.5	2615	2715	<u>2465</u>	∞	2465	125
200	2760	∞	2834	2927.5	2820	3010	<u>2760</u>	2760	150
225	∞	∞	3177.5	3240	3125	<u>3115</u>	3305	3115	125
250	∞	∞	3510	3582.5	3445	3427	<u>3410</u>	3410	150
275	∞	∞	∞	3915	3787.5	3740	<u>3722.5</u>	3722.5	150
300	∞	∞	∞	∞	4120	4082.5	<u>4035</u>	4035	150
325	∞	∞	∞	∞	∞	4415	<u>4377.5</u>	4377.5	150
350	∞	∞	∞	∞	∞	∞	<u>4710</u>	4710	150

Ahora se debe minimizar la función 2.13 para $50 \leq P_3 \leq 175 \text{ MW}$ y $D = 300$ y 325 MW .

$$f_3 = f_2(D - P_3) + F_3(P_3) \quad (2.13)$$

Asignar la tercera unidad para los dos diferentes niveles de demanda solo se requiere dos filas en la tabla 2.5

Tabla 2.5 Asignación de la unidad 3

D	$f_2(D)$	$P_3 = 0$	50	75	100	125	150	175	f_3	P_3^*
(MW)	(MBtu / h)	$F_3(P_3) = \infty$	806	1108.5	1411	1704.5	1998	2358	(MBtu / h)	(MW)
300	4035	∞	4216	4223.5	4171	4169.5	<u>4168</u>	4323	4168	150
325	4377.5	∞	4528.5	4518.5	4526	4464	<u>4463</u>	4528	4463	150

Por lo tanto entre los niveles de demanda de 300 y 325 MW, la unidad marginal es la unidad 2 (Esto es, toma toda la demanda adicional entre 300 y 325 MW). Se puede, por lo tanto, interpolar para encontrar el costo a un nivel de carga de 310 MW, o a un nivel de salida de la unidad 2 de 110 MW, El resultado de un nivel de demanda de 310 MW es:

$$P_1 = 50, P_2 = 110, \text{ y } P_3 = 150 \text{ para un costo total de } 4286 \text{ MBtu/h}$$



CAPÍTULO 3

DESPACHO DE POTENCIA REACTIVA EN REDES DE DISTRIBUCIÓN

3.1. Introducción

El control de voltaje y de potencia reactiva en un sistema de distribución es muy importante para la operación diaria. Este control puede disminuir las pérdidas de potencia real en el sistema si se realiza un despacho adecuado de las fuentes de potencia reactiva en la red. Los dispositivos que pueden ser controlados de forma manual o automática son los capacitores a lo largo del alimentador y los cambiadores de derivación en el transformador de distribución.

El despacho de potencia reactiva se debe basar en un pronóstico de carga para un horizonte de 24 horas, así mismo se debe contar con los datos de la configuración de la red y las características de sus elementos para poder minimizar las pérdidas totales en el alimentador para el periodo de estudio.

Las restricciones que se deben tomar en cuenta son el máximo número de operaciones de conexión o desconexión para los capacitores, el número máximo de movimientos en la posición de los cambiadores de derivación del transformador en subestación y los límites de voltaje del alimentador.

3.2. Formulación del problema

Para ejemplificar la formulación se considera el sistema mostrado en la figura 3.1, que consta de una subestación con cambiador de derivación y cuatro bancos de capacitores.

Se supone que el alimentador tiene una carga que varía durante el día y que puede ser obtenida mediante un pronóstico de carga horaria para un horizonte de 24 horas.

Cada uno de los bancos de capacitores (C_1 , C_2 , C_3 , C_4) tiene la capacidad de ser conectado o desconectado individualmente de la red, estas operaciones son independientes entre ellas y a su vez, son independientes de los movimientos en los cambiadores de derivación

en la subestación. Todas estas operaciones pueden ser realizadas por un operador de forma remota.

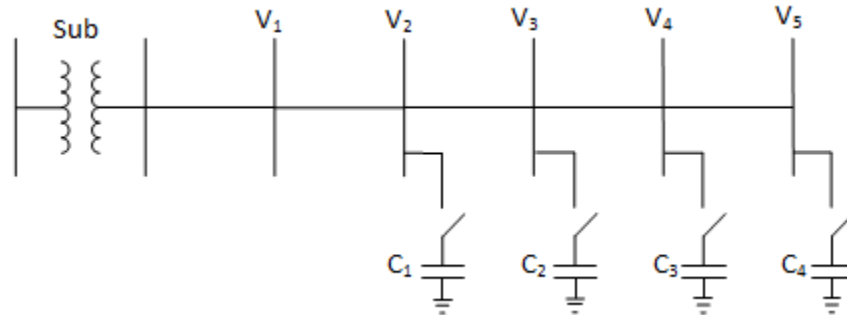


Figura 3.1 Subestación con alimentador de distribución.

El problema del despacho de potencia reactiva radica en determinar la posición más adecuada del cambiador de derivaciones en cada hora del día y el momento para conectar o desconectar los bancos de capacitores de forma tal que las pérdidas totales sean minimizadas y los voltajes se encuentren dentro de los límites de operación.

También es importante aclarar que debido a que las operaciones de conexión o desconexión de los bancos de capacitores y los movimientos de los cambiadores de derivación requieren de dispositivos mecánicos para realizarse, estos sufren un desgaste que los va deteriorando, por lo cual es necesario limitar también el número máximo de operaciones que se pueden realizar en cada dispositivo de control.

De esta forma, el despacho se debe formular como un problema de optimización, donde se debe obtener el estado de los elementos de control:

Sea TAP_n la posición del cambiador de derivaciones en la hora n :

$$TAP_n = \{-16, -15, \dots, 0, 1, \dots, 15, 16\} \quad (3.1)$$

Sea $X_i(n)$ para $(i = 1, 2, \dots, C_T)$ el estado del capacitor C_i en la hora n :

$$X_i(n) = \begin{cases} X_i(n) = 0, & \text{si } C_i \text{ esta desconectado en la hora } n \\ X_i(n) = 1, & \text{si } C_i \text{ esta conectado en la hora } n \end{cases} \quad (3.2)$$

El problema de optimización consiste en determinar el estado óptimo de $X_i(n)$ ($i = 1, 2, \dots, C_T$; $n = 1, 2, \dots, 24$) para cada capacitor C_i en cada hora n , tal que, las pérdidas totales P_{loss} durante todo el periodo de estudio sean minimizadas y estén sujetas a los límites de voltaje y restricción en el número máximo permitido de operaciones de



cada capacitor y el número máximo de movimientos permitidos del cambiador de derivaciones.

La formulación matemática del problema puede ser puesta en los siguientes términos:

Minimizar:

$$P_{loss} = \sum_{n=1}^{24} P_n [X_1(n), X_2(n), \dots, X_{C_t}(n), TAP_n] \quad (3.3)$$

Sujeto a:

$$\sum_{n=1}^{24} |TAP_n - TAP_{n-1}| \leq K_T \quad (3.4)$$

$$\sum_{n=1}^{24} |X_i(n) \oplus X_i(n-1)| \leq K_C \quad i = 1, 2, \dots, C_t \quad (3.5)$$

$$V^{min} \leq V_j \leq V^{max} \quad j = 1, 2, \dots, N \quad (3.6)$$

Donde:

P_{loss}	= Pérdidas totales del alimentador en un periodo de 24 horas
$P_n [X_1(n), X_2(n), \dots, X_{C_t}(n), TAP_n]$	= Pérdidas del alimentador en la hora n cuando el estado del banco de capacitores C_i es $X_i(n)$ y el cambiador de derivaciones está en la posición TAP_n .
C_T	= Número total de capacitores
TAP_n	= Posición del cambiador de derivaciones en la hora n.
K_T	= Límite de movimientos del cambiador de derivaciones en un día.
$X_i(n)$	= Estado del capacitor i en la hora n.
\oplus	= Operación OR exclusiva, $X_i(n) \oplus X_i(n-1) = 1$, sí $X_i(n) \neq X_i(n-1)$, $X_i(n) \oplus X_i(n-1) = 0$, sí $X_i(n) = X_i(n-1)$.
K_C	= Límite de operaciones del capacitor en un día.
V^{min}	= Voltaje mínimo permitido de V_j .



- V_j = Voltaje del nodo j .
- V^{max} = Voltaje mínimo permitido de V_j .
- N = Número de nodos del sistema.

De la ecuación 3.3 se observa que es necesario analizar las combinaciones resultantes de los posibles estados de cada banco de capacitores y la posición del cambiador de derivaciones durante el día para seleccionar una trayectoria que incluya las combinaciones que en conjunto sumen las menores pérdidas a lo largo del día.

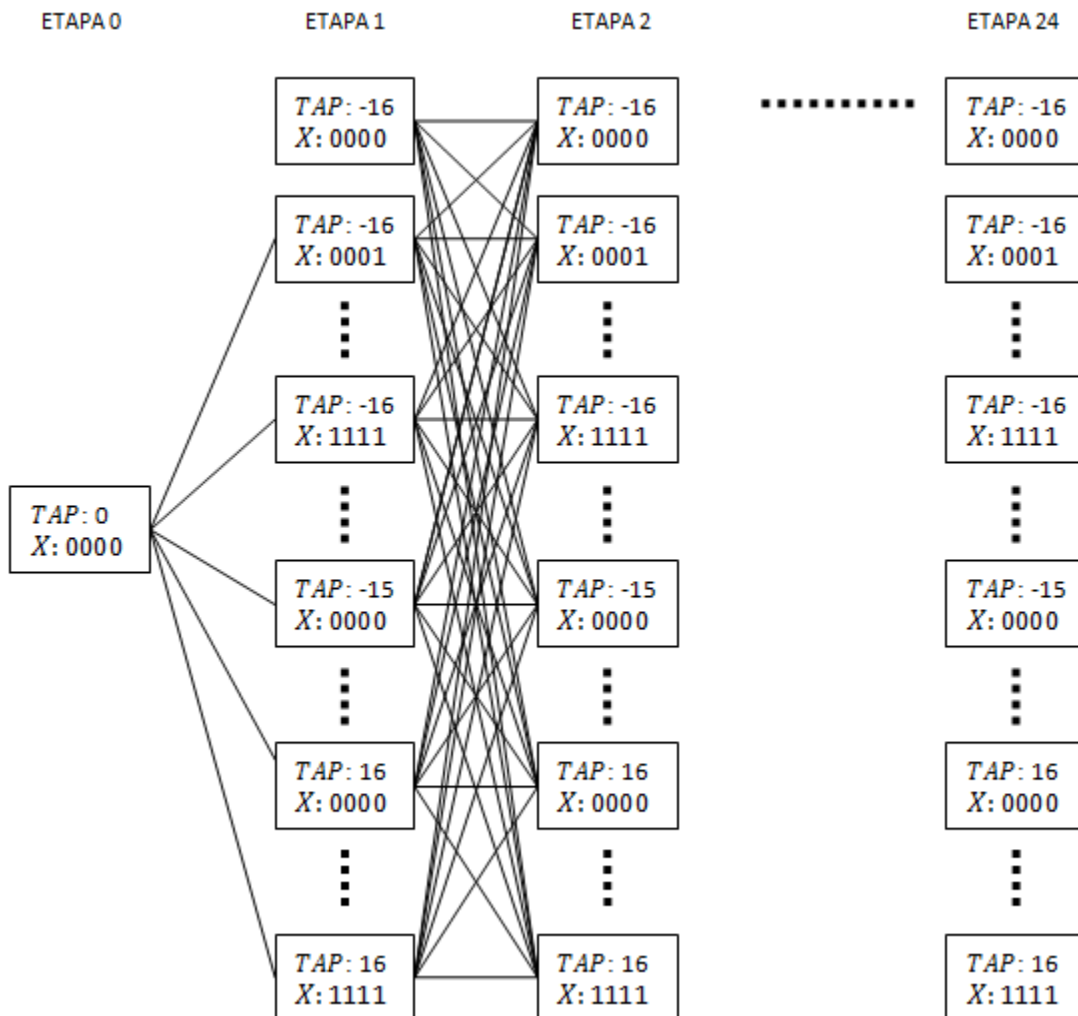


Figura 3.2 Diagrama de transición de estados de capacitores.



Para el sistema de la figura 3.1, se observa que cuenta con cuatro bancos de capacitores y un cambiador de derivaciones con 33 posiciones, por lo que existen $2^4 \cdot 33 = 16 \cdot 33 = 528$ posibles combinaciones de estados en cada etapa (hora) del problema. Estos estados pueden mostrarse en un diagrama de transición de estados [9] como se ve en la figura 3.2 para poder visualizar las posibles trayectorias que se generan para este problema. Se considera que previo a la etapa 1, los capacitores se encuentran desconectados y el cambiador de derivaciones se encuentra en la posición 0, y a partir de la etapa 1 cada banco de capacitores puede operarse de manera independiente al igual que el cambiador de derivaciones por lo que se generan $(258)^{24} (\cong 7.56 \times 10^{57})$ posibles trayectorias a revisar para las 24 etapas.

Analizar cada una de las trayectorias posibles requiere de un gran esfuerzo computacional, por lo que es necesario implementar un algoritmo que reduzca el número de combinaciones.

Una forma de reducir el número de combinaciones a analizar es descomponer el problema de coordinación en dos problemas que se pueden resolver por separado, uno a nivel alimentador (con los bancos de capacitores) y otro a nivel subestación (con el cambiador de derivación), debido a que las áreas en las que los voltajes se ven influenciados por las operaciones de los dispositivos de control son limitadas.

De esta forma se divide la coordinación en dos problemas que se resuelven por separado y cuyas soluciones recursivas dan una solución coordinada.

La formulación matemática del problema a nivel subestación puede ser puesta en los siguientes términos:

Minimizar:

$$P_{loss} = \sum_{n=1}^{24} P_n[TAP_n] \quad (3.7)$$

Sujeto a las restricciones expuestas previamente en las ecuaciones 3.4 y 3.6.

$$\sum_{n=1}^{24} |TAP_n - TAP_{n-1}| \leq K_T \quad (3.4)$$

$$V^{min} \leq V_j \leq V^{max} \quad j = 1, 2, \dots, N \quad (3.6)$$

Donde:

$P_n[TAP_n]$ = Pérdidas del alimentador en la hora n cuando el cambiador de derivaciones está en la posición TAP_n .



De la misma forma el problema a nivel alimentador se puede expresar de la siguiente manera

Minimizar:

$$P_{loss} = \sum_{n=1}^{24} P_n [X_1(n), X_2(n), \dots, X_{C_t}(n)] \quad (3.8)$$

Sujeto a las restricciones expuestas previamente en las ecuaciones 3.5 y 3.6.

$$\sum_{n=1}^{24} |X_i(n) \oplus X_i(n-1)| \leq K_C \quad i = 1, 2, \dots, C_t \quad (3.5)$$

$$V^{min} \leq V_j \leq V^{max} \quad j = 1, 2, \dots, N \quad (3.6)$$

Donde

$P_n [X_1(n), X_2(n), \dots, X_{C_t}(n)]$ = Pérdidas del alimentador en la hora n cuando el estado del banco de capacitores C_i es $X_i(n)$

3.2.1. Control de voltaje

Para mantener los circuitos de distribución en límites permisibles existen diversas formas de proveer un control de voltaje [2], como lo son:

- Uso de reguladores de voltaje en generación.
- Aplicación de equipo regulador de voltaje en subestaciones de distribución.
- Instalación de capacitores en subestaciones de distribución.
- Balancear las cargas en los alimentadores primarios.
- Incrementar el calibre del conductor del alimentador.
- Cambiar las secciones monofásicas del alimentador a secciones bifásicas o trifásicas.
- Transferir cargas a nuevos alimentadores.
- Instalar nuevas subestaciones y alimentadores primarios.
- Incrementar el nivel de voltaje.
- Instalación de reguladores de voltaje en alimentadores primarios.
- Instalación de capacitores en derivación en alimentadores primarios.
- Instalación de capacitores en serie en alimentadores primarios.

Sin embargo, estos puntos están más dirigidos a una etapa de planeación de la red, en el caso de la operación de la red se requieren estrategias que coordinen las fuentes de potencia reactiva del sistema, que en el caso de la red de distribución estos elementos son



los bancos de capacitores y los transformadores con cambiador de derivación y reguladores de voltaje.

Gracias a que los bancos de capacitores tienen la capacidad no solo de mejorar el perfil de voltaje sino también de aportar flujo de potencia reactiva a la red son los dispositivos más usados en el control de voltaje en sistemas de distribución, sin embargo se pueden mejorar los resultados si se coordina la operación de bancos de capacitores junto con los cambiadores de derivación en los transformadores de subestación.

3.2.2. Límites de voltaje para la operación.

Generalmente se ocupa la norma ANSI C84.1 [19] para determinar los límites de voltaje permitidos en la red. Este estándar especifica rangos operacionales aceptables en dos ubicaciones en sistemas eléctricos de potencia:

Voltaje de servicio: Es el voltaje en el punto donde el sistema eléctrico del proveedor y el del usuario se interconectan. Normalmente esto es en el medidor. Este voltaje es responsabilidad del proveedor.

Voltaje de utilización: Voltaje en las terminales de línea del equipo de utilización. Este voltaje es responsabilidad del usuario.

La norma de ANSI especifica caídas de voltaje dentro de los centros de consumo, así que el voltaje de servicio requerido es más estricto que el de utilización. La misma norma ANSI define dos rangos de voltaje. Una simplificación de los rangos de voltaje es:

Rango A: Los sistemas del suministro eléctrico deben ser diseñados y operados de modo que la mayoría de los voltajes de servicio estén dentro de los límites especificados para el Rango A. La ocurrencia de voltajes fuera de estos límites debe ser poco frecuente.

Rango B: Son voltajes arriba y debajo de los límites del Rango A. Cuando estos voltajes ocurran, se emprenderán las medidas correctivas dentro de un tiempo razonable para mejorar los voltajes para llegar al Rango A.

Para sistemas de baja tensión la norma considera los valores mostrados en la tabla 3.1.

Tabla 3.1 Rangos de Voltaje para 120 V de la norma ANSI C84.1

Rango	Voltaje de servicio		Voltaje de Utilización	
	Mínimo	Máximo	Mínimo	Máximo
A	114 (-5%)	126 (+5%)	110 (-8.3%)	125 (+4.2%)
B	110 (-8.3%)	127 (+5.8%)	106 (-11.7%)	127 (+5.8%)



En el caso de media tensión (definida por la norma para voltajes de 1kV a 100kV) se dan límites más estrictos para los rangos A y B. En el caso del Rango A es de -2.5 % a +5 % y para el rango B es de -5 % a +5.8 % como se deduce de la tabla 3.2.

Tabla 3.2 Voltajes nominales y rango de voltajes de la norma ANSI para media tensión

VOLTAGE CLASS	Nominal System Voltage			Voltage		
	(Note a)			Maximum	Minimum	
	2-wire	3-wire	4-wire	Utilization and Service Voltage (Note c)	Service Voltage	Utilization Voltage
Range A		2400		2520	2340	2160
		4160	4160Y/2400	4370/2520	4050Y/2340	3740Y/2160
		4800		4370	4050	3740
		6900		5040	4680	4320
			8320Y/4800	7240	6730	6210
			12000Y/6930	8730Y/5040	8110Y/4680	
			12470Y/7200	12600Y/7270	11700Y/6760	
			13200Y/7620	13090Y/7560	12160Y/7020	
		13800	13800Y/7970	13860Y/8000	12870Y/7430	
			20780Y/12000	14490Y/8370	13460Y/7770	12420
Range B		2400		2540	2280	2080
		4160	4160Y/2400	4400Y/2540	3950Y/2280	3600Y/2080
		4800		4400	3950	3600
		6900		5080	4560	4160
			8320Y/4800	7260	6560	5940
			12000Y/6930	8800Y/5080	7900Y/4560	
			12470Y/7200	12700Y/7330	11400Y/6580	
			13200Y/7620	13200Y/7620	11850Y/6840	
		13800	13800Y/7970	13970Y/8070	12504Y/7240	
			20780Y/12000	14520Y/8380	13110Y/7570	11880
		22860Y/13200	14520	13110		
	23000		22000Y/12700	19740Y/11400		
		24940Y/14400	24200Y/13970	21720Y/12540		
		34500Y/19920	24340	21850		
	34500		26400Y/15240	23690Y/13680		
			36510Y/21080	32780Y/18930		
			36510	32780		

3.3. Metodología empleada

La metodología propuesta consiste en atacar el problema en dos fases que se coordinan resolviendo cada una por separado y uniendo sus resultados hasta que el resultado converja. Los pasos a seguir se puede describen de la siguiente manera:



- Paso 1: Ingreso de datos del sistema (configuración de la red, pronóstico de carga) y condiciones de la simulación (valor de las restricciones).
- Paso 2: Se considera que los capacitores se encuentran desconectados antes del periodo de estudio y se asigna la posición 0 al cambiador de derivaciones para cada hora del día.
- Paso 3: Obtener los estados óptimos de los bancos de capacitores para cada hora del día que cumplan con las restricciones en las ecuaciones 3.5 y 3.6 sin afectar las posiciones seleccionadas para el cambiador de derivaciones.
- Paso 4: Obtener las posiciones adecuadas del cambiador de derivaciones para cada hora del día que cumplan con las restricciones en las ecuaciones 3.4 y 3.6 considerando que los capacitores se conectan o desconectan de acuerdo al resultado obtenido en el paso previo.
- Paso 5: Repetir los pasos 3 y 4 hasta que converja la solución, es decir, que los resultados obtenidos tanto de los estados óptimos de capacitores y las posiciones del cambiador de derivación sean los mismos que los obtenidos antes de ejecutar al paso 3.

Tanto en el paso 3 como en el paso 4, se hace emplea una rutina de flujos de potencia realizada por Iván Archundia en [20], esta rutina fue modificada y adaptada en este trabajo para evaluar las diferentes combinaciones de los elementos de control y las variaciones de carga a lo largo del día. Los códigos empleados se encuentran en el Apéndice B, mientras que en el Apéndice C se muestra el formato del archivo de entrada y en el Apéndice D se muestran los archivos de salida obtenidos de las simulaciones.

En la figura 3.3 se muestra un diagrama de flujo simplificado del programa que se desarrolló, y posteriormente se desglosará el método con el que se realiza cada despacho.

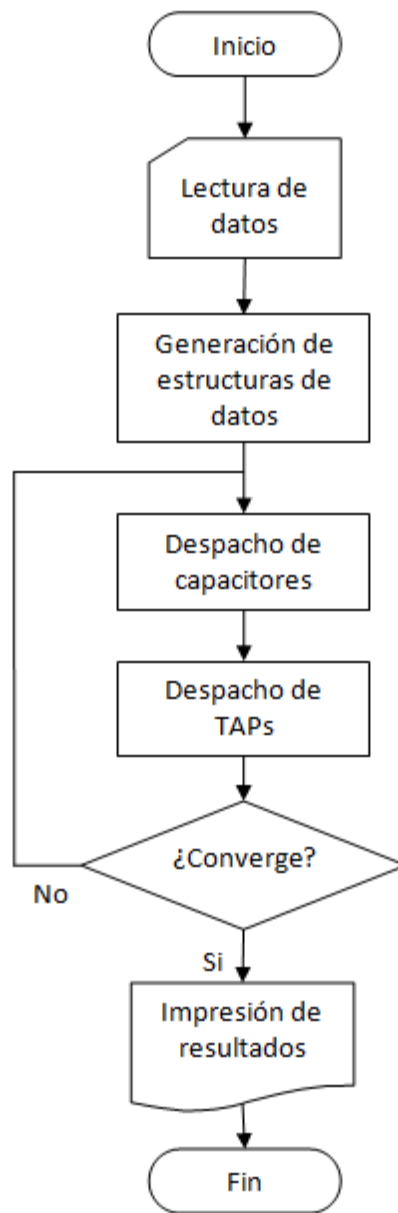


Figura 3.3 Diagrama de flujo simplificado del programa realizado

3.4. Despacho de capacitores

Para poder tomar en cuenta las restricciones en el número máximo de operaciones diarias en los capacitores, se adopta un sistema de estados modificados, donde en cada estado se



representa con un dígito el número de operaciones que se han realizado durante el día para cada capacitor.

De esta forma el escenario en donde todos los capacitores se encuentran desconectados puede ser representado mediante [0000] o [2222] dependiendo del estado predecesor que tenga. Es probable que al usar este sistema de estados modificados se termine aumentando el número de trayectorias a analizar, porque ahora una misma condición de los capacitores tiene más de un estado que lo puede representar, por lo que es necesario trabajar con cuidado este sistema.

Para disminuir el esfuerzo computacional que se requiere para analizar cada una de las trayectorias, se propone almacenar en cada etapa solo un número determinado de estados que satisfagan las restricciones y que tengan menos pérdidas. Como el número de combinaciones crece con el número de capacitores en la red y el número de operaciones diarias permitidas por capacitor, no es posible fijar un número de estados a guardar para cualquier caso a analizar, por lo que se tuvo la necesidad de ocupar una fórmula que calculara el número de combinaciones o escenarios guardados en cada etapa, y en base a diferentes pruebas realizadas se propone el uso de la fórmula 3.9 para determinar el número de estados almacenados en cada etapa:

$$N_{esc} = 2 * C_T * K_C \quad (3.9)$$

Donde:

N_{esc}	= Escenarios guardados por etapa
C_T	= Número total de capacitores
K_C	= Límite de operaciones del capacitor en un día.

La generación de estados se realiza mediante una subrutina que revisa los estados almacenados en la etapa anterior del despacho y marca los posibles estados sucesores de estos en un arreglo de dimensión 2^{C_T} . Esto se considera primeramente porque el estado de los capacitores es binario, siendo 0 para el estado desconectado y 1 para el estado conectado, por lo que se puede representar la combinación de todos los capacitores mediante un número binario de C_T cifras. Por ejemplo, para 3 capacitores, se puede tener la representación mostrada en la tabla 3.3:

De esta forma se puede ver que se tienen 2^{C_T} posibles estados, por lo que si se almacenan todos los estados posibles de los capacitores para cada escenario de forma



individual se necesita una matriz de dimensiones $[2^{C_T}, C_T]$, pero si se considera que cada escenario puede estar representado por un número decimal que puede ser convertido a binario ya no se requiere el uso de esta matriz para conocer los estados de los capacitores en cada escenario a analizar, y en su lugar se usa un arreglo de tamaño 2^{C_T} .

Tabla 3.3 Representación de estados para tres capacitores

Número equivalente de escenario	Estado del capacitor		
	1	2	3
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Para la etapa 1, como se considera que el estado inicial de todos los capacitores es desconectado, todos los estados se analizan. A partir de la etapa 2 se analiza si a partir de cada estado almacenado en la etapa anterior se puede cambiar de estado sin exceder el número máximo de operaciones diarias permitidas para cada capacitor, por ejemplo, para una red con tres capacitores y una restricción de dos operaciones diarias permitidas por capacitor, se muestran en la tabla 3.4 tres estados almacenados en la etapa k-1 y sus posibles sucesores para la etapa k.

Tabla 3.4 Posibles estados sucesores en la etapa k para una red con $C_T = 3$ y $K_C = 2$

Estado almacenado en la etapa k-1	Posibles estados modificados sucesores en la etapa k	Posibles estados ordinarios sucesores en la etapa k	Número equivalente de escenario
0 2 1	0 2 1	0 0 1	1
	0 2 2	0 0 0	0
	1 2 1	1 0 1	5
	1 2 2	1 0 0	4
2 0 2	2 0 2	0 0 0	0
	2 1 2	0 1 0	2
2 1 2	2 1 2	0 1 0	2
	2 2 2	0 0 0	0



De esta forma vemos que de los 7 posibles escenarios, solo 5 (0, 1, 2, 4, 5) son posibles sucesores en la etapa k , por lo que no sería necesario analizar los dos estados restantes. Este análisis para un mayor número de capacitores puede reducir considerablemente el número de estados a analizar, lo cual disminuye el esfuerzo computacional que se requiere.

Para la búsqueda de una trayectoria óptima en algunos trabajos [3, 9, 12] esta se realiza cuando se han seleccionado los estados óptimos de cada una de las etapas, se inicia un barrido desde la última etapa hacia la etapa previa buscando las rutas posibles entre los estados almacenados, se suman las pérdidas de cada ruta posible y se selecciona la que menos pérdidas presenta. Esto se realiza analizando desde la última etapa cuales estados de la etapa previa pueden ser sus predecesores y así hasta llegar a la primera etapa, cuando se encuentra una ruta viable se obtienen sus pérdidas y se selecciona la más óptima. Esta búsqueda se vuelve demasiado larga a medida que aumenta el número de capacitores en la red o cuando se permite más operaciones diarias por capacitor.

El principio de optimalidad de Bellman dice que “cualquier subsecuencia de una secuencia óptima debe ser a su vez, una secuencia óptima” [17], por lo que se puede analizar la trayectoria por partes. En este trabajo se usa el método de programación dinámica progresiva, que propone hacer el barrido de rutas óptimas desde que se terminan de seleccionar los estados con menores pérdidas de la etapa 2, esto es, para cada estado de la etapa 2 se busca cual es su predecesor más óptimo de los almacenados en la etapa 1, se marca y se acumulan las pérdidas de ambas etapas, al llegar a la etapa 3 se analiza cual de los estados almacenados en la etapa 2 es un posible predecesor y tiene menor cantidad de pérdidas acumuladas y así ya no es necesario buscar de nuevo el estado predecesor más adecuado de los estados de la etapa 2. Esto reduce en gran medida el número de análisis hechos, ya que, por ejemplo, para una red donde se almacenan 10 estados en 24 etapas se tendrían 10^{24} trayectorias a analizar y de esta forma se analizan $10 * 10 * 23 = 2300$ trayectorias, ya que en la etapa 1 no se busca un estado predecesor.

Para hacer esta selección de trayectorias se ocupa una subrutina que compara cada estado recién almacenado con los estados almacenados en la etapa previa para ver si puede ser un estado sucesor de estos sin exceder el número de operaciones diarias permitidas. Para esto se hace uso de otro arreglo de dimensión $2 * 2^{CT}$ en el cual se almacenan los posibles estados modificados que pueden ser sucesores de los estados previos, si algún estado modificado se repite solo almacena el que tenga menos pérdidas acumuladas. Por lo que al final del análisis, se ordenan los estados modificados y se usan solo los que tengan las menores pérdidas acumuladas.

En la figura 3.4 se muestra un diagrama de flujo que representa como se realiza el despacho de capacitores.

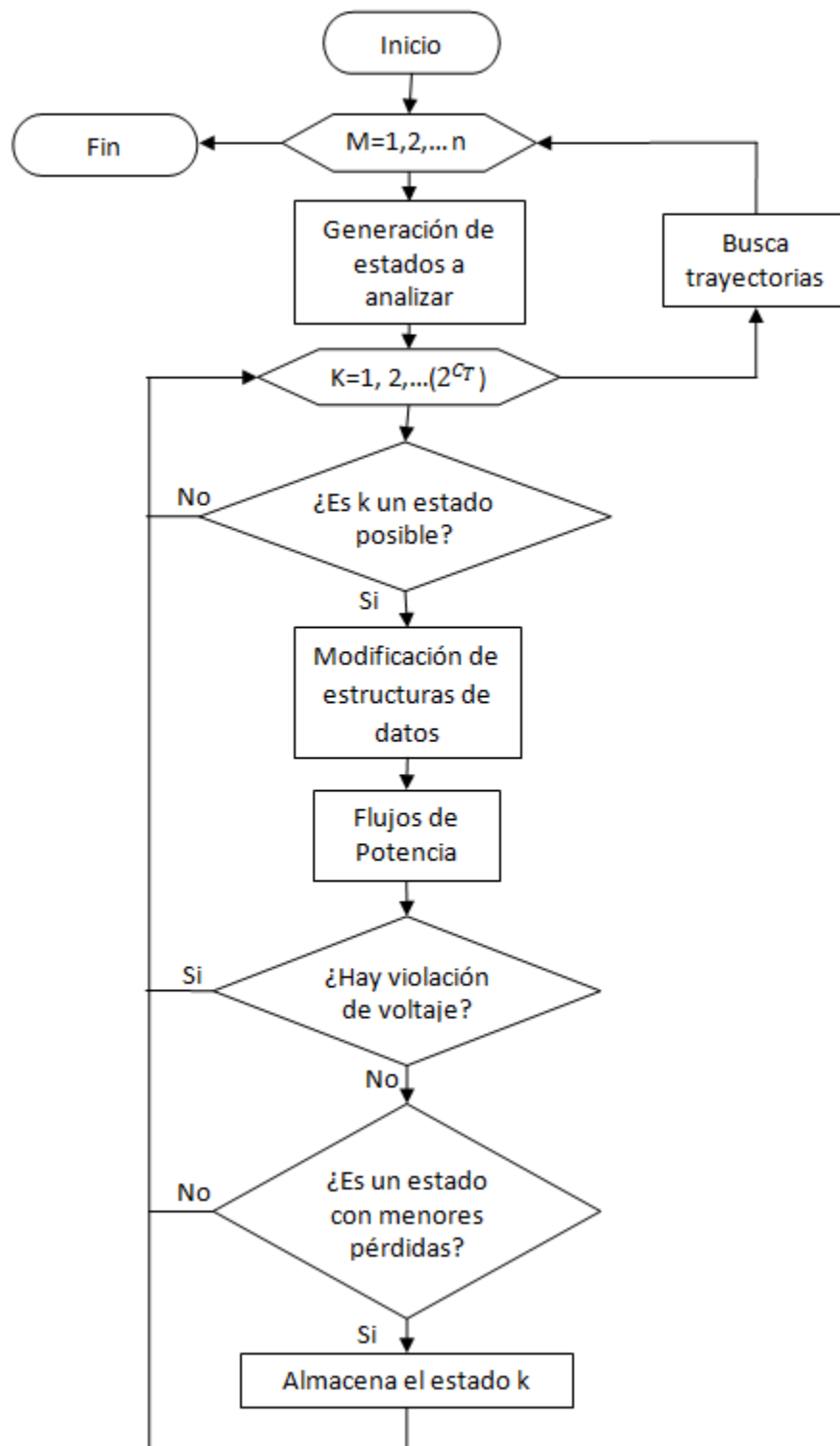


Figura 3.4 Diagrama de flujo del despacho de capacitores



3.5. Despacho de cambiadores de derivación.

Una vez resuelto el despacho de capacitores se cuenta ya con una base que disminuye el alto número de combinaciones posibles de configuraciones de los elementos de la red, por lo que se puede analizar el efecto del movimiento del cambiador de derivaciones en subestación ya con los estados obtenidos para cada capacitor en cada etapa del problema. Dicho efecto es más ligero en comparación a la conexión de un banco de capacitores, ya que por lo general, como se abordó en el capítulo 2, cada movimiento del cambiador de derivaciones modifica el voltaje en un $5/8$ % ó 0.00625 por unidad.

Al realizar pruebas para buscar la posición idónea del cambiador de derivaciones se observó que en la gran mayoría de los casos, la posición con menos pérdidas no variaba bruscamente de una hora a otra, sin embargo, al analizar una trayectoria que incluyera las posiciones óptimas del cambiador de derivaciones en cada hora la suma de estas variaciones generaba un número considerable de movimientos que podrían ser no deseables desde el punto de vista operativo, por lo que se propone analizar las posiciones del cambiador de derivaciones como puntos de una función y de esta forma obtener una línea de tendencia mediante el método de mínimos cuadrados [21] con lo que se obtiene un polinomio de primer grado. Para obtener las posiciones del cambiador de derivaciones se redondea la evaluación del polinomio para cada hora y se analiza si el punto corresponde a una solución válida previamente almacenada, si no es así, se busca la posición válida más cercana a ese valor para esa hora. Dado a que la función obtenida se ajusta hacia los puntos óptimos en cada hora, el resultado que otorga no difiere demasiado en pérdidas al obtenido con las posiciones óptimas, por lo que se considera que es aceptable ya que reduce considerablemente el número de movimientos del cambiador de derivaciones a lo largo del día.

Es importante señalar que la contabilización del número de movimientos realizados en un día se considera a partir de la posición del cambiador de derivaciones considerada en la etapa 1, por lo que el movimiento que se debe hacer desde la posición 0 a la posición seleccionada en la etapa 1 no se cuenta como parte de los movimientos realizados, ya que esto limitaría en gran medida alcanzar las posiciones más altas en el cambiador de derivaciones si así se requiriera. Es por esto que se propone el uso de una línea de tendencia, para que solo si la posición adecuada del cambiador de derivaciones es consistente de una etapa a otra sea esta seleccionada, y en caso de que sea una posición que este alejada de las posiciones óptimas en etapas contiguas, esta no ocasione un exceso de movimientos en el cambiador de derivación.

Las figuras 3.5 y 3.6 muestran los resultados obtenidos en la iteración 1 y en la iteración 2 respectivamente para una de las simulaciones realizadas, en ellas se pueden observar las



posiciones del cambiador de derivaciones con menos pérdidas en cada hora, la línea de tendencia resultante y las posiciones finales que se consideran para el cambiador de derivaciones después de evaluar y analizar la función.

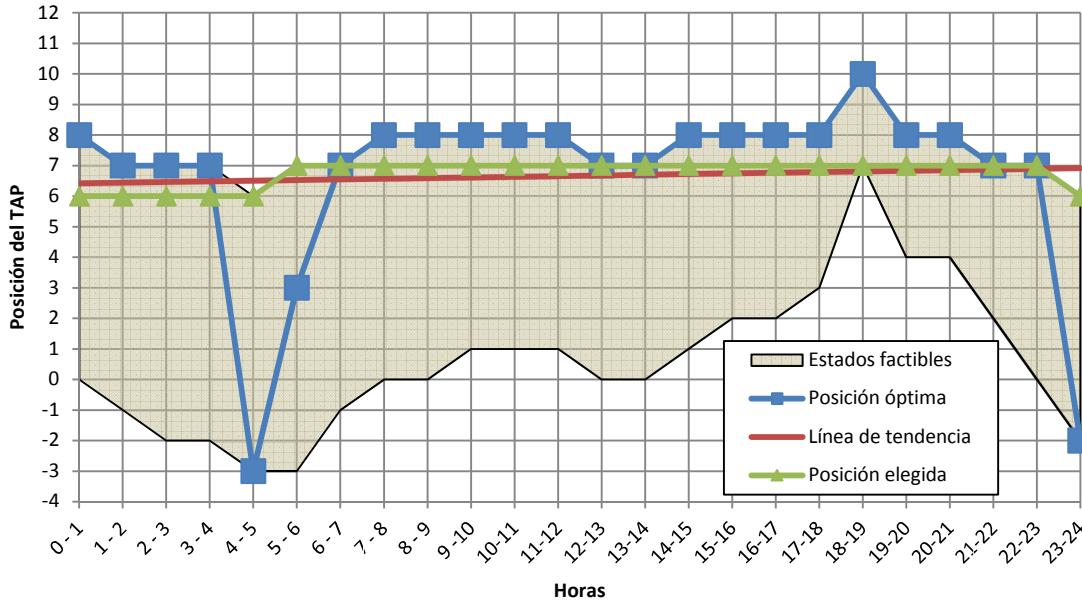


Figura 3.5 Posiciones del cambiador de derivaciones en la iteración 1

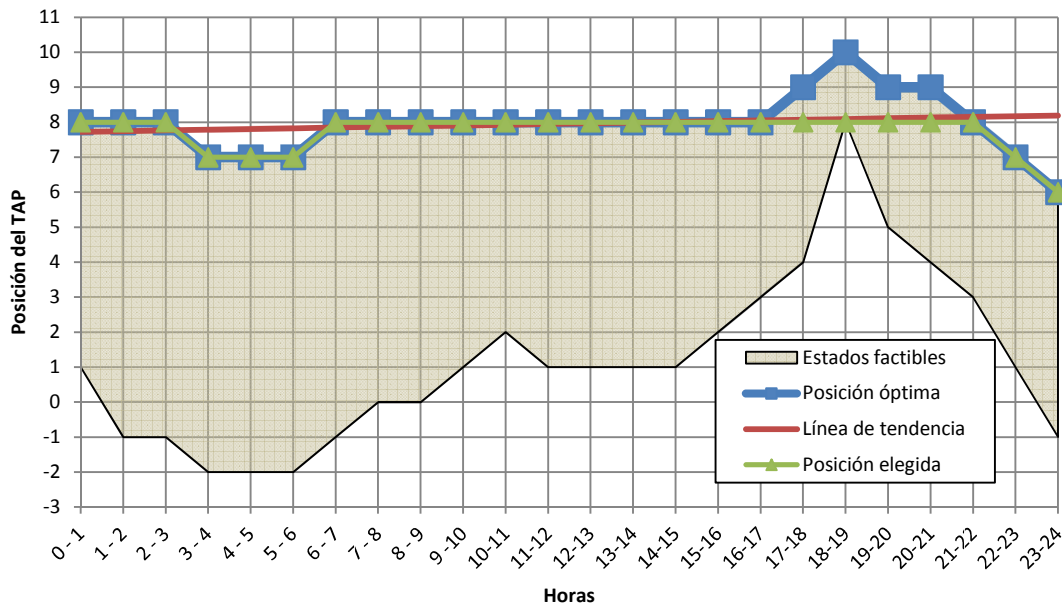


Figura 3.6 Posiciones del cambiador de derivaciones en la iteración 2



El diagrama de flujo para el despacho de cambiadores de derivación está representado en la figura 3.7.

