

ESCUELA SUPERIOR POLITECNICA DEL LITORAL

FACULTAD DE INGENIERIA ELECTRICA

"EL METODO DE NEWTON - RAPHSON EN LA RESOLUCION DEL
PROBLEMA DEL FLUJO DE POTENCIA."

Tesis de Grado

Previa a la obtención del título de:

INGENIERO EN ELECTRICIDAD

ESPECIALIZACION : POTENCIA

Presentada por:

OTTO ISMAEL ALVARADO MORENO

GUAYAQUIL - ECUADOR

1.985

AGRADECIMIENTO

- AL ING. CRISTOBAL MERA G., Director de Tesis y al ING. GUILLERMO ROMERO, por la valiosa ayuda que me brindaron en la elaboración de esta tesis.
- A los Ingenieros: Flor María Dávila, Eduardo Rodríguez, Antonio Ycaza; y, al señor Daniel Rodríguez, por sus contribuciones al desarrollo de este trabajo.

DEDICATORIA


- A LA MEMORIA DE MI PADRE (+)
- A MI MADRE
- A MIS HERMANOS
- A MIS AMIGOS



ING. GUSTAVO BERMUDEZ FLORES
SUB-DECANO DE LA FACULTAD DE
INGENIERIA ELECTRICA



ING. CRISTOBAL MERA G.
DIRECTOR DE TESIS



ING. ALBERTO HANZE BELLO
MIEMBRO PRINCIPAL



ING. JORGE FLORES MACIAS
MIEMBRO PRINCIPAL

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
Departamento de Ingeniería Eléctrica
BIBLIOTECA
Inv. No. POT - 067

DECLARACION EXPRESA

"LA RESPONSABILIDAD POR LOS HECHOS, IDEAS Y DOCTRINAS EXPUESTOS EN ESTA TESIS, ME CORRESPONDEN EXCLUSIVAMENTE; Y, EL PATRIMONIO INTELECTUAL DE LA MISMA, A LA ESCUELA - SUPERIOR POLITÉCNICA DEL LITORAL".

(Reglamento de Exámenes y Títulos Profesionales de la ESPOL).

Otto Alvarado

OTTO ISMAEL ALVARADO MORENO

RESUMEN

El presente trabajo contiene las bases teóricas del método de Newton - Raphson, y se incluye también las técnicas, que han hecho que este método se imponga en la resolución del problema del flujo de potencia.

En el Capítulo I, se dan breves definiciones de un Sistema de Potencia, del flujo de potencia y sus objetivos, se explica como han ido evolucionando los métodos de resolución de flujo de potencia, y se modela los componentes de un sistema de potencia.

En el Capítulo II, se enfoca el problema del flujo de potencia y se plantea como resolverlo, se explica la teoría general del método de Newton - Raphson y aplicada al flujo de potencia, se dan las ventajas y desventajas del método de Newton - Raphson.

En el Capítulo III, se explica en forma breve como optimizar memoria aprovechando las características que tiene un

Sistema Eléctrico de Potencia, y se dan las técnicas numéricas que se usan para resolver en la forma más óptima las ecuaciones del método de Newton - Raphson, se dan las técnicas de programación y almacenamiento para aprovechar la dispersidad de la matriz Jacobiano.

En el Capítulo IV, se aplica el método de Newton - Raphson en la resolución del flujo de potencia de un ejemplo, usando una versión del programa de la Compañía Eléctrica de Philadelphia ambientado en el sistema 4341, también se incluye - aquí la estructura del programa, la función de las principales subrutinas y el manual del usuario.

INDICE GENERAL

	<u>PAGS.</u>
RESUMEN -----	VI
INDICE GENERAL -----	VIII
INDICE DE FIGURAS -----	XII
INDICE DE TABLAS -----	XV
INTRODUCCION -----	17
CAPITULO I	
FUNDAMENTOS BASICOS DEL FLUJO DE POTENCIA	
1.1. DEFINICION DE UN SISTEMA ELECTRICO DE POTENCIA ----	19
1.2. DEFINICION Y OBJETIVOS DEL FLUJO DE POTENCIA ----	20
1.3. RESUMEN HISTORICO DE LOS METODOS EMPLEADOS EN LA RESOLUCION DEL FLUJO DE POTENCIA -----	22
1.4. MODELO DE LOS COMPONENTES DE UN SISTEMA DE PO TENCIA -----	25
1.4.1. Generadores -----	26
1.4.2. Líneas de Transmisión -----	27
1.4.3. Transformadores -----	29
1.4.4. Capacitores y Reactores en paralelo -----	36

CAPITULO II

ANALISIS DEL PROBLEMA DEL FLUJO DE POTENCIA

2.1. PLANTEAMIENTO DEL PROBLEMA -----	38
2.2. ALTERNATIVAS DE SOLUCION -----	44
2.3. EL METODO DE NEWTON - RAPHSON -----	45
2.3.1. Caso de una dimensión -----	46
2.3.2. Caso de n dimensiones -----	52
2.4. EL METODO DE NEWTON - RAPHSON PARA EL FLUJO DE POTENCIA -----	57
2.4.1. Derivación y análisis de las ecuaciones de Newton - Raphson para el flujo de poten cia -----	58
2.4.2. Estrategia de solución por medio del méto do de Newton - Raphson -----	74
2.4.3. Ventajas y desventajas del método de New ton - Raphson sobre los métodos iterativos (Gauss - Seidel).	77

CAPITULO III

TECNICAS NUMERICAS Y DE PROGRAMACION

3.1. ALMACENAMIENTO APROVECHANDO SIMETRIA Y DISPERSIDAD	82
3.1.1. Almacenamiento aprovechando simetría -----	83
3.1.2. Almacenamiento aprovechando dispersidad ---	85

	<u>PAGS.</u>
3.2. TECNICAS NUMERICAS -----	88
3.2.1. Eliminación de Gauss -----	90
3.2.2. Optimización de dispersidad y secuencia - de procesamiento -----	105
3.2.3. Variación de dispersidad con secuencia de procesamiento -----	107
3.2.4. Esquemas de ordenamiento de nodos -----	115
3.3. TECNICAS DE PROGRAMACION -----	124
3.3.1. Tablas de almacenamiento para un ejemplo dado -----	126
3.3.2. Cálculo de elementos del Jacobiano -----	130
3.3.3. Esquema de triangularización -----	131
3.3.4. Esquema de almacenamiento para Jacobiano Triangulado -----	135
3.3.5. Esquema de Direccionamiento -----	138
3.3.6. Esquema de eliminación Gaussiana (Ejemplo)---	140
3.3.7. Esquema de eliminación Gaussiana para Jacobia no del flujo de potencia -----	149

CAPITULO IV

APLICACION DEL METODO DE NEWTON - RAPHSON EN LA SOLUCION DEL FLUJO DE POTENCIA

4.1. ESTRUCTURA DEL PROGRAMA A USAR -----	156
---	-----

	<u>PAGS.</u>
4.2. FUNCION DE PRINCIPALES SUBROUTINAS -----	159
4.3. MANUAL DEL USUARIO -----	169
4.4. EJEMPLO DE APLICACION -----	243
CONCLUSIONES Y RECOMENDACIONES -----	261
BIBLIOGRAFIA -----	266

I N T R O D U C C I O N

Día a día aumenta la dependencia de la sociedad a los servicios de energía eléctrica, por lo cual se hace necesario dedicar mayores esfuerzos en desarrollar y mejorar aún más los sistemas eléctricos de potencia, los cuales son cada vez más grandes y complejos.