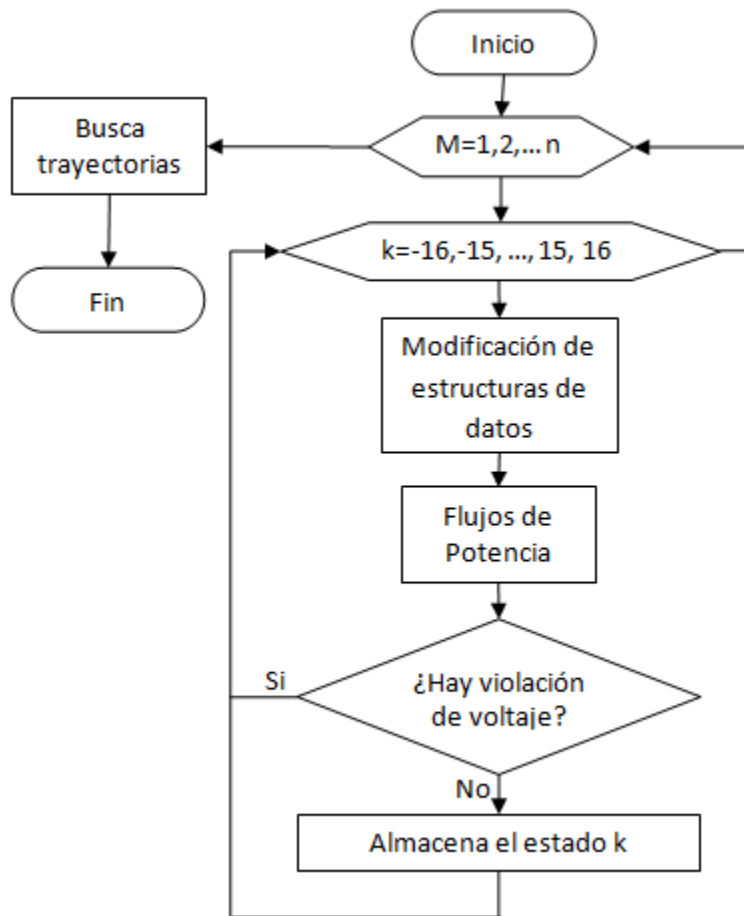


Figura 3.7 Diagrama de flujo para el despacho de cambiadores en derivación

CAPÍTULO 4

PRUEBAS Y RESULTADOS

4.1. Descripción de la red de prueba

El sistema que se eligió para probar la metodología propuesta fue diseñado por Magdy M. Salama en [4], y consiste en un sistema radial de 33 nodos, cuyo alimentador opera a un voltaje de 11kV. Este sistema se usa en [6] para resolver el problema de localización de capacitores y el resultado al que se llega es el que se considera para realizar la coordinación de fuentes de potencia reactiva. Este sistema se muestra en la figura 4.1.

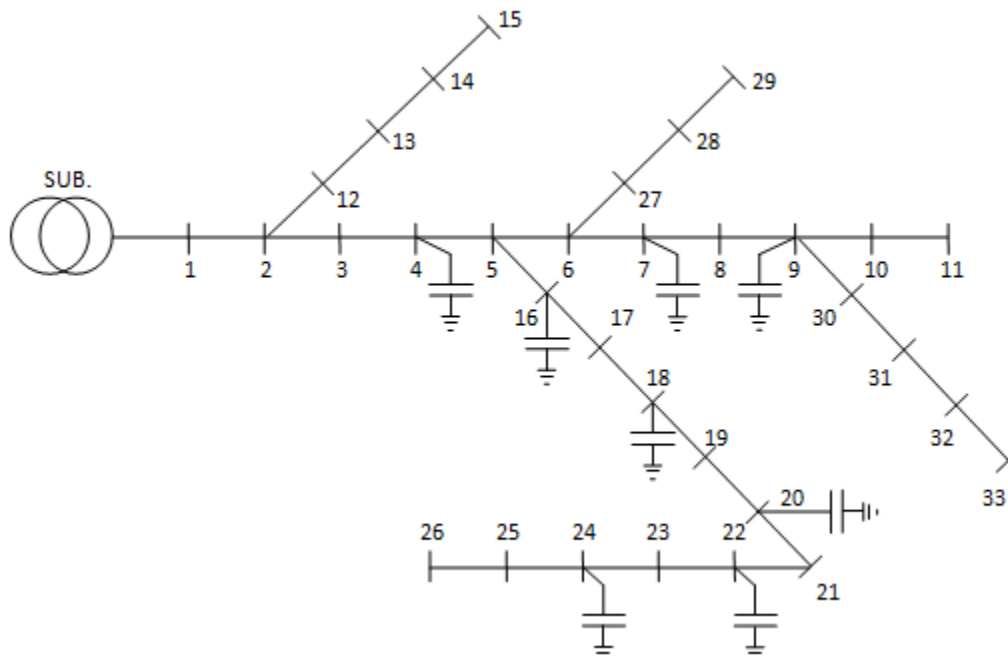


Figura 4.1 Red de prueba de 33 nodos de Salama

Los datos correspondientes a las impedancias de cada sección del alimentador y las potencias correspondientes a las cargas conectadas en cada nodo se muestran en la tabla 4.1.



Tabla 4.1 Datos por sección de la red de Salama

Del nodo	A nodo	R(Ω)	X(Ω)	P(kW)	Q(kVAr)
0	1	0.1170	0.0480	230.0	142.5
1	2	0.1073	0.0440	-	-
2	3	0.1645	0.0457	230.0	142.5
3	4	0.1495	0.0415	230.0	142.5
4	5	0.1495	0.0415	-	-
5	6	0.3144	0.0540	-	-
6	7	0.2096	0.0360	230.0	142.5
7	8	0.3144	0.0540	230.0	142.5
8	9	0.2096	0.0360	-	-
9	10	0.1310	0.0225	230.0	142.5
10	11	0.1048	0.0180	137.0	84.0
2	12	0.1572	0.0270	72.0	45.0
12	13	0.2096	0.0360	72.0	45.0
13	14	0.1048	0.0180	72.0	45.0
14	15	0.0524	0.0090	13.5	7.5
5	16	0.1794	0.0498	230.0	142.5
16	17	0.1645	0.0457	230.0	142.5
17	18	0.2079	0.0473	230.0	142.5
18	19	0.1890	0.0430	230.0	142.5
19	20	0.1890	0.0430	230.0	142.5
20	21	0.2620	0.0450	230.0	142.5
21	22	0.2620	0.0450	230.0	142.5
22	23	0.3144	0.0540	230.0	142.5
23	24	0.2096	0.0360	230.0	142.5
24	25	0.1310	0.0225	230.0	142.5
25	26	0.1048	0.0180	137.0	85.0
6	27	0.1572	0.0270	75.0	48.0
27	28	0.1572	0.0270	75.0	48.0
28	29	0.1572	0.0270	75.0	48.0
9	30	0.1572	0.0270	57.0	34.5
30	31	0.2096	0.0360	57.0	34.5
31	32	0.1572	0.0270	57.0	34.5
32	33	0.1048	0.0180	57.0	34.5

Los valores de los bancos de capacitores obtenidos en [6], así como el nodo donde se encuentran, se muestran en la tabla 4.2.



Tabla 4.2 Datos de los bancos de capacitores usados en la red de Salama

Nodo	kVAr
4	300
7	300
9	300
16	300
18	300
20	300
22	300
24	300

4.2. Perfil de carga considerado.

Para poder realizar una coordinación con un horizonte de 24 horas se determinó un perfil de carga proporcional a la carga máxima para cada hora del día como se puede apreciar en la figura 4.2, y que se asemeja a las curvas típicas de perfil de carga. Para las simulaciones realizadas las cargas se consideraron de potencia real y potencia compleja constante (modelo PQ).

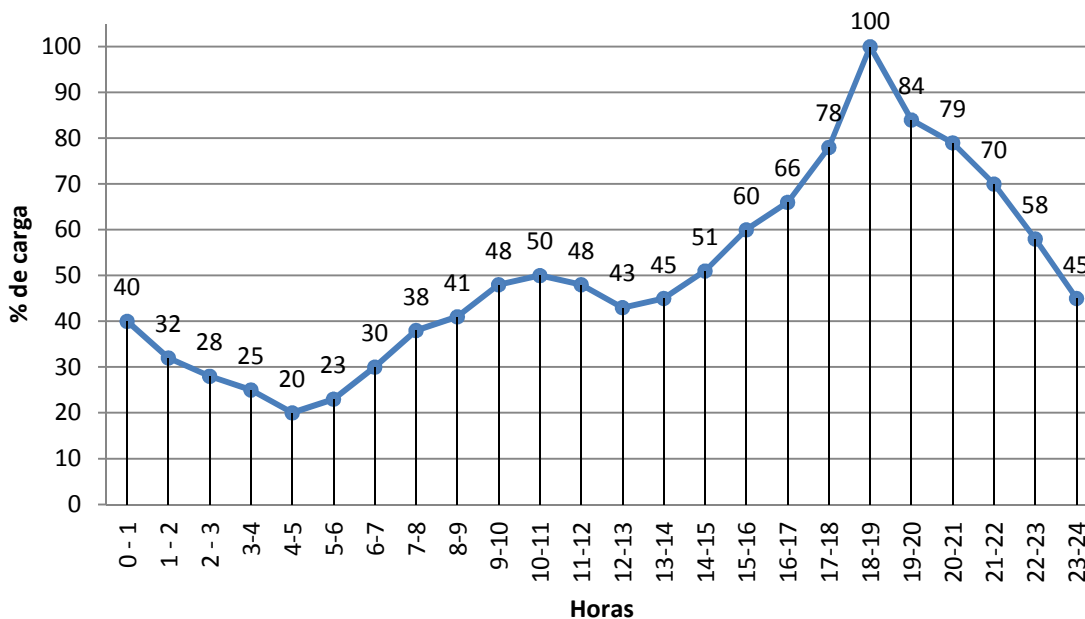


Figura 4.2 Perfil de carga considerado

4.3. Resultados obtenidos

Para tener una base contra la cual analizar los resultados obtenidos, se analizó primeramente el comportamiento de la red sin operar los dispositivos de control, por lo que se cuantificaron las pérdidas de potencia real en la red durante el día dejando fijo el cambiador de derivaciones en la posición 0, y los bancos de capacitores se simularon estando desconectados durante todo el día y también dejándolos conectados todo el día, esto se puede apreciar en la figura 4.3.

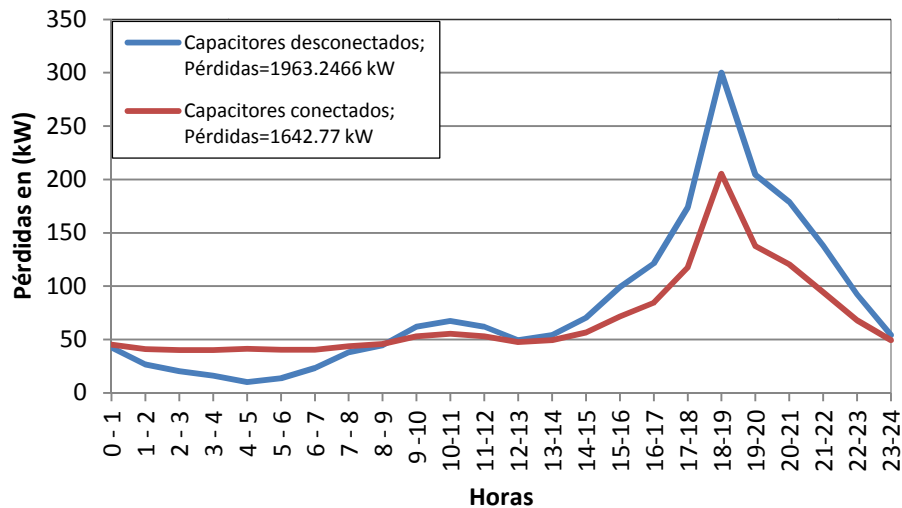


Figura 4.3 Pérdidas de la red durante el día sin despacho

De la figura 4.3 se puede observar que dejar conectados los bancos de capacitores durante todo el día no es una solución adecuada al problema del despacho, ya que existen lapsos de tiempo donde esta condición de operación representa más pérdidas a una condición donde los bancos de capacitores están conectados. De esta forma se ve la necesidad de buscar los estados óptimos de cada banco de capacitores para cada hora del día.

Para probar la metodología propuesta, el sistema se analizó con diferentes valores en el número máximo de operaciones permitidas por banco de capacitores, este límite se varió desde $K_C=1$ hasta $K_C=6$, sin embargo el número de movimientos permitidos por día para el cambiador de derivaciones se mantuvo en $K_T=5$ por dos razones: la primera razón para fijar el valor de la restricción K_T fue para analizar el efecto de la restricción K_C en la metodología propuesta; la segunda razón es que una vez que se realizó este análisis, se observó que en la mayoría de las simulaciones, las posiciones obtenidas para el cambiador



de derivación se mantenían predominantemente fijas, por lo que se decidió que no era necesario ampliar el valor de la restricción K_T .

Al inicio del desarrollo de la metodología se realizaron simulaciones haciendo exclusivamente el despacho de capacitores, y al comparar los resultados obtenidos de estas pruebas iniciales contra los que se generaron con el despacho coordinado, se observa una mejora notable en la disminución de pérdidas, esto se ve reflejado en la figura 4.4 se observan las pérdidas obtenidas en el sistema entre ambos casos con diferentes criterios de límite K_c . De esta forma se observa como el despacho coordinado representa una mejora como estrategia de control en redes de distribución.

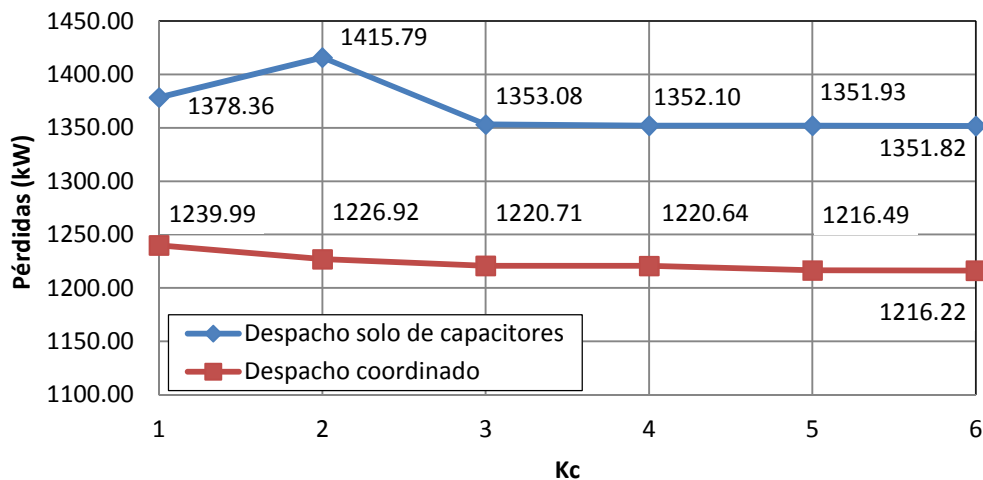


Figura 4.4 Pérdidas obtenidas con dos estrategias de control y diferentes valores de K_c

En la figura 4.4 se nota que las pérdidas se pueden reducir si se aumenta el número de operaciones máximas permitidas, sin embargo, es notable que se llega a un punto en el que por más que se incremente el número de operaciones máximas permitidas, la disminución de las pérdidas es mínima, sin embargo, los tiempos de ejecución son mayores porque mientras mayor sea el número de operaciones permitidas, mayor es el número de combinaciones que se analizan de acuerdo a la ecuación 3.9. Este aumento en el tiempo de ejecución se puede apreciar en la figura 4.5.

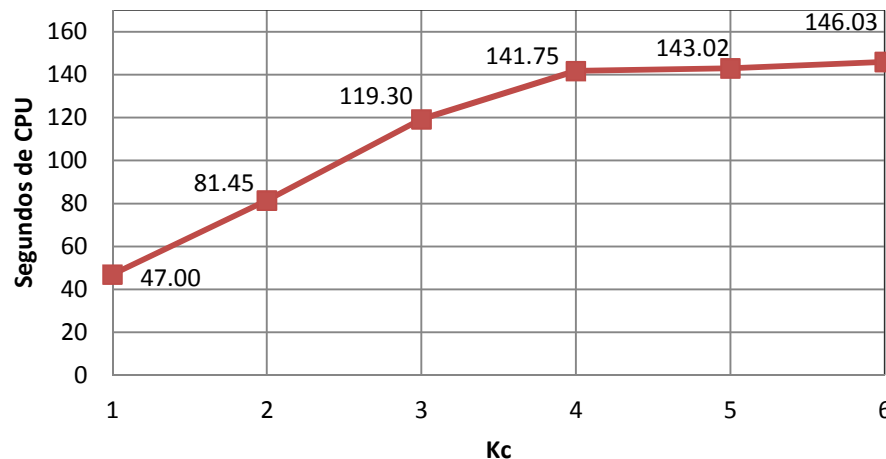


Figura 4.5 Tiempos de ejecución del despacho coordinado con diferentes valores de K_c

De las figuras 4.4 y 4.5 se puede concluir que incluso con un mínimo de operaciones realizadas durante el día se puede obtener un despacho en redes de distribución que permita disminuir las pérdidas de la red.

A continuación se muestran de la figura 4.6 hasta la 4.11 los despachos obtenidos tanto de bancos de capacitores como del cambiador de derivaciones, considerando diferentes valores en el número de operaciones máximas permitidas, para todos los casos se considero como estado inicial del despacho con todos los capacitores desconectados, y el cambiador de derivaciones en una posición de 0. Los archivos de salida de cada uno de estos casos se encuentran en el Apéndice D.

En estas figuras se puede apreciar primeramente que las posiciones seleccionadas del cambiador de derivaciones en cada simulación son consistentes entre sí, la simulación con más variaciones fue la obtenida con una restricción de $K_c = 1$ (figura 4.6), esto se debe a que el estado de los bancos de capacitores cerca del final del día ya no se podían volver a operar por lo que permanecían conectados aunque la carga disminuyera. Para el resto de las simulaciones el cambiador de derivaciones solo se movía dos veces e incluso para los casos con $K_c = 5$ y $K_c = 6$ (figuras 4.10 y 4.11) durante todo el día el cambiador de derivaciones no cambia de posición.

Para los bancos de capacitores se nota que conforme el número de operaciones máximas permitidas se aumenta también conservan una tendencia, la cual es muy clara en las figuras 4.10 y 4.11 donde el despacho para los capacitores C_1 , C_5 y C_8 resultaron idénticos, sin embargo el resto de los capacitores también repiten partes de los despachos previos con cambios menores. Estas semejanzas son igualmente notorias en las figuras 4.8 y 4.9 donde también el despacho obtenido muestra mucha consistencia entre los dos.

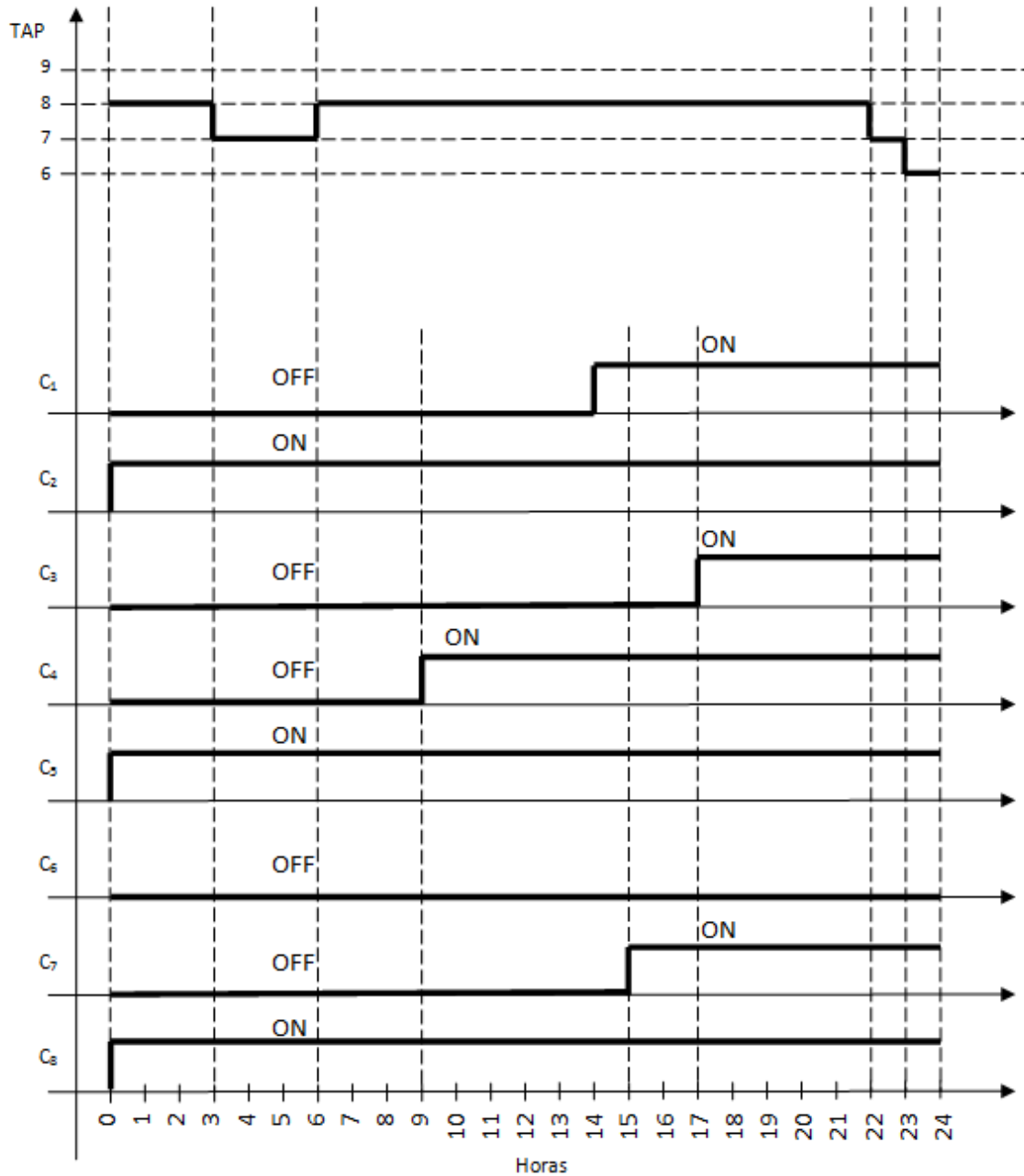


Figura 4.6 Despacho obtenido con una restricción $K_c=1$

Quizá en el caso de las figuras 4.6 y 4.7 sea menos notoria la similitud con el resto de los despachos obtenidos, esto se puede deber a que al tener un número de operaciones máximas permitidas pequeño no se pueden adaptar tanto a los cambios de la carga.

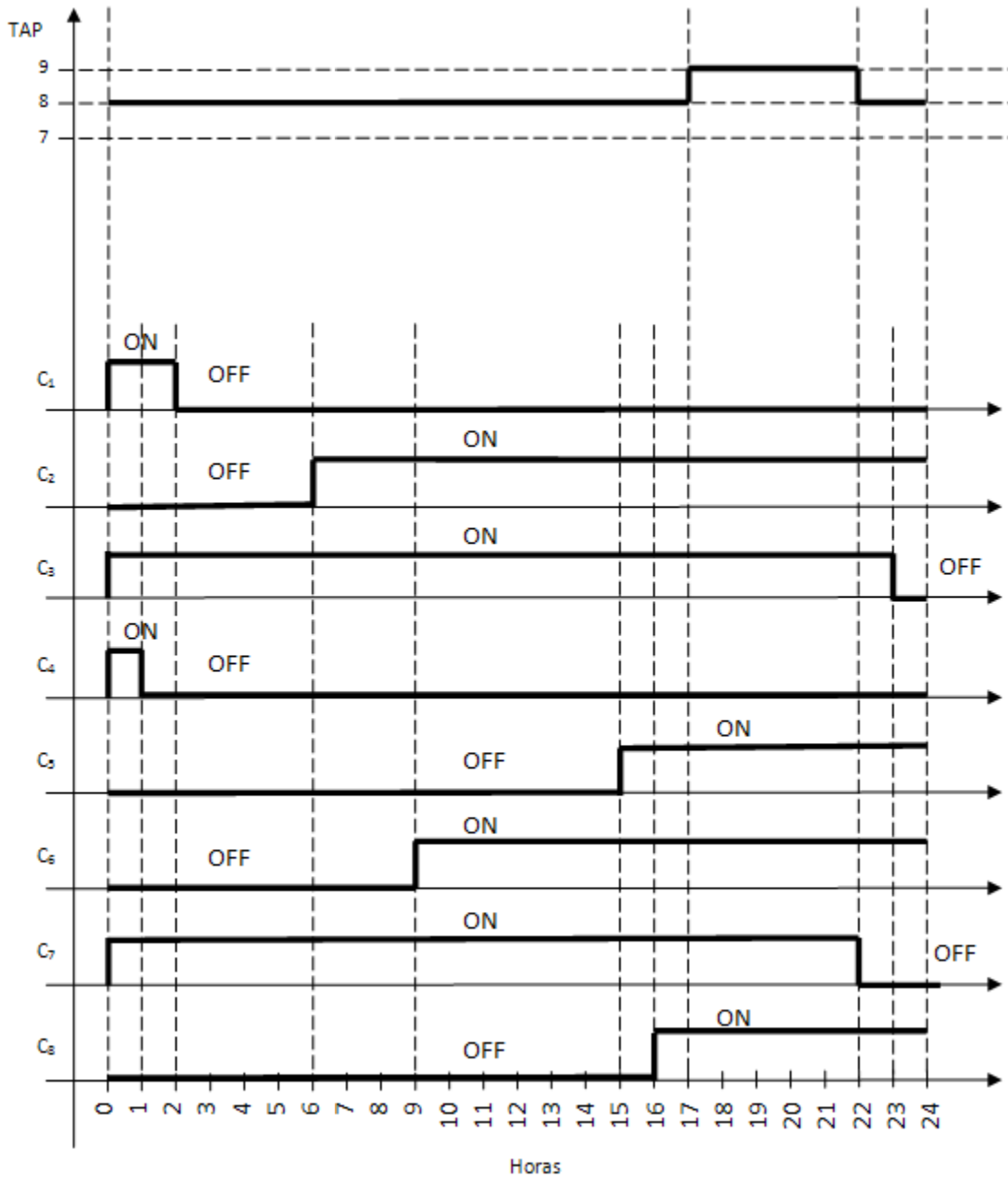


Figura 4.7 Despacho obtenido con una restricción $K_c=2$

Como se ve en la figura 4.4 las pérdidas totales no son muy diferentes en cada caso y además si se considera el deterioro en la vida útil que pueden sufrir los equipos por una constante operación de los mismos, es más recomendable implementar los despachos con menos operaciones de los equipos.

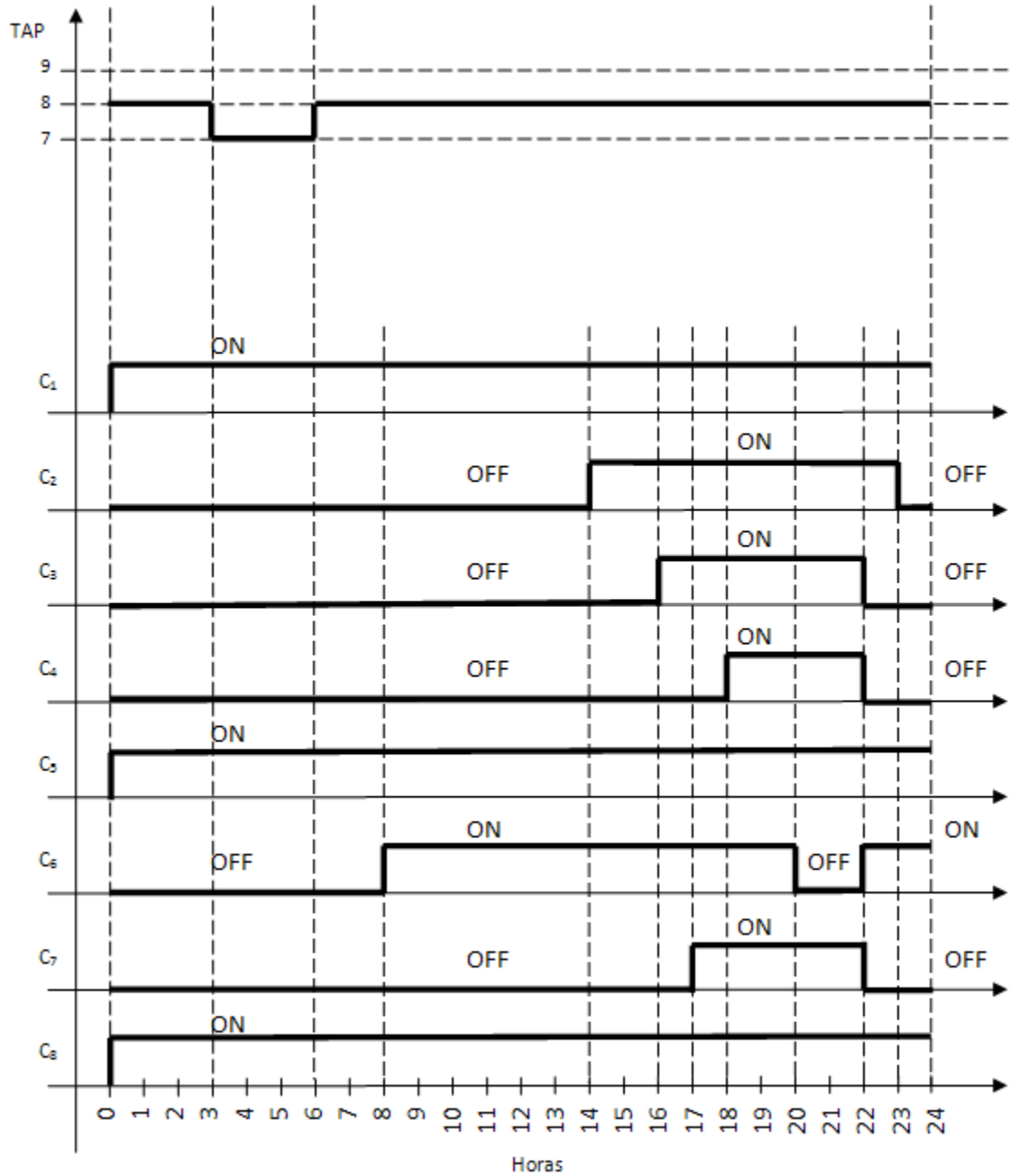


Figura 4.8 Despacho obtenido con una restricción $K_c=3$

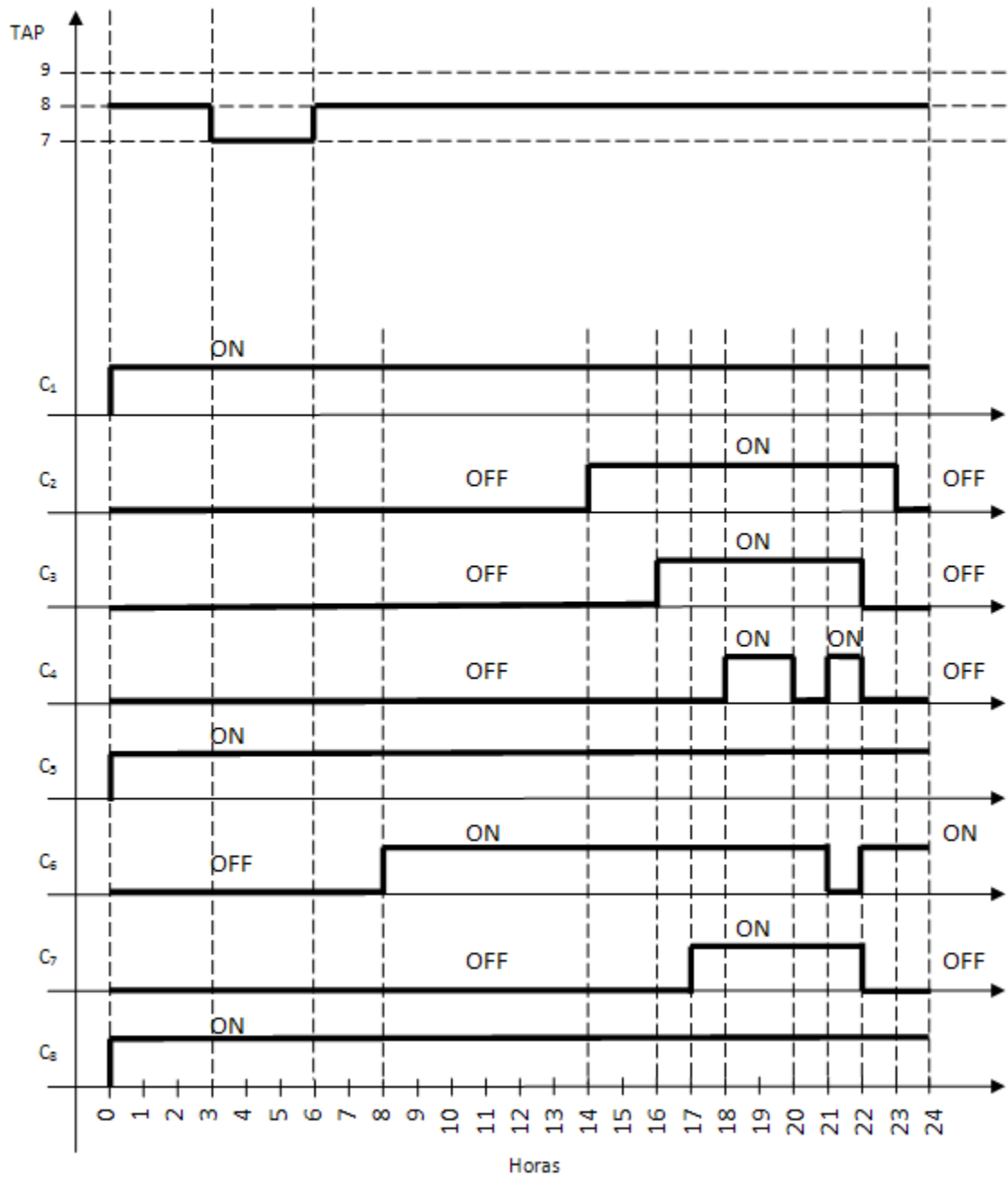


Figura 4.9 Despacho obtenido con una restricción $K_c=4$

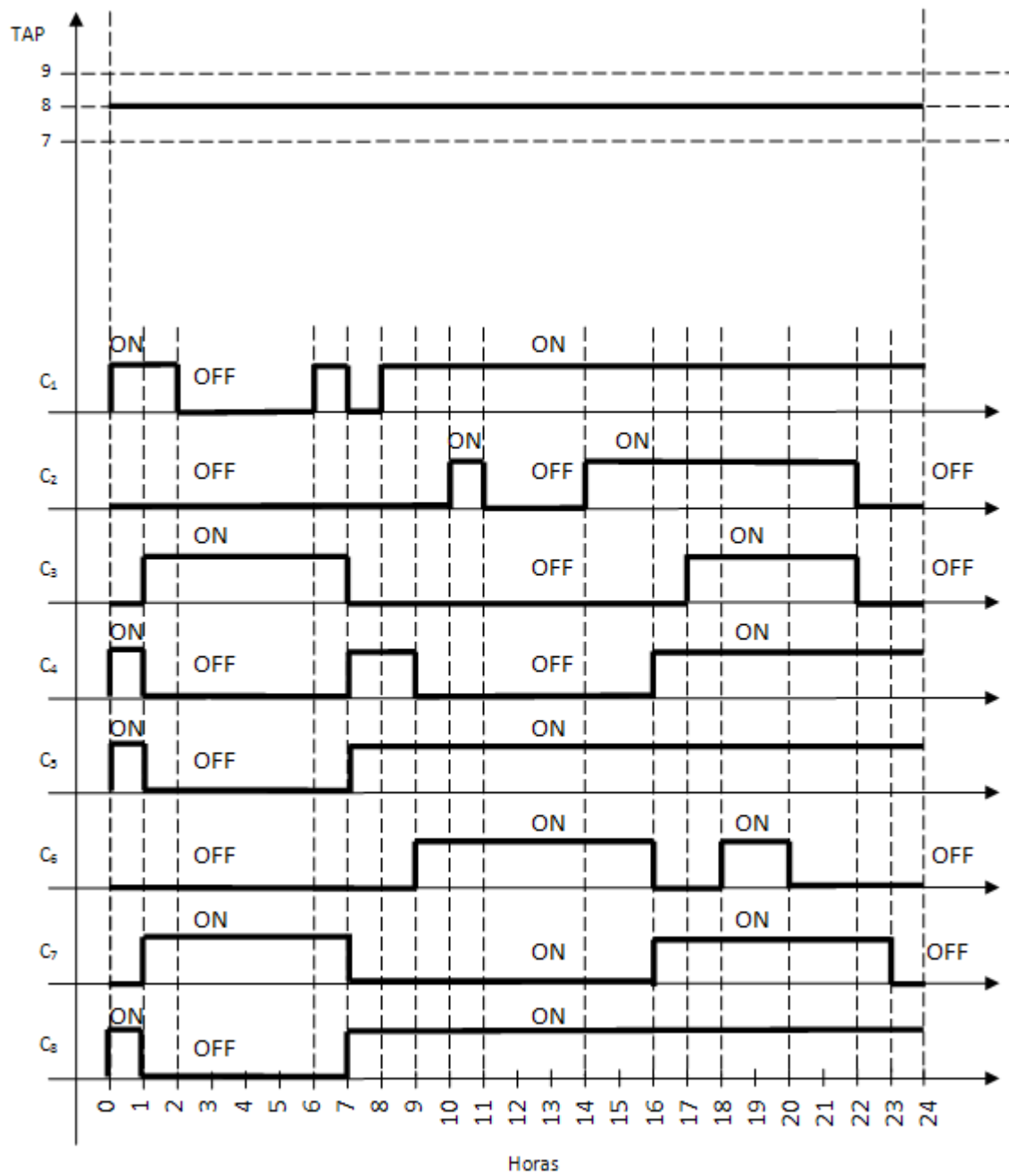


Figura 4.10 Despacho obtenido con una restricción $K_c=5$

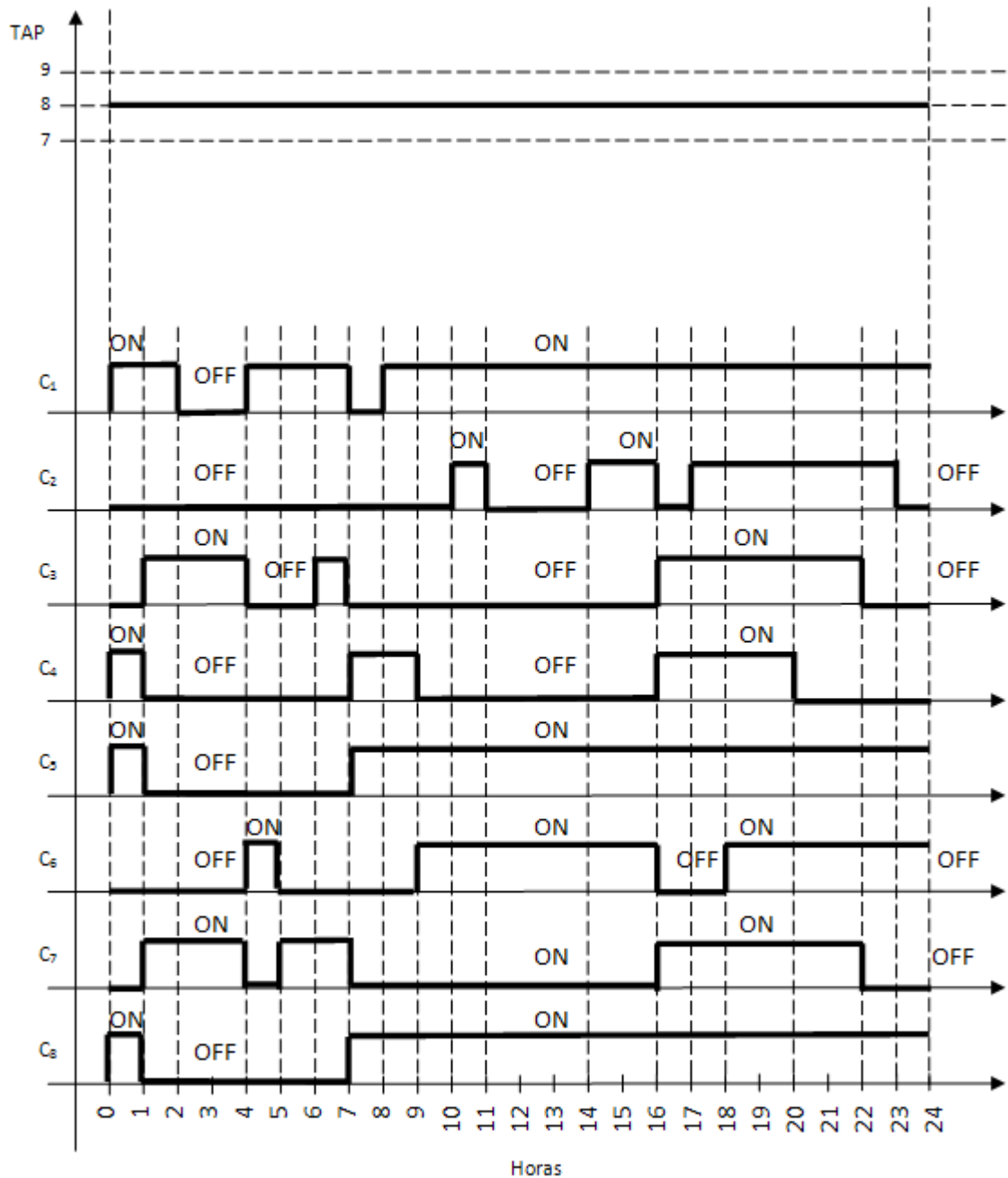


Figura 4.11 Despacho obtenido con una restricción $K_c=6$

Finalmente, para observar el comportamiento de la red una vez aplicada la metodología de coordinación, se presentan en la figura 4.12 y 4.13 las gráficas del perfil de voltaje en p.u. para los nodos 9 y 26 respectivamente para tres simulaciones.

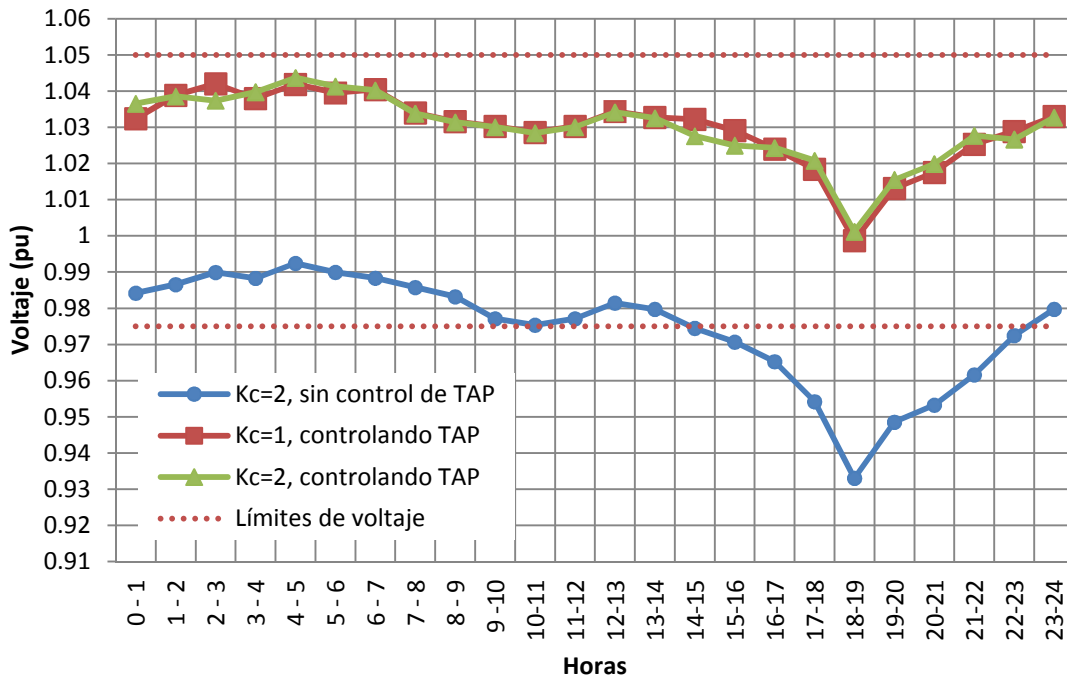


Figura 4.12 Perfil de voltaje del nodo 9 en diferentes simulaciones

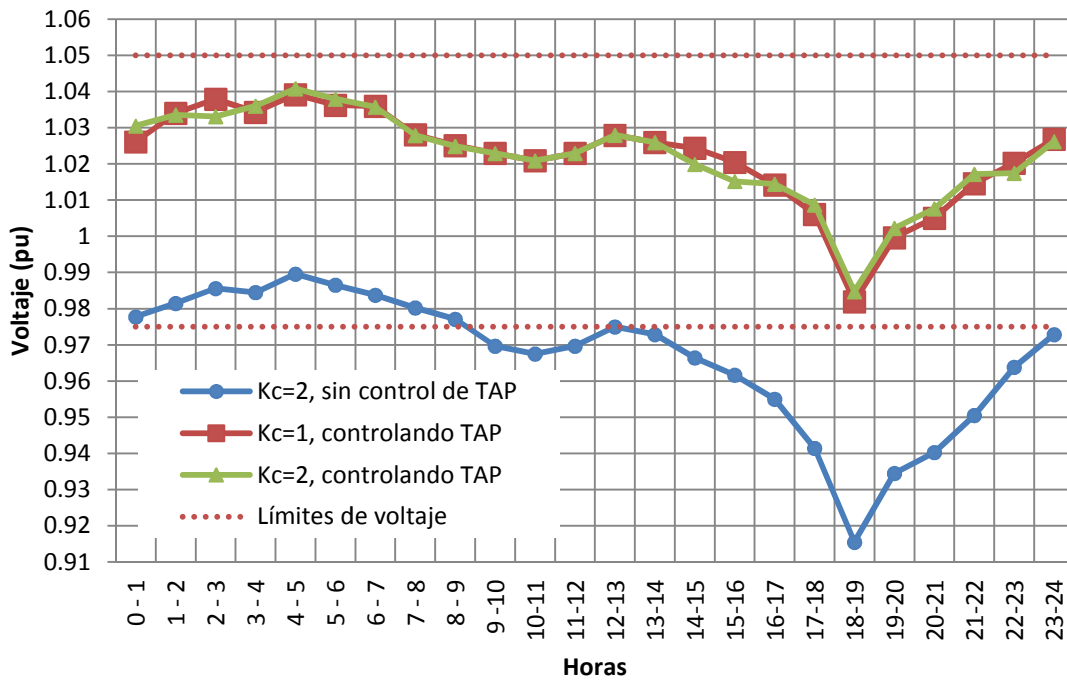


Figura 4.13 Perfil de voltaje del nodo 26 en diferentes simulaciones



De las figuras 4.12 y 4.13 se puede observar que sin la consideración del control del cambiador de derivaciones, en gran parte del tiempo se opera en los límites de voltaje o incluso fuera de ellos, y sin embargo, cuando se realiza el despacho coordinado se puede operar siempre dentro de los límites de voltaje.



CAPÍTULO 5

CONCLUSIONES Y TRABAJOS FUTUROS

5.1. Conclusiones

La metodología propuesta fue verificada en una red de prueba, con lo cual se obtuvo el estado adecuado de los capacitores conectados a dicha red así como también la posición del cambiador de derivaciones en cada hora del día con diferentes condiciones.

Se observa la necesidad de coordinar los elementos de la red de distribución para tener un mejor aprovechamiento de la misma como se demuestra en la figura 4.4, donde se compara el despacho de capacitores contra un despacho coordinado.

Un punto importante a resaltar es que no se requiere un elevado número de operaciones de los capacitores a lo largo del día para disminuir las pérdidas, ya que si el despacho es adecuado, se observa que no hay necesidad de operar tanto los capacitores, porque se llega un punto en el cual la disminución de pérdidas del sistema es mínima por más que se aumente el número de operaciones permitidas por día y sin embargo los tiempos de ejecución de la metodología propuesta aumentan debido al gran número de combinaciones que se deben analizar.

De la misma manera se obtuvo un despacho de los cambiadores de derivación donde a pesar de no incluir todas las posiciones óptimas, se obtienen resultados muy similares con un número menor de operaciones diarias de los cambiadores de derivación por lo que se valida el método propuesto.

5.2. Recomendaciones para trabajos futuros

- Desarrollar un programa de coordinación de fuentes de potencia reactiva para la operación en tiempo real que sirva de apoyo a los operadores de las redes de distribución.



- Incorporar una rutina de pronóstico de carga basada en datos históricos que permita obtener resultados más específicos para algún día en particular.
- Implementar técnicas como redes neuronales o algoritmos genéticos que puedan reducir el tiempo de cómputo requerido para obtener la solución del problema.



REFERENCIAS

- [1] W. H. Kersting, "Distribution System Modelling and Analysis", CRC Press 2002.

- [2] T.Gönen, "Electric Power Distribution System Engineering", McGraw – Hill, 1986.

- [3] Y. Y. Hsu, H. C. Kuo, "Dispatch of capacitors on distribution system using dynamic programming", IEE PROCEEDINGS-C, Vol. 140, No. 6, pp. 433 – 438, 1993.

- [4] M. M. A. Salama, A. Y. Chikhani, "A Simplified Network Approach to the Var Control Problem for Radial Distribution Systems", IEEE Transactions on Power Delivery, Vol. 8, No 3, pp. 1529 - 1535, 1993.

- [5] G. Estrada Soria, "Metodología Técnico-económica de Localización de Capacitores en Sistemas de Distribución para la reducción de Pérdidas Eléctricas", Tesis de Maestría, Instituto Tecnológico de Morelia, PGIIE, Febrero 2003.

- [6] M. Galarza Espín, C. Riofrío Reyes , "Localización Óptima de Capacitores en Redes Primarias de Distribución Mediante Técnicas Heurísticas", Jornadas de Ingeniería Eléctrica y Electrónica (FIEE), 2006.

- [7] O. Urióstegui Miranda, "Minimización de pérdidas en redes de distribución empleando programación dinámica", Tesis de Maestría, Instituto Politécnico Nacional, Sección de Posgrado e Investigación, ESIME, Junio de 2009.

- [8] Z. Hu, X. Wang, H. Chen, G.A. Taylor, "Volt/VAr control in distribution systems using a time interval based approach", IEE Proc. - Gener. Transm. Distrib., Vol. 150, No. 5, pp. 548 – 554, 2003.



- [9] F. C. Lu and Y. Y. Hsu, "Reactive Power/Voltage control in a distribution substation using dynamic programming", IEE Proceedings. – Generation, Transmission and Distribution, Vol. 142, No. 6, pp. 639 - 645, 1995.
- [10] Z. Gu, D. T. Rizy, "Neural Networks for combined control of capacitor banks and voltage regulators in distribution systems", IEEE Transactions on Power Delivery, Vol. 11, No. 4, pp. 1921 – 1928, 1996.
- [11] Y. Liu, X. Qiu, "Optimal Reactive Power and Voltage Control for Radial Distribution System", IEEE Power Engineering Society Summer Meeting, Vol. 1, pp. 85 - 90, 2000.
- [12] R.-H. Liang, C.-K. Cheng. "Dispatch of Main Transformer ULTC and Capacitors in a Distribution System", IEEE Transactions on Power Delivery, Vol. 16, No. 4, pp. 625 - 630, 2001.
- [13] C.-C. Kuo, P.-H. Chen, C.-L. Hsu, "The Reactive Power and Voltage Control of Distribution Systems Using the Normalized Weighting Method", IEEE/PES Transmission and Distribution Conference & Exhibition: Asia and Pacific, pp. 1 - 6, 2005.
- [14] R. Guerequeta, A. Vallecillo, "Técnicas de Diseño de Algoritmos", Servicio de Publicaciones de la Universidad de Málaga, 2000.
- [15] S. E. Dreyfus, A. M. Law, "Art And Theory Of Dynamic Programming", Academic Press, New York, 1977.
- [16] "Conceptos avanzados de planteamiento operativo", Seminario del Programa de Actividades Regionales en el Subsector Eléctrico del Istmo Centroamericano (PARSEICA), 1993.
- [17] R. Bellman, S. E. Dreyfus, "Applied Dynamic Programming", Princeton University Press, 1962.



- [18] A. J. Wood, B. F. Wollenberg, "Power generation, operation, and control", John Wiley & sons, 1996.
- [19] American Nation Standard for Electric Power – Systems and Equipment Voltage Ratings (60 Hertz), ANSI C84.1-1995, National Electrical Manufacturers Association, Rosslyn, Virginia, 1996.
- [20] I. Archundia Aranda, Programa "RADIFLUX 2.0" para resolver Flujos de potencia en redes radiales de distribución, Instituto Politécnico Nacional, Sección de Posgrado e Investigación, ESIME, 2010, iarchundia@ipn.mx.
- [21] S. C. Chapra, R. P. Canale, "Numerical Methods for Engineers", Mc Graw – Hill, 2006.
- [22] W. H. Kersting and W. Howard Phillips, "Distribution Feeder Line Models", IEEE Transactions on Industry Applications, Vol. 31, No. 4, pp. 715 - 720, 1995.
- [23] T. A. Short, "Electric Power Distribution Equipment and Systems", CRC Press 2004.
- [24] IEEE Standard Requirements, Terminology, and Test Code for Step-Voltage and Induction-Voltage Regulators, ANSI/IEEE C57.15-1986, Institute of Electrical and Electronic Engineers, New York, 1988.
- [25] C. S. Cheng and D. Shirmohammadi, "A Three Phase Power Flow Method for Real Time Distribution System Analysis", IEEE Transactions on Power Systems, Vol. 10, No. 2, pp. 671 – 679, 1995.



APÉNDICE A

MODELADO Y ANÁLISIS DE LOS ELEMENTOS DE DISTRIBUCIÓN.

A.1. Introducción

Un sistema de distribución de energía eléctrica comprende todo el conjunto de elementos mediante los cuales se conduce la energía eléctrica desde una subestación de potencia hasta el punto de consumo. Estos elementos pueden ser, líneas primarias de distribución, transformadores de distribución y líneas secundarias de distribución. Cada subestación de distribución atiende a uno o más alimentadores primarios y, con escasas excepciones, los alimentadores son radiales, lo que significa que solo existe un camino para que la potencia fluya de la subestación de distribución al usuario [22].

A.2. Modelo π de la línea

Para el análisis de sistemas con alimentadores trifásicos se usan matrices de impedancias que representan a los elementos de la red de acuerdo a su configuración física y a sus propiedades.

A.2.1. Impedancia serie de líneas aéreas y subterráneas

La impedancia serie de una línea de distribución de una, dos, o tres fases consiste en la resistencia de los conductores y de las reactancias inductivas propia y mutua resultante de los campos magnéticos que rodean a los conductores. El componente resistencia de los conductores normalmente se obtiene de tablas de datos de conductores [23]. La matriz de impedancia de fase trifásica desarrollada es:

$$[z_{abc}] = \begin{bmatrix} z_{aa} & z_{ab} & z_{ac} \\ z_{ba} & z_{bb} & z_{bc} \\ z_{ca} & z_{cb} & z_{cc} \end{bmatrix} \Omega / \text{milla} \quad (\text{A.1})$$



Para una línea de distribución que no está transpuesta, los términos de la diagonal de la ecuación A.1 no serán iguales entre sí, y los términos fuera de la diagonal no serán tampoco iguales entre sí. Sin embargo, la matriz será simétrica.

Para líneas bifásicas y líneas monofásicas en un sistema estrella aterrizada, se forman matrices de dimensiones 2x2 y de un elemento sencillo. Estas matrices pueden expandirse a matrices de 3x3 en el “marco de referencia de fase” sumando filas y columnas que consisten en elementos cero para las fases inexistentes. Por ejemplo, una línea de bifásica con fases a y c; la matriz de impedancias sería:

$$[z_{abc}] = \begin{bmatrix} z_{ab} & 0 & z_{ac} \\ 0 & 0 & 0 \\ z_{ca} & 0 & z_{cc} \end{bmatrix} \Omega / \text{milla} \quad (\text{A.2})$$

La matriz de impedancias para la fase b de una línea monofásica sería

$$[z_{abc}] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & z_{bb} & 0 \\ 0 & 0 & 0 \end{bmatrix} \Omega / \text{milla} \quad (\text{A.3})$$

A.2.2. Admitancia en derivación de líneas aéreas o subterráneas

La admitancia en derivación en una línea consiste en una conductancia y una susceptancia capacitiva. Normalmente la conductancia es ignorada debido a que es muy pequeña comparada con la susceptancia capacitiva. La capacitancia en una línea es el resultado de la diferencia de potencial entre conductores. La matriz de admitancia en derivación trifásica resultante tiene el siguiente modelo:

$$[y_{abc}] = \begin{bmatrix} y_{aa} & y_{ab} & y_{ac} \\ y_{ba} & y_{bb} & y_{bc} \\ y_{ca} & y_{cb} & y_{cc} \end{bmatrix} \mu S / \text{milla} \quad (\text{A.4})$$

Para líneas bifásicas o monofásicas, la matriz de admitancias en derivación tendrá elementos cero para las filas y columnas correspondientes a las fases inexistentes.

A.3. Modelo π exacto de la línea

La figura A.1 muestra la representación del modelo exacto de una línea aérea o subterránea trifásica. Cuando un segmento de línea es bifásica, o monofásica, los renglones y columnas de los valores de admitancia e impedancia de las fases que faltan serán cero.

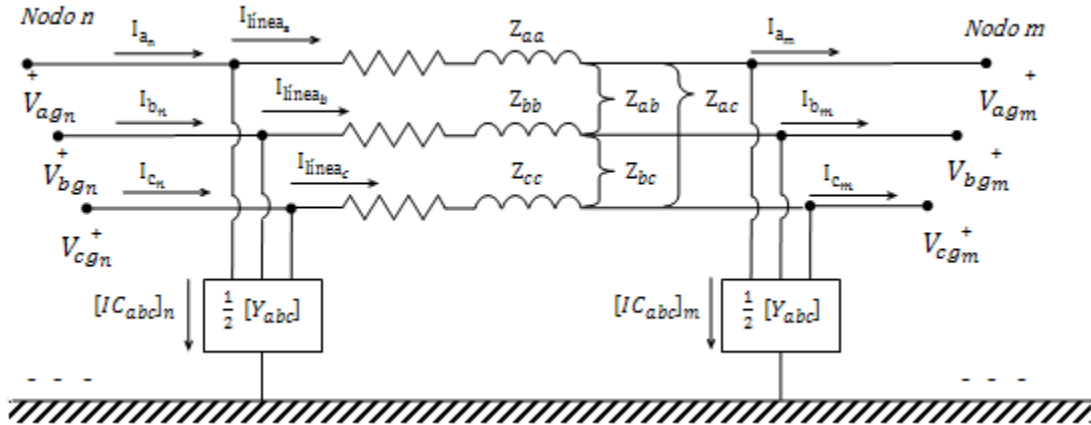


Figura A.1 Modelo π exacto de una línea trifásica

De esta forma se puede desarrollar un conjunto de ecuaciones para modelar todos los segmentos de líneas subterráneas y aéreas. Los valores de las impedancias y admitancias representan las impedancias y admitancias totales de la línea.

$$[Z_{abc}] = \begin{bmatrix} Z_{aa} & Z_{ab} & Z_{ac} \\ Z_{ba} & Z_{bb} & Z_{bc} \\ Z_{ca} & Z_{cb} & Z_{cc} \end{bmatrix} \Omega \quad (A.5)$$

$$[Y_{abc}] = \begin{bmatrix} Y_{aa} & Y_{ab} & Y_{ac} \\ Y_{ba} & Y_{bb} & Y_{bc} \\ Y_{ca} & Y_{cb} & Y_{cc} \end{bmatrix} \mu S \quad (A.6)$$

Las ecuaciones que relacionan los voltajes y corrientes de entrada (nodo n) a los voltajes y corrientes de salida (nodo m) se desarrollan y usando las leyes de Kirchhoff al modelo resulta:

$$[VLG_{abc}]_m = [a][VLG_{abc}]_n - [b][I_{abc}]_n \quad (A.7)$$



$$[I_{abc}]_m = -[c][VLG_{abc}]_n + [d][I_{abc}]_n \quad (A.8)$$

Donde:

$$[a] = [U] + \frac{1}{2}[Z_{abc}][Y_{abc}] \quad (A.9)$$

$$[b] = [Z_{abc}] \quad (A.10)$$

$$[c] = [Y_{abc}] + \frac{1}{4}[Y_{abc}][Z_{abc}][Y_{abc}] \quad (A.11)$$

$$[d] = [U] + \frac{1}{2}[Z_{abc}][Y_{abc}] \quad (A.12)$$

Algunas veces es necesario calcular los voltajes del nodo m en función de los voltajes en el nodo n y las corrientes que entran al nodo m . Así sucede en la técnica iterativa de flujos de potencia por el método de barrido regresivo-progresivo.

$$[VLG_{abc}]_m = [A][AVG_{abc}]_n - [B][I_{abc}]_m \quad (A.13)$$

Donde:

$$[A] = [a]^{-1} \quad (A.14)$$

$$[B] = [a]^{-1}[b] \quad (A.15)$$

Los voltajes de línea a línea son calculados mediante:

$$\begin{bmatrix} V_{ab} \\ V_{bc} \\ V_{ca} \end{bmatrix}_m = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_{ag} \\ V_{bg} \\ V_{cg} \end{bmatrix}_m = [D][VLG_{abc}]_m \quad (A.16)$$

Debido a que los acoplamientos mutuos entre fases en los segmentos de línea, no son iguales, habrá valores diferentes de caída de voltaje en cada una de las tres fases. Como resultado, los voltajes en un alimentador de distribución resultan desbalanceados incluso cuando las cargas son balanceadas

A.4. Modelado de transformadores trifásicos

Se puede aplicar una variedad de conexiones de transformador. La carga puede ser trifásica pura o una combinación de carga monofásica de iluminación y una carga trifásica tal como un motor de inducción. En el análisis de un alimentador de distribución, es importante que las conexiones de los diferentes transformadores trifásicos se modelen correctamente. En [1] se desarrollan modelos para las conexiones trifásicas siguientes:

- Delta – Estrella aterrizada
- Estrella – Delta
- Estrella aterrizada – Estrella Aterrizada
- Delta – Delta
- Estrella Abierta – Delta Abierta

En la figura A.2 se definen los diferentes voltajes y corrientes para todos los bancos de transformación conectados entre el lado fuente nodo n y el nodo lado carga m . La notación es tal que las letras mayúsculas A, B, C, N siempre se referirán al lado *fuentes* (nodo n) del banco y las letras minúsculas a, b, c y n siempre se referirán al lado *carga* (nodo m) del banco. Se considera que todas las variaciones de la conexión estrella-delta están conectadas en la conexión "Norma Americana de Treinta-Grados".

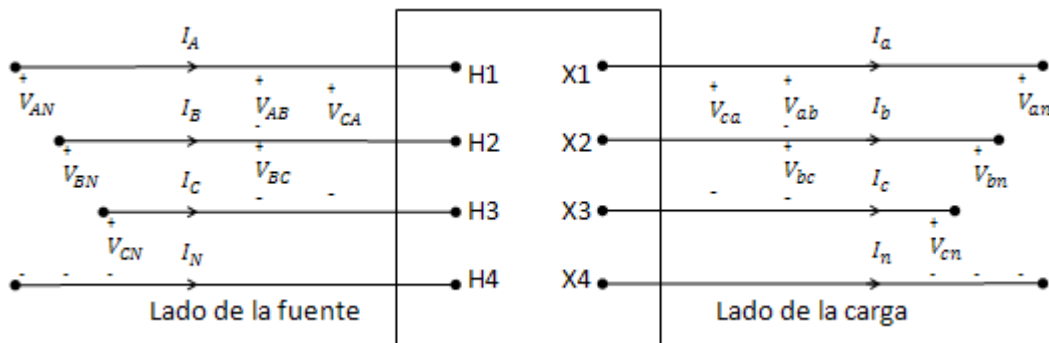


Figura A.2 Banco de transformación trifásico en general

Las ecuaciones matriciales para calcular las tensiones y corrientes en el nodo m como una función de las tensiones y corrientes en el nodo n están dadas por:

$$[VLN_{abc}] = [A_t][VLN_{ABC}] - [B_t][I_{abc}] \quad (A.17)$$

$$[I_{ABC}] = [d_t][ID_{ABC}] \quad (A.18)$$



Las matrices A_t , B_t y d_t son matrices constantes y sus elementos son diferentes para cada tipo de conexión de los bancos de transformación.

Las matrices $[VLN_{ABC}]$ y $[VLN_{abc}]$ representan los voltajes de línea a neutro para una conexión estrella aislada ó los voltajes de línea a tierra para una conexión estrella aterrizada. Para una conexión en delta, las matrices de voltaje representan voltajes de línea a neutro “equivalentes”. Las matrices de corriente representan las corrientes de línea independientemente de la conexión del devanado del transformador.

Los voltajes conocidos de línea a línea se transforman a voltajes equivalentes de línea a neutro mediante:

$$[VLN_{ABC}] = [W][VLL_{ABC}] \quad (A.19)$$

Donde:

$$[W] = \frac{1}{3} \begin{bmatrix} 2 & 1 & 0 \\ 0 & 2 & 1 \\ 1 & 0 & 2 \end{bmatrix} \quad (A.20)$$

La tabla A.1 condensa las matrices constantes A_t , B_t y d_t para las conexiones más comunes de transformadores trifásicos obtenidas en [1].

Tabla A.1 Matrices generalizadas para diferentes conexiones de transformadores

Delta - Estrella aterrizada	
$n_t = \frac{VLL_{no\ min\ al\ -lado\ de\ alta}}{VLN_{no\ min\ al\ _lado\ de\ baja}}$	
$A_t = \frac{1}{n_t} \begin{bmatrix} 1 & 0 & -1 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}$	$d_t = \frac{1}{n_t} \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \end{bmatrix}$
$B_t = \begin{bmatrix} Zt_a & 0 & 0 \\ 0 & Zt_b & 0 \\ 0 & 0 & Zt_c \end{bmatrix}$	



Estrella –Delta	
$n_t = \frac{VLN_{no\ min\ al\ _lado_alta}}{VLL_{no\ min\ al\ _lado_baja}}$	
$A_t = \frac{1}{3n_t} \begin{bmatrix} 2 & 1 & 0 \\ 0 & 2 & 1 \\ 1 & 0 & 2 \end{bmatrix}$	$d_t = \frac{1}{3n_t} \begin{bmatrix} 1 & -1 & 0 \\ 1 & 2 & 0 \\ -2 & -1 & 0 \end{bmatrix}$
$B_t = \frac{1}{9} \begin{bmatrix} 2Zt_{ab} + Zt_{bc} & 2Zt_{bc} - 2Zt_{ab} & 0 \\ 2Zt_{bc} - 2Zt_{ca} & 4Zt_{bc} - Zt_{ca} & 0 \\ Zt_{ab} - 4Zt_{ca} & -Zt_{ab} - 2Zt_{ca} & 0 \end{bmatrix}$	
Estrella aterrizada - Estrella aterrizada	
$n_t = \frac{VLG_{no\ min\ al\ _lado_alta}}{VLG_{no\ min\ al\ _lado_baja}}$	
$A_t = \frac{1}{n_t} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$d_t = \frac{1}{n_t} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
$B_t = \begin{bmatrix} Zt_a & 0 & 0 \\ 0 & Zt_b & 0 \\ 0 & 0 & Zt_c \end{bmatrix}$	
Delta – Delta	
$n_t = \frac{VLL_{no\ min\ al\ _lado_alta}}{VLL_{no\ min\ al\ _lado_baja}}$	
$A_t = \frac{1}{3n_t} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$	$d_t = \frac{1}{n_t} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
$B_t = \frac{1}{3(Zt_{ab} + Zt_{bc} + Zt_{ca})} \begin{bmatrix} 2Zt_{ab} & Zt_{bc} & 0 \\ 0 & 2Zt_{bc} & Zt_{ca} \\ Zt_{ab} & 0 & 2Zt_{ca} \end{bmatrix} \begin{bmatrix} Zt_{ca} & -Zt_{bc} & 0 \\ Zt_{ca} & Zt_{ab} + Zt_{ca} & 0 \\ -(Zt_{ab} + Zt_{bc}) & -Zt_{bc} & 0 \end{bmatrix}$	



Estrella abierta - Delta abierta	
$n_t = \frac{VLN_{no\ min\ al\ _lado_alta}}{VLL_{no\ min\ al\ _lado_baja}}$	
$A_t = \frac{1}{3n_t} \begin{bmatrix} 2 & 1 & 0 \\ -1 & 1 & 0 \\ -1 & -2 & 0 \end{bmatrix}$	$d_t = \frac{1}{n_t} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}$
$B_t = \frac{1}{3} \begin{bmatrix} 2Zt_{ab} & 0 & -Zt_{bc} \\ -Zt_{ab} & 0 & -Zt_{bc} \\ -Zt_{ab} & 0 & 2Zt_{bc} \end{bmatrix}$	

A.5. Modelado de cargas

Las cargas en un sistema de distribución generalmente se expresan como la potencia que se consume en un punto. Estas cargas pueden ser trifásicas, bifásicas o monofásicas y a su vez pueden estar balanceadas o desbalanceadas. Otro aspecto a considerar es que pueden ser cargas puntuales (que se conectan en un punto específico de la red) o cargas distribuidas (que se encuentran a lo largo de una sección de la línea), en el caso de estas últimas, se puede considerar que toda la carga se encuentra concentrada en un nodo ficticio localizado a la mitad de la línea o tomar la mitad de las cargas distribuidas en cada uno de los extremos de la línea. Finalmente las cargas se modelan como cargas de potencia constante (PQ), corriente constante (I) o impedancia constante (Z), cada modelo tiene un comportamiento diferente en la red e incluso se pueden considerar cargas mixtas que incluyan dos o los tres tipos de carga.

En la tabla A.2 se condensan las ecuaciones para obtener las corrientes de línea de cada configuración de carga obtenidas en [1].

Tabla A.2 Ecuaciones de corriente de línea para diferentes tipos de carga.

Conexión Tipo	ESTRELLA	DELTA
PQ	$I_{L_a} = \frac{ S_a }{ V_{an} } \angle \delta_a - \theta_a = I_{L_a} \angle \alpha_a$ $I_{L_b} = \frac{ S_b }{ V_{bn} } \angle \delta_b - \theta_b = I_{L_b} \angle \alpha_b$ $I_{L_c} = \frac{ S_c }{ V_{cn} } \angle \delta_c - \theta_c = I_{L_c} \angle \alpha_c$	$I_{L_{ab}} = \frac{ S_{ab} }{ V_{ab} } \angle \delta_{ab} - \theta_{ab} = I_{L_{ab}} \angle \alpha_{ab}$ $I_{L_{bc}} = \frac{ S_{bc} }{ V_{bc} } \angle \delta_{bc} - \theta_{bc} = I_{L_{bc}} \angle \alpha_{bc}$ $I_{L_{ca}} = \frac{ S_{ca} }{ V_{ca} } \angle \delta_{ca} - \theta_{ca} = I_{L_{ca}} \angle \alpha_{ca}$
Z	$I_{L_a} = \frac{ V_{an} }{ Z_a } \angle \delta_a - \theta_a = I_{L_a} \angle \alpha_a$ $I_{L_b} = \frac{ V_{bn} }{ Z_b } \angle \delta_b - \theta_b = I_{L_b} \angle \alpha_b$ $I_{L_c} = \frac{ V_{cn} }{ Z_c } \angle \delta_c - \theta_c = I_{L_c} \angle \alpha_c$	$I_{L_{ab}} = \frac{ V_{ab} }{ Z_{ab} } \angle \delta_{ab} - \theta_{ab} = I_{L_{ab}} \angle \alpha_{ab}$ $I_{L_{bc}} = \frac{ V_{bc} }{ Z_{bc} } \angle \delta_{bc} - \theta_{bc} = I_{L_{bc}} \angle \alpha_{bc}$ $I_{L_{ca}} = \frac{ V_{ca} }{ Z_{ca} } \angle \delta_{ca} - \theta_{ca} = I_{L_{ca}} \angle \alpha_{ca}$
I	$I_{L_a} = I_{L_a} \angle \delta_a - \theta_a$ $I_{L_b} = I_{L_b} \angle \delta_b - \theta_b$ $I_{L_c} = I_{L_c} \angle \delta_c - \theta_c$	$I_{L_{ab}} = I_{L_{ab}} \angle \delta_{ab} - \theta_{ab}$ $I_{L_{bc}} = I_{L_{bc}} \angle \delta_{bc} - \theta_{bc}$ $I_{L_{ca}} = I_{L_{ca}} \angle \delta_{ca} - \theta_{ca}$

A.6. Modelado de capacitores en derivación

Estos dispositivos se usan para el control de voltaje y compensación de potencia reactiva. Estos bancos de capacitores se modelan como susceptancias constantes conectadas tanto en estrella como en delta.

La susceptancia de un banco de capacitores en estrella se determina mediante:

$$B_C = \frac{kVAr}{kV_{LN}^2 \cdot 1000} S \quad (A.21)$$

Las corrientes de línea se obtienen de:

$$I_{C_a} = jB_a \cdot V_{an} \quad (A.22)$$

$$IC_b = jB_b \cdot V_{bn} \quad (A.23)$$

$$IC_c = jB_c \cdot V_{cn} \quad (A.24)$$

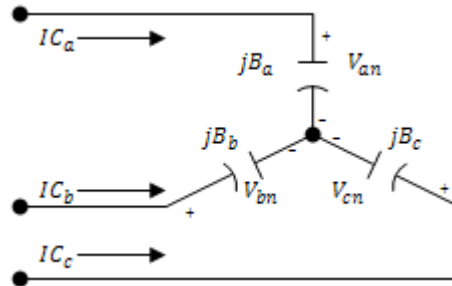


Figura A.3 Banco de capacitores en estrella

La susceptancia de un banco de capacitores en delta se determina mediante:

$$B_C = \frac{kVAr}{kV_{LL}^2 \cdot 1000} S \quad (A.25)$$

Y finalmente las corrientes de línea se obtienen de:

$$\begin{bmatrix} IL_a \\ IL_b \\ IL_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \cdot \begin{bmatrix} IL_{ab} \\ IL_{bc} \\ IL_{ca} \end{bmatrix} \quad (A.26)$$

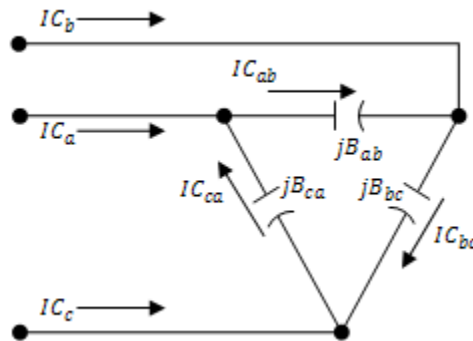


Figura A.4 Banco de capacitores en delta

A.7. Modelado de reguladores de voltaje

Los reguladores de voltaje son autotransformadores con un ajustador automático de cambiadores de derivación (TAP). Comúnmente, los reguladores proveen un rango de -10% a 10% a 32 pasos de $\frac{5}{8}$ % por paso. Estos reguladores pueden ayudar a mejorar el perfil de voltaje en puntos específicos de la red. Existen dos conexiones para estos reguladores, la conexión Tipo A y la conexión Tipo B de acuerdo a la norma ANSI/IEEE C57.15 [24].

A.7.1. Regulador monofásico Tipo A

El circuito equivalente detallado y el circuito equivalente abreviado de un regulador de voltaje a pasos Tipo A en posición elevador se muestran en la figura A.5. El circuito primario del sistema se conecta directamente al devanado en derivación del regulador Tipo A. El devanado serie se conecta al devanado en derivación y, a su vez, mediante el cambiador de derivaciones, al circuito regulado. En esta conexión la excitación del núcleo varía porque el devanado en derivación se conecta directamente al circuito primario.

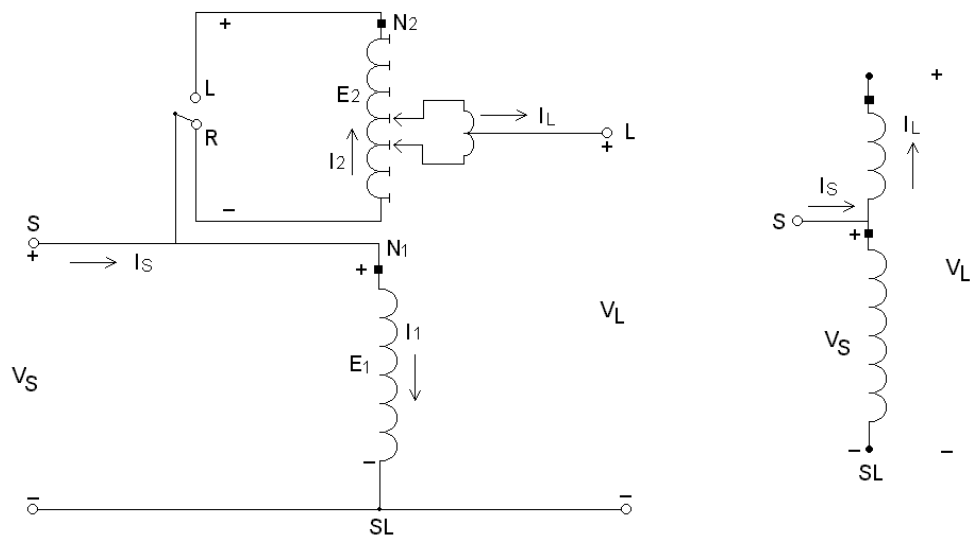


Figura A.5 Regulador monofásico Tipo A en posición elevador

A.7.2. Regulador monofásico Tipo B.

La conexión más común para los reguladores de voltaje a pasos es el Tipo B. Los circuitos equivalentes detallado y abreviado de un regulador de voltaje a pasos Tipo B en la posición elevador se muestran en la figura A.6. El circuito primario del sistema se conecta, mediante el cambiador de derivaciones, al devanado serie del regulador en la conexión

Tipo B. El devanado serie se conecta al devanado en paralelo, que se conecta directamente al circuito regulado. En un regulador Tipo B, la excitación del núcleo del regulador es constante porque el devanado en derivación se conecta al circuito regulado.

Las ecuaciones de voltaje y corriente para el regulador en posición elevador se muestran en la tabla A.3.

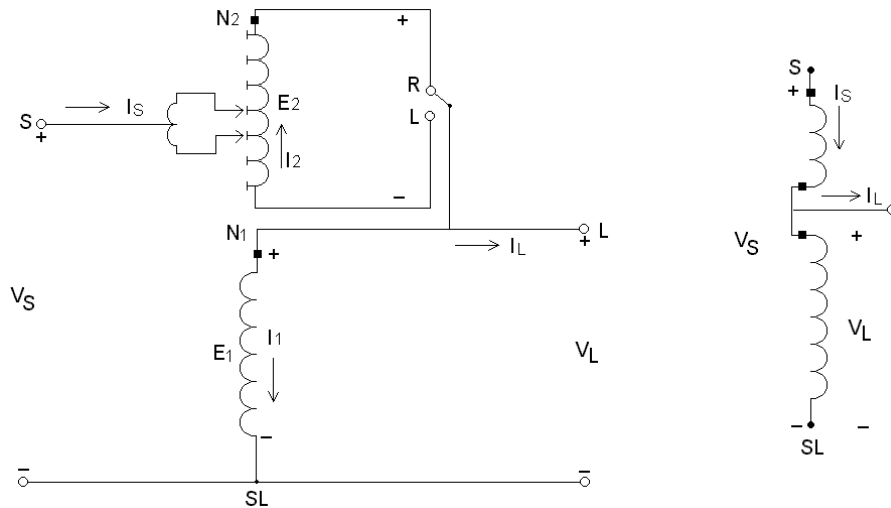


Figura A.5.6 Regulador monofásico Tipo B en posición elevador

Tabla A.5.3 Ecuaciones de voltaje y corriente para el regulador en posición elevador

Ecuaciones de voltaje	Ecuaciones de corriente
$\frac{E_1}{N_1} = \frac{E_2}{N_2}$	$N_1 I_1 = N_2 I_2$
$V_s = E_1 - E_2$	$I_L = I_s - I_1$
$V_L = E_1$	$I_2 = I_s$
$E_2 = \frac{N_2}{N_1} E_1 = \frac{N_2}{N_1} V_L$	$I_1 = \frac{N_2}{N_1} I_2 = \frac{N_2}{N_1} I_s$
$V_s = \left(1 - \frac{N_2}{N_1}\right) V_L$	$I_L = \left(1 - \frac{N_2}{N_1}\right) I_s$
$V_s = a_R V_L$	$I_L = a_R I_s$



$$a_R = 1 - \frac{N_2}{N_1}$$

En el caso de la posición reductor la dirección de las corrientes del devanado serie y en derivación cambian, pero la polaridad del voltaje de los dos devanados permanece igual. Las ecuaciones para esta posición se muestran en la tabla A.4.

Tabla A.5.4 Ecuaciones de voltaje y corriente para el regulador en posición reductor.

Ecuaciones de voltaje	Ecuaciones de corriente
$\frac{E_1}{N_1} = \frac{E_2}{N_2}$	$N_1 I_1 = N_2 I_2$
$V_s = E_1 + E_2$	$I_L = I_s + I_1$
$V_L = E_1$	$I_2 = I_s$
$E_2 = \frac{N_2}{N_1} E_1 = \frac{N_2}{N_1} V_L$	$I_1 = \frac{N_2}{N_1} I_2 = \frac{N_2}{N_1} I_s$
$V_s = \left(1 + \frac{N_2}{N_1}\right) V_L$	$I_L = \left(1 + \frac{N_2}{N_1}\right) I_s$
$V_s = a_R V_L$	$I_L = a_R I_s$
$a_R = 1 + \frac{N_2}{N_1}$	

De estas ecuaciones se observa que la única diferencia para las ecuaciones de corriente y voltaje del regulador Tipo B en las posiciones elevador y reductor, es el signo de la relación de transformación. Sin embargo para ponerlo en función de la posición del cambiador de derivaciones, se considera que cada posición cambia el voltaje un $\frac{5}{8} \%$ o 0.00625 por unidad. Por lo tanto la relación de transformación se puede expresar como

$$a_R = 1 \mp 0.00625 \cdot Tap \tag{A.27}$$

Para ambos reguladores, el Tipo A y el Tipo B, la relación entre el voltaje y corriente de la fuente al voltaje y corriente de carga se da mediante las ecuaciones A.28 y A.29:



$$\text{Tipo A: } V_S = \frac{1}{a_R} V_L \quad I_S = a_R I_L \quad (\text{A.28})$$

$$\text{Tipo B: } V_S = a_R V_L \quad I_S = \frac{1}{a_R} I_L \quad (\text{A.29})$$

Por consiguiente, las constantes generalizadas para el regulador monofásico de voltaje a pasos son:

$$\text{Tipo A: } a = \frac{1}{a_R} \quad b = 0 \quad c = 0 \quad d = a_R \quad (\text{A.30})$$

$$\text{Tipo B: } a = a_R \quad b = 0 \quad c = 0 \quad d = \frac{1}{a_R} \quad (\text{A.31})$$

Donde a_R se da por la ecuación A.27 y el signo se asigna de acuerdo a la tabla A.5.