Este incremento en tamaño y complejidad de tales sistemas, hacen necesaria una continua revisión y actualización de los conocimientos sobre los tópicos relacionados con el Análisis de los Sistemas Eléctricos de Potencia, y sobre todo en las bases de los análisis que son punto de partida para estudios más complejos.

El estudio de flujo de carga o flujo de potencia es precisamente uno de dichos tópicos, y es uno de los más importantes, por lo cual en las últimas décadas grandes esfuerzos han sido dedicados a resolver el problema del flujo de potencia en una forma más eficiente, utilizando técnicas numéricas y de programación adecuadas y mejoradas, con el objeto de vencer los problemas e inconvenientes causados por el aumento de tamaño y

complejidad de los Sistemas de Potencia.

El método de Newton - Raphson es una de dichas técnicas numéricas, pero si este hubiera sido confinado al algoritmo básico de Newton, su aplicación a la resolución de flujo de grandes sistemas de potencia hubiera sido muy difícil, por no decir sino imposible, así que esto exigió que se desarrollaran sofisticadas técnicas computacionales y de programación, que transforman este método en el más poderoso de los métodos de resolución de flujo de potencia.

Además en la actualidad practicamente todas las técnicas empleadas en los Análisis de los Sistemas Eléctricos de Potencia, y en especial en los flujos de potencia, están basadas en el desarrollo de modelos matemáticos y técnicas computacionales que aprovechan las características del sistema de potencia a resolver, por lo cual esta tesis presenta el método de Newton - Raphson con las técnicas matemáticas y de programación que han hecho que este método sea ampliamente aceptado en la resolución del flujo de potencia.

CAPITULO I

FUNDAMENTOS BASICOS DEL FLUJO DE POTENCIA

1.1. DEFINICION DE UN SISTEMA ELECTRICO DE POTENCIA

La producción y distribución de la energía eléctrica tiene como objetivo principal dar comodidad a la vida humana, el consumo de dicha energía eléctrica, representa la carga en un sistema eléctrico. Para que la energía eléctrica producida en los centros de generación llegue a la carga, se requiere de una red eléctrica⁽¹⁾ que está constituida en forma general por nodos o barras y ramales, pudiendo estos ramales ser serie como líneas, transformadores y a veces capacitores y reactores, o en paralelo como lo son los capacitores y reactores conectados entre un nodo y tierra.

A la configuración que incluye la generación, carga y red eléctrica se denomina un Sistema Eléctrico de Potencia, ρ en forma abreviada un Sistema de Poten-

cia, puede ser tan sencillo como el mostrado en la figura N° 1.1., que consta de un generador, línea de transmisión y carga, o puede ser tan complejo de tal forma que esté constituido por una multitud de ramales, nodos, centros de generación y cargas.

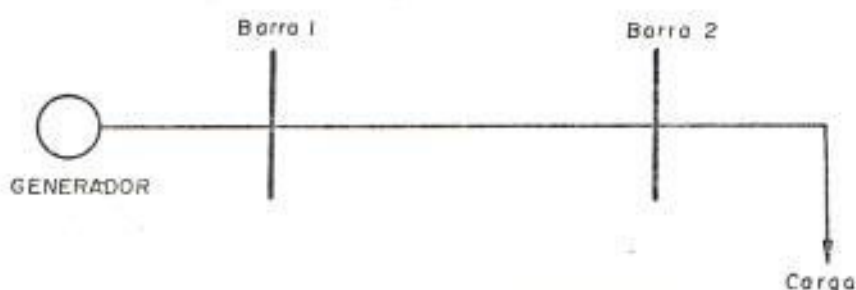


FIGURA N° 1.1. SISTEMA DE POTENCIA SENCILLO

1.2. DEFINICION Y OBJETIVOS DEL FLUJO DE POTENCIA

Bajo condiciones normales un Sistema Eléctrico de Potencia opera en su modo de estado estable, y el cálculo básico requerido para determinar las características de este estado en el llamado flujo de potencia⁽²⁾.

El objetivo del flujo de potencia es determinar las características de operación de estado estable de un Sistema Eléctrico de Potencia para un conjunto de condiciones

dadas de generación y carga.

La solución del flujo de potencia dará información de las magnitudes y ángulos del voltaje en cada uno de los nodos o barras del Sistema de Potencia estudiado, los flujos de potencia activa y reactiva y las pérdidas de potencia sobre cada uno de los ramales serie del sistema, además la potencia reactiva generada o absorbida en los generadores con potencia reactiva controlable.

Los estudios de flujo de potencia pueden ser realizados para la planificación de sistemas de potencia, planificación operacional, y en operación y control del sistema, con estos estudios se puede investigar lo siguiente⁽¹⁾:

Las condiciones óptimas de operación y distribución de cargas del sistema.

El efecto del rearrreglo de circuitos e incorporación de nuevas cargas.

Los efectos de pérdidas temporales de generadores o líneas de transmisión.

Las pérdidas óptimas de potencia.

La influencia del cambio de tamaño en los conductores.

La posición óptima del cambiador de taps de los transformadores.

dores.

Las posibles ampliaciones del Sistema Eléctrico de Potencia.

La programación óptima de despacho de carga.

1.3. RESUMEN HISTORICO DE LOS METODOS EMPLEADOS EN LA RESOLUCION DEL FLUJO DE POTENCIA⁽³⁾

Inicialmente la resolución de los flujos de potencia eran obtenidos usando los analizadores de redes, e incluso hasta algún tiempo después del advenimiento del computador digital, pero el trabajo laborioso y costoso - que demandaba el uso de los analizadores, volcaron las miradas al computador en busca de precisión y rapidez para realizar los cálculos.