Tabla A.5.5 Convención de signos para a_R de la ecuación A.27

Posición \ Tipo	Tipo A	Tipo B
Elevador	+	-
Reductor	-	+

A.7.3. Reguladores trifásicos

Al conectar tres reguladores monofásicos de forma externa se pueden formar reguladores trifásicos. De esta forma se pueden tener diferentes conexiones, las conexiones más típicas son:

- Monofásico
- Dos reguladores conectados en estrella abierta
- Tres reguladores conectados en estrella aterrizada
- Dos reguladores conectados en delta abierta
- Tres reguladores conectados en delta cerrada

Las ecuaciones con las que se obtienen las corrientes de línea se mantienen en la misma forma que se maneja en las ecuaciones A.7 y A.8, por lo que las matrices generalizadas se



expresan en la tabla A.6, las matrices b y c se consideran cero debido a que la impedancia serie y la admitancia en derivación de cada regulador se desprecia.

Tabla A.5.6 Matrices generalizadas para reguladores

Conexión	Tipo A	Tipo B
Estrella aterrizada	$a = \begin{bmatrix} \frac{1}{a_{Ra}} & 0 & 0 \\ 0 & \frac{1}{a_{Rb}} & 0 \\ 0 & 0 & \frac{1}{a_{Rc}} \end{bmatrix}$	$a = \begin{bmatrix} a_{Ra} & 0 & 0 \\ 0 & a_{Rb} & 0 \\ 0 & 0 & a_{Rc} \end{bmatrix}$
	$d = \begin{bmatrix} a_{Ra} & 0 & 0 \\ 0 & a_{Rb} & 0 \\ 0 & 0 & a_{Rc} \end{bmatrix}$	$d = \begin{bmatrix} \frac{1}{a_{Ra}} & 0 & 0 \\ 0 & \frac{1}{a_{Rb}} & 0 \\ 0 & 0 & \frac{1}{a_{Rc}} \end{bmatrix}$
Delta cerrada	$a = \begin{bmatrix} \frac{1}{a_{R_{ab}}} & 1 - \frac{1}{a_{R_{bc}}} & 0 \\ 0 & \frac{1}{a_{R_{bc}}} & 1 - \frac{1}{a_{R_{ca}}} \\ 1 - \frac{1}{a_{R_{ab}}} & 0 & \frac{1}{a_{R_{ca}}} \end{bmatrix}$	$a = \begin{bmatrix} a_{R_{ab}} & 1 - a_{R_{bc}} & 0 \\ 0 & a_{R_{bc}} & 1 - a_{R_{ca}} \\ 1 - a_{R_{ab}} & 0 & a_{R_{ca}} \end{bmatrix}$
	$d^{-1} = \begin{bmatrix} \frac{1}{a_{R_{ab}}} & 1 - \frac{1}{a_{R_{bc}}} & 0 \\ 0 & \frac{1}{a_{R_{bc}}} & 1 - \frac{1}{a_{R_{ca}}} \\ 1 - \frac{1}{a_{R_{ab}}} & 0 & \frac{1}{a_{R_{ca}}} \end{bmatrix}$	$d^{-1} = \begin{bmatrix} a_{R_{ab}} & 1 - a_{R_{bc}} & 0 \\ 0 & a_{R_{bc}} & 1 - a_{R_{ca}} \\ 1 - a_{R_{ab}} & 0 & a_{R_{ca}} \end{bmatrix}$
Delta abierta	$a = \begin{bmatrix} a_{R_{ab}} & 0 & 0 \\ 0 & a_{R_{cb}} & 0 \\ -a_{R_{ab}} & -a_{R_{cb}} & 0 \end{bmatrix}$	$a = \begin{bmatrix} \frac{1}{a_{R_{ab}}} & 0 & 0 \\ 0 & \frac{1}{a_{R_{cb}}} & 0 \\ \frac{-1}{a_{R_{ab}}} & \frac{-1}{a_{R_{cb}}} & 0 \end{bmatrix}$



	$d = \begin{bmatrix} a_{R_{ab}} & 0 & 0 \\ -a_{R_{ab}} & 0 & -a_{R_{cb}} \\ 0 & 0 & a_{R_{cb}} \end{bmatrix}$	$d = \begin{bmatrix} \frac{1}{a_{R_{ab}}} & 0 & 0 \\ -1 & 0 & -1 \\ a_{R_{ab}} & a_{R_{cb}} & \frac{1}{a_{R_{cb}}} \\ 0 & 0 & \frac{1}{a_{R_{cb}}} \end{bmatrix}$
--	---	---

A.8. Flujos de potencia trifásicos por el método de barrido regresivo/progresivo

Debido a que las redes de transmisión y de distribución tienen topologías diferentes y al extenso grupo de elementos que se deben modelar en las redes de distribución, no es común usar los métodos de solución de flujos de potencia tradicionales de redes de transmisión, por lo que en su lugar se usa una técnica iterativa especialmente diseñada para un sistema radial que puede resolver sistemas trifásicos tanto balanceados como desbalanceados [25].

En este método primeramente se enumeran los nodos de la red iniciando con el nodo donde se encuentra la subestación, que a su vez se considera el nivel inicial, y a partir de ese nodo se buscan los nodos adyacentes que formen el siguiente nivel y así sucesivamente hasta que se tenga el nivel de cada nodo.

Como consideración inicial del problema los voltajes de todos los nodos de la red se suponen iguales al del nodo inicial, si existe algún transformador conectado se modifica el voltaje aguas debajo de acuerdo a la relación de transformación de ese equipo.

Una vez realizado lo anterior, se comienza la solución iterativa de este algoritmo, que consiste en tres pasos.

1. Cálculo de las corrientes nodales.

De acuerdo a las ecuaciones descritas en los apartados A.4 y A.5 se calculan las corrientes de las cargas y los capacitores en derivación que estuviesen conectados en cada nodo. Esto se expresa mediante la ecuación A.32.

$$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_m^{(k)} = \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{carga\ m}^{(k)} + \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{derivación\ m}^{(k)} \quad (A.32)$$



Donde:

$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_m^{(k)}$ Vector de corriente inyectada en el nodo m en la iteración k.

$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{carga\ m}^{(k)}$ Vector de corriente de la carga conectada en el nodo m en la iteración k.

$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{derivación\ m}^{(k)}$ Vector de corriente de los capacitores en derivación conectados al nodo m en la iteración k.

2. Cálculo de las corrientes de rama mediante un barrido regresivo.

Desde los nodos del último nivel hasta el nodo inicial, se suman las corrientes nodales de los nodos aguas abajo. Esto se representa mediante la ecuación A.33

$$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{nm}^{(k)} = [d] \left(\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_m^{(k)} + \sum \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{mp}^{(k)} \right) \quad (A.33)$$

Donde:

$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{nm}^{(k)}$ Vector de corrientes totales entre los nodos n y m

$[d]$ Matriz generalizada para el segmento nm. Esta depende del tipo de elemento conectado entre los nodos n y m.

$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_m^{(k)}$ Vector de corriente inyectada en el nodo m en la iteración k.



$\sum \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}_{mp}^{(k)}$ Sumatoria de los vectores de corrientes de todas los elementos conectados al nodo m desde un nivel superior.

3. Cálculo de los voltajes nodales mediante un barrido progresivo.

Con todas las corrientes de rama calculadas se actualizan los voltajes nodales desde el nodo inicial hasta los nodos en el último nivel. Esto se realiza mediante la ecuación A.34, donde las matrices A y B son las matrices generalizadas del elemento correspondiente conectado entre los nodos n y m

$$[V_{LG\ abc}]_m = [A][V_{LG\ ABC}]_n - [B][I_{abc}]_{nm} \quad (A.34)$$

Estos tres pasos se repiten hasta que el sistema converja. El criterio de convergencia se basa en la diferencia de potencia por fase entre la última iteración y la anterior.



APÉNDICE B

CÓDIGOS

B.1. Principal.f90

```

=====
!
!RADIFLUX 2.0
!
! FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION-inclusion de redes malladas
!
!05 feb 2010
!20 jul 2010-modificaciones para archivos de reconfig. & restaur.
!
!DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
!
!JUNIO 2012
=====
!
!IVAN ARCHUNDIA ARANDA
!iarchundia@ipn.mx
!
!RAUL GUILLERMO JARA SIXTOS
!rjaras0400@ipn.mx
!
=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
=====
!
PROGRAM RADIFLUX
!-----
USE DFLIB
USE DFPORT

use estructuras
use estructuras2
use estructuras3 !15 feb 2010

use rutinas_de_asignacion
use ConstantesG14
use ConstantesG2
use ConstantesG3
use Mcons
use PrintSalidaDetallada
USE DATA_DESPACHO

use BackForw
!-----
IMPLICIT NONE

!Declaracion de variables
!.....
character(20)      :: filename,archsai !Nombre del archivo de entrada
character(len=60)  :: header         !Encabezado del archivo de entrada
integer           :: n2              !Número de nodos de la red (incluido en nodo raiz)
integer           :: m2              !Numero de mallas
real, dimension(6) :: vrootr         !Voltaje del nodo Raiz (kV-LN para todos los
casos) vector de lectura
!complex, dimension(3) :: vrootc      !Procesa el anterior para hacerlo complejo y
asignarlo al nodo 1 (raiz)

```



```
type(lectura)::temp                                !Contenedor de Lectura de nodo
type(lista), allocatable, dimension(:)::nodo      !Datos de la Red para Lista de Barrido
type(lectura_mallas)::temp2                       !Contenedor de lectura de malla
type(lista_mallas), allocatable, dimension(:)::malla !Datos de estructura de malla
type(THV_est)::thv                                !*****
integer, allocatable, dimension(:)::VISIT !auxiliar para marcar nodos visitados (puede
estar en la estructura nodal) ojo!!
complex, allocatable, dimension(:,:):: Zthevenin
complex, allocatable, dimension(:,:):: Zthevenin_KPZ
complex, allocatable, dimension(:,:):: Ythevenin_KPZ
complex, allocatable, dimension(:,:):: Ythevenin
real, allocatable, dimension(:,:):: KPZ !Matriz de Compactacion
complex, allocatable, dimension(:)::Vthevenin
complex, allocatable, dimension(:)::Ithevenin
!.....
integer :: maxIter = 300 !No. Máximo de Iteraciones Permitidas (Cambiar
desde aqui)
real :: tolerancia = 0.01 !Tolerancia de error del proceso iterativo ((Cambiar
desde aqui)
integer :: iteraciones =0 !Iteraciones del algoritmo de Barrido (algoritmo de
flujos o iterativo)
integer :: converge =1 !numero acumulativo de convergencia de nodos (para
converger debe alcanzar el máximo n2)
!.....
integer :: iteracionesMacro=0
integer :: convergeMacro= -1 !ojo en el monofasico estaba en -1, for prevent one or no
meshes
!.....
integer::k,m, p, t, , q ,r, xx, yy, zz !contador (es)- variable (s) auxiliares
(no usar "i" ni "j")
integer:: extremo1=0, extremo2=0 !15feb2010
INTEGER :: CONTT,CONTC,caps
REAL(4) I, TA(2)
contt=0
contc=0
!-----
!-----INICIA CODIFICACION DE PROGRAMA PRINCIPAL-----
!-----
WRITE(*, '(/)')
WRITE(*, *) 'PROGRAMA DE DESPACHO DE POTENCIA REACTIVA PARA SISTEMAS DE DISTRIBUCION'
WRITE(*, '(/)')

WRITE(*, *) 'INSTITUTO POLITECNICO NACIONAL'
WRITE(*, *) 'ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA'
WRITE(*, *) 'SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION'

WRITE(*, '(/)')
WRITE(*, *) 'Programadores:Ivan Archundia Aranda, Raul Guillermo Jara Sixtos'
WRITE(*, *) ' iarchundia@ipn.mx rjaras0400@ipn.mx'
WRITE(*, '(/)')
!-----
!Introduccion y Apertura del Archivo de Entrada
WRITE(*, *) 'INTRODUCIR NOMBRE DEL ARCHIVO DE ENTRADA:' !introducir con extension
READ(*, *) fileName
!TRYTP: CONTINUE
CLOSE(UNIT=1)
open(1,file=fileName)
!-----
!lectura y asignación
read(1,*)header
read(1,*)n2
READ(1,*)transformers
READ(1,*)capacitors
READ(1,*)on_off
READ(1,*)tap_lim
! LEE EL TOTAL DE ESCENARIOS
READ(1,*)escenarios
ALLOCATE(porcentaje_de_carga(ESCENARIOS))
IF ( escenarios < on_off ) THEN
WRITE(*, *)'ERROR, EL NUMERO DE OPERACIONES ON_OFF DEBE SER'
WRITE(*, *)'MENOR O IGUAL AL NUMERO DE ESCENARIOS'
WRITE(*, *)' CORREGIR ARCHIVO DE DATOS '
PAUSE
ENDIF
! LEE EL PORCENTAJE DE CARGA DE CADA ESCENARIO.
```




```

nodo(k)%Z4, &
&
)
nodo(k)%d1, nodo(k)%A1, nodo(k)%B1,
nodo(k)%d4, nodo(k)%A4, nodo(k)%B4
)
call dab2 (nodo(k)%Z2, nodo(k)%d2, nodo(k)%A2, nodo(k)%B2)
call dab3 (nodo(k)%tybr,nodo(k)%nt, nodo(k)%Z3, nodo(k)%d3, nodo(k)%A3, nodo(k)%B3)
!Inicializacion voltajes nodales , corrientes nodales (normales y primadas) , corrientes
de incoming branch
!y admitancia total de linea NODAL Y2 (charge admittance)
nodo(k)%Y2=CMPLX(0.0, 0.0)
nodo(k)%V1=CMPLX(0.0, 0.0)
nodo(k)%V2=CMPLX(0.0, 0.0)
nodo(k)%V3=CMPLX(0.0, 0.0)
nodo(k)%V4=CMPLX(0.0, 0.0)
nodo(k)%I1=CMPLX(0.0, 0.0)
nodo(k)%I2=CMPLX(0.0, 0.0)
nodo(k)%I3=CMPLX(0.0, 0.0)
nodo(k)%I4=CMPLX(0.0, 0.0)
nodo(k)%Ibreak=CMPLX(0.0, 0.0) !23feb2010 -----""/&/&/%/&
nodo(k)%J1=CMPLX(0.0, 0.0)
nodo(k)%J2=CMPLX(0.0, 0.0)
nodo(k)%J3=CMPLX(0.0, 0.0)
nodo(k)%J4=CMPLX(0.0, 0.0)
nodo(k)%JP1=CMPLX(0.0, 0.0)
nodo(k)%JP2=CMPLX(0.0, 0.0)
nodo(k)%JP3=CMPLX(0.0, 0.0)
nodo(k)%JP4=CMPLX(0.0, 0.0)

enddo LECTURA_Y_ASIGNACION

!-----
CALCULO_Y2: DO k=1,n2

call calculoY2 (nodo(k)%Y, nodo(nodo(k)%pre)%Y2, nodo(k)%Y2)

ENDDO CALCULO_Y2

!-----

VOLTAJES_INICIALES: do k=1, n2

!Establecimiento de voltajes nodales iniciales
!Se calculan con un barrido progresivo en vacio tomando en cuenta los reguladores
!(Aqui se aprovecha que las condiciones iniciales de corriente son cero, porque así fueron
!inicializadas y en este caso permite hacer el barrido en vacio)

IF (k==1) then

nodo(k)%V1= Vrootc
nodo(k)%V2= Vrootc
nodo(k)%V3= Vrootc
nodo(k)%V4= Vrootc

ELSE

nodo(k)%V1= MATMUL (nodo(k)%A1, nodo(nodo(k)%pre)%V4)
nodo(k)%V2= MATMUL (nodo(k)%A2, nodo(k)%V1)
nodo(k)%V3= MATMUL (nodo(k)%A3, nodo(k)%V2)
nodo(k)%V4= MATMUL (nodo(k)%A4, nodo(k)%V3)

ENDIF

ENDDO VOLTAJES_INICIALES

!-----

VOLTAJES_REFERENCIA: do k=1, n2
!Establecimiento de voltajes nodales de referencia
!Se calculan con un barrido progresivo en vacio tomando SIN TOMAR en cuenta los reguladores
!Estos voltajes se supone serviran para calcular los voltajes en pu para cada nodo.
!Se supone que se aprovecha el hecho de que la fuente es simetrica y balanceada--ojo-- yuo
know!
!(Aqui se aprovecha que las condiciones iniciales de corriente son cero, porque así fueron
!inicializadas y en este caso permite hacer el barrido en vacio)

```



```

IF (k==1) then

nodo(k)%VR1= Vrootc
nodo(k)%VR2= Vrootc
nodo(k)%VR3= Vrootc
nodo(k)%VR4= Vrootc

ELSE

nodo(k)%VR1= MATMUL(U3C      , nodo(nodo(k)%pre)%VR4)
nodo(k)%VR2= MATMUL(nodo(k)%A2, nodo(k)%VR1)
nodo(k)%VR3= MATMUL(nodo(k)%A3, nodo(k)%VR2)
nodo(k)%VR4= MATMUL(U3C      , nodo(k)%VR3)

ENDIF

ENDDO VOLTAJES_REFERENCIA
!-----
!Imprime Entrada

!call EntradaDetallada (filename,header, Vrootr,Vrootc, n2, nodo)

!-----
!Algoritmo de Barrido
!#####
#
!05 FEBRERO 2010
!Anexión para MAYATOR

read(1,*)m2

CONDITION1: if(m2>0) then

allocate(malla(m2))

LecturaAsignacionMayas:do k=1, m2
  read(1,*) temp2%m
  read(1,*) temp2%from
  read(1,*) temp2%to
  read(1,*) temp2%Z
  read(1,*) temp2%Y

  malla(k)%m   = temp2%m
  malla(k)%from = temp2%from
  malla(k)%to   = temp2%to
  call conversor_Zm(temp2%Z, malla(k)%Z)
  call conversor_Y (temp2%Y, malla(k)%Y)

enddo LecturaAsignacionMayas
!suma la capacitancia de las mallas a los nodos de la red radial
do zz=1, m2

nodo(malla(zz)%from)%Y2= nodo(malla(zz)%from)%Y2 + (1.0 / 2.0)*malla(zz)%Y
nodo(malla(zz)%to )%Y2= nodo(malla(zz)% to )%Y2 + (1.0 / 2.0)*malla(zz)%Y

enddo
!***** 15 feb 2010
allocate( thv%I(m2), thv%V(m2), thv%Z(m2,m2) )
!Se inicializan las corrientes thv y los voltajes thv con cero
do k=1, m2
thv%I(k)%v31=cplx(0.0, 0.0) !corregido 25feb2010
thv%V(k)%v31=cplx(0.0, 0.0)
enddo
!Se inicializa las impeadncias thv%Z con cero por si las moscas (por si no hay mallas)
do k=1, m2
  do m=1, m2
    thv%Z(k,m)%m33=cplx(0.0, 0.0)
  enddo
enddo

!Se forma ahora la matriz thv%Z, compound con el algoritmo monofásico
!-----
!Se mete a la diagonal el valor de las impedancias de rama

do k=1, m2

```



```

        thv%Z(k,k)%m33 = malla(k)%Z
enddo
!asigna memoria e inicializa el vector de visitas de nodo
allocate(VISIT(n2))
VISIT=0
!-----
EXTERIOR: DO p=1, m2
VISIT=0

!Empieza con la ruta del "FROM"

t= malla(p)%from

do while(t>1) !Haga mientras no se ha alcanzado el nodo raiz
    thv%Z(p,p)%m33=thv%Z(p,p)%m33 + nodo(t)%Z
    visit(t)=1
    t=nodo(t)%pre
enddo
!y se sigue con la ruta del "TO"
t= malla(p)%to
do while(t>1) !Haga mientras no se ha alcanzado el nodo raiz
IF(visit(t)==0) then
    thv%Z(p,p)%m33=thv%Z(p,p)%m33 + nodo(t)%Z
    visit(t)=1
    t=nodo(t)%pre
ELSE !el nodo ya fue visitao
    thv%Z(p,p)%m33=thv%Z(p,p)%m33 - nodo(t)%Z
    visit(t)=0
    t=nodo(t)%pre
ENDIF
enddo

        INTERIOR:DO q=p+1, m2
        extremo1=0
        extremo2=0
        !***
        !determinación de mallas internas
        !-----
        t= malla(p)%from
        do while(t>1)
        if(t==malla(q)%from .OR. t==malla(q)%to) extremo1=t
        t=nodo(t)%pre
        enddo

        t= malla(p)%to
        do while(t>1)
        if(t==malla(q)%from .OR. t==malla(q)%to) extremo2=t
        t=nodo(t)%pre
        enddo
        !-----

        t= malla(q)%from
        do while(t>1)
        if(t==malla(p)%from .OR. t==malla(p)%to) extremo1=t
        t=nodo(t)%pre
        enddo

        t= malla(q)%to
        do while(t>1)
        if(t==malla(p)%from .OR. t==malla(p)%to) extremo2=t
        t=nodo(t)%pre
        enddo
        !***

        t= malla(q)%from
        do while(t>1)
        if (visit(t)==1) thv%Z(p,q)%m33= thv%Z(p,q)%m33 + nodo(t)%Z
        t=nodo(t)%pre
        enddo

        t= malla(q)%to
        do while(t>1)

        if(extremo1>0 .AND. extremo2>0 .AND. extremo1/=extremo2) then
            if(visit(t)==1) thv%Z(p,q)%m33= thv%Z(p,q)%m33 + nodo(t)%Z
            t=nodo(t)%pre

```



```

else
    if(visit(t)==1)      thv%Z(p,q)%m33= thv%Z(p,q)%m33 - nodo(t)%Z
                        t=nodo(t)%pre
    endif
enddo

!CORRECCION PARA Numeros NEGATIVOS
do xx=1, 3
do yy=1, 3
if ( REAL( thv%Z(p,q)%m33(xx, yy)) < 0.0 ) thv%Z(p,q)%m33(xx, yy)= -
thv%Z(p,q)%m33(xx, yy)      !ojo aca, no es tan directo

enddo
enddo

!WHERE ( real(thv%Z(p,q)%m33) < 0.0)  thv%Z(p,q)%m33= -thv%Z(p,q)%m33
!simetria

thv%Z(q,p)%m33 =thv%Z(p,q)%m33

ENDDO INTERIOR

ENDDO EXTERIOR
!!print*, 'matriz z'
!!print*, thv%Z
!Se asigna espacio para la matriz thevenin bruta y para los vectores de voltaje y corriente thevenin

allocate(Zthevenin (3*m2,3*m2))

allocate( vthevenin (3*m2) )
allocate( ithevenin (3*m2) )

!Se inicializan los vectores de voltaje/corriente Thevenin

vthevenin = cmplx(0.0, 0.0)
ithevenin = cmplx(0.0, 0.0)

!y se copian valores a Zthevenin (desde la matriz compound)

Do p=1, m2
Do q=1, m2

    zthevenin(3*p-2:3*p , 3*q-2:3*q) = thv%Z(p,q)%m33

Enddo
Enddo
!!call WRCRN('Zthevenin',3*m2, 3*m2, Zthevenin, 3*m2, 0)      !quitar, solo prueba
!-----
!PROCESO DE KOMPACTAZION*****
!Se cuentan los elementos no nulos en la diagonal de Zth

r=0
do k=1,3*m2
    IF( Zthevenin(k,k) /= cmplx(0.0, 0.0) ) r=r+1
enddo
!!print*, 'valor de R--*--*--*--*--*--*--*--*--*--*'
!!print*, r

!Se asigna memoria para la matriz de compactacion y las matrices compactadas de Z e Y

allocate(KPZ(r,3*m2))
allocate(Zthevenin_KPZ(r,r))
allocate(Ythevenin_KPZ(r,r))
allocate(Ythevenin (3*m2,3*m2))

!se llama a rutina de Kompactación para conocer la matriz KPZ .....18feb2010

KPZ=0.0      !INICIALIZACION DE KOMPACTAZION
call KOMPACTAZION(Zthevenin,3*m2,r,KPZ)

! y se hacen las transformaciones

```



```

Zthevenin_KPZ = matmul(matmul(KPZ,Zthevenin),transpose(KPZ))

call LINGC(r,Zthevenin_KPZ,r,Ythevenin_KPZ,r)      !obtiene la matriz Ythevenin compactada
invertiendo Zthevenin

Ythevenin = matmul(matmul(transpose(KPZ),Ythevenin_KPZ),KPZ)  !and ahora se descompacta
para Y

endif CONDITION1

!normalizacion a cantidaes reales      (de kV a V  y de kW a w)

NORMALIZACION: do k=1,n2

nodo(k)%V1=      nodo(k)%V1      * 1000.00
nodo(k)%V2=      nodo(k)%V2      * 1000.00
nodo(k)%V3=      nodo(k)%V3      * 1000.00
nodo(k)%V4=      nodo(k)%V4      * 1000.00

nodo(k)%VR1=     nodo(k)%VR1     * 1000.00
nodo(k)%VR2=     nodo(k)%VR2     * 1000.00
nodo(k)%VR3=     nodo(k)%VR3     * 1000.00
nodo(k)%VR4=     nodo(k)%VR4     * 1000.00

nodo(k)%S1YS =   nodo(k)%S1YS   * 1000.00
nodo(k)%S1YI =   nodo(k)%S1YI   * 1000.00
nodo(k)%S1YZ =   nodo(k)%S1YZ   * 1000.00
nodo(k)%S1DS =   nodo(k)%S1DS   * 1000.00
nodo(k)%S1DI =   nodo(k)%S1DI   * 1000.00
nodo(k)%S1DZ =   nodo(k)%S1DZ   * 1000.00

nodo(k)%S2YS =   nodo(k)%S2YS   * 1000.00
nodo(k)%S2YI =   nodo(k)%S2YI   * 1000.00
nodo(k)%S2YZ =   nodo(k)%S2YZ   * 1000.00
nodo(k)%S2DS =   nodo(k)%S2DS   * 1000.00
nodo(k)%S2DI =   nodo(k)%S2DI   * 1000.00
nodo(k)%S2DZ =   nodo(k)%S2DZ   * 1000.00

nodo(k)%S3YS =   nodo(k)%S3YS   * 1000.00
nodo(k)%S3YI =   nodo(k)%S3YI   * 1000.00
nodo(k)%S3YZ =   nodo(k)%S3YZ   * 1000.00
nodo(k)%S3DS =   nodo(k)%S3DS   * 1000.00
nodo(k)%S3DI =   nodo(k)%S3DI   * 1000.00
nodo(k)%S3DZ =   nodo(k)%S3DZ   * 1000.00

nodo(k)%S4YS =   nodo(k)%S4YS   * 1000.00
nodo(k)%S4YI =   nodo(k)%S4YI   * 1000.00
nodo(k)%S4YZ =   nodo(k)%S4YZ   * 1000.00
nodo(k)%S4DS =   nodo(k)%S4DS   * 1000.00
nodo(k)%S4DI =   nodo(k)%S4DI   * 1000.00
nodo(k)%S4DZ =   nodo(k)%S4DZ   * 1000.00

nodo(k)%Y2 =     nodo(k)%Y2     * 1.0E-6  !se normaliza a Siemens por que estaba en
micro-siemens

enddo NORMALIZACION

archsal=TRIM(filename) // '_desp' //''.res'
OPEN (UNIT= 101, FILE= archsal)
!SE COPIAN LOS VALORES ORIGINALES DE CAPACITORES Y CARGAS

do k=1,CAPACITORS
  DO caps=1,3
    CAP_OR(K,CAPS)=NODO(UB_CAP(K))%S4YZ(CAPS)
    CAP_OR2(K,CAPS)=NODO(UB_CAP(K))%SBC(CAPS)
  END DO
end do

do k=1,n2
  DO caps=1,3
    LOADC_OR(K,CAPS)=NODO(k)%S4YS(CAPS)
    LOADC_OR2(K,CAPS)=NODO(k)%SC(CAPS)
  END DO !FALTA AGREGAR CARGA DISTRIBUIDA
end do

```



```

ALLOCATE(TRAYC_ANT(ESCAPENARIOS),TRAYC(ESCAPENARIOS),TRAYT(TRANSFORMERS,ESCAPENARIOS),TRAYT_ANT(T
RANSFORMERS,ESCAPENARIOS))
TRAYC=on_off+1
TRAYT=0
TRAYT_ANT=-20

```

```

call BackwardForward(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2, Ithevenin,
Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro ) !23feb2010
archsal=TRIM(filename) // '_base' //'_res'
call SalidaDetallada(archsal, n2,m2, nodo, iteraciones, iteracionesMacro, converge,
header)

```

```

DO K=1,10
  Q=1
  TRAYC_ANT=TRAYC
  IF (CAPACITORS.GT.0) THEN
    DO P=1,escenarios
      IF (TRANSFORMERS.GT.0) THEN
        IF(TRAYT_ANT(1,P).NE.TRAYT(1,P)) THEN
          call DESPACHOC(nodo, n2, maxIter, iteraciones, converge, tolerancia,
m2, Ithevenin, Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro )
          Q=0
          EXIT
        END IF
      ELSE
        call DESPACHOC(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2,
Ithevenin, Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro )
        EXIT
      END IF
    END DO
    IF(Q.EQ.1)THEN
      WRITE(101,*)
      WRITE(101,*) 'CONVERGE EN ',K-1,' ITERACIONES'
      CALL SALIDATOTAL(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2,
Ithevenin, Ythevenin, Vthevenin, malla, thv, iteracionesMacro,
convergeMacro,filename,header )
      EXIT
    END IF
    TRAYT_ant=TRAYT
    IF (TRANSFORMERS.GT.0) THEN
      CALL DESPACHOT(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2,
Ithevenin, Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro )
    END IF
  END DO

```

```

WRITE(*,'(/)')

```

```

I = ETIME(TA)
write(*,*) 'Program has used', I, 'seconds of CPU time.'
write(101,*) 'Program has used', I, 'seconds of CPU time.'
!-----
END PROGRAM RADIFLUX

```

```

SUBROUTINE KOMPACTAZION(M,n,r,KPZ)

```

```

!Argumentos de Entrada Salida
complex, intent(in ), dimension(n,n)::M           !Matriz a compactar
integer, intent(in )           ::n                !Orden de la matriz "M"
integer, intent(in )           ::r                !No. Renglones de la Matriz KPZ
real, intent(out), dimension(r,n)::KPZ          !Matriz de Compactación
!Variable(s) internas
integer:: i,j
!-----**-----**-----**-----**-----**-----**-----**-----**-----**

```

```

i=1
do j=1, n
if(M(j,j) /= cmplx(0.0, 0.0)) then
  KPZ(i,j)= 1.0           !cmplx(1.0, 0.0)
  i=i+1

```



endif
enddo

END SUBROUTINE KOMPACTAZION

B.2. Barrido.f90

```

=====
!
!RADIFLUX 1.0
!
! FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
!20 de DICIEMBRE de 2008
!
=====
!
!IVAN ARCHUNDIA ARANDA
!iarchundia@ipn.mx
!
=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!
=====
MODULE BackForw                                !Algoritmo BackwardForward Trifásico
!+++++
CONTAINS

Subroutine BackwardForward(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2,
Ithevenin, Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro )
!23feb2010

USE MSIMSL
USE estructuras
USE estructuras2
USE estructuras3
USE DATA_DESPACHO
implicit none

!Declaracion de Variables I/O
integer, intent(in)::n2
type(lista), dimension(n2),intent(inout) :: nodo    !INOUT para meter todo y sacar voltaje

integer, intent(in)    :: maxIter
integer, intent(inout) :: iteraciones

integer, intent(inout) :: converge
real    , intent(in)   :: tolerancia

!23feb2010
integer, intent(in)                :: m2
complex, dimension(3*m2)           , intent(inout):: Ithevenin
complex, dimension(3*m2, 3*m2)    , intent(in  ):: Ythevenin
complex, dimension(3*m2)           , intent(inout):: Vthevenin

type(lista_mallas), dimension(m2), intent(in):: malla    !24feb2010
type(THV_est), intent(inout)      :: thv                !24feb2010--
!03mar2010

integer, intent(inout)::iteracionesMacro
integer, intent(inout)::convergeMacro

!Declaracion de Variables internas
integer ::k, t    !Contador(es)

complex, dimension(3):: V4antes
complex, dimension(3):: ERROR

```



```
!%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
complex, dimension(3):: Slos
```

```
do k=2, n2
  do t=1,3
    if( nodo(k)%Z3(t,t) == CMLX(0.0, 0.0) .AND. nodo(k)%tybr /= -1 ) nodo(k)%V4(t)=
nodo(k)%VR4(t)
  enddo
enddo
```

```
Slos = (0.0 , 0.0)
```

```
MACROITERACIONES: do while(convergeMacro<m2 .AND. iteracionesMacro<MaxIter)
```

```
convergeMacro=0
```

```
converge=1 !se pone aqui provisionalmente porque converge a la primera MacroIteracion &
!y no deja que el ciclo de Macroiteraciones vuelva a entrar, !pudiera
inicializarse tambien con cero.
```

```
CICLO_DE_ITERACIONES: DO WHILE ( converge < n2 .and. iteraciones < maxIter)
```

```
converge=1
```

```
!=====
!=====
```

```
Inyecciones_Nodales: DO k=1,n2
```

```
!!!Contribucion de las cargas conectadas en Estrella en todos los modelos (ZIP)
```

```
nodo(k)%I2= CONJG (nodo(k)% S2YS / nodo(k)% V2) + &
CONJG (nodo(k)% S2YI / nodo(k)% VR2) * &
EXP (CMLX(0.0, ATAN2( IMAG(nodo(k)% VR2),REAL(nodo(k)% VR2))-ATAN2(
IMAG(nodo(k)% V2),REAL(nodo(k)% V2)) ) ) + &
CONJG (nodo(k)% S2YZ / nodo(k)% V2) * (ABS( nodo(k)% V2 / nodo(k)% VR2)) ** 2.0
```

```
!!!Contribucion de la delta con potencia constante
```

```
nodo(k)%I2(1) = nodo(k)%I2 (1) + &
CONJG(nodo(k)% S2DS(1) / (nodo(k)% V2(1)-nodo(k)% V2(2))) - CONJG(nodo(k)%
S2DS(3) / (nodo(k)% V2(3)-nodo(k)% V2(1)))
nodo(k)%I2(2) = nodo(k)%I2 (2) + &
CONJG(nodo(k)% S2DS(2) / (nodo(k)% V2(2)-nodo(k)% V2(3))) - CONJG(nodo(k)%
S2DS(1) / (nodo(k)% V2(1)-nodo(k)% V2(2)))
nodo(k)%I2(3) = nodo(k)%I2 (3) + &
CONJG(nodo(k)% S2DS(3) / (nodo(k)% V2(3)-nodo(k)% V2(1))) - CONJG(nodo(k)%
S2DS(2) / (nodo(k)% V2(2)-nodo(k)% V2(3)))
```

```
!!!Contribucion de la delta con corriente constante
```

```
nodo(k)%I2(1) = nodo(k)%I2 (1) + &
CONJG(nodo(k)% S2DI(1) / (nodo(k)% VR2(1)-nodo(k)% VR2(2))) * &
EXP (CMLX( 0.0, atan2(IMAG(nodo(k)% VR2(1)-nodo(k)% VR2(2)), REAL(nodo(k)% VR2(1)-
nodo(k)% VR2(2)))- atan2(IMAG(nodo(k)% V2(1)-nodo(k)% V2(2)), REAL(nodo(k)% V2(1)-nodo(k)%
V2(2)))) - &
CONJG(nodo(k)% S2DI(3) / (nodo(k)% VR2(3)-nodo(k)% VR2(1))) * &
EXP (CMLX( 0.0, atan2(IMAG(nodo(k)% VR2(3)-nodo(k)% VR2(1)), REAL(nodo(k)% VR2(3)-
```




nodo(k)% VR2(1))- atan2(IMAG(nodo(k)% V2(3)-nodo(k)% V2(1)), REAL(nodo(k)% V2(3)-nodo(k)% V2(1))))

nodo(k)%I2(2) = nodo(k)%I2 (2) + &

CONJG(nodo(k)% S2DI(2) / (nodo(k)% VR2(2)-nodo(k)% VR2(3))) * &

EXP (CMLPX(0.0, atan2(IMAG(nodo(k)% VR2(2)-nodo(k)% VR2(3)), REAL(nodo(k)% VR2(2)-nodo(k)% VR2(3))) - atan2(IMAG(nodo(k)% V2(2)-nodo(k)% V2(3)), REAL(nodo(k)% V2(2)-nodo(k)% V2(3)))) - &

CONJG(nodo(k)% S2DI(1) / (nodo(k)% VR2(1)-nodo(k)% VR2(2))) * &

EXP (CMLPX(0.0, atan2(IMAG(nodo(k)% VR2(1)-nodo(k)% VR2(2)), REAL(nodo(k)% VR2(1)-nodo(k)% VR2(2))) - atan2(IMAG(nodo(k)% V2(1)-nodo(k)% V2(2)), REAL(nodo(k)% V2(1)-nodo(k)% V2(2))))

nodo(k)%I2(3) = nodo(k)%I2 (3) + &

CONJG(nodo(k)% S2DI(3) / (nodo(k)% VR2(3)-nodo(k)% VR2(1))) * &

EXP (CMLPX(0.0, atan2(IMAG(nodo(k)% VR2(3)-nodo(k)% VR2(1)), REAL(nodo(k)% VR2(3)-nodo(k)% VR2(1))) - atan2(IMAG(nodo(k)% V2(3)-nodo(k)% V2(1)), REAL(nodo(k)% V2(3)-nodo(k)% V2(1)))) - &

CONJG(nodo(k)% S2DI(2) / (nodo(k)% VR2(2)-nodo(k)% VR2(3))) * &

EXP (CMLPX(0.0, atan2(IMAG(nodo(k)% VR2(2)-nodo(k)% VR2(3)), REAL(nodo(k)% VR2(2)-nodo(k)% VR2(3))) - atan2(IMAG(nodo(k)% V2(2)-nodo(k)% V2(3)), REAL(nodo(k)% V2(2)-nodo(k)% V2(3))))

!!!Contribucion de la delta con impedancia constante

nodo(k)%I2(1) = nodo(k)%I2 (1) + &
 CONJG(nodo(k)% S2DZ(1) / (nodo(k)% V2(1)-nodo(k)% V2(2))) * &
 (ABS((nodo(k)% V2(1)-nodo(k)% V2(2)) / (nodo(k)% VR2(1)-nodo(k)% VR2(2))))**2.0 &
 -CONJG(nodo(k)% S2DZ(3) / (nodo(k)% V2(3)-nodo(k)% V2(1))) * &
 (ABS((nodo(k)% V2(3)-nodo(k)% V2(1)) / (nodo(k)% VR2(3)-nodo(k)% VR2(1))))**2.0

nodo(k)%I2(2) = nodo(k)%I2 (2) + &
 CONJG(nodo(k)% S2DZ(2) / (nodo(k)% V2(2)-nodo(k)% V2(3))) * &
 (ABS((nodo(k)% V2(2)-nodo(k)% V2(3)) / (nodo(k)% VR2(2)-nodo(k)% VR2(3))))**2.0 &
 -CONJG(nodo(k)% S2DZ(1) / (nodo(k)% V2(1)-nodo(k)% V2(2))) * &
 (ABS((nodo(k)% V2(1)-nodo(k)% V2(2)) / (nodo(k)% VR2(1)-nodo(k)% VR2(2))))**2.0

nodo(k)%I2(3) = nodo(k)%I2 (3) + &
 CONJG(nodo(k)% S2DZ(3) / (nodo(k)% V2(3)-nodo(k)% V2(1))) * &
 (ABS((nodo(k)% V2(3)-nodo(k)% V2(1)) / (nodo(k)% VR2(3)-nodo(k)% VR2(1))))**2.0 &
 -CONJG(nodo(k)% S2DZ(2) / (nodo(k)% V2(2)-nodo(k)% V2(3))) * &
 (ABS((nodo(k)% V2(2)-nodo(k)% V2(3)) / (nodo(k)% VR2(2)-nodo(k)% VR2(3))))**2.0

!%%
 !%%

!!!Contribucion de las cargas conectadas en Estrella en todos los modelos (ZIP)

nodo(k)%I4= CONJG (nodo(k)% S4YS / nodo(k)% V4) + &
 CONJG (nodo(k)% S4YI / nodo(k)% VR4) * &
 EXP (CMLPX(0.0, ATAN2(IMAG(nodo(k)% VR4),REAL(nodo(k)% VR4))-ATAN2(IMAG(nodo(k)% V4),REAL(nodo(k)% V4)))) + &



CONJG (nodo(k)% S4YZ / nodo(k)% V4) * (ABS(nodo(k)% V4 / nodo(k)% VR4)) ** 2.0

!!!Contribucion de la delta con potencia constante

nodo(k)%I4(1) = nodo(k)%I4 (1) + &
CONJG(nodo(k)% S4DS(1) / (nodo(k)% V4(1)-nodo(k)% V4(2))) - CONJG(nodo(k)%
S4DS(3) / (nodo(k)% V4(3)-nodo(k)% V4(1)))

nodo(k)%I4(2) = nodo(k)%I4 (2) + &
CONJG(nodo(k)% S4DS(2) / (nodo(k)% V4(2)-nodo(k)% V4(3))) - CONJG(nodo(k)%
S4DS(1) / (nodo(k)% V4(1)-nodo(k)% V4(2)))

nodo(k)%I4(3) = nodo(k)%I4 (3) + &
CONJG(nodo(k)% S4DS(3) / (nodo(k)% V4(3)-nodo(k)% V4(1))) - CONJG(nodo(k)%
S4DS(2) / (nodo(k)% V4(2)-nodo(k)% V4(3)))

!!!Contribucion de la delta con corriente constante

nodo(k)%I4(1) = nodo(k)%I4 (1) + &
CONJG(nodo(k)% S4DI(1) / (nodo(k)% VR4(1)-nodo(k)% VR4(2))) * &

EXP (CMPLX(0.0, atan2(IMAG(nodo(k)% VR4(1)-nodo(k)% VR4(2)), REAL(nodo(k)% VR4(1)-
nodo(k)% VR4(2)))- atan2(IMAG(nodo(k)% V4(1)-nodo(k)% V4(2)), REAL(nodo(k)% V4(1)-nodo(k)%
V4(2)))) - &

CONJG(nodo(k)% S4DI(3) / (nodo(k)% VR4(3)-nodo(k)% VR4(1))) * &

EXP (CMPLX(0.0, atan2(IMAG(nodo(k)% VR4(3)-nodo(k)% VR4(1)), REAL(nodo(k)% VR4(3)-
nodo(k)% VR4(1)))- atan2(IMAG(nodo(k)% V4(3)-nodo(k)% V4(1)), REAL(nodo(k)% V4(3)-nodo(k)%
V4(1)))) - &

nodo(k)%I4(2) = nodo(k)%I4 (2) + &

CONJG(nodo(k)% S4DI(2) / (nodo(k)% VR4(2)-nodo(k)% VR4(3))) * &

EXP (CMPLX(0.0, atan2(IMAG(nodo(k)% VR4(2)-nodo(k)% VR4(3)), REAL(nodo(k)% VR4(2)-
nodo(k)% VR4(3)))- atan2(IMAG(nodo(k)% V4(2)-nodo(k)% V4(3)), REAL(nodo(k)% V4(2)-nodo(k)%
V4(3)))) - &

CONJG(nodo(k)% S4DI(1) / (nodo(k)% VR4(1)-nodo(k)% VR4(2))) * &

EXP (CMPLX(0.0, atan2(IMAG(nodo(k)% VR4(1)-nodo(k)% VR4(2)), REAL(nodo(k)% VR4(1)-
nodo(k)% VR4(2)))- atan2(IMAG(nodo(k)% V4(1)-nodo(k)% V4(2)), REAL(nodo(k)% V4(1)-nodo(k)%
V4(2)))) - &

nodo(k)%I4(3) = nodo(k)%I4 (3) + &

CONJG(nodo(k)% S4DI(3) / (nodo(k)% VR4(3)-nodo(k)% VR4(1))) * &

EXP (CMPLX(0.0, atan2(IMAG(nodo(k)% VR4(3)-nodo(k)% VR4(1)), REAL(nodo(k)% VR4(3)-
nodo(k)% VR4(1)))- atan2(IMAG(nodo(k)% V4(3)-nodo(k)% V4(1)), REAL(nodo(k)% V4(3)-nodo(k)%
V4(1)))) - &

CONJG(nodo(k)% S4DI(2) / (nodo(k)% VR4(2)-nodo(k)% VR4(3))) * &

EXP (CMPLX(0.0, atan2(IMAG(nodo(k)% VR4(2)-nodo(k)% VR4(3)), REAL(nodo(k)% VR4(2)-
nodo(k)% VR4(3)))- atan2(IMAG(nodo(k)% V4(2)-nodo(k)% V4(3)), REAL(nodo(k)% V4(2)-nodo(k)%
V4(3)))) - &

!!!Contribucion de la delta con impedancia constante

nodo(k)%I4(1) = nodo(k)%I4 (1) + &
CONJG(nodo(k)% S4DZ(1) / (nodo(k)% V4(1)-nodo(k)% V4(2))) * &
(ABS((nodo(k)% V4(1)-nodo(k)% V4(2)) / (nodo(k)% VR4(1)-nodo(k)%
VR4(2))))**2.0 &



```

-CONJG(nodo(k)% S4DZ(3) / (nodo(k)% V4(3)-nodo(k)% V4(1))) * &
VR4(1)))**2.0 (ABS((nodo(k)% V4(3)-nodo(k)% V4(1)) / (nodo(k)% VR4(3)-nodo(k)%

nodo(k)%I4(2) = nodo(k)%I4 (2) + &
CONJG(nodo(k)% S4DZ(2) / (nodo(k)% V4(2)-nodo(k)% V4(3))) * &
VR4(3)))**2.0 & (ABS((nodo(k)% V4(2)-nodo(k)% V4(3)) / (nodo(k)% VR4(2)-nodo(k)%

-CONJG(nodo(k)% S4DZ(1) / (nodo(k)% V4(1)-nodo(k)% V4(2))) * &
VR4(2)))**2.0 (ABS((nodo(k)% V4(1)-nodo(k)% V4(2)) / (nodo(k)% VR4(1)-nodo(k)%

nodo(k)%I4(3) = nodo(k)%I4 (3) + &
CONJG(nodo(k)% S4DZ(3) / (nodo(k)% V4(3)-nodo(k)% V4(1))) * &
VR4(1)))**2.0 & (ABS((nodo(k)% V4(3)-nodo(k)% V4(1)) / (nodo(k)% VR4(3)-nodo(k)%

-CONJG(nodo(k)% S4DZ(2) / (nodo(k)% V4(2)-nodo(k)% V4(3))) * &
VR4(3)))**2.0 (ABS((nodo(k)% V4(2)-nodo(k)% V4(3)) / (nodo(k)% VR4(2)-nodo(k)%

!!!Contribucion de la Admitancia de Carga
nodo(k)%I4 = nodo(k)%I4 + MATMUL(nodo(k)%Y2 , nodo(k)%V4)

!contribucion de la corriente de los break points (mayator) 23feb2010
nodo(k)%I4 = nodo(k)%I4 + nodo(k)%Ibreak

END DO Inyecciones_Nodales

!=====
!=====
!Inicializacion de corrientes de incoming branch (amperes)

DO k=1, n2

nodo(k)%JP4 = nodo(k)%I4
nodo(k)%J4 = MATMUL(nodo(k)%d4 , nodo(k)%JP4)

nodo(k)%JP3 = nodo(k)%I3 + nodo(k)%J4
nodo(k)%J3 = MATMUL(nodo(k)%d3 , nodo(k)%JP3)

nodo(k)%JP2 = nodo(k)%I2 + nodo(k)%J3
nodo(k)%J2 = MATMUL(nodo(k)%d2 , nodo(k)%JP2)

nodo(k)%JP1 = nodo(k)%I1 + nodo(k)%J2
nodo(k)%J1 = MATMUL(nodo(k)%d1 , nodo(k)%JP1)

enddo

!=====
!=====
!Barrido Regresivo

Backward_Sweep: DO k=n2, 2, -1

nodo(nodo(k)%pre)%JP4 = nodo(nodo(k)%pre)%JP4 + nodo(k)%J1
nodo(nodo(k)%pre)%J4 = nodo(nodo(k)%pre)%J4 + MATMUL(nodo(nodo(k)%pre)%d4 ,
nodo(k)%J1)

nodo(nodo(k)%pre)%JP3 = nodo(nodo(k)%pre)%JP3 + MATMUL(nodo(nodo(k)%pre)%d4 ,
nodo(k)%J1)
nodo(nodo(k)%pre)%J3 = nodo(nodo(k)%pre)%J3 + MATMUL(nodo(nodo(k)%pre)%d3 ,
MATMUL(nodo(nodo(k)%pre)%d4 , nodo(k)%J1))

nodo(nodo(k)%pre)%JP2 = nodo(nodo(k)%pre)%JP2 + MATMUL(nodo(nodo(k)%pre)%d3 ,
MATMUL(nodo(nodo(k)%pre)%d4 , nodo(k)%J1))
nodo(nodo(k)%pre)%J2 = nodo(nodo(k)%pre)%J2 + MATMUL(nodo(nodo(k)%pre)%d2 ,
MATMUL(nodo(nodo(k)%pre)%d3 , MATMUL(nodo(nodo(k)%pre)%d4 , nodo(k)%J1)))

nodo(nodo(k)%pre)%JP1 = nodo(nodo(k)%pre)%JP1 + MATMUL(nodo(nodo(k)%pre)%d2 ,
MATMUL(nodo(nodo(k)%pre)%d3 , MATMUL(nodo(nodo(k)%pre)%d4 , nodo(k)%J1)))
nodo(nodo(k)%pre)%J1 = nodo(nodo(k)%pre)%J1 + MATMUL(nodo(nodo(k)%pre)%d1 ,
MATMUL(nodo(nodo(k)%pre)%d2 , MATMUL(nodo(nodo(k)%pre)%d3 , MATMUL(nodo(nodo(k)%pre)%d4 ,

```



```

nodo(k)%J1)))
ENDDO Backward_Sweep

!=====
!=====
!Barrido Progresivo

Forward_Sweep: DO k=2,n2

!-----
V4antes=nodo(k)%V4
!-----
      nodo(k)%V1= MATMUL(nodo(k)%A1, nodo(nodo(k)%pre)%V4) - MATMUL(nodo(k)%B1, nodo(k)%JP1)
      nodo(k)%V2= MATMUL(nodo(k)%A2, nodo(k)%V1)           - MATMUL(nodo(k)%B2, nodo(k)%JP2)
      nodo(k)%V3= MATMUL(nodo(k)%A3, nodo(k)%V2)           - MATMUL(nodo(k)%B3, nodo(k)%JP3)
      nodo(k)%V4= MATMUL(nodo(k)%A4, nodo(k)%V3)           - MATMUL(nodo(k)%B4, nodo(k)%JP4)
!-----

ERROR = V4antes - nodo(k)%V4

if( abs(ERROR(1))<= tolerancia .AND. abs(ERROR(2)) <= tolerancia .AND. abs(ERROR(3)) <=
tolerancia) converge=converge+1

ENDDO Forward_Sweep

!=====
!=====

iteraciones=iteraciones + 1

ENDDO CICLO_DE_ITERACIONES

!=====
!=====

ibreak_condition: if(m2>0) then

!Update of Brakpoint Currents---
23feb2010*****

UPDATE_BKP_CURRENTS: do k=1, m2

      thv%V(k)%v31 = ( nodo(malla(k)%from)%V4 - MATMUL (malla(k)%Z , thv%I(k)%v31)) -
nodo(malla(k)%to)%V4

end do UPDATE_BKP_CURRENTS

do k=1, m2
      vthevenin(3*k-2:3*k) = thv%V(k)%v31
enddo
!*****
Ithevenin = Ithevenin + MATMUL ( Ythevenin, vthevenin )
!*****

do k=1, m2
nodo(malla(k)%from)%Ibreak = Ithevenin(3*k-2:3*k)
nodo(malla(k)%to)%Ibreak   = - Ithevenin(3*k-2:3*k)
enddo

parche: do k=1, m2
      thv%I(k)%v31 = Ithevenin(3*k-2:3*k)
end do parche

DO k=1, m2

if( abs(thv%V(k)%v31(1))<= tolerancia .AND. abs(thv%V(k)%v31(2)) <= tolerancia .AND.
abs(thv%V(k)%v31(3)) <= tolerancia) convergeMacro=convergeMacro+1

```



ENDDO

iteracionesMacro=iteracionesMacro+1

ENDIF iBREAK_CONDITION

ENDDO MACROITERACIONES

!=====
!Calculo de Flujos en Lineas y Cargas Totales- Una vez conocidos los voltajes nodales y corrientes de rama y carga

!calculo de potencias de carga

```
do k=1, n2
do t=1, 3
nodo(k)%SI1(t)= nodo(k)%V1(t)* CONJG (nodo(k)%I1(t))
nodo(k)%SI2(t)= nodo(k)%V2(t)* CONJG (nodo(k)%I2(t))
nodo(k)%SI3(t)= nodo(k)%V3(t)* CONJG (nodo(k)%I3(t))
nodo(k)%SI4(t)= nodo(k)%V4(t)* CONJG (nodo(k)%I4(t))
enddo
enddo
```

!calculo de caidas de tension en branches!

```
do k=2, n2
nodo(k)%e1= MATMUL(nodo(k)%B1, nodo(k)%JP1)
nodo(k)%e2= MATMUL(nodo(k)%B2, nodo(k)%JP2)
nodo(k)%e3= MATMUL(nodo(k)%B3, nodo(k)%JP3)
nodo(k)%e4= MATMUL(nodo(k)%B4, nodo(k)%JP4)
enddo
```

!calculo de potencias de perdida en branches

```
do k=2, n2
do t=1, 3
nodo(k)%SJP1(t)= nodo(k)%e1(t)* CONJG (nodo(k)%JP1(t))
nodo(k)%SJP2(t)= nodo(k)%e2(t)* CONJG (nodo(k)%JP2(t))
nodo(k)%SJP3(t)= nodo(k)%e3(t)* CONJG (nodo(k)%JP3(t))
nodo(k)%SJP4(t)= nodo(k)%e4(t)* CONJG (nodo(k)%JP4(t))
enddo
enddo
```

!Se suma la carga total por rama-nodo

```
do k=1, n2
nodo(k)%SI= nodo(k)%SI1 + nodo(k)%SI2 + nodo(k)%SI3 + nodo(k)%SI4
nodo(k)%SJP= nodo(k)%SJP1 + nodo(k)%SJP2 + nodo(k)%SJP3 + nodo(k)%SJP4
enddo
```

!=====
!=====
!ciclo de refinamiento de ceros

```
do k=2, n2
do t=1,3
if( nodo(k)%Z3(t,t) == CMPLX(0.0, 0.0) .AND. nodo(k)%tybr /= -1 ) nodo(k)%V4(t)=
CMPLX(0.00, 0.00)
enddo
enddo
```

```
do k=2, n2
Slos= Slos + nodo(k)%SJP
enddo
Slos = Slos / 1000.00
```



```

!loss=Real(S1os(1)+S1os(2)+S1os(3))
!=====
!=====
END Subroutine BackwardForward
!=====
!=====
END MODULE BackForw

```

B.3. ConstantesG2.f90

```

!=====
!=====
!RADIFLUX 1.0
!
!FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
!20 de DICIEMBRE de 2008
!=====
!
!IVAN ARCHUNDIA ARANDA
!iarchundia@ipn.mx
!
!=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!=====

```

Module ConstantesG2

```

CONTAINS
!*****
subroutine dAB2 (Z2, d2, A2, B2)

USE MCons

implicit none

!Declaracion de variables I/O

complex, dimension(3,3), intent(in):: Z2
complex, dimension(3,3), intent(out):: d2
complex, dimension(3,3), intent(out):: A2
complex, dimension(3,3), intent(out):: B2

!El elemento 2 siempre será una línea sin importar si hay o no carga distribuida
d2= U3C
A2=d2
B2=Z2

End subroutine dAB2
!*****

!*****
End Module ConstantesG2

```

B.4. ConstantesG3.f90

```

!=====
!=====
!RADIFLUX 1.0

```



```

!
! FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
! 20 de DICIEMBRE de 2008
!=====
!
! IVAN ARCHUNDIA ARANDA
! iarchundia@ipn.mx
!
!=====
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!=====

Module ConstantesG3
!*****
CONTAINS
!*****

subroutine dAB3 (tybr,nt, Z3, d3, A3, B3)

use MCons

implicit none

!Declaracion de variables I/O

integer, intent(in):: tybr
real, intent(in):: nt
complex, dimension(3,3), intent(in):: Z3

complex, dimension(3,3), intent(out):: d3
complex, dimension(3,3), intent(out):: A3
complex, dimension(3,3), intent(out):: B3

selector_Constantes: SELECT CASE (tybr)

CASE(1) !Línea Aérea, Línea Subterranea -----
d3=U3C
A3=d3
B3=Z3

CASE(2) !Transformador D-Yg-----
d3=CMPLX(0.0, 0.0)

d3(1,1)=CMPLX( 1.0, 0.0)
d3(1,2)=CMPLX(-1.0, 0.0)
d3(2,2)=CMPLX( 1.0, 0.0)
d3(2,3)=CMPLX(-1.0, 0.0)
d3(3,1)=CMPLX(-1.0, 0.0)
d3(3,3)=CMPLX( 1.0, 0.0)

d3=(1.0 / nt) * d3
A3=CMPLX(0.0, 0.0)

A3(1,1)=CMPLX( 1.0, 0.0)
A3(1,3)=CMPLX(-1.0, 0.0)
A3(2,1)=CMPLX(-1.0, 0.0)
A3(2,2)=CMPLX( 1.0, 0.0)
A3(3,2)=CMPLX(-1.0, 0.0)
A3(3,3)=CMPLX( 1.0, 0.0)

A3=(1.0 / nt) * A3
B3=Z3

CASE(3) !Transformador Yg-Yg-----

```



```
d3=U3C
d3=(1.0 / nt) * d3
A3=d3
B3=Z3

CASE DEFAULT
d3=U3C
A3=U3C
B3=CMPLX(0.0, 0.0)

END SELECT Selector_Constantes

End subroutine dAB3

!*****

End Module ConstantesG3
```

B.5. ConstantesG14.f90

```
!=====
!=====
!RADIFLUX 1.0
!
!FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
!20 de DICIEMBRE de 2008
!=====
!
!IVAN ARCHUNDIA ARANDA
!iarchundia@ipn.mx
!
!=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!=====

Module ConstantesG14

contains

!*****

subroutine dAB14( lc_Reg, ty_Reg, cx_Reg, ar, Z1, Z4, &
                d1, A1, B1, &
                d4, A4, B4 )

USE MCons

IMPLICIT NONE

!Declaracion de variables I/O

integer, intent (in):: lc_Reg
integer, intent (in):: ty_Reg
integer, intent (in):: cx_Reg
real, dimension (3), intent(in) :: ar
complex, dimension(3,3), intent(in):: Z1
complex, dimension(3,3), intent(in):: Z4

complex, dimension(3,3), intent(out):: d1
complex, dimension(3,3), intent(out):: A1
complex, dimension(3,3), intent(out):: B1

complex, dimension(3,3), intent(out):: d4
```




```

complex, dimension(3,3), intent(out):: A4
complex, dimension(3,3), intent(out):: B4

!Decalracion de variables internas

!AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
!Se inicializan las constantes de los 2 reguladores como si fueran lineas sin pérdidas
d1= U3C
A1= d1
B1= Z1 !siempre sera cero cuando el regulador sea sin sin pérdidas

d4= U3C
A4= d4
B4= Z4 !siempre sera cero cuando el regulador sea sin sin pérdidas

!Y se modifican dependiendo de si el reg está al principio o al final

IF (lc_Reg==1) call modify_dAB14 ( ar, d1, A1)
IF (lc_Reg==2) call modify_dAB14 ( ar, d4, A4)

!-----
!+++++
CONTAINS
!+++++
!-----

subroutine modify_dAB14 (ar, d, A)

implicit none

real, dimension(3), intent(in):: ar
complex, dimension(3,3), intent(inout):: d,A

Tipo: SELECT CASE (ty_Reg) !modifica las Cns. Gen de acuerdo al tipo de regulador, A o B

CASE(1) !El regulador es tipo A

    Conexion: SELECT CASE (cx_Reg) !Modifica de acuerdo a la conexion
        CASE(1) !El regulador esta conectado en estrella
            d(1,1)= ar(1) * d(1,1)
            d(2,2)= ar(2) * d(2,2)
            d(3,3)= ar(3) * d(3,3)

            A(1,1)= ar(1) * A(1,1)
            A(2,2)= ar(2) * A(2,2)
            A(3,3)= ar(3) * A(3,3)

        CASE(2) !El regulador esta conectado en delta
            !Todavia no esta definifo!!!!!(12 nov2008)

        END SELECT Conexion

CASE(2) !El regulador es tipo B

    Conetsion: SELECT CASE (cx_Reg) !Modifica de acuerdo a la conexion
        CASE(1) !El regulador esta conectado en estrella
            d(1,1)= (1.0 / ar(1) ) * d(1,1)
            d(2,2)= (1.0 / ar(2) ) * d(2,2)
            d(3,3)= (1.0 / ar(3) ) * d(3,3)

            A(1,1)= (1.0 / ar(1) ) * A(1,1)
            A(2,2)= (1.0 / ar(2) ) * A(2,2)
            A(3,3)= (1.0 / ar(3) ) * A(3,3)

```



```

CASE(2) !El regulador esta conectado en delta
!Todavia no esta definifo!!!!!! (12 nov2008)
END SELECT Conetsion
END SELECT Tipo
end subroutine modify_dAB14
ENDsubroutine dAB14
End Module ConstantesG14

```

B.6. Estructuras.f90

```

=====
!RADIFLUX 1.0
!
!FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
!20 de DICIEMBRE de 2008
!
=====
!IVAN ARCHUNDIA ARANDA
!iarchundia@ipn.mx
!
=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!
=====
Module estructuras
implicit none
!*****
type::lectura
!-----
character(10):: busName !nombre del nodo
integer:: n !numero de nodo
integer:: pre !predecesor
integer:: niv !nivel
integer:: tybr !tipo de rama (line, transformer...)
real:: nt !relacion de espiras (de primario a secundario)
real, dimension(12):: Z !Impedancia of incoming branch
real, dimension(6):: Y !Admitancia of incoming branch microsiemens
integer:: cx_Sc !conexion de la carga concentrada(1:estrella, 2:delta)
integer:: md_Sc !modelo de la carga concentrada(1:S , 2: I, 3: Z)
Potencia, Corriente o Impedancia constante
real, dimension(6):: Sc !Carga concentrada del nodo en kw
integer:: cx_Sd !conexion de la carga distribuida(1:estrella, 2:delta)
integer:: md_Sd !modelo de la carga distribuida(1:S , 2: I, 3: Z)
Potencia, Corriente o Impedancia constante
real, dimension(6):: Sd !Carga distribuida del nodo en kw
integer:: cx_BC !Conexion del Banco de capacitores; 1:estrella, 2: delta
(no acoplamiento)
real, dimension(3):: SBC !Potencia reactiva del banco de capacitores en kVAR
integer:: lc_Reg !Ubicacion del Regulador (1: principio, 2: Final)
integer:: ty_Reg !Tipo de Regulador (1: Tipo A, 2: Tipo B. See Kersting)
integer:: cx_Reg !Conexion del Regulador (1: Estrella, 2: Delta)

```



```

real, dimension(3):: Tap      !Ajuste de Tap de regulador para fases a-b-c.
end type lectura
!*****
type::lista
!-----
character(10)::      busName      !nombre del nodo
integer::n
integer::pre
integer::niv
integer::tybr
real::      nt

complex, dimension(3,3):: Z
complex, dimension(3,3):: Y

integer::          cx_Sc
integer::          md_Sc
complex, dimension(3):: Sc

integer::          cx_Sd
integer::          md_Sd
complex, dimension(3):: Sd

integer::          cx_BC
complex, dimension(3):: SBC

integer::          tc_Reg
integer::          ty_Reg
integer::          cx_Reg
real, dimension(3):: Tap
!-----
complex, dimension(3)      :: V1, V2, V3, V4
complex, dimension(3)      :: VR1, VR2, VR3, VR4

complex, dimension(3,3)    :: Y2          !Admitancia total de carga, en el nodo
complex, dimension(3,3)    :: Z1, Z2, Z3, Z4

complex, dimension(3)      :: S1YS, S2YS, S3YS, S4YS
complex, dimension(3)      :: S1YI, S2YI, S3YI, S4YI
complex, dimension(3)      :: S1YZ, S2YZ, S3YZ, S4YZ

complex, dimension(3)      :: S1DS, S2DS, S3DS, S4DS
complex, dimension(3)      :: S1DI, S2DI, S3DI, S4DI
complex, dimension(3)      :: S1DZ, S2DZ, S3DZ, S4DZ

!-----
complex, dimension(3,3)::d1, d2, d3, d4      !for backward sweep
complex, dimension(3,3)::A1, A2, A3, A4      !for forward sweep
complex, dimension(3,3)::B1, B2, B3, B4      !for forward sweep

COMPLEX, dimension(3):: I1, I2, I3, I4      !Inyeccion de corrientes nodal (total)
COMPLEX, dimension(3):: J1, J2, J3, J4      !Corriente de incoming BRANCH
COMPLEX, dimension(3):: JP1, JP2, JP3, JP4  !Corriente de incoming BRANCH primadas

real, dimension(3):: ar                      !ar del regulador (see Kersting)

complex, dimension(3)::Ibreak                !Inyeccion de corrientes de Brakpoints
23feb2010

!-----
complex, dimension(3):: SI1, SI2, SI3, SI4
complex, dimension(3):: SJP1, SJP2, SJP3, SJP4
complex, dimension(3):: e1, e2, e3, e4

complex, dimension(3):: SI
complex, dimension(3):: SJP
!-----
end type lista
!*****

```



end module estructuras

B.7. Estructuras2.f90

```

=====
!
!RADIFLUX 1.0
!
!FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
!20 de DICIEMBRE de 2008
!
=====
!
!IVAN ARCHUNDIA ARANDA
!iarchundia@ipn.mx
!
=====
!
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!
=====
!
!05feb07

Module Estructuras2
implicit none

!*****

type::lectura_mallas
!-----

integer::      m          !numero de malla
integer::      from      !FROM
integer::      to        !TO

real, dimension(12)::Z      !Impedancia of incoming branch
real, dimension(6)::Y      !Admitancia of incoming branch microSiemens

end type lectura_mallas

!*****

type::lista_mallas
!-----

integer::m
integer::from
integer::to

complex, dimension(3,3):: Z
complex, dimension(3,3):: Y

!-----
end type lista_mallas

!*****

END Module Estructuras2

```

B.8. Estructuras3.f90

```

=====
!
!RADIFLUX 1.0
!

```



```
! FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
! 20 de DICIEMBRE de 2008
!=====
!
! IVAN ARCHUNDIA ARANDA
! iarchundia@ipn.mx
!
!=====
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!=====
```

Module Estructuras3 !15 feb 2010

implicit none

```
!-----
type:: vec31
   complex, dimension(3)::v31
end type vec31
```

```
type:: mat33
   complex, dimension(3,3)::m33
end type mat33
```

```
type::THV_est                                !thevenin_estructura
   type(vec31), pointer, dimension(1):: I
   type(vec31), pointer, dimension(1):: V
   type(mat33), pointer, dimension(1,1):: Z
   !type(mat33), pointer, dimension(1,1):: Y
end type THV_est
```

!*****

```
!-----
End Module Estructuras3
```

B.9. Estructuras4.f90

```
!=====
!=====
! RADIFLUX 1.0
!
! FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
! 20 de DICIEMBRE de 2008
!=====
!
! IVAN ARCHUNDIA ARANDA
! iarchundia@ipn.mx
!
!=====
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!=====
```

Module Estructuras4 !29MARZO 2011

implicit none

```
!-----
!29marzo2011---Estructura de Línea
```

type:: line

!*****

```
integer::          n          !número de línea
```



```

character(10)::      from      !self explained
character(10)::      to        !self explained

!*****
complex, dimension(3):: Jfrom    !self explained   (rectangular)
complex, dimension(3):: Jto      !self explained   (rectangular)

real, dimension(3)::   Jfrom_mag  !idem, pero magnitud (polar)
real, dimension(3)::   Jfrom_ang   !idem, pero angulo   (polar)

real, dimension(3)::   Jto_mag     !idem, pero magnitud (polar)
real, dimension(3)::   Jto_ang     !idem, pero angulo   (polar)

!*****
complex, dimension(3):: Sfrom      !Self explained
complex, dimension(3):: Sto        !Self explained
!*****

```

end type line

!-----
End Module Estructuras4

B.10. Mcons.f90

```

!=====
!=====
!RADIFLUX 1.0
!
!FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
!20 de DICIEMBRE de 2008
!=====
!
!IVAN ARCHUNDIA ARANDA
!iarchundia@ipn.mx
!
!=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!=====

```

```

module MCons
!Matrices constantes

```

IMPLICIT NONE

```

!-----
!Matriz unidad de 3x3 compleja
complex, dimension(3,3), parameter:: &
U3C=reshape((/(1.0, 0.0),(0.0, 0.0),(0.0, 0.0), (0.0, 0.0), (1.0, 0.0),(0.0, 0.0),(0.0,
0.0),(0.0, 0.0),(1.0, 0.0)/),(/3,3/))
!-----

```

end module MCons

B.11. PrintSalidaDetallada.f90

```

!=====
!=====
!RADIFLUX 1.0
!
!FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
!20 de DICIEMBRE de 2008
!Mod. 29 julio 2010 (argumentos)

```



```

=====
!
!IVAN ARCHUNDIA ARANDA
!iarchundia@ipn.mx
!
=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!
=====
!

module PrintsSalidaDetallada
!
contains

Subroutine SalidaDetallada (n_archivo ,n2,m2, nodo, iteraciones, iteracionesMacro,
converge, header)

USE MSIMSL

use estructuras
use UtileriasComplejas2

Implicit none

!Declaracion de variables de I/O
character(20), intent(in)           :: n_archivo
integer, intent(in)                :: n2
integer, intent(in)                :: m2
type(lista), dimension(:), intent(in)  :: nodo

integer, intent(in):: iteraciones
integer, intent(in):: iteracionesMacro
integer, intent(in):: converge
character(len=60), intent (in)       :: header

!Declaracion de variables internas
integer:: k,p !contador (es)

complex, dimension(3):: Sint = (0.0 , 0.0)
complex, dimension(3):: Sout = (0.0 , 0.0)
complex, dimension(3):: Slos = (0.0 , 0.0)

!Apertura de archivo de Resultados de Flujos de Potencia
open(21, file=n_archivo)

!=====
!Impresion de Encabezado de Reporte

!-----
write(21, '(/)')
write(21, ('=====
=)')
write(21, ('
                                RESULTADOS DE FLUJOS DE POTENCIA
=)')
write(21, ('
                                radiflux v2.1
=)')
write(21, ('=====
=)')
!-----
write(21, '(/)')

write(21, ('RED           :", A60)') header
!write(21, ('ARCHIVO DE ENTRADA:", A)')n_archivo
write(21, '(/)')
!=====
!
IMPRIMIR: IF (converge<n2) then
write(21, ('RED           :", A60)') header
!write(21, ('ARCHIVO DE ENTRADA:", A)')n_archivo

write(21, '(/)')
```



```

    if (m2==0) then
    write(21,('NO SE ALCANZO LA CONVERGENCIA EN", 2X, I3,2X,"ITERACIONES" ))ITERACIONES
    else
    write(21,('NO SE ALCANZO LA CONVERGENCIA EN", 2X, I3,2X,"ITERACIONES"
))iteracionesMacro
    endif
!write(21,('Nodos en los que se alcanzo la convergencia:", 2X, I3 ))CONVERGE
ELSE
    if (m2==0) then
    write(21,('SE ALCANZO LA CONVERGENCIA EN",2X, I3,2X,"ITERACIONES" ))ITERACIONES
    else
    write(21,('SE ALCANZO LA CONVERGENCIA EN",2X, I3,2X,"ITERACIONES" ))iteracionesMacro
    endif
!write(21,('Nodos en los que se alcanzo la convergencia:", 2X, I3 ))CONVERGE
write(21,('(/)')
!=====
!=====
Sint= nodo(1)%V1 * CONJG (nodo(1)%J1)
do k=2, n2
Sout= Sout + nodo(k)%SI
Slos= Slos + nodo(k)%SJP
enddo
!Normalizacion
Sint = Sint / 1000.00      !Se normaliza a KVA
Sout = Sout / 1000.00
Slos = Slos / 1000.00
!=====
!IMPRESION DE RESUMEN TOTAL DEL SISTEMA

!-----
write(21,('=====  
=")')
write(21,('          R E S U M E N   G E N E R A L  
=")')
write(21,('=====  
=")')
!-----
!write(21,('(/)')
!write(21,('-----")')
WRITE(21,('1X,"          ", T22, "kw", T34, "kVar", T47, "kVA")')
write(21,('-----")')
write(21,('2X, "ENTRADA :", T18, F9.3, T30, F9.3, T43, F9.3)')
Real(Sint(1)+Sint(2)+Sint(3)), Imag(Sint(1)+Sint(2)+Sint(3)), Abs(Sint(1)+Sint(2)+Sint(3))
write(21,('2X, "CARGA :", T18, F9.3, T30, F9.3, T43, F9.3)')
Real(Sout(1)+Sout(2)+Sout(3)), Imag(Sout(1)+Sout(2)+Sout(3)), Abs(Sout(1)+Sout(2)+Sout(3))
write(21,('2X, "PERDIDAS:", T18, F9.3, T30, F9.3, T43, F9.3)')
Real(Slos(1)+Slos(2)+Slos(3)), Imag(Slos(1)+Slos(2)+Slos(3)), Abs(Slos(1)+Slos(2)+Slos(3))
write(21,('-----")')
write(21,('(/)')
!-----
write(21,('=====  
=")')
write(21,('          P E R F I L   D E   V O L T A J E S  
=")')
write(21,('=====  
=")')
!-----
write(21,('(/)')
!IMPRESION DE PERFIL DE VOLTAJE EN FORMA POLAR, CON UNIDADES REALES
write(21,('PERFIL DE VOLTAJE -- (Forma Polar, kv @ °)')
write(21,('-----")')
write(21,('2X, "-----N O D O-----", 4X, "----F A S E  A----", 4X, "----F A S E  B----", 4X, "----F A  
S E  C----")')
write(21,('3X, "NO.", 5X, "NOMBRE", 7X, "kv", 6X, "GRADOS", 6X, "kv", 6X, "GRADOS", 6X, "kv",  
6X, "GRADOS", 6X)')
write(21,('-----")')

do k=1, n2
CALL RecPol (21, nodo(k)%n, nodo(k)%busName, nodo(k)%v4 / 1000.00)      !lo divido
entre 1000.00 para que sea kv
enddo
write(21,('-----")')
write(21,('(/)')

!IMPRESION DE PERFIL DE VOLTAJE EN FORMA POLAR, CON VALORES DE MAGNITUD EN P.U. Y ANGULO EN

```




```

GRADOS
write(21,('PERFIL DE VOLTAJE -- (Forma Polar, PU @ °)'))

write(21,('-----'))
write(21,('2X,"-----N O D O-----", 4X,"---F A S E A---", 4X,"---F A S E B---",4X,"---F A
S E C---"))
write(21,('3X,"NO.",5X,"NOMBRE",7X,"PU", 6X,"GRADOS", 6X,"PU", 6X,"GRADOS", 6X,"PU",
6X,"GRADOS", 6X)')
write(21,('-----'))

do k=1, n2
CALL RecPol_pu (21, nodo(k)%n, nodo(k)%busName, nodo(k)%V4 / abs(nodo(k)%VR4))
enddo
write(21,('-----'))
write(21,('(/)'))

!-----
write(21,('=====
=))')
write(21,('                                PERFIL DE CORRIENTES
'))
write(21,('=====
=))')
!-----
write(21,('(/)'))

!IMPRESION DE CORRIENTES NODALES EN FORMA POLAR, CON UNIDADES REALES
write(21,('CORRIENTES NODALES -- (Forma Polar, A @ °)'))

write(21,('-----'))
write(21,('2X,"-----N O D O-----", 4X,"---F A S E A---", 4X,"---F A S E B---",4X,"---F A
S E C---"))
write(21,('3X,"NO.",5X,"NOMBRE",7X,"amps. ", 3X,"grados", 7X,"amps. ", 2X,"grados",
6X,"amps. ", 2X,"grados", 6X)')
write(21,('-----'))

do k=1, n2
CALL RecPol (21, nodo(k)%n, nodo(k)%busName, nodo(k)%I4 )
enddo
write(21,('-----'))
write(21,('(/)'))

!IMPRESION DE CORRIENTES NODALES DE SUBRAMA (CARGA DISTRIBUIDA) EN FORMA POLAR, CON
UNIDADES REALES
write(21,('CORRIENTES NODALES DE SUBRAMA-- (Forma Polar, A @ °) *'))

write(21,('-----'))
write(21,('2X,"-----N O D O-----", 4X,"---F A S E A---", 4X,"---F A S E B---",4X,"---F A
S E C---"))
write(21,('3X,"NO.",5X,"NOMBRE",7X,"amps. ", 3X,"grados", 7X,"amps. ", 2X,"grados",
6X,"amps. ", 2X,"grados", 6X)')
write(21,('-----'))

do k=1, n2

if (nodo(k)%I2(1) /= cmplx(0.0, 0.0) .or. nodo(k)%I2(2) /= cmplx(0.0, 0.0) .or.
nodo(k)%I2(3) /= cmplx(0.0, 0.0) ) THEN
CALL RecPol (21, nodo(k)%n, nodo(k)%busName, nodo(k)%I2 )
ENDIF

enddo
write(21,('-----'))
write(21,('(*En nodo ficticio en la linea cuando hay carga distribuida)'))
write(21,('(/)'))

!IMPRESION DE CORRIENTES DE RAMA EN FORMA POLAR, CON UNIDADES REALES
write(21,('CORRIENTES DE RAMA -- (Forma Polar, A @ °)'))

write(21,('-----'))
write(21,('2X,"-----N O D O-----", 4X,"---F A S E A---", 4X,"---F A S E B---",4X,"---F A
S E C---"))
write(21,('3X,"NO.",5X,"NOMBRE",7X,"amps. ", 3X,"grados", 7X,"amps. ", 2X,"grados",
6X,"amps. ", 2X,"grados", 6X)')
write(21,('-----'))

```



```

do k=1, n2
CALL RecPol (21, nodo(k)%n, nodo(k)%busName, nodo(k)%J2 )

if (nodo(k)%tybr >= 2) call RecPol_1 (21, nodo(k)%JP3)

enddo
write(21,('-----'))')
write(21,('Notas: Para líneas con carga distribuida, se despliega el valor de corriente
'))')
write(21,('      antes del nodo ficticio (valor mas grande)'))')
write(21,('      Para transformadores, se despliegan consecutivamente los valores de
'))')
write(21,('      corriente de entrada y de salida
'))')
write(21,('(/)'))')
!-----
write(21,('=====
=))')
write(21,('                                PERFIL DE POTENCIAS
'))')
write(21,('=====
=))')
!-----
write(21,('(/)'))')
!IMPRESION DE POTENCIAS NODALES FINALES
write(21,('POTENCIAS NODALES DE CARGA (kw+jkvar)'))')
write(21,('-----'))')
write(21,('2X, "-----N O D O-----", T22, " -F A S E  A- ", T42, " -F A S E  B- ", T62,
"-F A S E  C- "))')
write(21,('3X, "NO.", T10, "NOMBRE",  T22, " KW      KVAR",  T42, " KW      KVAR", T62,
" KW      KVAR"  )')
write(21,('-----'))')

DO k=1, n2
write(21,('1X,I5,1X, A10,T18,2(1X,F9.2), 2(1X,F9.2), 2(1X,F9.2))') nodo(k)%n,
nodo(k)%busName, (nodo(k)%SI1(p)/ 1000.00+nodo(k)%SI4(p)/ 1000.00 , p=1,3)
ENDDO

write(21,('-----'))')
write(21,('(/)'))')

!IMPRESION DE POTENCIAS DE RAMA (PERDIDAS)
write(21,('PERDIDAS (kw+jkvar)'))')
write(21,('-----'))')
write(21,('2X, "-----N O D O-----", T22, " -F A S E  A- ", T42, " -F A S E  B- ", T62,
"-F A S E  C- "))')
write(21,('3X, "NO.", T10, "NOMBRE",  T22, " KW      KVAR",  T42, " KW      KVAR", T62,
" KW      KVAR"  )')
write(21,('-----'))')

DO k=2, n2      !AQUI SE INICIA EN 2 BECAUSE THE FIRST NODE DON'T HAVE REAL LINE

write(21,('1X,I5,1X, A10,T18,2(1X,F9.2), 2(1X,F9.2), 2(1X,F9.2))') nodo(k)%n,
nodo(k)%busName, (nodo(k)%SJP2(p)/ 1000.00+nodo(k)%SJP3(p)/ 1000.00 , p=1,3)

ENDDO

write(21,('-----'))')
write(21,('(/)'))')

endif IMPRIMIR

End Subroutine SalidaDetallada

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!11
end module PrintSalidaDetallada