El primer método verdaderamente práctico para una solución digital automática apareció en 1.956⁽⁴⁾, fue desarrollado por J.B. Ward y H.W. Hale, posteriormente A.F. Glim y G.W. Stagg, R.J. Brown y W.F. Tinney presentan métodos de cálculo iterativos en base a la resolución de la ecuación $\bar{I} = \bar{Y} * \bar{E}$, que finalmente terminarán por converger en la solución del problema planteado, estos métodos son conocidos con el nombre de métodos iterativos de la matriz admitancia (\bar{Y}), o también como métodos de desplazamientos sucesivos. Para su tiempo, estos métodos cumplen con los requeri

mientos de memoria que en forma limitada poseía la primera generación de computadores y también sirven plenamente en la resolución de flujos, aunque tienen características de lenta convergencia y en ocasiones no pueden resolver - Sistemas de Potencia denominados mal condicionados.

El deseo de cubrir estas deficiencias, conduce al desarrollo del método de la matriz impedancia (Z), el cual converge en una forma más confiable, pero sacrifica algunas de las ventajas de los métodos iterativos de la matriz admittancia especialmente cuando se trata de resolver grandes Sistemas de Potencia, pues los requerimientos de memoria son demasiado grandes y demora mucho su ejecución, todo esto debido a la misma característica de la matriz impedancia que es una matriz llena, haciendo imposible tratar de reducir sus necesidades de memoria.

Así es que en la década del 60 nuevamente se trata de enfocar la formulación matemática en base a la matriz admittancia, pero usando el algoritmo de Newton - Raphson⁽⁵⁾ en lugar del de Gauss - Seidel que usaban los métodos de desplazamiento sucesivo, el método de Newton - Raphson demostró tener poderosas propiedades de convergencia⁽⁶⁾, pero computacionalmente no era competitivo, debido a su difícil programación y a considerables requisitos de memoria, aunque cubría la deficiencia de convergencia lenta de los

métodos iterativos anteriores y resolvía problemas que no podían ser resueltos por estos métodos, quedó relegado a un segundo plano.

A mediados de los años 60 se consiguen considerables adelantos en los cálculos de redes de sistemas de Potencia, con el desarrollo por Tinney y otros autores de técnicas para el manejo y cálculo con matrices dispersas donde la dispersidad^(7,9,10) es precisamente una de las características de las matrices admitancias y jacobiano que representan una red de un Sistema Eléctrico de Potencia, es así como con estas técnicas se logró reducir dramáticamente la velocidad de cálculo y requerimientos de memoria del método de Newton⁽⁸⁾, lo cual conjuntamente con su poder de convergencia determinó la aceptación del método de Newton-Raphson para la resolución del flujo de potencia.

En la década del 70 una vez impuesto el método de Newton-Raphson se profundizó sobre este, mejorándolo y sirviendo como punto de partida para el desarrollo de los métodos desacoplados, que son métodos de resolución aproximados, los autores de estos métodos son principalmente S.T. Despotovic, B.S. Babic, B. Stott y O. Alsac, con estos métodos se gana velocidad de cálculo y reducción de memoria, pero se sacrifica exactitud e incrementa el número de iteraciones para la convergencia con respecto al método de Newton - Raphson.

Posteriormente, métodos de programación no lineal y métodos híbridos han sido desarrollados, pero estos han sido tomados - en cuenta únicamente con interés académico y no han sido aceptados para uso en la industria de potencia eléctrica.

Finalmente el método de Newton - Raphson y otro método derivado de este que es el desacoplado rápido, se han impuesto como métodos de resolución exactos y aproximados respectivamente en el flujo de potencia.

1.4. MODELO DE LOS COMPONENTES DE UN SISTEMA DE POTENCIA

Los componentes más comunes de un Sistema de Potencia que son de interés en el análisis de estado estable son:

- Los generadores
- Transformadores
- Líneas de transmisión
- Capacitores y reactores en paralelo
- Cargas

En la mayoría de aplicaciones de análisis de flujo de potencia, la operación balanceada del Sistema de Potencia es asumida.

En algunos casos especiales la operación desbalanceada es dada, sin embargo consideraremos aquí operación balanceada, bajo es

ta condición el modelo de una fase será representativo de un componente trifásico.

1.4.1. Generadores

Para el estudio del flujo de potencia el generador - sincrónico tiene el modelo aproximado dado en la figura N° 1.2.a., la cual muestra los límites operacionales del generador sincrónico en estado estable, donde la línea cde viene dada por la capacidad de la turbina que se encuentra con la línea de fp. nominal del generador, y que coincide en este punto con la capacidad nominal en MVA del generador, y que está limitado por la capacidad térmica del generador, es decir por la elevación de temperatura del estator y del rotor, esta es la zona bc, desde b hasta a el calentamiento del rotor es el que decide la curva.

El límite ef viene dado por las pérdidas de sincronismo a fp. en adelante, teóricamente la curva de P_{\min} es cero, pero aquí se da un valor de potencia activa mínimo que depende del tipo de turbina que mueve al generador.

La figura N° 1.2.b., muestra la representación unifilar del generador para el flujo de potencia, el vol

taje del generador se mantendrá constante siempre y cuando exista suficiente reactivo sin violar los límites para mantenerlo, caso contrario el reactivo del generador se fija a uno de los límites permitiéndosele variar a la magnitud del voltaje.

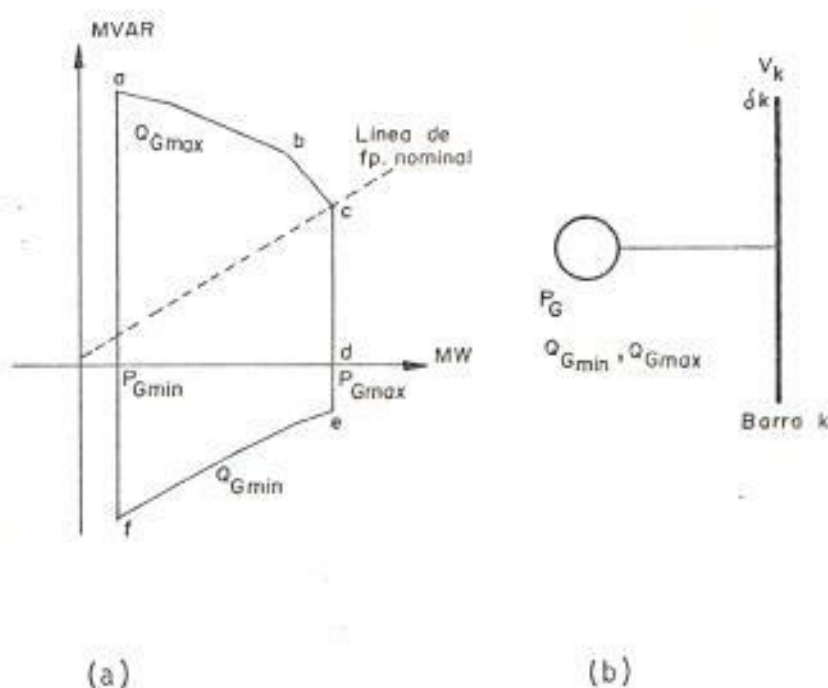


FIGURA N° 1.2. MODELO DEL GENERADOR

1.4.2. Líneas de Transmisión

Bajo condiciones de estado estable podemos resolver las ecuaciones de ondas para líneas de transmisión y reducir el modelo de la línea de transmisión a un equivalente π simétrico y concentrado como el representado en la figura N° 1.3.