```

B.12. Rutinas.f90

```

!=====
!=====
!RADIFLUX 1.0

```



```

!
! FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
! 20 de DICIEMBRE de 2008
!=====
!
! IVAN ARCHUNDIA ARANDA
! iarchundia@ipn.mx
!
!=====
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!=====

module rutinas_de_asignacion
!rutinas de conversión de datos para lista de barrido
CONTAINS
!*****
!*****
!*****
subroutine conversor_Z(Ztemp, Z, Z1, Z2, Z3, Z4)

implicit none
!Declaración de variables I/O
real, dimension(12), intent(in)      ::Ztemp

complex, dimension(3,3), intent(out)::Z
complex, dimension(3,3), intent(out)::Z1, Z2, Z3, Z4

!Declaración de variables internas

real:: FZ2 = 1.0 / 4.0
real:: FZ3 = 3.0 / 4.0

!Inicialización de impedancias serie

Z = (0.0, 0.0)

Z1= (0.0, 0.0)  !siempre cero (Regulador sin pérdidas)
Z2= (0.0, 0.0)  !inicializa hasta saber si hay carga distribuida en la línea
Z3= (0.0, 0.0)  !inicializa hasta saber si hay carga distribuida en la línea
Z4= (0.0, 0.0)  !siempre cero (Regulador sin pérdidas)
!Se asigna la entrada de datos al contenedor Z que es una matriz compleja de 3X3

Z(1,1)= cplx(Ztemp(1) , Ztemp(2) )
Z(1,2)= cplx(Ztemp(3) , Ztemp(4) )
Z(1,3)= cplx(Ztemp(5) , Ztemp(6) )
Z(2,2)= cplx(Ztemp(7) , Ztemp(8) )
Z(2,3)= cplx(Ztemp(9) , Ztemp(10) )
Z(3,3)= cplx(Ztemp(11), Ztemp(12) )

Z(2,1)= cplx(Ztemp(3) , Ztemp(4) )
Z(3,1)= cplx(Ztemp(5) , Ztemp(6) )
Z(3,2)= cplx(Ztemp(9) , Ztemp(10) )

!Asignación de la impedancia serie de entrada a Z3, como si "TODA" la carga fuese
concentrada
Z3=Z
!Finalmente se refinan las Impedancias Z2 y Z3 con los factores auxiliares, repartiendo la
impedancia entre los elementos 2 y 3,
!Incluso si no hubiese carga distribuida (realmente los elementos están en serie)

Z2= FZ2 * Z3
Z3= FZ3 * Z3

end subroutine conversor_Z
!*****
subroutine conversor_Y(Ytemp, Y)

implicit none

real, dimension(6), intent(in)::Ytemp
complex, dimension(3,3), intent(out)::Y

```



```

Y=cplx(0.0, 0.0)

Y(1,1)=cplx(0.0, 1.0*Ytemp(1))
Y(1,2)=cplx(0.0, 1.0*Ytemp(2))
Y(1,3)=cplx(0.0, 1.0*Ytemp(3))
Y(2,2)=cplx(0.0, 1.0*Ytemp(4))
Y(2,3)=cplx(0.0, 1.0*Ytemp(5))
Y(3,3)=cplx(0.0, 1.0*Ytemp(6))

Y(2,1)=cplx(0.0, 1.0*Ytemp(2))
Y(3,1)=cplx(0.0, 1.0*Ytemp(3))
Y(3,2)=cplx(0.0, 1.0*Ytemp(5))

end subroutine conversor_Y
!*****

SUBROUTINE conversor_carga( cx_Sc, md_Sc, Sc, cx_Sd, md_Sd, Sd, cx_Bc, Sbc, &
                           S1YS, S2YS, S3YS, S4YS, &
                           S1YI, S2YI, S3YI, S4YI, &
                           S1YZ, S2YZ, S3YZ, S4YZ, &
                           S1DS, S2DS, S3DS, S4DS, &
                           S1DI, S2DI, S3DI, S4DI, &
                           S1DZ, S2DZ, S3DZ, S4DZ, &
                           VSc, VSd, VSbc )

implicit none

!Declaracion de variables I/O

integer, intent (in):: cx_Sc
integer, intent (in):: md_Sc
real, dimension(6), intent(in):: Sc

integer, intent (in):: cx_Sd
integer, intent (in):: md_Sd
real, dimension(6), intent(in):: Sd

integer, intent (in):: cx_Bc
real, dimension(3), intent(in):: Sbc

complex, dimension(3), intent(out):: S1YS, S2YS, S3YS, S4YS
complex, dimension(3), intent(out):: S1YI, S2YI, S3YI, S4YI
complex, dimension(3), intent(out):: S1YZ, S2YZ, S3YZ, S4YZ

complex, dimension(3), intent(out):: S1DS, S2DS, S3DS, S4DS
complex, dimension(3), intent(out):: S1DI, S2DI, S3DI, S4DI
complex, dimension(3), intent(out):: S1DZ, S2DZ, S3DZ, S4DZ

complex, dimension (3), intent(out) :: VSc !Vector de carga concentrada
complex, dimension (3), intent(out) :: VSd !Vector de carga distribuida
complex, dimension (3), intent(out) :: VSbc !Vector de carga de Banco Capacitivo
!Declaracion de variables internas
real:: FS2 = 2.0 / 3.0 !factores auxiliares para modelar lineas con carga distribuida
real:: FS4 = 1.0 / 3.0 !ver Kersting (estos son para cargas)

!.....
!
VSc(1)=CMPLX(Sc(1),Sc(2))
VSc(2)=CMPLX(Sc(3),Sc(4))
VSc(3)=CMPLX(Sc(5),Sc(6))

VSd(1)=CMPLX(Sd(1),Sd(2))
VSd(2)=CMPLX(Sd(3),Sd(4))
VSd(3)=CMPLX(Sd(5),Sd(6))

VSbc(1)= CMPLX( 0.0 , -Sbc(1) )
VSbc(2)= CMPLX( 0.0 , -Sbc(2) )
VSbc(3)= CMPLX( 0.0 , -Sbc(3) )

!.....
!
!Inicializacion de cargas

S1YS = (0.0, 0.0)

```



S2YS = (0.0, 0.0)
S3YS = (0.0, 0.0)
S4YS = (0.0, 0.0)

S1YI = (0.0, 0.0)
S2YI = (0.0, 0.0)
S3YI = (0.0, 0.0)
S4YI = (0.0, 0.0)

S1YZ = (0.0, 0.0)
S2YZ = (0.0, 0.0)
S3YZ = (0.0, 0.0)
S4YZ = (0.0, 0.0)

S1DS = (0.0, 0.0)
S2DS = (0.0, 0.0)
S3DS = (0.0, 0.0)
S4DS = (0.0, 0.0)

S1DI = (0.0, 0.0)
S2DI = (0.0, 0.0)
S3DI = (0.0, 0.0)
S4DI = (0.0, 0.0)

S1DZ = (0.0, 0.0)
S2DZ = (0.0, 0.0)
S3DZ = (0.0, 0.0)
S4DZ = (0.0, 0.0)

!
,

IF (cx_Sc == 1 .AND. md_Sc == 1) S4YS = VSc
IF (cx_Sc == 1 .AND. md_Sc == 2) S4YI = VSc
IF (cx_Sc == 1 .AND. md_Sc == 3) S4YZ = VSc

IF (cx_Sc == 2 .AND. md_Sc == 1) S4DS = VSc
IF (cx_Sc == 2 .AND. md_Sc == 2) S4DI = VSc
IF (cx_Sc == 2 .AND. md_Sc == 3) S4DZ = VSc

IF (cx_Sd == 1) **THEN**

SELECT CASE (md_Sd)

CASE (1)

S2YS = FS2 * VSd
S4YS = S4YS + FS4 * VSd

CASE (2)

S2YI = FS2 * VSd
S4YI = S4YI + FS4 * VSd

CASE (3)

S2YZ = FS2 * VSd
S4YZ = S4YZ + FS4 * VSd

ENDSELECT

ENDIF

IF (cx_Sd == 2) **THEN**

SELECT CASE (md_Sd)

CASE (1)

S2DS = FS2 * VSd
S4DS = S4DS + FS4 * VSd

CASE (2)

S2DI = FS2 * VSd



```

S4DI = S4DI + FS4 * VSd
CASE (3)
S2DZ = FS2 * VSd
S4DZ = S4DZ + FS4 * VSd

ENDSELECT
ENDIF

IF (CX_BC == 1) S4YZ = S4YZ + VSBC
IF (CX_BC == 2) S4DZ = S4DZ + VSBC

end subroutine conversor_carga
!*****
subroutine arReg (Tap, ar)

implicit none

!Declaración de variables I/O
real, dimension(3), intent(in) :: Tap
real, dimension(3), intent(out) :: ar

!::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
:
ar(1) = 1.0 + 0.00625 * Tap(1)
ar(2) = 1.0 + 0.00625 * Tap(2)
ar(3) = 1.0 + 0.00625 * Tap(3)

end subroutine arReg
!*****

subroutine conversor_voltaje(Vrootr, Vrootc)
implicit none

!Declaración de variables I/O
real, dimension(6), intent(in):: vrootr !kv polar
complex, dimension(3), intent(out)::Vrootc !kv rectangular
real, parameter::pi=3.141593

!Declaración de variables internas
real, dimension(3)::mag,tetha !contenedor de magnitud y angulos en
radianes

!::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
:
!Se inicializa la salida con valor nulo
Vrootc=cplx(0.0, 0.0)

!Se preprocesan los datos de entrada
mag(1)=Vrootr(1)
mag(2)=Vrootr(3)
mag(3)=Vrootr(5)

tetha(1)=vrootr(2)*(pi/180.0)
tetha(2)=vrootr(4)*(pi/180.0)
tetha(3)=vrootr(6)*(pi/180.0)

!Se llena el vector complejo
Vrootc(1)=cplx(mag(1)*cos(tetha(1)), mag(1)*sin(tetha(1)))
Vrootc(2)=cplx(mag(2)*cos(tetha(2)), mag(2)*sin(tetha(2)))
Vrootc(3)=cplx(mag(3)*cos(tetha(3)), mag(3)*sin(tetha(3)))

end subroutine conversor_voltaje

!*****
subroutine calculoY2 ( Y, Y2pre, Y2k)

```



```
implicit none
```

```
!Decalaracion de variables I/O
```

```
complex, dimension(3,3), intent(in):: Y
```

```
complex, dimension(3,3), intent(out):: Y2pre
```

```
complex, dimension(3,3), intent(out):: Y2k
```

```
!AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

```
^
```

```
Y2k = Y2k + (1.0 / 2.0) * Y
```

```
Y2pre= Y2pre + (1.0 / 2.0) * Y
```

```
end subroutine calculoY2
```

```
!*****
```

```
subroutine conversor_Zm(Ztemp, Z) !subrutina para convertir la entrada de la impedancia de mallas a complejos
```

```
!05feb2010
```

```
implicit none
```

```
!Declaracion de variables I/O
```

```
real, dimension(12), intent(in) ::Ztemp
```

```
complex, dimension(3,3), intent(out)::Z
```

```
!Inicializacion de impedancias serie
```

```
Z = (0.0, 0.0)
```

```
!Se asigna la entrada de datos al contenedor Z que es una matriz compleja de 3X3
```

```
Z(1,1)= cplx(Ztemp(1) , Ztemp(2) )  
Z(1,2)= cplx(Ztemp(3) , Ztemp(4) )  
Z(1,3)= cplx(Ztemp(5) , Ztemp(6) )  
Z(2,2)= cplx(Ztemp(7) , Ztemp(8) )  
Z(2,3)= cplx(Ztemp(9) , Ztemp(10) )  
Z(3,3)= cplx(Ztemp(11), Ztemp(12) )
```

```
Z(2,1)= cplx(Ztemp(3) , Ztemp(4) )  
Z(3,1)= cplx(Ztemp(5) , Ztemp(6) )  
Z(3,2)= cplx(Ztemp(9) , Ztemp(10) )
```

```
end subroutine conversor_Zm
```

```
!*****  
!*****  
!*****
```

```
end module rutinas_de_asignacion
```

B.13. Utileriascomplejas1.f90

```
!=====  
!=====  
!RADIFLUX 1.0  
!  
!FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION  
!  
!20 de DICIEMBRE de 2008  
!=====  
!  
!IVAN ARCHUNDIA ARANDA  
!iarchundia@ipn.mx  
!  
!=====  
!  
!INSTITUTO POLITECNICO NACIONAL  
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA  
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION  
!=====
```



```

!=====
module UtileriasComplejas1

contains
subroutine RectangularPolar (unit, V, VC)
!Declaracion de variables I/O
integer, intent(in) :: unit
character, intent(in) :: V
complex, dimension(3), intent(in) :: VC
!Declaracion de variables internas
real, dimension(3) :: mag
real, dimension(3) :: ang
integer :: k
!-----

mag= abs(VC)

do k=1, 3
IF ( Imag( VC(k) ) /= 0.0 .OR. Real( VC(k) ) /= 0.0 ) then
      ang(k) = ATAN2D( Imag(VC(k)), Real (VC(k)) )
ELSE
      ang(k)= 0.0
endif
enddo
!-----
write(unit,'(1X, A5 ,":", 2X, F10.5, 2X,"@", F10.5, 4X , F10.5, 2X , "@", F10.5, 4X, F10.5,
2X,"@", F10.5 )') &
      V, mag(1), ang(1), mag(2), ang(2), mag(3), ang(3)

end subroutine RectangularPolar
!*****
!*****
end module UtileriasComplejas1

```

B.14. Utileriascomplejas2.f90

```

!=====
!=====
!RADIFLUX 1.0
!
!FLUJOS DE POTENCIA EN REDES RADIALES DE DISTRIBUCION
!
!20 de DICIEMBRE de 2008
!=====
!
!IVAN ARCHUNDIA ARANDA
!iarchundia@ipn.mx
!
!=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!=====

module UtileriasComplejas2

contains
subroutine RecPol (unit, n, BusName, V)
!Declaracion de variables I/O
integer, intent(in) :: unit !unidad para impresion
integer, intent(in) :: n !nodo numero...
character(10), intent(in) :: BusName !nombre del nodo
complex, dimension(3), intent(in) :: V !voltaje de entrada en forma rectangular
!Declaracion de variables internas
real, dimension(3) :: mag
real, dimension(3) :: ang
integer :: k
!-----

mag= abs(V)

```




```

do      k=1, 3
  IF ( Imag( V(k) ) /= 0.0 .OR. Real( V(k) ) /= 0.0 ) then
    ang(k) = ATAN2D( Imag(V(k)), Real (V(k)) )
  ELSE
    ang(k)= 0.0
  endif
enddo
!-----
write(unit,'(2X, I5,2X, A10 ,3X, F7.3, 1X,"@",1X, F7.2, 3X , F7.3, 1X ,"@",1X, F7.2, 3X,
F7.3, 1X,"@",1X, F7.2 )') &
n, BusName, mag(1), ang(1), mag(2), ang(2), mag(3), ang(3)
end subroutine RecPol

!*****
subroutine RecPol_pu (unit, n, BusName, V)
!Declaracion de variables I/O
integer, intent(in) :: unit !unidad para impresion
integer, intent(in) :: n !nodo numero...
character(10), intent(in) :: BusName !nombre del nodo
complex,dimension(3),intent(in) :: V !voltaje de entrada en forma rectangular
!Declaracion de variables internas
real, dimension(3):: mag
real, dimension(3):: ang
integer:: k
!-----
mag= abs(V)
do      k=1, 3
  IF ( Imag( V(k) ) /= 0.0 .OR. Real( V(k) ) /= 0.0 ) then
    ang(k) = ATAN2D( Imag(V(k)), Real (V(k)) )
  ELSE
    ang(k)= 0.0
  endif
enddo
!-----
write(unit,'(2X, I5,2X, A10 ,3X, F7.5, 1X,"@",1X, F7.2, 3X , F7.5, 1X ,"@",1X, F7.2, 3X,
F7.5, 1X,"@",1X, F7.2 )') &
n, BusName, mag(1), ang(1), mag(2), ang(2), mag(3), ang(3)
end subroutine RecPol_pu
!*****
subroutine RecPol_1 (unit,V)
!Declaracion de variables I/O
integer, intent(in) :: unit !unidad para impresion
complex,dimension(3),intent(in) :: V !voltaje de entrada en forma rectangular
!Declaracion de variables internas
real, dimension(3):: mag
real, dimension(3):: ang
integer:: k
!-----
mag= abs(V)
do      k=1, 3
  IF ( Imag( V(k) ) /= 0.0 .OR. Real( V(k) ) /= 0.0 ) then
    ang(k) = ATAN2D( Imag(V(k)), Real (V(k)) )
  ELSE
    ang(k)= 0.0
  endif
enddo
!-----

```



```
write(unit,'(2X, "      ",2X, "      ",3X, F7.3, 1X,"@",1X, F7.2, 3X , F7.3, 1X
,"@",1X, F7.2, 3X, F7.3, 1X,"@",1X, F7.2 )') &
mag(1), ang(1), mag(2), ang(2), mag(3), ang(3)
end subroutine RecPol_1

!*****
end module UtileriasComplejas2
```

B.15. Activar.f90

```
=====
!
! DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
!
! JUNIO 2012
!
!
! RAUL GUILLERMO JARA SIXTOS
! rjaras0400@ipn.mx
!
!
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!
=====
```

```
SUBROUTINE ACTIVAR(N)
!ESTA FUNCION ACTIVA LOS ESTADOS POSIBLES EN EL ESCENARIO
!PARA LA CORRIDA DE FLUJOS
USE DATA_DESPACHO
IMPLICIT NONE
```

```
INTEGER :: P,I,K,ACTIVOS,SALIDA
INTEGER :: N
```

```
INTEGER, ALLOCATABLE :: BC(:,:),CD(:,:)
ALLOCATE (BC(1,capacitors),CD(1,capacitors))
BC=0
CD=0
EDOS_ACT=2
ACTIVOS=256
```

```
DO I=1,NESC*2
  IF(SPERDIDAS(I).GT.0) THEN
    DO p=0,2**CAPACITORS-1
      DO K=1,CAPACITORS
        BC(1,K)=MOD(EDOS_ANT(N,I,K),2)
        CD(1,K)=EDOS_ANT(N,I,K)
      END DO
      CALL TABLA_DE_ESCENARIOS(p)
      SALIDA=1
      DO K=1,CAPACITORS
        IF(BC(1,k).NE.AB(1,K)) THEN
          CD(1,K)=CD(1,K)+1
        END IF
        IF(CD(1,K).GT.ON_OFF) THEN
          SALIDA=0
          EXIT
        EXIT
      END IF
    END DO !K
    IF (SALIDA.EQ.1) THEN
      IF (EDOS_ACT(P+1).EQ.0) THEN
        ACTIVOS=ACTIVOS+1
      END IF
      EDOS_ACT(P+1)=1
    ELSE
      IF(EDOS_ACT(P+1).EQ.2) THEN
        EDOS_ACT(P+1)=0
```



```

ACTIVOS=ACTIVOS-1
      END IF
    END IF
  END DO !P
END IF
END DO !I

IF (EDOS_ACT(1).EQ.0) THEN
  EDOS_ACT(1)=2
END IF
IF (EDOS_ACT(2**CAPACITORS).EQ.0) THEN
  EDOS_ACT(2**CAPACITORS)=2
END IF

DEALLOCATE (BC,CD)
59 FORMAT (A6,8I3)
60 FORMAT (A6,X,I3,2X,8I3)
57 FORMAT (I9,A4,8I3)
END SUBROUTINE ACTIVAR

```

B.16. Actualiza.f90

```

=====
!
! DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
! JUNIO 2012
=====
! RAUL GUILLERMO JARA SIXTOS
! rjaras0400@ipn.mx
!
=====
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
=====
SUBROUTINE ACTUALIZA(M)
! DESPUES DE ELEGIR LOS NESC ESTADOS CON MENOS PERDIDAS DE LA CORRIDA DE FLUJOS
! ESTA RUTINA ACTUALIZA LOS ESTADOS DE ACUERDO AL NUMERO DE CAMBIOS
USE DATA_DESPACHO

IMPLICIT NONE

INTEGER :: M,N,K,I,J,L,DIF,CAMBIO,CONTA,SNEG,SPOS,DES1,DES2,MARCA
REAL :: PERD
REAL, ALLOCATABLE :: PERDIDAS_ACUM(:)

INTEGER :: ESC_NEW(NESC*2,CAPACITORS),ESC_AP(NESC*2),ESC_BIN(NESC*2),ESC_AUX(CAPACITORS)
REAL :: ESC_PERD(NESC*2),ESC_OR(NESC*2)

ALLOCATE (PERDIDAS_ACUM(INT(NESC*2)))

PERDIDAS_ACUM=0

IF (M.NE.1) THEN

  ! *****
  ! BUSCA TRAYECTORIAS EN CADA ETAPA
  PERDIDAS_ACUM=-99
  CONTA=1
  ESC_NEW=ON_OFF+1
  ESC_AP=0
  ESC_PERD=-99
  ESC_BIN=2**CAPACITORS+1
  ESC_OR=-99
  DO i=1,nesc*2
    IF (PERDIDAS_X_ESCENARIO(M,I).GT.0) THEN
      DO k=1,nesc*2

```



```
IF (SPERDIDAS(K).GT.0) THEN
MARCA=1
DO j=1,capacitors
DIF=EDOS_ANT(M,I,J)-EDOS_ANT(M-1,K,J)
IF( (DIF.NE.1) .AND. (DIF.NE.0)) THEN
MARCA=0

IF (CONTA.LE.NESC*2) THEN
DO n=1,capacitors

DIF=MOD(EDOS_ANT(M,I,N),2)-MOD(EDOS_ANT(M-1,K,N),2)
IF( DIF.EQ.0) THEN
ESC_AUX(N)=EDOS_ANT(M-1,K,N)
ELSE
ESC_AUX(N)=EDOS_ANT(M-1,K,N)+1
IF (ESC_AUX(N).GT.ON_OFF) THEN

CONTA=CONTA-1
EXIT
END IF
END IF

END DO !N
CAMBIO=0
IF (N.GT.CAPACITORS) THEN

DO n=1,conta
IF (ESC_BIN(n).EQ.EDOS_ANT_BIN(M,I)) THEN
PERD=PERDIDAS_X_ESCENARIO(M,I)+SPERDIDAS(K)
DES1=1
DO L=1,CAPACITORS
IF(ESC_AUX(L).NE.ESC_NEW(N,L))THEN
DES1=0
EXIT
END IF
END DO

IF (DES1.EQ.1) THEN

IF (ESC_PERD(N).GT.0) THEN
IF (PERD.LT.ESC_PERD(N)) THEN
CAMBIO=N

ELSE
CAMBIO=-N
END IF

ELSE
CAMBIO=N
END IF
exit
END IF
END IF

END DO
IF (CAMBIO.EQ.0) THEN
DO n=1,capacitors
ESC_NEW(conta,N)=ESC_AUX(N)
END DO

ESC_PERD(conta)=PERDIDAS_X_ESCENARIO(M,I)+SPERDIDAS(K)
ESC_BIN(conta)=EDOS_ANT_BIN(M,I)
ESC_AP(conta)=K
ESC_OR(conta)=PERDIDAS_X_ESCENARIO(M,I)
CONTA=CONTA+1
ELSE IF(CAMBIO.GT.0)THEN
DO n=1,capacitors
ESC_NEW(CAMBIO,N)=ESC_AUX(N)
END DO

ESC_PERD(CAMBIO)=PERDIDAS_X_ESCENARIO(M,I)+SPERDIDAS(K)
ESC_BIN(CAMBIO)=EDOS_ANT_BIN(M,I)
ESC_AP(CAMBIO)=K
ESC_OR(CAMBIO)=PERDIDAS_X_ESCENARIO(M,I)

END IF
```



```

        CONTA=CONTA-1
    END IF
    IF(CONTA.GE.NESC*2)THEN
        CALL SORT2(
        ESC_PERD,NESC*2,ESC_NEW,CAPACITORS,ESC_BIN,ESC_AP,ESC_OR)
    END IF
ELSE
    PERD=PERDIDAS_X_ESCENARIO(M,I)+SPERDIDAS(K)
    IF (PERD.LT.ESC_PERD(NESC*2)) THEN
        DO n=1,capacitors
            DIF=MOD(EDOS_ANT(M,I,N),2)-MOD(EDOS_ANT(M-1,K,N),2)
            IF( DIF.EQ.0) THEN
                ESC_AUX(N)=EDOS_ANT(M-1,K,N)
            ELSE
                ESC_AUX(N)=EDOS_ANT(M-1,K,N)+1
                IF (ESC_AUX(N).GT.ON_OFF) THEN
                    CONTA=CONTA-1
                    EXIT
                END IF
            END IF
        END DO !N
        CAMBIO=0
        IF (N.GT.CAPACITORS) THEN
            DO n=1,NESC*2
                IF (ESC_BIN(n).EQ.EDOS_ANT_BIN(M,I)) THEN
                    DES1=1
                    DO L=1,CAPACITORS
                        IF(ESC_AUX(L).NE.ESC_NEW(N,L))THEN
                            DES1=0
                            EXIT
                        END IF
                    END DO
                    IF (DES1.EQ.1) THEN
                        IF (PERD.LT.ESC_PERD(N)) THEN
                            CAMBIO=N
                        ELSE
                            CAMBIO=-N
                        END IF
                    END IF
                    exit
                END IF
            END DO
            IF (CAMBIO.EQ.0) THEN
                DO n=1,capacitors
                    ESC_NEW(NESC*2,N)=ESC_AUX(N)
                END DO
                N=NESC*2
            ESC_PERD(N)=PERDIDAS_X_ESCENARIO(M,I)+SPERDIDAS(K)
            ESC_BIN(N)=EDOS_ANT_BIN(M,I)
            ESC_AP(N)=K
            ESC_OR(N)=PERDIDAS_X_ESCENARIO(M,I)
            CALL SORT2(
            ESC_PERD,n,ESC_NEW,CAPACITORS,ESC_BIN,ESC_AP,ESC_OR)
            ELSE IF(CAMBIO.GT.0)THEN
                DO n=1,capacitors
                    ESC_NEW(CAMBIO,N)=ESC_AUX(N)
                END DO
                N=NESC*2
            ESC_PERD(CAMBIO)=PERDIDAS_X_ESCENARIO(M,I)+SPERDIDAS(K)
            ESC_BIN(CAMBIO)=EDOS_ANT_BIN(M,I)
            ESC_AP(CAMBIO)=K
            ESC_OR(CAMBIO)=PERDIDAS_X_ESCENARIO(M,I)
            CALL SORT2(
            ESC_PERD,n,ESC_NEW,CAPACITORS,ESC_BIN,ESC_AP,ESC_OR)
        END IF

```



```

                                END IF
                            END IF
                        END IF
                        CONTA=CONTA+1
                        EXIT
                    END IF
                END DO !J
                IF (MARCA.EQ.1 ) THEN
                    PERD=PERDIDAS_X_ESCENARIO(M,I)+SPERDIDAS(K)

                    IF (PERDIDAS_ACUM(I).GT.0) THEN
                        IF(PERD.LT.PERDIDAS_ACUM(I)) THEN
                            PERDIDAS_ACUM(I)=PERD
                            TRAYM(M-1,I)=K
                        END IF
                    ELSE
                        PERDIDAS_ACUM(I)=PERD
                        TRAYM(M-1,I)=K
                    END IF
                END IF
            END IF
        END DO !K
    END DO !i
ELSE !M
    DO K=1,NESC*2
        PERDIDAS_ACUM(K)=PERDIDAS_X_ESCENARIO(1,K)
    END DO

    END IF !M
    SPERDIDAS=PERDIDAS_ACUM
    MARCA=INT(NESC*2)

    IF(M.GT.1)THEN
        CALL SORT2may( SPERDIDAS,MARCA,EDOS_ANT(m, :, :),CAPACITORS,EDOS_ANT_BIN(m, :),TRAYM(M-1, :),PERDIDAS_X_ESCENARIO(M, :))
        DO J=1,NESC*2
            IF(SPERDIDAS(J).LT.0) THEN
                CAMBIO=J-1
                EXIT
            END IF
        END DO

        DO J=1,NESC*2-CAMBIO
            SPERDIDAS(CAMBIO+J)=ESC_PERD(J)
            EDOS_ANT_BIN(m,CAMBIO+J)=ESC_BIN(J)
            TRAYM(M-1,CAMBIO+J)=ESC_AP(J)
            PERDIDAS_X_ESCENARIO(M,CAMBIO+J)=ESC_OR(J)
            DO K=1,CAPACITORS
                EDOS_ANT(m,CAMBIO+J,K)=ESC_NEW(J,K)
            END DO
        END DO
    END DO
    END IF
    DEALLOCATE (PERDIDAS_ACUM)
END SUBROUTINE ACTUALIZA

```

B.17. Backtrack.f90

```

=====
!
!
!DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
!
!JUNIO 2012
!
=====
!
!RAUL GUILLERMO JARA SIXTOS
!rjaras0400@ipn.mx

```



```

!
!=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!
SUBROUTINE BACKTRACKT(TRANS)

USE DATA_DESPACHO

IMPLICIT NONE

INTEGER:: TRANS
INTEGER :: TEMP
REAL, ALLOCATABLE :: LOSSV(:)
INTEGER, ALLOCATABLE :: TRAY(:),OP_ACUM(:)

REAL(8), ALLOCATABLE :: sumx(:),sumxy(:),mx(:, :)
INTEGER :: i,j,k,n
REAL(8), ALLOCATABLE :: pol(:)

ALLOCATE (TRAY(escenarios),LOSSV(escenarios),OP_ACUM(escenarios))

n=1
ALLOCATE (sumx(2*n+1),sumxy(n+1),mx(n+1,n+1),pol(n+1))
pol=0.0
sumx=0.0
sumxy=0.0
mx=0.0
DO i=1,2*n+1
  DO j=1,escenarios
    sumx(i)=sumx(i)+j**(i-1)
  END DO
END DO
DO i=1,n+1
  DO j=1,escenarios
    sumxy(i)=sumxy(i)+j**(i-1)*TRAYT(TRANS,j)
  END DO
END DO
DO i=1,n+1
  k=i
  DO j=1,n+1
    mx(i,j)=sumx(k)
    k=k+1
  END DO
END DO
CALL GAUSS(n+1, mx, sumxy,pol)

DO n=1,escenarios
  temp=NINT((POL(2)*n+POL(1)))

  IF (temp.LT.mini(n)) THEN
    TRAYT(TRANS,n)=mini(n)
  ELSE IF (temp.GT.maxi(n)) THEN
    TRAYT(TRANS,n)=maxi(n)
  ELSE
    TRAYT(TRANS,n)=temp
  END IF
END DO

59 FORMAT (24(I2,X))

DEALLOCATE (TRAY,LOSSV)

END SUBROUTINE BACKTRACKT

```

B.18. Data_despacho.f90

```

!=====
!
!

```



```

!DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
!
!JUNIO 2012
!=====
!
!RAUL GUILLERMO JARA SIXTOS
!rjaras0400@ipn.mx
!
!=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!
MODULE DATA_DESPACHO

IMPLICIT NONE

INTEGER,ALLOCATABLE :: EDOS_ACT(:)
INTEGER,ALLOCATABLE :: EDOS_ANT(:,:,:), EDOS_ANT_BIN(:,:)
INTEGER,ALLOCATABLE :: AB(:,:)
REAL, ALLOCATABLE :: PERDIDAS_X_ESCENARIO(:,:)
INTEGER, ALLOCATABLE :: PREVIO(:,:)
REAL,ALLOCATABLE :: SPERDIDAS(:)
INTEGER :: NESC
INTEGER, ALLOCATABLE :: TOPTIMAC(:),TRAYC_ANT(:),TRAYC(:),TRAYT(:,:),TRAYT_ant(:,:)
!,mtray(:,:)
REAL :: LOSST,loss
INTEGER:: capacitores
INTEGER:: TRANSFORMERS
INTEGER :: on_off,escenarios,tap_lim
REAL, ALLOCATABLE :: porcentaje_de_carga(:)

INTEGER, ALLOCATABLE :: UB_TRANS(:),UB_CAP(:),POSIBLEM(:,:,:),mini(:),maxi(:)
COMPLEX, allocatable :: CAP_OR(:,:),LOADC_OR(:,:),CAP_OR2(:,:),LOADC_OR2(:,:)
INTEGER, ALLOCATABLE :: TRAYM(:,:)
complex, dimension(3) :: Vrootc

END MODULE DATA_DESPACHO

```

B.19. Despachoc.f90

```

!=====
!=====
!
!DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
!
!JUNIO 2012
!=====
!
!RAUL GUILLERMO JARA SIXTOS
!rjaras0400@ipn.mx
!
!=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!=====
!
SUBROUTINE DESPACHOC(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2, Ithevenin,
Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro )

use rutinas_de_asignacion
use ConstantesG14
use ConstantesG2
use ConstantesG3
USE MSIMSL
USE estructuras
USE estructuras2
USE estructuras3
USE DATA_DESPACHO
use BackForw
use UtileriasComplejas2

```




implicit none

```
!Declaracion de variables I/O
integer, intent(in)::n2
type(lista), dimension(n2), intent(inout) :: nodo !INOUT para meter todo y sacar voltaje
```

```
integer, intent(in) :: maxIter
integer, intent(inout) :: iteraciones
```

```
integer, intent(inout) :: converge
real , intent(in) :: tolerancia
```

```
!23feb2010
```

```
integer, intent(in) :: m2
complex, dimension(3*m2) , intent(inout):: Ithevenin
complex, dimension(3*m2, 3*m2), intent(in ):: Ythevenin
complex, dimension(3*m2) , intent(inout):: Vthevenin
```

```
type(lista_mallas), dimension(m2), intent(in):: malla !24feb2010
type(THV_est), intent(inout) :: thv !24feb2010--
!03mar2010
```

```
integer, intent(inout)::iteracionesMacro
integer, intent(inout)::convergeMacro
```

```
!Declaracion de variables internas
integer ::k, t,caps !Contador(es)
```

```
complex, dimension(3):: V4antes
complex, dimension(3):: ERROR
```

```
INTEGER:: m,n,p,j,l,auxpos,POS ,salto,amax
```

```
REAL:: LOSS_ON,LOSS_OFF,maxp,Vpu(3)
```

```
INTEGER, ALLOCATABLE :: TRAYECTORIA(:)
```

```
NESC=(2*(CAPACITORS))*ON_OFF
```

```
if (nesc.LT.1)then
  nesc=1
end if
M=INT(NESC*2)
```

```
ALLOCATE(AB(1,CAPACITORS),EDOS_ACT(2**CAPACITORS),EDOS_ANT(escenarios,M,capacitors),PERDIDA
S_X_ESCENARIO(escenarios,M),EDOS_ANT_BIN(escenarios,M))
ALLOCATE (PREVIO(escenarios,M),SPERDIDAS(M),TRAYECTORIA(ESCENARIOS),TRAYM(ESCENARIOS,M))
! ,mtray(nesc,nesc)
```

```
AB = 0
EDOS_ACT=1
EDOS_ANT=ON_OFF+1
EDOS_ANT_BIN=0
SPERDIDAS=0
```

```
loss_off = 0.0
loss_on = 0.0
PERDIDAS_X_ESCENARIO=-99
previo=0
trayectoria=0
```

```
TRAYM=0
```

```
DO m = 1,escenarios ! RESUELVE FLUJOS DE POTENCIA PARA TODOS LOS ESCENARIOS
! write(*,77) "Escenario ",m
77 FORMAT (A10,2X,I2)
IF(M.GT.1) THEN
  CALL ACTIVAR(M-1) ! Marca los posibles estados a seguir
ELSE
  EDOS_ACT=1
END IF
```



```

auxpos=0
do k=1,n2
  DO caps=1,3
    NODO(k)%S4YS(CAPS)=LOADC_OR(K,CAPS)*porcentaje_de_carga(m)/100.0
    NODO(k)%SC(CAPS)=LOADC_OR2(K,CAPS)*porcentaje_de_carga(m)/100.0
  END DO !FALTA AGREGAR CARGA DISTRIBUIDA
end do

!*****
!asignacion de taps para cada escenario

DO k=1,transformers
  DO p=1,3
    nodo(UB_TRANS(k))%Tap(p)=TRAYT(k,m)
  END DO
!
! WRITE(*,78) ,K,nodo(UB_TRANS(k))%Tap(1)
78 FORMAT (15X,'TAP ',I2,'=' ',2X,F5.1)
  call arReg (nodo(UB_TRANS(k))%Tap, nodo(UB_TRANS(k))%ar)

  call dAB14( nodo(UB_TRANS(k))%lc_Reg, nodo(UB_TRANS(k))%ty_Reg,
nodo(UB_TRANS(k))%cx_Reg, nodo(UB_TRANS(k))%ar, nodo(UB_TRANS(k))%Z1, nodo(UB_TRANS(k))%Z4,
&
nodo(UB_TRANS(k))%A1, nodo(UB_TRANS(k))%B1, &
nodo(UB_TRANS(k))%A4, nodo(UB_TRANS(k))%B4
)
nodo(UB_TRANS(k))%d1,
nodo(UB_TRANS(k))%d4,
)
  call dAB2 (nodo(UB_TRANS(k))%Z2, nodo(UB_TRANS(k))%d2, nodo(UB_TRANS(k))%A2,
nodo(UB_TRANS(k))%B2)
  call dAB3 (nodo(UB_TRANS(k))%tybr,nodo(UB_TRANS(k))%nt, nodo(UB_TRANS(k))%Z3,
nodo(UB_TRANS(k))%d3, nodo(UB_TRANS(k))%A3, nodo(UB_TRANS(k))%B3)

END DO

DO n = 0,(2**CAPACITORS-1) !Genera los estados de los capacitores

IF (EDOS_ACT(n+1).NE.0) THEN
  CALL TABLA_DE_ESCENARIOS(N)

  do k=1,CAPACITORS
    DO caps=1,3
      NODO(UB_CAP(K))%S4YZ(CAPS)=CAP_OR(K,CAPS)*AB(1,k)
      NODO(UB_CAP(K))%SBC(CAPS)=CAP_OR2(K,CAPS)*AB(1,k)
    END DO
  end do

  iteraciones=0
  converge=1
  convergeMacro=-1
  SALTO=0
  call BackwardForward(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2,
Ithevenin, Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro )
  do k=1, n2
    Vpu=abs(nodo(k)%V4 / abs(nodo(k)%VR4)
    DO p=1,3
      IF( ((Vpu(p).LT.0.90) .OR. (Vpu(p).GT. 1.08)) .AND. (Vpu(p).GT. 0.1) )
THEN
        SALTO=1
        EXIT
      END IF
    END DO
    IF(SALTO.EQ.1)THEN
      exit
    end if
  end do
  IF(SALTO.EQ.0)THEN

    IF(EDOS_ACT(N+1).NE.2)THEN
      IF(auxpos.LT.NESC) THEN
        auxpos=auxpos+1
        PERDIDAS_X_ESCENARIO(m,auxpos) = loss
        DO p=1,capacitors
          EDOS_ANT(m, auxpos, p)=AB(1, p)
          EDOS_ANT_BIN(m, auxpos)=N
        END DO
      END IF
    END IF
  END IF
END IF

```



```

        IF(auxpos.EQ.NESC) THEN
            maxp=PERDIDAS_X_ESCENARIO(m,1)
            amax=1
            DO p=2,NESC
                IF(PERDIDAS_X_ESCENARIO(m,p).GT.maxp) THEN
                    maxp=PERDIDAS_X_ESCENARIO(m,p)
                    amax=p
                END IF
            END DO
        END IF
    ELSE
        ! 4 es el numero de estados a guardar
        IF(loss.LT.maxp) THEN
            PERDIDAS_X_ESCENARIO(M,amax) = loss ! si este estado tiene
            menos perdidas que alguno almacenado para este escenario
            DO p=1,capacitors ! almacena las perdidas y
                EDOS_ANT(m,amax,p)=AB(1,p)
            END DO
            EDOS_ANT_BIN(m,amax)=n
            maxp=PERDIDAS_X_ESCENARIO(m,1)
            amax=1
            DO p=2,NESC
                IF(PERDIDAS_X_ESCENARIO(m,p).GT.maxp) THEN
                    maxp=PERDIDAS_X_ESCENARIO(m,p)
                    amax=p
                END IF
            END DO
        END IF !loss.LT.maxp

        END IF
        END IF !EDOS_ACT(N+1).NE.2
        END IF !SALTO.EQ.0
        IF (N.eq.0) THEN ! Guarda las perdidas para un estado all_off
            LOSS_OFF=LOSS_OFF+loss
        END IF
        IF (N.eq.(2**capacitors-1)) THEN ! Guarda las perdidas para un estado all_off
            LOSS_ON=LOSS_ON+loss
        END IF
    END IF
END DO !N

CALL SORTC(
PERDIDAS_X_ESCENARIO(M,1:NESC),NESC,EDOS_ANT(m,1:NESC,1:CAPACITORS),CAPACITORS,EDOS_ANT_BIN
(m,1:NESC))
CALL ACTUALIZA(M)

END DO !M

IF(SPERDIDAS(1).GT.0) THEN
    loss=Sperdidas(1)
    POS=1

    DO k=2,NESC*2
        IF((loss.GT.SPERDIDAS(K)).AND.(SPERDIDAS(K).GT.0)) THEN
            loss=Sperdidas(k)
            pos=k
        END IF
    END DO

    TRAYECTORIA(ESCENARIOS)=POS

    DO M=ESCENARIOS-1,1,-1
        TRAYECTORIA(M)=TRAYM(M,POS)
        POS=TRAYECTORIA(M)
    END DO

    WRITE (101,*)
    ! WRITE (101,*) 'EL NUMERO DE ESTADOS ALMACENADOS EN CADA ESCENARIO DE CARGA ES: ',NESC

```



```

! WRITE(101,87) LOSS_OFF
! WRITE(101,88) LOSS_ON
WRITE(101,89) LOSS

WRITE(101,90)
DO M=1, ESCENARIOS
  WRITE(101,91) M, PERDIDAS_X_ESCENARIO(M, TRAYECTORIA(M))
  TRAYC(M)=EDOS_ANT_BIN(m, TRAYECTORIA(M))
  DO k=1, capacitors
    WRITE(101,93) EDOS_ANT(m, TRAYECTORIA(M), K)
  END DO
  WRITE(101,95)
  DO k=1, transformers
    WRITE(101,94) TRAYT(K, M)
  END DO
END DO