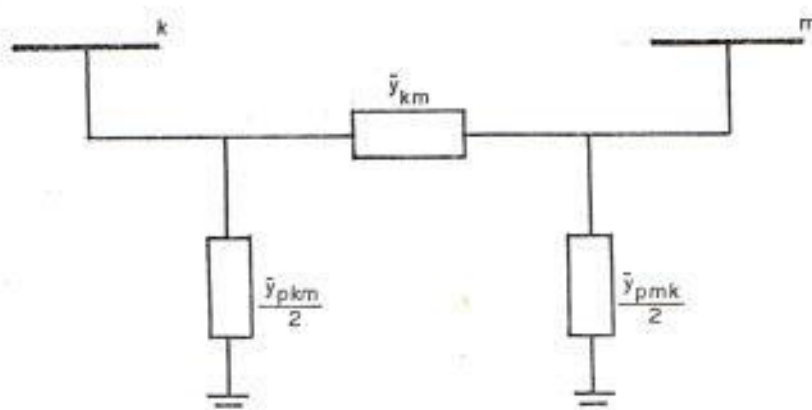


FIGURA N^o 1.3. MODELO EN EQUIVALENTE π DE UNA LINEA DE TRANSMISION

Donde:

$$\bar{y}_{km} = \frac{1}{\bar{z}_{km}} = g_{km} + j b_{km}$$

Admitancia total serie de la línea conectada entre los nodos k y m.

$$\bar{z}_{km} = r_{km} + j x_{km}$$

Impedancia total serie de la línea entre los nodos k y m.

$$\bar{y}_{pkm} = \bar{y}_{pmk}$$

Admitancia total paralelo de la línea entre los nodos k y m.

Para el caso de líneas cortas

$$\bar{y}_{pkm} = \bar{y}_{pmk} = 0$$

1.4.3. Transformadores

Los transformadores de potencia con taps que pueden ser movidos fuera de su posición nominal, pueden ser modelados como se muestra en la figura N° 1.4., es decir la impedancia de cortocircuito \bar{z}_{cc} o admitancia de cortocircuito \bar{y}_{cc} del transformador en serie con un autotransformador ideal de relación $a:1$ donde a es la posición del tap en por unidad, sin embargo el modelo que debe ser usado para el flujo de potencia es el de un circuito equivalente en π como se muestra en la figura N° 1.5., para obtener este modelo procedemos de la siguiente forma:

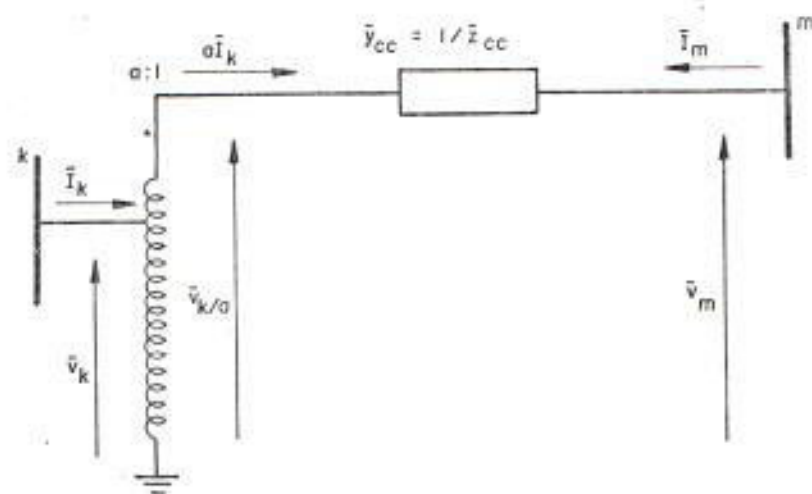


FIGURA N° 1.4. REPRESENTACION DEL TRANSFORMADOR CON TAP FUERA DE SU POSICION NOMINAL.

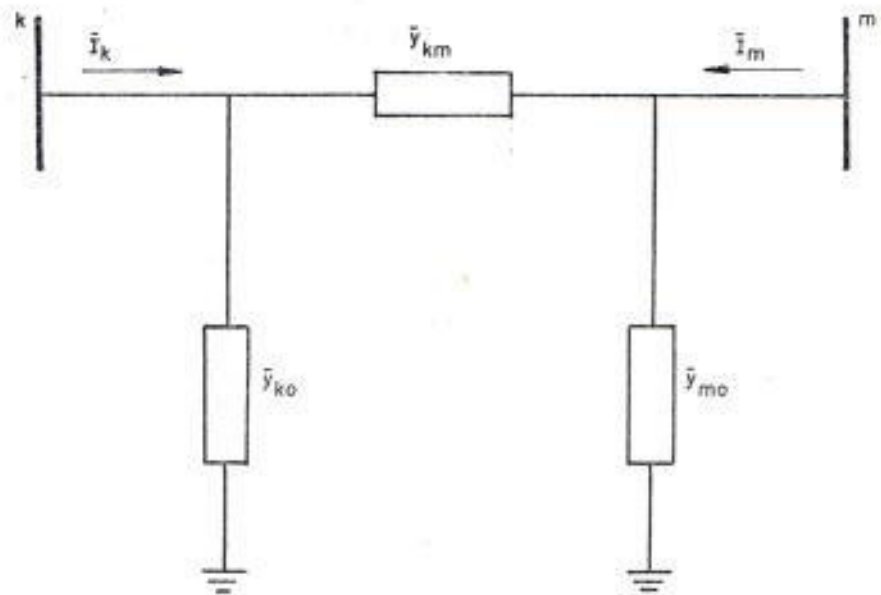


FIGURA N° 1.5. EQUIVALENTE EN π DEL TRANSFORMADOR

$$a\bar{I}_k = \left(\frac{\bar{V}_k}{a} - \bar{V}_m\right) \bar{y}_{cc}$$

$$a\bar{I}_k = \frac{\bar{y}_{cc}}{a} \bar{V}_k - \bar{y}_{cc} \bar{V}_m$$

$$\bar{I}_k = \frac{\bar{y}_{cc}}{a^2} \bar{V}_k - \frac{\bar{y}_{cc}}{a} \bar{V}_m$$

$$\bar{I}_m = -a\bar{I}_k$$

$$\bar{I}_m = -\frac{\bar{y}_{cc}}{a} \bar{V}_k + \bar{y}_{cc} \bar{V}_m$$

Del equivalente π tenemos que:

$$\bar{I}_k = \bar{y}_{ko} \bar{V}_k + (\bar{V}_k - \bar{V}_m) \bar{y}_{km}$$

$$\bar{I}_k = (\bar{y}_{ko} + \bar{y}_{km}) \bar{V}_k - \bar{y}_{km} \bar{V}_m$$

$$\bar{I}_m = (\bar{V}_m - \bar{V}_k) \bar{y}_{km} + \bar{y}_{mo} \bar{V}_m$$

$$\bar{I}_m = -\bar{y}_{km} \bar{V}_k + (\bar{y}_{km} + \bar{y}_{mo}) \bar{V}_m$$

Con lo que tendremos los siguientes pares de ecuaciones:

$$\bar{I}_k = \frac{\bar{y}_{cc}}{a^2} \bar{V}_k - \frac{\bar{y}_{cc}}{a} \bar{V}_m$$

$$\bar{I}_m = -\frac{\bar{y}_{cc}}{a} \bar{V}_k + \bar{y}_{cc} \bar{V}_m$$

$$\bar{I}_k = (\bar{y}_{ko} + \bar{y}_{km}) \bar{V}_k - \bar{y}_{km} \bar{V}_m$$

$$\bar{I}_m = -\bar{y}_{km} \bar{V}_k + (\bar{y}_{km} + \bar{y}_{mo}) \bar{V}_m$$

Comparando los coeficientes de ambos pares de ecuaciones tenemos:

$$\bar{y}_{km} = \frac{\bar{y}_{cc}}{a}$$

$$\bar{y}_{ko} = \frac{1}{a} \left(\frac{1}{a} - 1 \right) \bar{y}_{cc}$$

$$\bar{y}_{mo} = \left(1 - \frac{1}{a} \right) \bar{y}_{cc}$$

Luego el modelo del transformador con tap cambiabile fuera de la posición nominal queda como está mostrado en la figura N° 1.6.

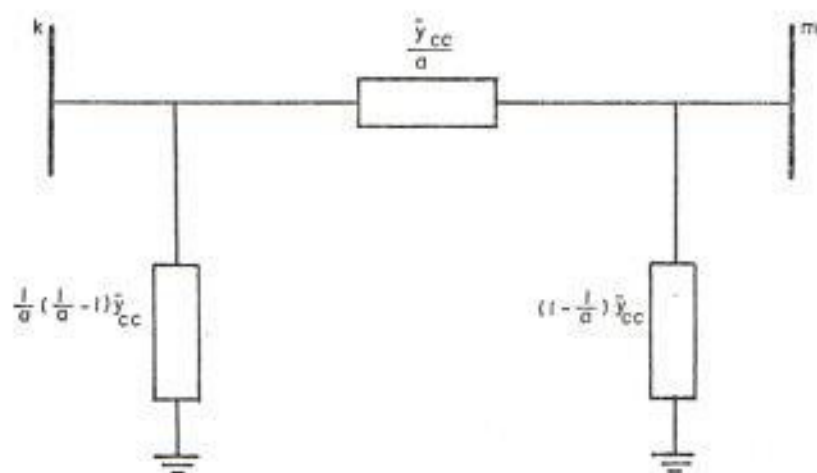


FIGURA N° 1.6. MODELO DEL TRANSFORMADOR CON TAP CAMBIABLE FUERA DE SU POSICIÓN NOMINAL

Si se trata de un transformador desfasador, es decir con una relación de transformación compleja $\bar{a} = \alpha + j\beta = a(\cos \theta + j \sin \theta)$, el modelo puede ser representado como en la figura N° 1.7., es de hacer notar que aquí la relación entre voltajes

y corrientes primarias y secundarias son diferentes, y que además tiene un comportamiento no bilateral es decir $\bar{y}_{km} \neq \bar{y}_{mk}$

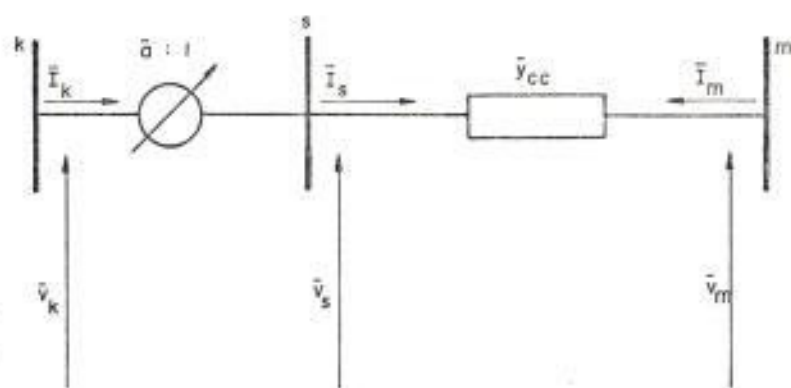


FIGURA N° 1.7. REPRESENTACION DE UN TRANSFORMADOR DESFASADOR.