WRITE(101,92)
DO M=1, ESCENARIOS
  WRITE(101,91) M, PERDIDAS_X_ESCENARIO(M, TRAYECTORIA(M))
  DO k=1, capacitors
    WRITE(101,93) MOD(EDOS_ANT(m, TRAYECTORIA(M), K), 2)
  END DO
  WRITE(101,95)
  DO k=1, transformers
    WRITE(101,94) TRAYT(K, M)
  END DO
END DO

END IF

DEALLOCATE (AB, EDOS_ACT, EDOS_ANT, PERDIDAS_X_ESCENARIO, EDOS_ANT_BIN)
DEALLOCATE (PREVIO, SPERDIDAS, TRAYECTORIA)
DEALLOCATE (TRAYM)

54 FORMAT (' ', '$)
55 FORMAT (I1, $)
56 FORMAT (5X, F8.3, $)
59 FORMAT (5X, I8, $)
!87 FORMAT (/, 'LAS PERDIDAS CON TODOS LOS CAPACITORES APAGADOS EN TODOS LOS ESCENARIOS ES
:', F15.5)
!88 FORMAT (/, 'LAS PERDIDAS CON TODOS LOS CAPACITORES CONECTADOS EN TODOS LOS ESCENARIOS ES
:', F15.5)
89 FORMAT (/, 'LAS PERDIDAS TOTALES CON LA TRAYECTORIA OPTIMA SON:', F15.5)
90 FORMAT (/, ' LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES: ', 2/, '
ESCENARIO', 5X, 'PERDIDAS', 5X, 'POSICION DE LOS CAPACITORES ', 'POSICION DE LOS TAPS')
91 FORMAT (/, I2, 'H', 5X, F15.5, 4X, ' ', '$)
92 FORMAT (2/, ' LA TRAYECTORIA OPTIMA EN ESTADOS NORMALES ES: ', 2/, '
ESCENARIO', 5X, 'PERDIDAS', 5X, 'POSICION DE LOS CAPACITORES ', 'POSICION DE LOS TAPS')
93 FORMAT (I2, ' ', '$)
94 FORMAT (' ', I6, '$)
95 FORMAT (' ', '$)
END SUBROUTINE DESPACHOC

```

B.20. Despachot.f90

```

=====
!
!
! DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
!
! JUNIO 2012
!
=====
!
! RAUL GUILLERMO JARA SIXTOS
! rjaras0400@ipn.mx
!
=====
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!
=====

```



SUBROUTINE DESPACHOT(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2, Ithevenin, Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro)

```

use rutinas_de_asignacion
use ConstantesG14
use ConstantesG2
use ConstantesG3
USE MSIMSL
USE estructuras
USE estructuras2
USE estructuras3
USE DATA_DESPACHO
use BackForw
use UtileriasComplejas2
implicit none

!Declaracion de Variables I/O
integer, intent(in)::n2
type(lista), dimension(n2), intent(inout) :: nodo !INOUT para meter todo y sacar voltaje

integer, intent(in) :: maxIter
integer, intent(inout) :: iteraciones

integer, intent(inout) :: converge
real , intent(in) :: tolerancia

!23feb2010
integer, intent(in) :: m2
complex, dimension(3*m2) , intent(inout):: Ithevenin
complex, dimension(3*m2, 3*m2), intent(in ):: Ythevenin
complex, dimension(3*m2) , intent(inout):: Vthevenin

type(lista_mallas), dimension(m2), intent(in):: malla !24feb2010
type(THV_est), intent(inout) :: thv !24feb2010--
!03mar2010

integer, intent(inout)::iteracionesMacro
integer, intent(inout)::convergeMacro

!Declaracion de Variables internas
integer ::k, t,caps !Contador(es)

complex, dimension(3):: V4antes
complex, dimension(3):: ERROR

INTEGER:: m,n,p,j,l,auxpos,POS,amax,salto
REAL:: LOSS_ON,LOSS_OFF,maxp,Vpu(3)
INTEGER, ALLOCATABLE :: TRAYECTORIA(:)

NESC=33

ALLOCATE(AB(1,CAPACITORS),EDOS_ANT(escenarios,NESC,transformers),PERDIDAS_X_ESCENARIO(escenarios,NESC) !,EDOS_ACT(2**CAPACITORS),EDOS_ANT_BIN(escenarios,NESC)
ALLOCATE
(PREVIO(escenarios,NESC),SPERDIDAS(NESC),TRAYECTORIA(ESCENARIOS),mini(ESCENARIOS),maxi(ESCENARIOS)) !,mtray(nesc,nesc)

!INICIALIZA VARIABLES
AB = 0
EDOS_ANT=0
SPERDIDAS=0
loss_off = 0.0
loss_on = 0.0
PERDIDAS_X_ESCENARIO=9999999
previo=0
trayectoria=0

!*****
!Se analiza cada regulador por separado
!*****
DO n = 1,transformers !Genera los estados de los taps

```



```

DO m = 1,escenarios ! RESUELVE FLUJOS DE POTENCIA PARA TODOS LOS ESCENARIOS
  auxpos=0
  do k=1,n2
    DO caps=1,3
      NODO(k)%S4YS(CAPS)=LOADC_OR(K,CAPS)*porcentaje_de_carga(m)/100.0
      NODO(k)%SC(CAPS)=LOADC_OR2(K,CAPS)*porcentaje_de_carga(m)/100.0
    END DO !FALTA AGREGAR CARGA DISTRIBUIDA
  end do
  CALL TABLA_DE_ESCENARIOS(TRAYC(m))

  do k=1,CAPACITORS
    DO caps=1,3
      NODO(UB_CAP(K))%S4YZ(CAPS)=CAP_OR(K,CAPS)*AB(1,k)
      NODO(UB_CAP(K))%SBC(CAPS)=CAP_OR2(K,CAPS)*AB(1,k)
    END DO
  end do
  maxp=99999
  amax=-18
  DO L=-16,16

    DO p=1,3
      nodo(UB_TRANS(n))%Tap(p)=L
    END DO

    call arReg (nodo(UB_TRANS(n))%Tap, nodo(UB_TRANS(n))%ar)
    call dAB14( nodo(UB_TRANS(n))%lc_Reg, nodo(UB_TRANS(n))%ty_Reg,
nodo(UB_TRANS(n))%cx_Reg, nodo(UB_TRANS(n))%ar, nodo(UB_TRANS(n))%z1, nodo(UB_TRANS(n))%z4,
&
nodo(UB_TRANS(n))%A1, nodo(UB_TRANS(n))%B1, &
nodo(UB_TRANS(n))%A4, nodo(UB_TRANS(n))%B4
)
!END DO
    call dAB2 (nodo(UB_TRANS(n))%z2, nodo(UB_TRANS(n))%d2, nodo(UB_TRANS(n))%A2,
nodo(UB_TRANS(n))%B2)
    call dAB3 (nodo(UB_TRANS(n))%tybr,nodo(UB_TRANS(n))%nt, nodo(UB_TRANS(n))%z3,
nodo(UB_TRANS(n))%d3, nodo(UB_TRANS(n))%A3, nodo(UB_TRANS(n))%B3)

    iteraciones=0
    converge=1
    convergeMacro=-1
    SALTO=0
    call BackwardForward(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2,
Ithevenin, Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro)
    do k=1, n2
      Vpu=abs(nodo(k)%V4 / abs(nodo(k)%VR4))
      DO p=1,3
        IF( ((Vpu(p).LT.0.975) .OR. (Vpu(p).GT. 1.05)) .AND. (Vpu(p).GT. 0.1)
) THEN
          SALTO=1
          EXIT
        END IF
      END DO
      IF(SALTO.EQ.1)THEN
        exit
      end if
    end do
    IF(SALTO.EQ.0)THEN
      if(auxpos.EQ.0) then
        mini(m)=L
      end if
      maxi(m)=L
      auxpos=auxpos+1
      PERDIDAS_X_ESCENARIO(m,auxpos) = loss
      EDOS_ANT(m,auxpos,n)=L
      IF (loss.LT.maxp) THEN
        maxp=loss
        amax=L
      END IF
    END IF
  END DO !L
  TRAYT(n,m)=AMAX

```



```

END DO !M
CALL BACKTRACKT(N)
END DO !N
!WRITE (101,*)
!WRITE (101,*) 'EL NUMERO DE ESTADOS ALMACENADOS EN CADA ESCENARIO DE CARGA ES: ',NESC
DEALLOCATE(AB,EDOS_ANT,PERDIDAS_X_ESCENARIO,mini,maxi)
DEALLOCATE (PREVIO,SPERDIDAS,TRAYECTORIA)
END SUBROUTINE DESPACHOT

```

B.21. Gauss.f90

```

=====
!
!DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
!
!JUNIO 2012
!
=====
!RAUL GUILLERMO JARA SIXTOS
!rjaras0400@ipn.mx
!
=====
!INSTITUTO POLITECNICO NACIONAL
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!
=====
SUBROUTINE GAUSS(orden, M1, VS,VR)
  IMPLICIT NONE
  INTEGER :: orden, i, j, k
  REAL*8 :: M1(orden,orden), VS(orden), VR(orden),MA(orden,orden+1)
  REAL*8 :: aux

  MA = 0.0
  DO i=1,orden
    MA(i,orden+1)= VS(i)
    DO j=1,orden
      MA(i,j)=M1(i,j)
    END DO
  END DO
  !DIAGONALIZA LA MATRIZ AUMENTADA
  DO i=1,orden
    aux=MA(i,i)
    DO j=i,orden+1
      MA(i,j)=MA(i,j)/aux
    END DO
  !HAZ CEROS ARRIBA Y ABAJO DEL PIVOTE
  DO j=1, orden
    IF (j .NE. i) THEN
      aux=MA(j,i)
      DO k=i,orden+1
        MA(j,k)=MA(j,k)-MA(i,k)*aux
      END DO
    END IF
  END DO
  !COPIA EL VECTOR AUMENTADO A VR
  DO i=1,orden
    VR(i)=MA(i,orden+1)
  END DO
END SUBROUTINE GAUSS

```



B.22. Salidatotal.f90

```

=====
!
! DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
!
! JUNIO 2012
!
=====
!
! RAUL GUILLERMO JARA SIXTOS
! rjaras0400@ipn.mx
!
=====
!
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!
=====

```

SUBROUTINE SALIDATOTAL(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2, Ithevenin, Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro, filename, header)

```

use rutinas_de_asignacion
use ConstantesG14
use ConstantesG2
use ConstantesG3
USE MSIMSL
USE estructuras
USE estructuras2
USE estructuras3
USE DATA_DESPACHO
use BackForw
use UtileriasComplejas2
use PrintSalidaDetallada

```

implicit none

```

character(20)          :: filename, archsal, ctemp
character(len=60)     :: header          !Encabezado del archivo de entrada

```

!Declaracion de variables I/O

```

integer, intent(in)::n2
type(lista), dimension(n2),intent(inout) :: nodo    !INOUT para meter todo y sacar voltaje

```

```

integer, intent(in)  :: maxIter
integer, intent(inout) :: iteraciones

```

```

integer, intent(inout) :: converge
real , intent(in)      :: tolerancia

```

!23feb2010

```

integer, intent(in)          :: m2
complex, dimension(3*m2)    , intent(inout):: Ithevenin
complex, dimension(3*m2, 3*m2), intent(in ):: Ythevenin
complex, dimension(3*m2)    , intent(inout):: Vthevenin

```

```

type(lista_mallas), dimension(m2), intent(in):: malla    !24feb2010
type(THV_est), intent(inout)          :: thv              !24feb2010--
!03mar2010

```

```

integer, intent(inout)::iteracionesMacro
integer, intent(inout)::convergeMacro

```

!Declaracion de Variables internas

```

integer ::k, t,caps    !Contador(es)

```

```

complex, dimension(3):: V4antes
complex, dimension(3):: ERROR

```




```

INTEGER :: M,auxpos,p,salto

ALLOCATE(AB(1,CAPACITORS))
WRITE(* ,*) 'Despacho obtenido por hora: '
WRITE(* ,*)
DO m = 1,escenarios ! RESUELVE FLUJOS DE POTENCIA PARA TODOS LOS ESCENARIOS
WRITE(* ,68) m
68 FORMAT ('HORA ',I3)
    auxpos=0

    do k=1,n2
        DO caps=1,3
            NODO(k)%S4YS(CAPS)=LOADC_OR(K,CAPS)*porcentaje_de_carga(m)/100.0
            NODO(k)%SC(CAPS)=LOADC_OR2(K,CAPS)*porcentaje_de_carga(m)/100.0
        END DO !FALTA AGREGAR CARGA DISTRIBUIDA
    end do
    CALL TABLA_DE_ESCENARIOS(TRAYC(m))
    WRITE(* ,70) TRAYC(m),AB
70 FORMAT (8X,"CAPACITORES: ",I4,2X,50I3)
    do k=1,CAPACITORS
        DO caps=1,3
            NODO(UB_CAP(K))%S4YZ(CAPS)=CAP_OR(K,CAPS)*AB(1,k)
            NODO(UB_CAP(K))%SBC(CAPS)=CAP_OR2(K,CAPS)*AB(1,k)
        END DO
    end do

    DO k=1,transformers
        DO p=1,3
            nodo(UB_TRANS(k))%Tap(p)=TRAYT(k,m)
        END DO
78 FORMAT (8X,'TAP ',I2,' = ',2X,F5.1)
        call arReg (nodo(UB_TRANS(k))%Tap, nodo(UB_TRANS(k))%ar)

        call dAB14( nodo(UB_TRANS(k))%lc_Reg, nodo(UB_TRANS(k))%ty_Reg,
nodo(UB_TRANS(k))%cx_Reg, nodo(UB_TRANS(k))%ar, nodo(UB_TRANS(k))%Z1, nodo(UB_TRANS(k))%Z4,
&
nodo(UB_TRANS(k))%A1, nodo(UB_TRANS(k))%B1, &
nodo(UB_TRANS(k))%A4, nodo(UB_TRANS(k))%B4
)
nodo(UB_TRANS(k))%d1,
nodo(UB_TRANS(k))%d4,

        call dAB2 (nodo(UB_TRANS(k))%Z2, nodo(UB_TRANS(k))%d2, nodo(UB_TRANS(k))%A2,
nodo(UB_TRANS(k))%B2)
        call dAB3 (nodo(UB_TRANS(k))%tybr,nodo(UB_TRANS(k))%nt, nodo(UB_TRANS(k))%Z3,
nodo(UB_TRANS(k))%d3, nodo(UB_TRANS(k))%A3, nodo(UB_TRANS(k))%B3)

        END DO
        iteraciones=0
        converge=1
        convergeMacro=-1
        SALTO=0
        call BackwardForward(nodo, n2, maxIter, iteraciones, converge, tolerancia, m2,
Ithevenin, Ythevenin, Vthevenin, malla, thv, iteracionesMacro, convergeMacro )

        archsal=TRIM(filename) // TRIM(ACHAR(64+m)) //' .res'
        call SalidaDetallada (archsal, n2,m2, nodo, iteraciones, iteracionesMacro,
converge, header)

    END DO !M
END SUBROUTINE SALIDATOTAL

```

B.23. Sort.f90

```

SUBROUTINE SORT (VECTOR,DIM,VECTOR2)

```

```

IMPLICIT NONE
INTEGER, INTENT(in) :: DIM
REAL, INTENT(inOUT),DIMENSION (DIM) :: VECTOR
INTEGER, INTENT(inOUT),DIMENSION (DIM) :: VECTOR2
INTEGER I, J, INDICE

```



```

REAL AUX

IF (DIM.GE.2) THEN
  DO J= DIM,2,-1
    INDICE = 1
    DO I =2,J
      IF (VECTOR(I).GT.VECTOR(INDICE)) THEN
        INDICE = I
      END IF
    ENDDO
    IF (INDICE.NE.J) THEN
      AUX = VECTOR(INDICE)
      VECTOR(INDICE) = VECTOR(J)
      VECTOR(J) = AUX
      AUX = VECTOR2(INDICE)
      VECTOR2(INDICE) = VECTOR2(J)
      VECTOR2(J) = AUX
    ENDIF
  END DO
END IF

END SUBROUTINE

```

B.24. Sort2.f90

```

=====
!
! DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
!
! JUNIO 2012
!
=====
!
! RAUL GUILLERMO JARA SIXTOS
! rjaras0400@ipn.mx
!
=====
!
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!
=====

```

```

SUBROUTINE SORT2 (VECTOR,DIM,VECTOR2,CAP,VECTOR3,VECTOR4,VECTOR5)

```

```

IMPLICIT NONE
INTEGER, INTENT(in) :: DIM,CAP
REAL, INTENT(inOUT),DIMENSION (DIM) :: VECTOR,VECTOR5
INTEGER, INTENT(inOUT),DIMENSION (DIM) :: VECTOR3,VECTOR4
INTEGER, INTENT(inOUT),DIMENSION (DIM,CAP) :: VECTOR2
INTEGER I, J, INDICE
REAL AUX

```

```

IF (DIM.GE.2) THEN
  DO J= DIM,2,-1
    INDICE = 1
    DO I =2,J
      IF (VECTOR(I).GT.VECTOR(INDICE)) THEN
        INDICE = I
      END IF
    ENDDO
    IF (INDICE.NE.J) THEN
      AUX = VECTOR(INDICE)
      VECTOR(INDICE) = VECTOR(J)
      VECTOR(J) = AUX
      AUX = VECTOR3(INDICE)
      VECTOR3(INDICE) = VECTOR3(J)
      VECTOR3(J) = AUX
      AUX = VECTOR4(INDICE)

```



```

        VECTOR4(INDICE) = VECTOR4(J)
        VECTOR4(J) = AUX
        AUX = VECTOR5(INDICE)
        VECTOR5(INDICE) = VECTOR5(J)
        VECTOR5(J) = AUX
        DO I=1,CAP
            AUX = VECTOR2(INDICE,I)
            VECTOR2(INDICE,I) = VECTOR2(J,I)
            VECTOR2(J,I) = AUX
        END DO
    ENDIF
END DO
END IF

END SUBROUTINE SORT2

```

B.25. Sort2may.f90

```

=====
!
! DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
!
! JUNIO 2012
!
=====
!
! RAUL GUILLERMO JARA SIXTOS
! rjaras0400@ipn.mx
!
=====
!
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
!
=====
SUBROUTINE SORT2MAY (VECTOR,DIM,VECTOR2,CAP,VECTOR3,VECTOR4,VECTOR5)

IMPLICIT NONE
INTEGER, INTENT(in) :: DIM,CAP
REAL, INTENT(inOUT),DIMENSION (DIM) :: VECTOR,VECTOR5
INTEGER, INTENT(inOUT),DIMENSION (DIM) :: VECTOR3,VECTOR4
INTEGER, INTENT(inOUT),DIMENSION (DIM,CAP) :: VECTOR2
INTEGER I, J, INDICE
REAL AUX

IF (DIM.GE.2) THEN
    DO J= DIM,2,-1
        INDICE = 1
        DO I =2,J
            IF (VECTOR(I).LT.VECTOR(INDICE)) THEN
                INDICE = I
            END IF
        ENDDO
        IF (INDICE.NE.J) THEN
            AUX = VECTOR(INDICE)
            VECTOR(INDICE) = VECTOR(J)
            VECTOR(J) = AUX
            AUX = VECTOR3(INDICE)
            VECTOR3(INDICE) = VECTOR3(J)
            VECTOR3(J) = AUX
            AUX = VECTOR4(INDICE)
            VECTOR4(INDICE) = VECTOR4(J)
            VECTOR4(J) = AUX
            AUX = VECTOR5(INDICE)
            VECTOR5(INDICE) = VECTOR5(J)
            VECTOR5(J) = AUX
            DO I=1,CAP
                AUX = VECTOR2(INDICE,I)
                VECTOR2(INDICE,I) = VECTOR2(J,I)

```



```

                VECTOR2(J,I) = AUX
            END DO
        ENDIF
    END DO
END IF

```

END SUBROUTINE SORT2MAY

B.26. Sortc.f90

```

=====
=====
! DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION
! JUNIO 2012
=====
! RAUL GUILLERMO JARA SIXTOS
! rjaras0400@ipn.mx
!
! INSTITUTO POLITECNICO NACIONAL
! ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
! SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION
=====
=====

```

SUBROUTINE SORTC (VECTOR,DIM,VECTOR2,CAP,VECTOR3)

```

IMPLICIT NONE
INTEGER, INTENT(in) :: DIM,CAP
REAL, INTENT(inOUT),DIMENSION (DIM) :: VECTOR
INTEGER, INTENT(inOUT),DIMENSION (DIM) :: VECTOR3
INTEGER, INTENT(inOUT),DIMENSION (DIM,CAP) :: VECTOR2
INTEGER I, J, INDICE
REAL AUX

```

```

IF (DIM.GE.2) THEN
    DO J= DIM,2,-1
        INDICE = 1
        DO I =2,J
            IF (VECTOR(I).GT.VECTOR(INDICE)) THEN
                INDICE = I
            END IF
        END DO
        IF (INDICE.NE.J) THEN
            AUX = VECTOR(INDICE)
            VECTOR(INDICE) = VECTOR(J)
            VECTOR(J) = AUX
            AUX = VECTOR3(INDICE)
            VECTOR3(INDICE) = VECTOR3(J)
            VECTOR3(J) = AUX
            DO I=1,CAP
                AUX = VECTOR2(INDICE,I)
                VECTOR2(INDICE,I) = VECTOR2(J,I)
                VECTOR2(J,I) = AUX
            END DO
        END IF
    END DO
END IF

```

END SUBROUTINE SORTC

B.27. Tabla_de_escenarios.f90

```

=====
=====
! DESPACHO DE POTENCIA RECTIVA EN REDES RADIALES DE DISTRIBUCION

```



```
!  
!JUNIO 2012  
!=====
```

```
!  
!RAUL GUILLERMO JARA SIXTOS  
!rjaras0400@ipn.mx  
!  
!=====
```

```
!  
!INSTITUTO POLITECNICO NACIONAL  
!ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA  
!SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION  
!=====
```

```
!  
!=====
```

```
SUBROUTINE TABLA_DE_ESCENARIOS(NUM)  
!ESTA SUBROUTINA OBTIENE EL ESCENARIO EQUIVALENTE AL NUMERO DE ESCENARIO  
!NUM ES EL NUMERO DE ESCENARIO EN BASE 10  
!LA RUTINA ALMACENA EN EL VECTOR AB EL ESTADO DE CADA CAPACITOR  
!ESTA RUTINA SE BASA EN UN SISTEMA DE BASE BINARIA PARA REPRESENTAR EL ESTADO DE CADA  
CAPACITOR  
USE DATA_DESPACHO  
IMPLICIT NONE  
  
INTEGER :: NUM,NUM2,TEMP,I  
INTEGER :: base  
INTEGER,ALLOCATABLE :: ESCENARIO(:,:) !ALLOCATE A NCAP  
ALLOCATE (ESCENARIO(1,capacitors))  
BASE=2  
NUM2=NUM  
ESCENARIO=0  
  DO i=CAPACITORS,1,-1  
    TEMP=MOD(NUM2,BASE)  
    NUM2=INT(NUM2/BASE)  
    ESCENARIO(1,i)=TEMP  
    IF(NUM2.EQ.0) THEN  
      EXIT  
    END IF  
  END DO ! i  
  
AB=ESCENARIO  
DEALLOCATE (ESCENARIO)  
END SUBROUTINE TABLA_DE_ESCENARIOS
```



APÉNDICE C

ARCHIVO DE ENTRADA

C.1. Formato de los datos de entrada.

Los datos de entrada para el programa son almacenados en un archivo de texto con nombre arbitrario, normalmente alusivo a la red por analizar, y con extensión .txt. Consta de un encabezado que puede contener información breve relacionada con la red por analizar, el número de nodos de la red, el voltaje del nodo raíz, y por cada nodo y por cada enlace que forma una malla, un bloque de datos que contiene toda la información topológica y eléctrica. Cabe resaltar que para hacer un archivo de datos de entrada, es necesario realizar una etapa previa de procesamiento de datos topológicos de la red para etiquetar cada nodo de la estructura radial con un número y ubicarlo en un nivel [25].

A continuación se describe cada uno de los rubros del archivo de entrada, donde cada dato está en formato libre.

Encabezado

Tipo carácter, longitud máxima: 60 (Sin espacios)

Número de nodos

Tipo entero, número máximo 99999

Número de cambiadores de derivación

Tipo entero, número máximo 99999

Número de bancos de capacitores

Tipo entero, número máximo 99999

Número máximo de operaciones de conexión/desconexión de los bancos de capacitores

Tipo entero, número máximo 99999

Número máximo de cambiadores de derivación

Tipo entero, número máximo 99999



Número de escenarios

Tipo entero, número máximo 99999

Porcentaje de carga para cada escenario

Tipo real

Voltaje de nodo raíz

Vector real de orden 6, que contiene el voltaje del nodo raíz de la siguiente forma.

Van	arg(Van)	Vbn	arg(Vbn)	Vcn	arg(Vcn)
-----	----------	-----	----------	-----	----------

Donde las magnitudes de los voltajes de fase están en kV y los argumentos en grados sexagesimales.

Boque de información por nodo

Cada nodo de la red contiene la siguiente información:

nomb											
n	pre(n)	niv(n)	ty	nt							
$r_{a,a}$	$x_{a,a}$	$r_{a,b}$	$x_{a,b}$	$r_{a,c}$	$x_{a,c}$	$r_{b,b}$	$x_{b,b}$	$r_{b,c}$	$x_{b,c}$	$r_{c,c}$	$x_{c,c}$
$b_{a,a}$	$b_{a,b}$	$b_{a,c}$	$b_{b,b}$	$b_{b,c}$	$b_{c,c}$						
cxLc	mdLc	PC ₁	QC ₁	PC ₂	QC ₂	PC ₃	QC ₃				
cxLd	mdLd	Pd ₁	Qd ₁	Pd ₂	Qd ₂	Pd ₃	Qd ₃				
cxBc	QBC ₁	QBC ₂	QBC ₃								
lcReg	tyReg	cxReg	tap ₁	tap ₂	tap ₃						

Donde:

nomb: (carácter) Nombre del bus. Longitud máxima: 10

n: (entero) Número de nodo asignado de acuerdo a jerarquización indicada en [25]

pre(n): (entero) Número del nodo que precede al nodo n.

nivel(n): (entero) Nivel o capa en la que se encuentra el nodo n, como se indica en [25]

ty: (entero) Tipo de rama que converge en el nodo n, asignada de acuerdo con siguiente:

-1 - Interruptor

0 -Nodo raíz (rama ficticia)

1 - Línea aérea o subterránea

2 - Transformador Reductor, Delta-Estrella aterrizada

3 - Transformador Reductor, Estrella aterrizada -Estrella aterrizada



nt: (real) Relación de espiras de primario a secundario para transformadores [1]. Para líneas, interruptores y nodo raíz, este valor se fija en 1.0.

$r_{aa}, X_{aa}, r_{ab}, X_{ab}, r_{ac}, X_{ac}, r_{bb}, X_{bb}, r_{bc}, X_{bc}, r_{cc}, X_{cc}$

Tipo real. Resistencia y Reactancia Inductiva en ohms del elemento serie que converge en el nodo n (diagonal principal y la triangular superior de la matriz de impedancia primitiva).

$b_{aa}, b_{ab}, b_{ac}, b_{bb}, b_{bc}, b_{cc}$

Tipo real. Susceptancia en micro siemens debido a la capacitancia de la línea. (Diagonal principal y la triangular superior de la matriz de admitancia primitiva). Para transformadores todos los valores son 0.0

cxLc

Tipo entero. Conexión de la carga concentrada en el nodo n. (1: Estrella, 2: Delta). El valor por omisión es 1.

mdLc

Tipo Entero. Modelo de la carga concentrada en el nodo n (1: Potencia constante, 2: Corriente constante, 3: Impedancia constante). El valor por omisión es 1.

$P_{c1}, Q_{c1}, P_{c2}, Q_{c2}, P_{c3}, Q_{c3}$

Tipo Entero. Potencia Real y reactiva en kW y kVAR, respectivamente de la carga concentrada en el nodo n. Si la carga estuviera conectada en estrella, los subíndices 1, 2, 3, corresponden a potencia de fase. Si estuviese conectada en delta, a potencia entre líneas. El valor por omisión es 0.0 en los seis casos.

cxLd

Similar a cxLc, pero para la carga distribuida del alimentador que converge en el nodo n. El valor por omisión es 1.

mdLd

Similar a mdLc, pero para la carga distribuida del alimentador que converge en el nodo n. El valor por omisión es 1.



$Pd_1, Qd_1, Pd_2, Qd_2, Pd_3, Qd_3$

Similar a $Pc_1, Qc_1, Pc_2, Qc_2, Pc_3, Qc_3$, pero para la carga distribuida del alimentador que converge en el nodo n. El valor por omisión es 0.0 en los seis casos.

cxBc

Tipo entero. Conexión del banco de capacitores conectado al nodo n. (1: Estrella, 2: Delta)

QBc_1, QB_2, QB_3

Potencia reactiva en kVAr del banco capacitivo. (Indicar con signo positivo). El valor por omisión es 0.0 en los tres casos.

IcReg

Tipo entero. Ubicación del Regulador en el alimentador que converge en el nodo n (1: Principio, 2: Final). El valor por omisión es 1.

TyReg

Tipo Entero. Tipo de Regulador (1: tipo A, 2: tipo B). El valor por omisión es 1.

cxReg

Tipo entero. Conexión del Regulador en el alimentador que converge en el nodo n (1: Estrella, Delta). El valor por omisión es 1.

tap₁, tap₂, tap₃

Tipo real. Posición de cada uno de los taps. El valor por omisión es 0.0 en los tres casos. Estos valores pueden ir precedidos por un signo negativo dependiendo de la operación específica del regulador, es decir, en elevación o reducción, y también del tipo de regulador. Para mayor detalle de modelado véase la referencia [1].

Número de mallas

Tipo entero, número máximo 99999

Boque de información por malla

Cada malla de la red contiene la siguiente información:



m												
from(m)												
To(m)												
$r'_{a,a}$	$x'_{a,a}$	$r'_{a,b}$	$x'_{a,b}$	$r'_{a,c}$	$x'_{a,c}$	$r'_{b,b}$	$x'_{b,b}$	$r'_{b,c}$	$x'_{b,c}$	$r'_{c,c}$	$x'_{c,c}$	
$b'_{a,a}$	$b'_{a,b}$	$b'_{a,c}$	$b'_{b,b}$	$b'_{b,c}$	$b'_{c,c}$							

Donde:

m: (entero) Número de malla (asignado en forma arbitraria). Cero en caso de no haber mallas

from(m): (entero) Nodo de la red radial del cual parte el enlace que forma la malla.

to(m): (entero) Nodo de la red radial al cual llega el enlace que forma la malla

nota: siempre $from(m) < to(m)$, según la numeración de la red radial ya jerarquizada

$r'_{aa}, x'_{aa}, r'_{ab}, x'_{ab}, r'_{ac}, x'_{ac}, r'_{bb}, x'_{bb}, r'_{bc}, x'_{bc}, r'_{cc}, x'_{cc}$

Tipo real. Resistencia y Reactancia Inductiva en ohms del enlace que forma la malla (diagonal principal y la triangular superior de la matriz de impedancia primitiva).

$b'_{aa}, b'_{ab}, b'_{ac}, b'_{bb}, b'_{bc}, b'_{cc}$

Tipo real. Susceptancia en micro siemens debido a la capacitancia del enlace que forma la malla. (Diagonal principal y la triangular superior de la matriz de admitancia primitiva).

C.2. Ejemplo de archivo de entrada para la red de Salama.