Donde:

$$\bar{a}_v = \bar{a} = \alpha + j\beta = \frac{\bar{V}_k}{\bar{V}_s}$$

Entre el lado k y s tenemos un transformador desfasador ideal, por lo cual $\bar{V}_k \bar{I}_k^* = \bar{V}_s \bar{I}_s^*$ de donde:

$$\frac{\bar{I}_k}{\bar{I}_s} = \left(\frac{\bar{V}_s}{\bar{V}_k} \right)^* = \left(\frac{1}{\bar{a}} \right)^* = \frac{1}{\bar{a}^*} = \frac{1}{\alpha - j\beta} = \frac{1}{\bar{a}_i}$$

Es decir:

$$\bar{a}_i = \alpha - j\beta$$

$$\bar{a}_v = \alpha + j\beta$$

De aquí:

$$\bar{I}_s = \bar{a}_i \bar{I}_k$$

$$\bar{V}_s = \frac{\bar{V}_k}{\bar{a}_v}$$

De la figura N° 1.7.

$$\bar{I}_s = (\bar{V}_s - \bar{V}_m) \bar{y}_{cc}$$

$$\bar{a}_i \bar{I}_k = \frac{\bar{V}_k}{\bar{a}_v} \bar{y}_{cc} - \bar{V}_m \bar{y}_{cc}$$

$$\bar{I}_k = \frac{\bar{y}_{cc}}{\bar{a}_v \bar{a}_i} \bar{V}_k - \frac{\bar{y}_{cc}}{\bar{a}_i} \bar{V}_m \quad (1.1)$$

$$\bar{I}_m = -\bar{I}_s = -\bar{a}_i \bar{I}_k$$

$$\bar{I}_m = -\frac{\bar{y}_{cc}}{\bar{a}_v} \bar{V}_k + \bar{y}_{cc} \bar{V}_m \quad (1.2)$$

Las ecuaciones (1.1) y (1.2) pueden ser colocadas como:

$$\begin{bmatrix} \bar{I}_k \\ \bar{I}_m \end{bmatrix} = \begin{bmatrix} \frac{\bar{y}_{cc}}{\bar{a}_v \bar{a}_i} & - \frac{\bar{y}_{cc}}{\bar{a}_i} \\ - \frac{\bar{y}_{cc}}{\bar{a}_v} & \bar{y}_{cc} \end{bmatrix} \begin{bmatrix} \bar{V}_k \\ \bar{V}_m \end{bmatrix} \quad (1.3)$$

De la ecuación (1.3), vemos que en el modelo del desfaseador se pierde la bilateralidad es decir la simetría.

Luego del equivalente π de la figura N° 1.5., tenemos:

$$\bar{I}_k = \bar{y}_{ko} \bar{V}_k + (\bar{V}_k - \bar{V}_m) \bar{y}_{km}$$

$$\bar{I}_k = (\bar{y}_{ko} + \bar{y}_{km}) \bar{V}_k - \bar{y}_{km} \bar{V}_m \quad (1.4)$$

$$\bar{I}_m = (\bar{V}_m - \bar{V}_k) \bar{y}_{mk} + \bar{y}_{mo} \bar{V}_m$$

$$\bar{I}_m = -\bar{y}_{mk} \bar{V}_k + (\bar{y}_{mo} + \bar{y}_{mk}) \bar{V}_m \quad (1.5)$$

Comparando las ecuaciones (1.1) con (1.4) y (1.2) -

con (1.5), respectivamente determinamos los parámetros del equivalente π para el transformador desfasador.

$$\bar{y}_{km} = \frac{\bar{y}_{cc}}{\bar{a}_i}$$

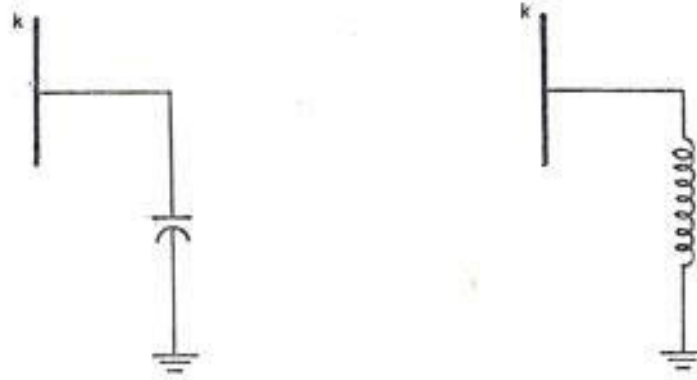
$$\bar{y}_{mk} = \frac{\bar{y}_{cc}}{\bar{a}_v}$$

$$\bar{y}_{ko} = \frac{1}{\bar{a}_i} \left(\frac{1}{\bar{a}_v} - 1 \right) \bar{y}_{cc}, \quad \bar{y}_{mo} = \left(1 - \frac{1}{\bar{a}_v} \right) \bar{y}_{cc}$$

1.4.4. Capacitores y Reactores en paralelo

Los capacitores en paralelo son considerados como fuente de reactivo en el problema del flujo de potencia, usualmente vienen dados en término de sus MVAR nominales y su representación se muestra en la figura N° 1.8.a.

En cambio los reactores en paralelo son modelados como cargas que consumen potencia reactiva, también viene dado en términos de sus MVAR nominales pero considerados con un signo negativo, su representación se muestra en la figura N° 1.8.b.



(a) Capacitor

(b) Reactor

FIGURA N° 1.8. MODELO DE CAPACITORES Y REACTORES EN PARALELO

CARGAS

Las cargas en un sistema de potencia en estado estable son normalmente especificadas por sus requerimientos de potencia activa y reactiva, suponiendo que estas permanecen constantes, a las pequeñas variaciones de voltaje y frecuencia esperados durante la operación normal de estado estable.

La carga conectada a una barra k cualquiera viene representada como se muestra en la figura N° 1.9.



FIGURA N° 1.9. REPRESENTACION DE LA CARGA EN UN SISTEMA DE POTENCIA.

CAPITULO II

ANALISIS DEL PROBLEMA DEL FLUJO DE POTENCIA

2.1. PLANTEAMIENTO DEL PROBLEMA

En breve resumen el problema del flujo de potencia consiste en la determinación del perfil de voltaje, tanto en magnitud y ángulo sobre cada una de las barras de un sistema de potencia y la determinación de los flujos de potencia sobre cada uno de los ramales de la red, satisfaciendo las condiciones impuestas sobre cada barra, como son generación y carga, y en el presente caso satisfaciendo restricciones de reactivo y voltaje.

A manera de ejemplo escojamos un sistema de potencia de cinco barras como el mostrado en la figura N° 2.1.