```

RED_SALAMA
35
1
8
1
5
24

40
32
28
25
20
23
30
38
41
48
50
48
43
45
51
    
```



60											
66											
78											
100											
84											
79											
70											
58											
45											
66.39528	0.00	66.39528	-120.00	66.39528	120.00						
SUB											
1	1	1	1	1	1.00000						
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
NO											
2	1	2	2	18.1078							
0.242000	1.936000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.242000	1.936000		
0.000000	0.000000	0.242000	1.936000								
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
N1											
3	2	3	1	1.00000							
0.1170	0.0480	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1170	0.0480	0.0000	0.0000
0.1170	0.0480										
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1	1	76.66	47.50	76.66	47.50	76.66	47.50	76.66	47.50	76.66	47.50
1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
N2											
4	3	4	1	1.00000							
0.1073	0.0440	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1073	0.0440	0.0000	0.0000
0.1073	0.0440										
	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
N3											
5	4	5	1	1.00000							
0.1645	0.0457	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1645	0.0457	0.0000	0.0000
0.1645	0.0457										
	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1	1	76.66	47.50	76.66	47.50	76.66	47.50	76.66	47.50	76.66	47.50
1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
N12											
6	4	5	1	1.00000							
0.1572	0.0270	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1572	0.0270	0.0000	0.0000
0.1572	0.0270										
	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1	1	24.00	15.00	24.00	15.00	24.00	15.00	24.00	15.00	24.00	15.00
1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
N4											
7	5	6	1	1.00000							
0.1495	0.0415	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1495	0.0415	0.0000	0.0000
0.1495	0.0415										
	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1	1	76.66	47.50	76.66	47.50	76.66	47.50	76.66	47.50	76.66	47.50
1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	100.00	100.00	100.00								
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
N13											
8	6	6	1	1.00000							



Coordinación de cambiadores de derivación de transformadores de subestación y capacitores para minimizar pérdidas en redes de distribución



0.2096	0.0360	0.0000	0.0000	0.0000	0.0000	0.2096	0.0360	0.0000	0.0000
0.2096	0.0360								
	1	1	24.00	15.00	24.00	15.00	24.00	15.00	
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	
1	1	1	0.00	0.00	0.00				
N5									
	9	7	7	1	1.00000				
0.1495	0.0415	0.0000	0.0000	0.0000	0.0000	0.1495	0.0415	0.0000	0.0000
0.1495	0.0415								
	1	1	0.00	0.00	0.00	0.00	0.00	0.00	
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	
1	1	1	0.00	0.00	0.00				
N14									
	10	8	7	1	1.00000				
0.1048	0.0180	0.0000	0.0000	0.0000	0.0000	0.1048	0.0180	0.0000	0.0000
0.1048	0.0180								
	1	1	24.00	15.00	24.00	15.00	24.00	15.00	
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	
1	1	1	0.00	0.00	0.00				
N6									
	11	9	8	1	1.00000				
0.3144	0.0540	0.0000	0.0000	0.0000	0.0000	0.3144	0.0540	0.0000	0.0000
0.3144	0.0540								
	1	1	0.00	0.00	0.00	0.00	0.00	0.00	
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	
1	1	1	0.00	0.00	0.00				
N16									
	12	9	8	1	1.00000				
0.1794	0.0498	0.0000	0.0000	0.0000	0.0000	0.1794	0.0498	0.0000	0.0000
0.1794	0.0498								
	1	1	76.66	47.50	76.66	47.50	76.66	47.50	
1	100.00	1	0.00	0.00	0.00	0.00	0.00	0.00	
1	1	1	0.00	0.00	0.00				
N15									
	13	10	8	1	1.00000				
0.0524	0.0090	0.0000	0.0000	0.0000	0.0000	0.0524	0.0090	0.0000	0.0000
0.0524	0.0090								
	1	1	4.50	2.50	4.50	2.50	4.50	2.50	
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	
1	1	1	0.00	0.00	0.00				
N7									
	14	11	9	1	1.00000				
0.2096	0.0360	0.0000	0.0000	0.0000	0.0000	0.2096	0.0360	0.0000	0.0000
0.2096	0.0360								
	1	1	76.66	47.50	76.66	47.50	76.66	47.50	
1	100.00	1	0.00	0.00	0.00	0.00	0.00	0.00	
1	1	1	0.00	0.00	0.00				
N27									
	15	11	9	1	1.00000				
0.0000	0.1572	0.0270	0.0000	0.0000	0.0000	0.0000	0.1572	0.0270	0.0000
	0.1572	0.0270							
	1	1	25.00	16.00	25.00	16.00	25.00	16.00	
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	
1	1	1	0.00	0.00	0.00				
N17									
	16	12	9	1	1.00000				
0.0000	0.1645	0.0457	0.0000	0.0000	0.0000	0.0000	0.1645	0.0457	0.0000
	0.1645	0.0457							
	1	1	76.66	47.50	76.66	47.50	76.66	47.50	



Coordinación de cambiadores de derivación de transformadores de subestación y capacitores para minimizar pérdidas en redes de distribución



1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
N8										
0.3144	17	14	10	1	1.00000					
0.3144	0.0540	0.0000	0.0000	0.0000	0.0000	0.0000	0.3144	0.0540	0.0000	0.0000
	1	1	76.66	47.50	76.66	47.50	76.66	47.50		0.0000
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
N28										
0.1572	18	15	10	1	1.00000					
0.1572	0.0270	0.0000	0.0000	0.0000	0.0000	0.0000	0.1572	0.0270	0.0000	0.0000
	1	1	25.00	16.00	25.00	16.00	25.00	16.00		0.0000
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
N18										
0.0000	19	16	10	1	1.00000					
0.0000	0.2079	0.0473	0.0000	0.0000	0.0000	0.0000	0.0000	0.2079	0.0473	0.0000
	1	1	76.66	47.50	76.66	47.50	76.66	47.50		0.0000
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	100.00	100.00	100.00	100.00	100.00	100.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
N9										
0.0000	20	17	11	1	1.00000					
0.0000	0.2096	0.0360	0.0000	0.0000	0.0000	0.0000	0.0000	0.2096	0.0360	0.0000
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		0.0000
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	100.00	100.00	100.00	100.00	100.00	100.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
N29										
0.1572	21	18	11	1	1.00000					
0.1572	0.0270	0.0000	0.0000	0.0000	0.0000	0.0000	0.1572	0.0270	0.0000	0.0000
	1	1	25.00	16.00	25.00	16.00	25.00	16.00		0.0000
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
N19										
0.0000	22	19	11	1	1.00000					
0.0000	0.1890	0.0430	0.0000	0.0000	0.0000	0.0000	0.0000	0.1890	0.0430	0.0000
	1	1	76.66	47.50	76.66	47.50	76.66	47.50		0.0000
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
N10										
0.0000	23	20	12	1	1.00000					
0.0000	0.1310	0.0225	0.0000	0.0000	0.0000	0.0000	0.0000	0.1310	0.0225	0.0000
	1	1	76.66	47.50	76.66	47.50	76.66	47.50		0.0000
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
N30										
0.0000	24	20	12	1	1.00000					
0.0000	0.1572	0.0270	0.0000	0.0000	0.0000	0.0000	0.0000	0.1572	0.0270	0.0000
	1	1	19.00	11.50	19.00	11.50	19.00	11.50		0.0000
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
N20										



Coordinación de cambiadores de derivación de transformadores de subestación y capacitores para minimizar pérdidas en redes de distribución



25	22	12	1	1.00000						
0.0000	0.1890	0.0430	0.0000	0.0000	0.0000	0.0000	0.0000	0.1890	0.0430	0.0000
	0.1890	0.0430								
	1	1	1	76.66	47.50	76.66	47.50	76.66	47.50	0.0000
	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
1	100.00	100.00	100.00							
1	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
N11										
26	23	13	1	1.00000						
0.0000	0.1048	0.0180	0.0000	0.0000	0.0000	0.0000	0.0000	0.1048	0.0180	0.0000
	0.1048	0.0180								
	1	1	1	45.66	28.00	45.66	28.00	45.66	28.00	0.0000
	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
1	0.00	0.00	0.00							
1	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
N31										
27	24	13	1	1.00000						
0.0000	0.2096	0.0360	0.0000	0.0000	0.0000	0.0000	0.0000	0.2096	0.0360	0.0000
	0.2096	0.0360								
	1	1	1	19.00	11.50	19.00	11.50	19.00	11.50	0.0000
	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
1	0.00	0.00	0.00							
1	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
N21										
28	25	13	1	1.00000						
0.0000	0.2620	0.0450	0.0000	0.0000	0.0000	0.0000	0.0000	0.2620	0.0450	0.0000
	0.2620	0.0450								
	1	1	1	76.66	47.50	76.66	47.50	76.66	47.50	0.0000
	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
1	0.00	0.00	0.00							
1	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
N32										
29	27	14	1	1.00000						
0.0000	0.1572	0.0270	0.0000	0.0000	0.0000	0.0000	0.0000	0.1572	0.0270	0.0000
	0.1572	0.0270								
	1	1	1	19.00	11.50	19.00	11.50	19.00	11.50	0.0000
	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
1	0.00	0.00	0.00							
1	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
N22										
30	28	14	1	1.00000						
0.0000	0.2620	0.0450	0.0000	0.0000	0.0000	0.0000	0.0000	0.2620	0.0450	0.0000
	0.2620	0.0450								
	1	1	1	76.66	47.50	76.66	47.50	76.66	47.50	0.0000
	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
1	100.00	100.00	100.00							
1	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
N33										
31	29	15	1	1.00000						
0.0000	0.1048	0.0180	0.0000	0.0000	0.0000	0.0000	0.0000	0.1048	0.0180	0.0000
	0.1048	0.0180								
	1	1	1	19.00	11.50	19.00	11.50	19.00	11.50	0.0000
	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
1	0.00	0.00	0.00							
1	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
N23										
32	30	15	1	1.00000						
0.0000	0.3144	0.0540	0.0000	0.0000	0.0000	0.0000	0.0000	0.3144	0.0540	0.0000
	0.3144	0.0540								
	1	1	1	76.66	47.50	76.66	47.50	76.66	47.50	0.0000
	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
1	0.00	0.00	0.00							
1	1	1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.0000
N24										
33	32	16	1	1.00000						
0.0000	0.2096	0.0360	0.0000	0.0000	0.0000	0.0000	0.0000	0.2096	0.0360	0.0000
	0.2096	0.0360								
	1	1	1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000



Coordinación de cambiadores de derivación de transformadores de subestación y capacitores para minimizar pérdidas en redes de distribución



	1	1	76.66	47.50	76.66	47.50	76.66	47.50		
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	100.00	100.00	100.00							
1	1	1	0.00	0.00	0.00					
N25										
	34	33	17	1	1.00000					
0.0000	0.1310	0.0225	0.0000	0.0000	0.0000	0.0000	0.0000	0.1310	0.0225	0.0000
	0.1310	0.0225								
	1	1	76.66	47.50	76.66	47.50	76.66	47.50		
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	0.00	0.00	0.00							
1	1	1	0.00	0.00	0.00					
N26										
	35	34	18	1	1.00000					
0.0000	0.1048	0.0180	0.0000	0.0000	0.0000	0.0000	0.0000	0.1048	0.0180	0.0000
	0.1048	0.0180								
	1	1	45.66	28.33	45.66	28.33	45.66	28.33		
	1	1	0.00	0.00	0.00	0.00	0.00	0.00		
1	0.00	0.00	0.00							
1	1	1	0.00	0.00	0.00					
0										



APÉNDICE D

ARCHIVOS DE SALIDA

D.1. Red de Salama con Kc=1 (Sin controlar el cambiador de derivaciones)

LAS PERDIDAS CON LA TRAYECTORIA OPTIMA ES : 1378.36023

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	30.04534	1	1	0	0	1	0	0	1	0
2H	19.64115	1	1	0	0	1	0	0	1	0
3H	15.80366	1	1	0	0	1	0	0	1	0
4H	13.50613	1	1	0	0	1	0	0	1	0
5H	10.75897	1	1	0	0	1	0	0	1	0
6H	12.24641	1	1	0	0	1	0	0	1	0
7H	17.61080	1	1	0	0	1	0	0	1	0
8H	27.09663	1	1	0	0	1	0	0	1	0
9H	31.60803	1	1	0	0	1	0	0	1	0
10H	43.26276	1	1	0	0	1	1	0	1	0
11H	47.00220	1	1	0	0	1	1	0	1	0
12H	43.26276	1	1	0	0	1	1	0	1	0
13H	34.97997	1	1	0	0	1	1	0	1	0
14H	38.11217	1	1	0	0	1	1	0	1	0
15H	48.96475	1	1	0	0	1	1	0	1	0
16H	68.51144	1	1	1	0	1	1	0	1	0
17H	83.11649	1	1	1	0	1	1	1	1	0
18H	117.41838	1	1	1	1	1	1	1	1	0
19H	205.23492	1	1	1	1	1	1	1	1	0
20H	137.69229	1	1	1	1	1	1	1	1	0
21H	120.61514	1	1	1	1	1	1	1	1	0
22H	94.38783	1	1	1	1	1	1	1	1	0
23H	67.94392	1	1	1	1	1	1	1	1	0
24H	49.53799	1	1	1	1	1	1	1	1	0

LA TRAYECTORIA OPTIMA NORMAL ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	30.04534	1	1	0	0	1	0	0	1	0
2H	19.64115	1	1	0	0	1	0	0	1	0
3H	15.80366	1	1	0	0	1	0	0	1	0
4H	13.50613	1	1	0	0	1	0	0	1	0
5H	10.75897	1	1	0	0	1	0	0	1	0
6H	12.24641	1	1	0	0	1	0	0	1	0
7H	17.61080	1	1	0	0	1	0	0	1	0
8H	27.09663	1	1	0	0	1	0	0	1	0
9H	31.60803	1	1	0	0	1	0	0	1	0
10H	43.26276	1	1	0	0	1	1	0	1	0
11H	47.00220	1	1	0	0	1	1	0	1	0
12H	43.26276	1	1	0	0	1	1	0	1	0
13H	34.97997	1	1	0	0	1	1	0	1	0
14H	38.11217	1	1	0	0	1	1	0	1	0
15H	48.96475	1	1	0	0	1	1	0	1	0



16H	68.51144	1	1	1	0	1	1	0	1	0
17H	83.11649	1	1	1	0	1	1	1	1	0
18H	117.41838	1	1	1	1	1	1	1	1	0
19H	205.23492	1	1	1	1	1	1	1	1	0
20H	137.69229	1	1	1	1	1	1	1	1	0
21H	120.61514	1	1	1	1	1	1	1	1	0
22H	94.38783	1	1	1	1	1	1	1	1	0
23H	67.94392	1	1	1	1	1	1	1	1	0
24H	49.53799	1	1	1	1	1	1	1	1	0

CONVERGE EN 1 ITERACION

Program has used 9.093750 seconds of CPU time.

D.2. Red de Salama con Kc=2 (Sin controlar el cambiador de derivaciones)

LAS PERDIDAS TOTALES CON LA TRAYECTORIA OPTIMA SON : 1415.79260

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	29.88384	1	0	0	1	1	0	0	1	0
2H	19.24579	1	0	1	2	2	0	0	1	0
3H	14.76325	1	0	1	2	2	0	0	1	0
4H	11.85837	2	0	1	2	2	0	0	1	0
5H	7.76997	2	0	1	2	2	0	0	1	0
6H	10.05919	2	0	1	2	2	0	0	1	0
7H	16.90375	2	1	1	2	2	0	0	1	0
8H	27.38506	2	1	1	2	2	1	0	1	0
9H	31.69353	2	1	1	2	2	1	0	1	0
10H	43.84109	2	1	1	2	2	1	0	1	0
11H	47.86318	2	1	1	2	2	1	0	1	0
12H	43.84109	2	1	1	2	2	1	0	1	0
13H	34.86214	2	1	1	2	2	1	0	1	0
14H	38.27102	2	1	1	2	2	1	0	1	0
15H	49.96797	2	1	1	2	2	1	0	1	0
16H	69.79525	2	1	1	2	2	1	1	1	0
17H	85.86977	2	1	1	2	2	1	1	1	0
18H	125.51465	2	1	1	2	2	1	1	1	0
19H	226.63547	2	1	1	2	2	1	1	1	0
20H	149.28958	2	1	1	2	2	1	1	1	0
21H	129.28873	2	1	1	2	2	1	1	1	0
22H	97.94999	2	1	1	2	2	1	1	1	0
23H	64.96902	2	1	1	2	2	1	1	1	0
24H	38.27102	2	1	1	2	2	1	2	1	0

LA TRAYECTORIA OPTIMA EN ESTADOS NORMALES ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	29.88384	1	0	0	1	1	0	0	1	0
2H	19.24579	1	0	1	0	0	0	0	1	0
3H	14.76325	1	0	1	0	0	0	0	1	0
4H	11.85837	0	0	1	0	0	0	0	1	0
5H	7.76997	0	0	1	0	0	0	0	1	0
6H	10.05919	0	0	1	0	0	0	0	1	0
7H	16.90375	0	1	1	0	0	0	0	1	0
8H	27.38506	0	1	1	0	0	1	0	1	0
9H	31.69353	0	1	1	0	0	1	0	1	0
10H	43.84109	0	1	1	0	0	1	0	1	0
11H	47.86318	0	1	1	0	0	1	0	1	0
12H	43.84109	0	1	1	0	0	1	0	1	0



13H	34.86214	0	1	1	0	0	1	0	1	0
14H	38.27102	0	1	1	0	0	1	0	1	0
15H	49.96797	0	1	1	0	0	1	0	1	0
16H	69.79525	0	1	1	0	0	1	1	1	0
17H	85.86977	0	1	1	0	0	1	1	1	0
18H	125.51465	0	1	1	0	0	1	1	1	0
19H	226.63547	0	1	1	0	0	1	1	1	0
20H	149.28958	0	1	1	0	0	1	1	1	0
21H	129.28873	0	1	1	0	0	1	1	1	0
22H	97.94999	0	1	1	0	0	1	1	1	0
23H	64.96902	0	1	1	0	0	1	1	1	0
24H	38.27102	0	1	1	0	0	1	0	1	0

CONVERGE EN 1 ITERACION

Program has used 20.04688 seconds of CPU time.

D.3. Red de Salama con Kc=3 (Sin controlar el cambiador de derivaciones)

LAS PERDIDAS CON LA TRAYECTORIA OPTIMA ES : 1353.07813

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	29.98858	1	0	0	1	1	0	1	0	0
2H	19.24807	1	0	1	2	2	0	1	0	0
3H	14.71505	1	0	1	2	2	0	1	0	0
4H	11.77399	2	0	1	2	2	0	1	0	0
5H	7.62283	2	0	1	2	2	0	1	0	0
6H	9.94970	2	0	1	2	2	0	1	0	0
7H	16.86889	3	0	1	2	2	0	1	0	0
8H	27.07871	3	0	2	2	3	1	2	1	0
9H	31.45054	3	0	2	2	3	1	2	1	0
10H	43.26281	3	1	2	2	3	1	2	1	0
11H	47.00228	3	1	2	2	3	1	2	1	0
12H	43.26281	3	1	2	2	3	1	2	1	0
13H	34.98000	3	1	2	2	3	1	2	1	0
14H	38.11221	3	1	2	2	3	1	2	1	0
15H	48.96483	3	1	2	2	3	1	2	1	0
16H	68.51144	3	1	3	2	3	1	2	1	0
17H	83.11649	3	1	3	2	3	1	3	1	0
18H	117.41838	3	1	3	3	3	1	3	1	0
19H	205.23492	3	1	3	3	3	1	3	1	0
20H	137.69229	3	1	3	3	3	1	3	1	0
21H	120.61514	3	1	3	3	3	1	3	1	0
22H	93.85809	3	1	3	3	3	2	3	1	0
23H	63.77288	3	2	3	3	3	2	3	1	0
24H	38.57732	3	2	3	3	3	2	3	2	0

LA TRAYECTORIA OPTIMA NORMAL ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	29.98858	1	0	0	1	1	0	1	0	0
2H	19.24807	1	0	1	0	0	0	1	0	0
3H	14.71505	1	0	1	0	0	0	1	0	0
4H	11.77399	0	0	1	0	0	0	1	0	0
5H	7.62283	0	0	1	0	0	0	1	0	0
6H	9.94970	0	0	1	0	0	0	1	0	0
7H	16.86889	1	0	1	0	0	0	1	0	0
8H	27.07871	1	0	0	0	1	1	0	1	0
9H	31.45054	1	0	0	0	1	1	0	1	0



10H	43.26281	1	1	0	0	1	1	0	1	0
11H	47.00228	1	1	0	0	1	1	0	1	0
12H	43.26281	1	1	0	0	1	1	0	1	0
13H	34.98000	1	1	0	0	1	1	0	1	0
14H	38.11221	1	1	0	0	1	1	0	1	0
15H	48.96483	1	1	0	0	1	1	0	1	0
16H	68.51144	1	1	1	0	1	1	0	1	0
17H	83.11649	1	1	1	0	1	1	1	1	0
18H	117.41838	1	1	1	1	1	1	1	1	0
19H	205.23492	1	1	1	1	1	1	1	1	0
20H	137.69229	1	1	1	1	1	1	1	1	0
21H	120.61514	1	1	1	1	1	1	1	1	0
22H	93.85809	1	1	1	1	1	0	1	1	0
23H	63.77288	1	0	1	1	1	0	1	1	0
24H	38.57732	1	0	1	1	1	0	1	0	0

CONVERGE EN 1 ITERACION

Program has used 21.92188 seconds of CPU time.

D.4. Red de Salama con Kc=4 (Sin controlar el cambiador de derivaciones)

LAS PERDIDAS CON LA TRAYECTORIA OPTIMA ES : 1352.10400

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	29.88384	1	0	0	1	1	0	0	1	0
2H	19.24579	1	0	1	2	2	0	0	1	0
3H	14.71505	1	0	1	2	2	0	1	2	0
4H	11.77399	2	0	1	2	2	0	1	2	0
5H	7.62283	2	0	1	2	2	0	1	2	0
6H	9.94970	2	0	1	2	2	0	1	2	0
7H	16.86889	3	0	1	2	2	0	1	2	0
8H	27.07871	3	0	2	2	3	1	2	3	0
9H	31.45054	3	0	2	2	3	1	2	3	0
10H	43.26281	3	1	2	2	3	1	2	3	0
11H	47.00228	3	1	2	2	3	1	2	3	0
12H	43.26281	3	1	2	2	3	1	2	3	0
13H	34.66145	3	2	2	2	3	1	2	3	0
14H	38.11266	3	2	2	2	3	1	2	3	0
15H	49.06571	3	2	2	3	3	1	2	3	0
16H	68.33212	3	2	3	3	3	2	3	3	0
17H	83.05814	3	3	3	3	3	2	3	3	0
18H	117.41838	3	3	3	3	3	3	3	3	0
19H	205.23492	3	3	3	3	3	3	3	3	0
20H	137.69229	3	3	3	3	3	3	3	3	0
21H	120.61514	3	3	3	3	3	3	3	3	0
22H	93.82827	3	3	3	4	3	3	3	3	0
23H	63.85559	3	3	3	4	3	3	4	3	0
24H	38.11221	3	3	4	4	3	3	4	3	0

LA TRAYECTORIA OPTIMA NORMAL ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	29.88384	1	0	0	1	1	0	0	1	0
2H	19.24579	1	0	1	0	0	0	0	1	0
3H	14.71505	1	0	1	0	0	0	1	0	0
4H	11.77399	0	0	1	0	0	0	1	0	0
5H	7.62283	0	0	1	0	0	0	1	0	0
6H	9.94970	0	0	1	0	0	0	1	0	0



7H	16.86889	1	0	1	0	0	0	1	0	0
8H	27.07871	1	0	0	0	1	1	0	1	0
9H	31.45054	1	0	0	0	1	1	0	1	0
10H	43.26281	1	1	0	0	1	1	0	1	0
11H	47.00228	1	1	0	0	1	1	0	1	0
12H	43.26281	1	1	0	0	1	1	0	1	0
13H	34.66145	1	0	0	0	1	1	0	1	0
14H	38.11266	1	0	0	0	1	1	0	1	0
15H	49.06571	1	0	0	1	1	1	0	1	0
16H	68.33212	1	0	1	1	1	0	1	1	0
17H	83.05814	1	1	1	1	1	0	1	1	0
18H	117.41838	1	1	1	1	1	1	1	1	0
19H	205.23492	1	1	1	1	1	1	1	1	0
20H	137.69229	1	1	1	1	1	1	1	1	0
21H	120.61514	1	1	1	1	1	1	1	1	0
22H	93.82827	1	1	1	0	1	1	1	1	0
23H	63.85559	1	1	1	0	1	1	0	1	0
24H	38.11221	1	1	0	0	1	1	0	1	0

CONVERGE EN 1 ITERACION

Program has used 41.26563 seconds of CPU time.

D.5. Red de Salama con Kc=5 (Sin controlar el cambiador de derivaciones)

LAS PERDIDAS CON LA TRAYECTORIA OPTIMA ES : 1351.93152

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	29.93433	1	0	0	0	1	1	0	1	0
2H	19.24579	1	0	1	0	2	2	0	1	0
3H	14.71505	1	0	1	0	2	2	1	2	0
4H	11.77399	2	0	1	0	2	2	1	2	0
5H	7.62283	2	0	1	0	2	2	1	2	0
6H	9.94970	2	0	1	0	2	2	1	2	0
7H	16.86889	3	0	1	0	2	2	1	2	0
8H	26.98671	3	0	2	1	3	2	2	3	0
9H	31.42079	3	0	2	1	3	2	2	3	0
10H	43.26281	3	1	2	2	3	3	2	3	0
11H	47.00228	3	1	2	2	3	3	2	3	0
12H	43.26281	3	1	2	2	3	3	2	3	0
13H	34.66145	3	2	2	2	3	3	2	3	0
14H	38.11221	3	3	2	2	3	3	2	3	0
15H	48.96483	3	3	2	2	3	3	2	3	0
16H	68.33212	3	4	3	3	3	4	3	3	0
17H	83.05814	3	5	3	3	3	4	3	3	0
18H	117.41838	3	5	3	3	3	5	3	3	0
19H	205.23492	3	5	3	3	3	5	3	3	0
20H	137.69229	3	5	3	3	3	5	3	3	0
21H	120.61514	3	5	3	3	3	5	3	3	0
22H	93.82827	3	5	3	4	3	5	3	3	0
23H	63.85559	3	5	3	4	3	5	4	3	0
24H	38.11221	3	5	4	4	3	5	4	3	0

LA TRAYECTORIA OPTIMA NORMAL ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	29.93433	1	0	0	0	1	1	0	1	0
2H	19.24579	1	0	1	0	0	0	0	1	0
3H	14.71505	1	0	1	0	0	0	1	0	0



4H	11.77399	0	0	1	0	0	0	1	0	0
5H	7.62283	0	0	1	0	0	0	1	0	0
6H	9.94970	0	0	1	0	0	0	1	0	0
7H	16.86889	1	0	1	0	0	0	1	0	0
8H	26.98671	1	0	0	1	1	0	0	1	0
9H	31.42079	1	0	0	1	1	0	0	1	0
10H	43.26281	1	1	0	0	1	1	0	1	0
11H	47.00228	1	1	0	0	1	1	0	1	0
12H	43.26281	1	1	0	0	1	1	0	1	0
13H	34.66145	1	0	0	0	1	1	0	1	0
14H	38.11221	1	1	0	0	1	1	0	1	0
15H	48.96483	1	1	0	0	1	1	0	1	0
16H	68.33212	1	0	1	1	1	0	1	1	0
17H	83.05814	1	1	1	1	1	0	1	1	0
18H	117.41838	1	1	1	1	1	1	1	1	0
19H	205.23492	1	1	1	1	1	1	1	1	0
20H	137.69229	1	1	1	1	1	1	1	1	0
21H	120.61514	1	1	1	1	1	1	1	1	0
22H	93.82827	1	1	1	0	1	1	1	1	0
23H	63.85559	1	1	1	0	1	1	0	1	0
24H	38.11221	1	1	0	0	1	1	0	1	0

CONVERGE EN 1 ITERACION

Program has used 42.62500 seconds of CPU time.

D.6. Red de Salama con Kc=6 (Sin controlar el cambiador de derivaciones)

LAS PERDIDAS CON LA TRAYECTORIA OPTIMA ES : 1351.81616

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	29.88384	1	0	0	1	1	0	0	1	0
2H	19.23383	2	0	1	1	2	0	0	1	0
3H	14.71505	3	0	1	2	2	0	1	2	0
4H	11.77399	4	0	1	2	2	0	1	2	0
5H	7.62283	4	0	1	2	2	0	1	2	0
6H	9.94970	4	0	1	2	2	0	1	2	0
7H	16.86889	5	0	1	2	2	0	1	2	0
8H	26.98671	5	0	2	3	3	0	2	3	0
9H	31.42079	5	0	2	3	3	0	2	3	0
10H	43.26281	5	1	2	4	3	1	2	3	0
11H	47.00228	5	1	2	4	3	1	2	3	0
12H	43.26281	5	1	2	4	3	1	2	3	0
13H	34.66145	5	2	2	4	3	1	2	3	0
14H	38.11221	5	3	2	4	3	1	2	3	0
15H	48.96483	5	3	2	4	3	1	2	3	0
16H	68.33212	5	4	3	5	3	2	3	3	0
17H	83.05814	5	5	3	5	3	2	3	3	0
18H	117.41838	5	5	3	5	3	3	3	3	0
19H	205.23492	5	5	3	5	3	3	3	3	0
20H	137.69229	5	5	3	5	3	3	3	3	0
21H	120.61514	5	5	3	5	3	3	3	3	0
22H	93.85809	5	5	3	5	3	4	3	3	0
23H	63.77242	5	6	3	5	3	5	4	3	0
24H	38.11266	5	6	4	6	3	5	4	3	0

LA TRAYECTORIA OPTIMA NORMAL ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
-----------	----------	-----------------------------	--	--	--	--	--	--	--	----------------------



1H	29.88384	1	0	0	1	1	0	0	1	0
2H	19.23383	0	0	1	1	0	0	0	1	0
3H	14.71505	1	0	1	0	0	0	1	0	0
4H	11.77399	0	0	1	0	0	0	1	0	0
5H	7.62283	0	0	1	0	0	0	1	0	0
6H	9.94970	0	0	1	0	0	0	1	0	0
7H	16.86889	1	0	1	0	0	0	1	0	0
8H	26.98671	1	0	0	1	1	0	0	1	0
9H	31.42079	1	0	0	1	1	0	0	1	0
10H	43.26281	1	1	0	0	1	1	0	1	0
11H	47.00228	1	1	0	0	1	1	0	1	0
12H	43.26281	1	1	0	0	1	1	0	1	0
13H	34.66145	1	0	0	0	1	1	0	1	0
14H	38.11221	1	1	0	0	1	1	0	1	0
15H	48.96483	1	1	0	0	1	1	0	1	0
16H	68.33212	1	0	1	1	1	0	1	1	0
17H	83.05814	1	1	1	1	1	0	1	1	0
18H	117.41838	1	1	1	1	1	1	1	1	0
19H	205.23492	1	1	1	1	1	1	1	1	0
20H	137.69229	1	1	1	1	1	1	1	1	0
21H	120.61514	1	1	1	1	1	1	1	1	0
22H	93.85809	1	1	1	1	1	0	1	1	0
23H	63.77242	1	0	1	1	1	1	0	1	0
24H	38.11266	1	0	0	0	1	1	0	1	0

CONVERGE EN 1 ITERACION

Program has used 43.26563 seconds of CPU time.

D.7. Red de Salama con Kc=1

LAS PERDIDAS TOTALES CON LA TRAYECTORIA OPTIMA SON: 1239.99207

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.35637	0	1	0	0	1	0	0	1	8
2H	17.54266	0	1	0	0	1	0	0	1	8
3H	13.86366	0	1	0	0	1	0	0	1	8
4H	11.69159	0	1	0	0	1	0	0	1	7
5H	8.87877	0	1	0	0	1	0	0	1	7
6H	10.41894	0	1	0	0	1	0	0	1	7
7H	15.60255	0	1	0	0	1	0	0	1	8
8H	24.59072	0	1	0	0	1	0	0	1	8
9H	28.81836	0	1	0	0	1	0	0	1	8
10H	39.29456	0	1	0	1	1	0	0	1	8
11H	42.77393	0	1	0	1	1	0	0	1	8
12H	39.29456	0	1	0	1	1	0	0	1	8
13H	31.54758	0	1	0	1	1	0	0	1	8
14H	34.48469	0	1	0	1	1	0	0	1	8
15H	44.41422	1	1	0	1	1	0	0	1	8
16H	62.10653	1	1	0	1	1	0	1	1	8
17H	75.10242	1	1	0	1	1	0	1	1	8
18H	105.83985	1	1	1	1	1	0	1	1	8
19H	185.25114	1	1	1	1	1	0	1	1	8
20H	124.31032	1	1	1	1	1	0	1	1	8
21H	108.75920	1	1	1	1	1	0	1	1	8
22H	84.71135	1	1	1	1	1	0	1	1	8
23H	60.62265	1	1	1	1	1	0	1	1	7
24H	42.71544	1	1	1	1	1	0	1	1	6

LA TRAYECTORIA OPTIMA EN ESTADOS NORMALES ES:



ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.35637	0	1	0	0	1	0	0	1	8
2H	17.54266	0	1	0	0	1	0	0	1	8
3H	13.86366	0	1	0	0	1	0	0	1	8
4H	11.69159	0	1	0	0	1	0	0	1	7
5H	8.87877	0	1	0	0	1	0	0	1	7
6H	10.41894	0	1	0	0	1	0	0	1	7
7H	15.60255	0	1	0	0	1	0	0	1	8
8H	24.59072	0	1	0	0	1	0	0	1	8
9H	28.81836	0	1	0	0	1	0	0	1	8
10H	39.29456	0	1	0	1	1	0	0	1	8
11H	42.77393	0	1	0	1	1	0	0	1	8
12H	39.29456	0	1	0	1	1	0	0	1	8
13H	31.54758	0	1	0	1	1	0	0	1	8
14H	34.48469	0	1	0	1	1	0	0	1	8
15H	44.41422	1	1	0	1	1	0	0	1	8
16H	62.10653	1	1	0	1	1	0	1	1	8
17H	75.10242	1	1	0	1	1	0	1	1	8
18H	105.83985	1	1	1	1	1	0	1	1	8
19H	185.25114	1	1	1	1	1	0	1	1	8
20H	124.31032	1	1	1	1	1	0	1	1	8
21H	108.75920	1	1	1	1	1	0	1	1	8
22H	84.71135	1	1	1	1	1	0	1	1	8
23H	60.62265	1	1	1	1	1	0	1	1	7
24H	42.71544	1	1	1	1	1	0	1	1	6

CONVERGE EN 3 ITERACIONES

Program has used 47.00000 seconds of CPU time.

D.8. Red de Salama con Kc=2

LAS PERDIDAS TOTALES CON LA TRAYECTORIA OPTIMA SON: 1226.92175

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.32844	1	0	1	1	0	0	1	0	8
2H	17.40940	1	0	1	2	0	0	1	0	8
3H	13.40880	2	0	1	2	0	0	1	0	8
4H	10.66625	2	0	1	2	0	0	1	0	8
5H	7.08280	2	0	1	2	0	0	1	0	8
6H	9.08596	2	0	1	2	0	0	1	0	8
7H	15.46812	2	1	1	2	0	0	1	0	8
8H	24.65645	2	1	1	2	0	0	1	0	8
9H	28.95909	2	1	1	2	0	0	1	0	8
10H	39.58716	2	1	1	2	0	1	1	0	8
11H	43.07542	2	1	1	2	0	1	1	0	8
12H	39.58716	2	1	1	2	0	1	1	0	8
13H	31.81790	2	1	1	2	0	1	1	0	8
14H	34.76394	2	1	1	2	0	1	1	0	8
15H	44.90228	2	1	1	2	0	1	1	0	8
16H	62.33857	2	1	1	2	1	1	1	0	8
17H	75.41999	2	1	1	2	1	1	1	1	8
18H	105.62035	2	1	1	2	1	1	1	1	9
19H	186.39600	2	1	1	2	1	1	1	1	9
20H	124.48561	2	1	1	2	1	1	1	1	9
21H	108.60662	2	1	1	2	1	1	1	1	9
22H	83.94275	2	1	1	2	1	1	1	1	9
23H	57.79259	2	1	1	2	1	1	2	1	8



24H 34.52013 2 1 2 2 1 1 2 1 8

LA TRAYECTORIA OPTIMA EN ESTADOS NORMALES ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.32844	1	0	1	1	0	0	1	0	8
2H	17.40940	1	0	1	0	0	0	1	0	8
3H	13.40880	0	0	1	0	0	0	1	0	8
4H	10.66625	0	0	1	0	0	0	1	0	8
5H	7.08280	0	0	1	0	0	0	1	0	8
6H	9.08596	0	0	1	0	0	0	1	0	8
7H	15.46812	0	1	1	0	0	0	1	0	8
8H	24.65645	0	1	1	0	0	0	1	0	8
9H	28.95909	0	1	1	0	0	0	1	0	8
10H	39.58716	0	1	1	0	0	1	1	0	8
11H	43.07542	0	1	1	0	0	1	1	0	8
12H	39.58716	0	1	1	0	0	1	1	0	8
13H	31.81790	0	1	1	0	0	1	1	0	8
14H	34.76394	0	1	1	0	0	1	1	0	8
15H	44.90228	0	1	1	0	0	1	1	0	8
16H	62.33857	0	1	1	0	1	1	1	0	8
17H	75.41999	0	1	1	0	1	1	1	1	8
18H	105.62035	0	1	1	0	1	1	1	1	9
19H	186.39600	0	1	1	0	1	1	1	1	9
20H	124.48561	0	1	1	0	1	1	1	1	9
21H	108.60662	0	1	1	0	1	1	1	1	9
22H	83.94275	0	1	1	0	1	1	1	1	9
23H	57.79259	0	1	1	0	1	1	0	1	8
24H	34.52013	0	1	0	0	1	1	0	1	8

CONVERGE EN 3 ITERACIONES

Program has used 81.45313 seconds of CPU time.

D.9. Red de Salama con Kc=3

LAS PERDIDAS TOTALES CON LA TRAYECTORIA OPTIMA SON: 1220.71045

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.56964	1	0	0	0	1	0	0	1	8
2H	17.49487	1	0	0	0	1	0	0	1	8
3H	13.68552	1	0	0	0	1	0	0	1	8
4H	11.42871	1	0	0	0	1	0	0	1	7
5H	8.45329	1	0	0	0	1	0	0	1	7
6H	10.09100	1	0	0	0	1	0	0	1	7
7H	15.48957	1	0	0	0	1	0	0	1	8
8H	24.73873	1	0	0	0	1	0	0	1	8
9H	28.64999	1	0	0	0	1	1	0	1	8
10H	39.21052	1	0	0	0	1	1	0	1	8
11H	42.71550	1	0	0	0	1	1	0	1	8
12H	39.21052	1	0	0	0	1	1	0	1	8
13H	31.39966	1	0	0	0	1	1	0	1	8
14H	34.36230	1	0	0	0	1	1	0	1	8
15H	44.32754	1	1	0	0	1	1	0	1	8
16H	61.91023	1	1	0	0	1	1	0	1	8
17H	75.17563	1	1	1	0	1	1	0	1	8
18H	105.78876	1	1	1	0	1	1	1	1	8
19H	180.29187	1	1	1	1	1	1	1	1	8
20H	123.27698	1	1	1	1	1	1	1	1	8



21H	108.75928	1	1	1	1	1	2	1	1	8
22H	84.71143	1	1	1	1	1	2	1	1	8
23H	57.60667	1	1	2	2	1	3	2	1	8
24H	34.36230	1	2	2	2	1	3	2	1	8

LA TRAYECTORIA OPTIMA EN ESTADOS NORMALES ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.56964	1	0	0	0	1	0	0	1	8
2H	17.49487	1	0	0	0	1	0	0	1	8
3H	13.68552	1	0	0	0	1	0	0	1	8
4H	11.42871	1	0	0	0	1	0	0	1	7
5H	8.45329	1	0	0	0	1	0	0	1	7
6H	10.09100	1	0	0	0	1	0	0	1	7
7H	15.48957	1	0	0	0	1	0	0	1	8
8H	24.73873	1	0	0	0	1	0	0	1	8
9H	28.64999	1	0	0	0	1	1	0	1	8
10H	39.21052	1	0	0	0	1	1	0	1	8
11H	42.71550	1	0	0	0	1	1	0	1	8
12H	39.21052	1	0	0	0	1	1	0	1	8
13H	31.39966	1	0	0	0	1	1	0	1	8
14H	34.36230	1	0	0	0	1	1	0	1	8
15H	44.32754	1	1	0	0	1	1	0	1	8
16H	61.91023	1	1	0	0	1	1	0	1	8
17H	75.17563	1	1	1	0	1	1	0	1	8
18H	105.78876	1	1	1	0	1	1	1	1	8
19H	180.29187	1	1	1	1	1	1	1	1	8
20H	123.27698	1	1	1	1	1	1	1	1	8
21H	108.75928	1	1	1	1	1	0	1	1	8
22H	84.71143	1	1	1	1	1	0	1	1	8
23H	57.60667	1	1	0	0	1	1	0	1	8
24H	34.36230	1	0	0	0	1	1	0	1	8

CONVERGE EN 3 ITERACIONES

Program has used 119.2969 seconds of CPU time.

D.10. Red de Salama con Kc=4

LAS PERDIDAS TOTALES CON LA TRAYECTORIA OPTIMA SON: 1220.63745

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.56964	1	0	0	0	1	0	0	1	8
2H	17.49487	1	0	0	0	1	0	0	1	8
3H	13.68552	1	0	0	0	1	0	0	1	8
4H	11.42871	1	0	0	0	1	0	0	1	7
5H	8.45329	1	0	0	0	1	0	0	1	7
6H	10.09100	1	0	0	0	1	0	0	1	7
7H	15.48957	1	0	0	0	1	0	0	1	8
8H	24.73873	1	0	0	0	1	0	0	1	8
9H	28.64999	1	0	0	0	1	1	0	1	8
10H	39.21052	1	0	0	0	1	1	0	1	8
11H	42.71550	1	0	0	0	1	1	0	1	8
12H	39.21052	1	0	0	0	1	1	0	1	8
13H	31.39966	1	0	0	0	1	1	0	1	8
14H	34.36230	1	0	0	0	1	1	0	1	8
15H	44.32754	1	1	0	0	1	1	0	1	8
16H	61.91023	1	1	0	0	1	1	0	1	8
17H	75.17563	1	1	1	0	1	1	0	1	8



18H	105.78876	1	1	1	0	1	1	1	1	8
19H	180.29187	1	1	1	1	1	1	1	1	8
20H	123.27698	1	1	1	1	1	1	1	1	8
21H	108.68616	1	1	1	2	1	1	1	1	8
22H	84.71143	1	1	1	3	1	2	1	1	8
23H	57.60667	1	1	2	4	1	3	2	1	8
24H	34.36230	1	2	2	4	1	3	2	1	8

LA TRAYECTORIA OPTIMA EN ESTADOS NORMALES ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.56964	1	0	0	0	1	0	0	1	8
2H	17.49487	1	0	0	0	1	0	0	1	8
3H	13.68552	1	0	0	0	1	0	0	1	8
4H	11.42871	1	0	0	0	1	0	0	1	7
5H	8.45329	1	0	0	0	1	0	0	1	7
6H	10.09100	1	0	0	0	1	0	0	1	7
7H	15.48957	1	0	0	0	1	0	0	1	8
8H	24.73873	1	0	0	0	1	0	0	1	8
9H	28.64999	1	0	0	0	1	1	0	1	8
10H	39.21052	1	0	0	0	1	1	0	1	8
11H	42.71550	1	0	0	0	1	1	0	1	8
12H	39.21052	1	0	0	0	1	1	0	1	8
13H	31.39966	1	0	0	0	1	1	0	1	8
14H	34.36230	1	0	0	0	1	1	0	1	8
15H	44.32754	1	1	0	0	1	1	0	1	8
16H	61.91023	1	1	0	0	1	1	0	1	8
17H	75.17563	1	1	1	0	1	1	0	1	8
18H	105.78876	1	1	1	0	1	1	1	1	8
19H	180.29187	1	1	1	1	1	1	1	1	8
20H	123.27698	1	1	1	1	1	1	1	1	8
21H	108.68616	1	1	1	0	1	1	1	1	8
22H	84.71143	1	1	1	1	1	0	1	1	8
23H	57.60667	1	1	0	0	1	1	0	1	8
24H	34.36230	1	0	0	0	1	1	0	1	8

CONVERGE EN 3 ITERACIONES

Program has used 141.7500 seconds of CPU time.

D.11. Red de Salama con Kc=5

LAS PERDIDAS TOTALES CON LA TRAYECTORIA OPTIMA SON: 1216.49292

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.21359	1	0	0	1	1	0	0	1	8
2H	17.40940	1	0	1	2	2	0	1	2	8
3H	13.40880	2	0	1	2	2	0	1	2	8
4H	10.66625	2	0	1	2	2	0	1	2	8
5H	7.08280	2	0	1	2	2	0	1	2	8
6H	9.08596	2	0	1	2	2	0	1	2	8
7H	15.35416	3	0	1	2	2	0	1	2	8
8H	24.56676	4	0	2	3	3	0	2	3	8
9H	28.52999	5	0	2	3	3	0	2	3	8
10H	39.21052	5	0	2	4	3	1	2	3	8
11H	42.65290	5	1	2	4	3	1	2	3	8
12H	39.21052	5	2	2	4	3	1	2	3	8
13H	31.39966	5	2	2	4	3	1	2	3	8
14H	34.36230	5	2	2	4	3	1	2	3	8



15H	44.32754	5	3	2	4	3	1	2	3	8
16H	61.91023	5	3	2	4	3	1	2	3	8
17H	75.10242	5	3	2	5	3	2	3	3	8
18H	105.83985	5	3	3	5	3	2	3	3	8
19H	180.29187	5	3	3	5	3	3	3	3	8
20H	123.27698	5	3	3	5	3	3	3	3	8
21H	108.75928	5	3	3	5	3	4	3	3	8
22H	84.71143	5	3	3	5	3	4	3	3	8
23H	57.79448	5	4	4	5	3	4	3	3	8
24H	34.32529	5	4	4	5	3	4	4	3	8

LA TRAYECTORIA OPTIMA EN ESTADOS NORMALES ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.21359	1	0	0	1	1	0	0	1	8
2H	17.40940	1	0	1	0	0	0	1	0	8
3H	13.40880	0	0	1	0	0	0	1	0	8
4H	10.66625	0	0	1	0	0	0	1	0	8
5H	7.08280	0	0	1	0	0	0	1	0	8
6H	9.08596	0	0	1	0	0	0	1	0	8
7H	15.35416	1	0	1	0	0	0	1	0	8
8H	24.56676	0	0	0	1	1	0	0	1	8
9H	28.52999	1	0	0	1	1	0	0	1	8
10H	39.21052	1	0	0	0	1	1	0	1	8
11H	42.65290	1	1	0	0	1	1	0	1	8
12H	39.21052	1	0	0	0	1	1	0	1	8
13H	31.39966	1	0	0	0	1	1	0	1	8
14H	34.36230	1	0	0	0	1	1	0	1	8
15H	44.32754	1	1	0	0	1	1	0	1	8
16H	61.91023	1	1	0	0	1	1	0	1	8
17H	75.10242	1	1	0	1	1	0	1	1	8
18H	105.83985	1	1	1	1	1	0	1	1	8
19H	180.29187	1	1	1	1	1	1	1	1	8
20H	123.27698	1	1	1	1	1	1	1	1	8
21H	108.75928	1	1	1	1	1	0	1	1	8
22H	84.71143	1	1	1	1	1	0	1	1	8
23H	57.79448	1	0	0	1	1	0	1	1	8
24H	34.32529	1	0	0	1	1	0	0	1	8

CONVERGE EN 3 ITERACIONES

Program has used 143.0156 seconds of CPU time

D.12. Red de Salama con Kc=6

LAS PERDIDAS TOTALES CON LA TRAYECTORIA OPTIMA SON: 1216.22046

LA TRAYECTORIA OPTIMA EN ESTADOS MODIFICADOS ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.21359	1	0	0	1	1	0	0	1	8
2H	17.40940	1	0	1	2	2	0	1	2	8
3H	13.40880	2	0	1	2	2	0	1	2	8
4H	10.66625	2	0	1	2	2	0	1	2	8
5H	6.97568	3	0	2	2	2	1	2	2	8
6H	9.13651	3	0	2	2	2	2	3	2	8
7H	15.35416	3	0	3	2	2	2	3	2	8
8H	24.56676	4	0	4	3	3	2	4	3	8
9H	28.52999	5	0	4	3	3	2	4	3	8
10H	39.21052	5	0	4	4	3	3	4	3	8
11H	42.65290	5	1	4	4	3	3	4	3	8



12H	39.21052	5	2	4	4	3	3	4	3	8
13H	31.39966	5	2	4	4	3	3	4	3	8
14H	34.36230	5	2	4	4	3	3	4	3	8
15H	44.32754	5	3	4	4	3	3	4	3	8
16H	61.91023	5	3	4	4	3	3	4	3	8
17H	74.98515	5	4	5	5	3	4	5	3	8
18H	105.83985	5	5	5	5	3	4	5	3	8
19H	180.29187	5	5	5	5	3	5	5	3	8
20H	123.27698	5	5	5	5	3	5	5	3	8
21H	108.68616	5	5	5	6	3	5	5	3	8
22H	84.83668	5	5	5	6	3	5	5	3	8
23H	57.60667	5	5	6	6	3	5	6	3	8
24H	34.36230	5	6	6	6	3	5	6	3	8

LA TRAYECTORIA OPTIMA EN ESTADOS NORMALES ES:

ESCENARIO	PERDIDAS	POSICION DE LOS CAPACITORES								POSICION DE LOS TAPS
1H	27.21359	1	0	0	1	1	0	0	1	8
2H	17.40940	1	0	1	0	0	0	1	0	8
3H	13.40880	0	0	1	0	0	0	1	0	8
4H	10.66625	0	0	1	0	0	0	1	0	8
5H	6.97568	1	0	0	0	0	1	0	0	8
6H	9.13651	1	0	0	0	0	0	1	0	8
7H	15.35416	1	0	1	0	0	0	1	0	8
8H	24.56676	0	0	0	1	1	0	0	1	8
9H	28.52999	1	0	0	1	1	0	0	1	8
10H	39.21052	1	0	0	0	1	1	0	1	8
11H	42.65290	1	1	0	0	1	1	0	1	8
12H	39.21052	1	0	0	0	1	1	0	1	8
13H	31.39966	1	0	0	0	1	1	0	1	8
14H	34.36230	1	0	0	0	1	1	0	1	8
15H	44.32754	1	1	0	0	1	1	0	1	8
16H	61.91023	1	1	0	0	1	1	0	1	8
17H	74.98515	1	0	1	1	1	0	1	1	8
18H	105.83985	1	1	1	1	1	0	1	1	8
19H	180.29187	1	1	1	1	1	1	1	1	8
20H	123.27698	1	1	1	1	1	1	1	1	8
21H	108.68616	1	1	1	0	1	1	1	1	8
22H	84.83668	1	1	1	0	1	1	1	1	8
23H	57.60667	1	1	0	0	1	1	0	1	8
24H	34.36230	1	0	0	0	1	1	0	1	8

CONVERGE EN 3 ITERACIONES

Program has used 146.0313 seconds of CPU time.