Para la solución de un sistema de potencia se definen tres tipos de barras, en las cuales se especificarán dos de las siguientes 4 variables P_k , Q_k , V_k , δ_k , donde:

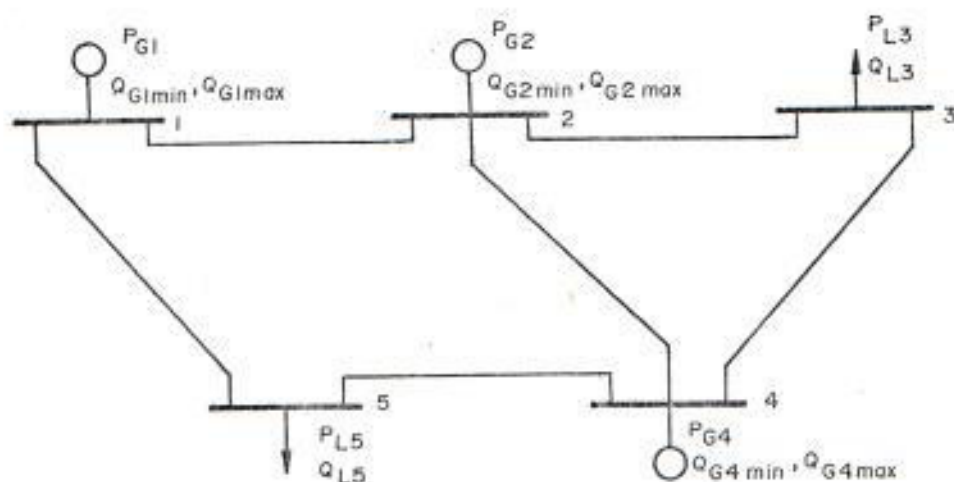


FIGURA Nº 2.1. SISTEMA DE POTENCIA DE 5 BARRAS

P_k es la potencia neta inyectada en la barra k y que se define como:

$$P_k = P_G - P_L$$

Donde:

P_G es la potencia activa generada en la barra k .

P_L potencia activa de carga conectada a la barra k .

Q_k es la potencia reactiva neta inyectada en la barra k y que se define como:

$$Q_k = Q_G - Q_L$$

Donde:

Q_G Potencia reactiva generada en la barra k , y

- Q_L potencia reactiva consumida por una carga en la barra k .
- V_k es la magnitud del voltaje en la barra k con respecto al nodo de referencia que es el neutro del sistema.
- δ_k es el ángulo de fase del voltaje de una barra k cualquiera con respecto a la barra oscilante.

Cada una de las barras del sistema de potencia se clasifica en una de las tres siguientes categorías, de acuerdo a las cantidades especificadas y son:

Barra oscilante: En ella se definen y se mantienen fijas la magnitud y ángulo del voltaje y son incógnitas P y Q .

El objeto de esta barra oscilante es compensar la diferencia de potencia producida entre la generación y carga del sistema, debido a las pérdidas en el sistema de transmisión.

Barra de Generación o Voltaje Controlado: En esta barra se especifican la magnitud del voltaje V_k y la potencia activa P_k y son incógnita el ángulo del voltaje δ_k y el reactivo Q_k .

Barra de carga: Aquí se especifican P_k y Q_k y son incógnitas V_k y δ_k .

Entonces en cada barra del sistema de potencia se puede

plantear la siguiente ecuación (2.1) que constituirán un conjunto de ecuaciones no lineales por resolver:

$$P_k + j Q_k = \bar{V}_k I_k^* \quad (2.1)$$

Donde:

I_k es la corriente inyectada en barra k y se relaciona con los voltajes de barra a través de la siguiente ecuación (2.2).

$$I_k = \sum_{m=1}^n \bar{Y}_{km} \bar{V}_m \quad (2.2)$$

El símbolo $\bar{}$ sobre una cantidad denota una cantidad compleja y el asterisco denota el conjugado de esa cantidad.

Si planteamos la ecuación para cada barra, se obtiene el siguiente sistema de ecuaciones: (2.2)

$$\begin{aligned} I_1 &= \bar{Y}_{11} \bar{V}_1 + \bar{Y}_{12} \bar{V}_2 + \dots + \bar{Y}_{1n} \bar{V}_n \\ I_2 &= \bar{Y}_{21} \bar{V}_1 + \bar{Y}_{22} \bar{V}_2 + \dots + \bar{Y}_{2n} \bar{V}_n \\ &\vdots \\ I_n &= \bar{Y}_{n1} \bar{V}_1 + \bar{Y}_{n2} \bar{V}_2 + \dots + \bar{Y}_{nn} \bar{V}_n \end{aligned} \quad (2.3)$$

En forma matricial:

$$\begin{bmatrix} \bar{I}_1 \\ \bar{I}_2 \\ \cdot \\ \cdot \\ \bar{I}_n \end{bmatrix} = \begin{bmatrix} \bar{Y}_{11} & \bar{Y}_{12} & \dots & \bar{Y}_{1n} \\ \bar{Y}_{21} & \bar{Y}_{22} & \dots & \bar{Y}_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \bar{Y}_{n1} & \bar{Y}_{n2} & \dots & \bar{Y}_{nn} \end{bmatrix} \begin{bmatrix} \bar{V}_1 \\ \bar{V}_2 \\ \cdot \\ \cdot \\ \bar{V}_n \end{bmatrix} \quad (2.4)$$

Expresando (2.4) en una notación más reducida tenemos :

$$|\bar{I}| = |\bar{Y}| |\bar{V}| \quad (2.5)$$

$|\bar{I}|$ es el vector de corrientes inyectadas del sistema.

$|\bar{V}|$ es el vector de voltajes de barra.

$|\bar{Y}|$ es la matriz admitancia de barra.

Los elementos de la diagonal de la matriz admitancia de barra \bar{Y}_{kk} están definidos como la suma de todas las admitancias de rama conectadas a la barra k es decir :

$$\bar{Y}_{kk} = \sum_{m \in k} \bar{y}_{km}$$

Donde:

$m \in k$ significa todos los ramales conectados a la

barra k inclusive los ramales paralelo entre la barra k y el neutro del sistema.

Los elementos \bar{Y}_{km} fuera de la diagonal de la matriz - admitancia de barra están definidos como el negativo de la admitancia del ramal conectado entre la barra k y m, es decir:

$$\bar{Y}_{km} = -\bar{y}_{km}$$

\bar{Y}_{km} será diferente de cero si dicho ramal o línea entre la barra k y m existe, de lo contrario será cero.

Reemplazando (2.2) en (2.1) tendremos:

$$P_k + jQ_k = \bar{V}_k \sum_{m=1}^n \bar{Y}_{km} \bar{V}_m^* \quad (2.6)$$

Donde:

$$\bar{V}_m = V_m e^{j\delta_m} = e_m + jf_m$$

$$\bar{V}_k = V_k e^{j\delta_k} = e_k + jf_k$$

$$\bar{Y}_{km} = Y_{km} e^{j\theta_{km}} = G_{km} + jB_{km}$$

Entonces para resolver el problema del flujo de poten

cia debemos resolver un juego de ecuaciones de la forma (2.6) que como son no lineales deberán resolverse mediante un proceso iterativo.

2.2. ALTERNATIVAS DE SOLUCION

El problema del flujo de potencia puede ser resuelto por varios métodos numéricos diferentes, los más importantes han sido los métodos de:

- Gauss
- Gauss - Seidel
- Newton - Raphson

A su vez, dentro de ellos hay diferentes variantes, todos ellos son métodos iterativos, ya que deben resolver un sistema de ecuaciones no lineales como (2.6), en cambio el método de Newton - Raphson a diferencia de los otros dos debe resolver adicionalmente un segundo sistema de ecuaciones lineales que se verán más adelante, y que son en realidad un sistema de ecuaciones variacionales formado a partir del sistema de ecuaciones no lineales por resolver y obtenido mediante diferenciación total de las variables.

Para la resolución de los sistemas de ecuaciones no lineales, debemos usar los llamados métodos iterativos, o

sea que resuelven las ecuaciones no lineales en forma indirecta mediante aproximaciones sucesivas hacia la solución, en cambio para la resolución del sistema de ecuaciones lineales podemos usar los llamados métodos directos, que consisten en obtener una solución directa de las incógnitas (sin iteración), estos métodos son los de eliminación de Gauss, factorización LU o mediante inversión de matrices.

El método de Newton - Raphson, es entonces un método en el cual el sistema de ecuaciones no lineales es resuelto en forma iterativa, pero en base a las correcciones realizadas a las incógnitas y obtenidas dichas correcciones mediante un método directo para la resolución del sistema de ecuaciones lineales, por lo cual este método tiene una mejor característica de convergencia que los dos primeros métodos; por esto a continuación se explicará este método.

2.3. EL METODO DE NEWTON - RAPHSON

El método de Newton - Raphson, es uno de los métodos ampliamente usados para la resolución de ecuaciones no lineales.

Este transforma el problema original no lineal en una secuencia de problemas lineales cuyas soluciones se aproximan a la solución del problema original. El método puede ser aplicado a una ecuación con una incógnita o a un sistema de ecuaciones simultáneas con tantas incógnitas como ecuaciones.

2.3.1. Caso de una Dimensión

Sea $F(x)$ una ecuación no lineal. Cualquier valor de x que satisfice $F(x) = 0$ es una raíz de $F(x)$. A fin de hallar una raíz particular, un valor inicial asumido para x en la vecindad de la raíz es necesitado.

Asumiendo que esta estimación inicial es x_0 , tendremos:

$$F(x_0) = \Delta F_0$$

Donde:

ΔF_0 es el residuo o error debido al hecho que x_0 no es la raíz.

Esta situación es mostrada graficamente en la figura N° 2.2.

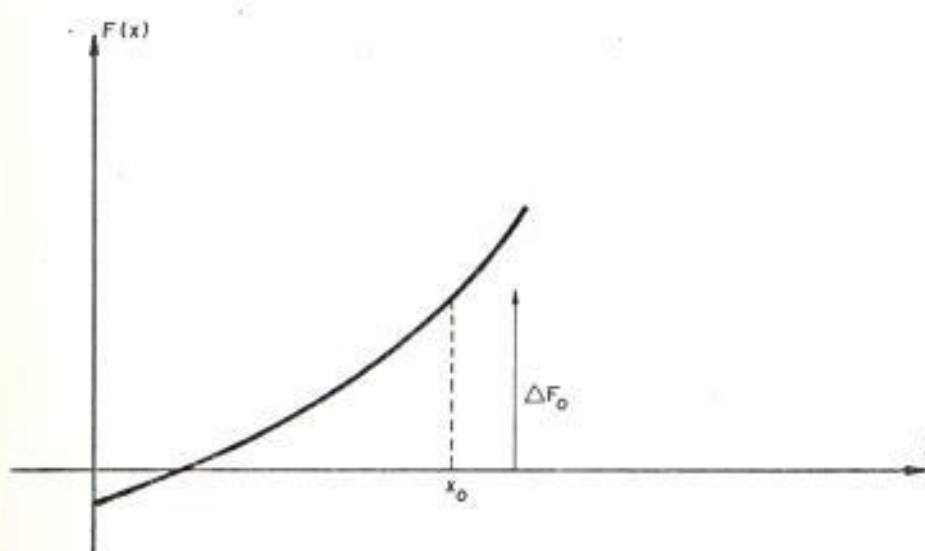


FIGURA N°2.2. ECUACION NO LINEAL EVALUADA EN SU ESTIMACION INICIAL

Si una tangente es trazada en el punto en la curva correspondiente a x_0 , la relación geométrica es como la que se muestra en la figura N^o 2.3., donde Δx es negativo como se muestra.

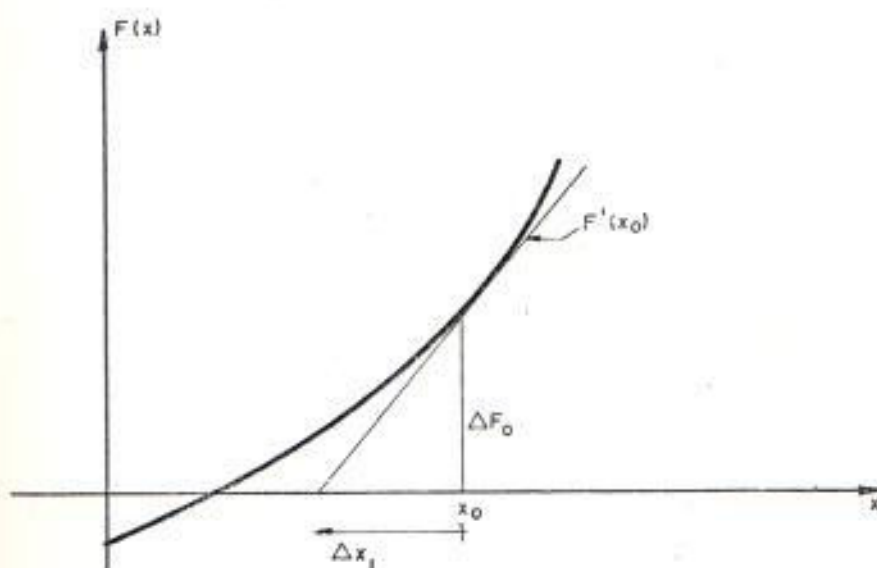


FIGURA N^o 2.3. PRIMERA APROXIMACION A LA SOLUCION DE LA ECUACION NO LINEAL $F(x)$

$F'(x_0)$ es la pendiente de la tangente sobre x_0 luego:

$$F'(x_0) = - \frac{\Delta F_0}{\Delta x_1}$$

El signo negativo es necesitado debido a que $F'(x)$ debe ser positivo cuando Δx es negativo.

Esto conduce directamente a un esquema iterativo como el mostrado en la figura N^o 2.4.

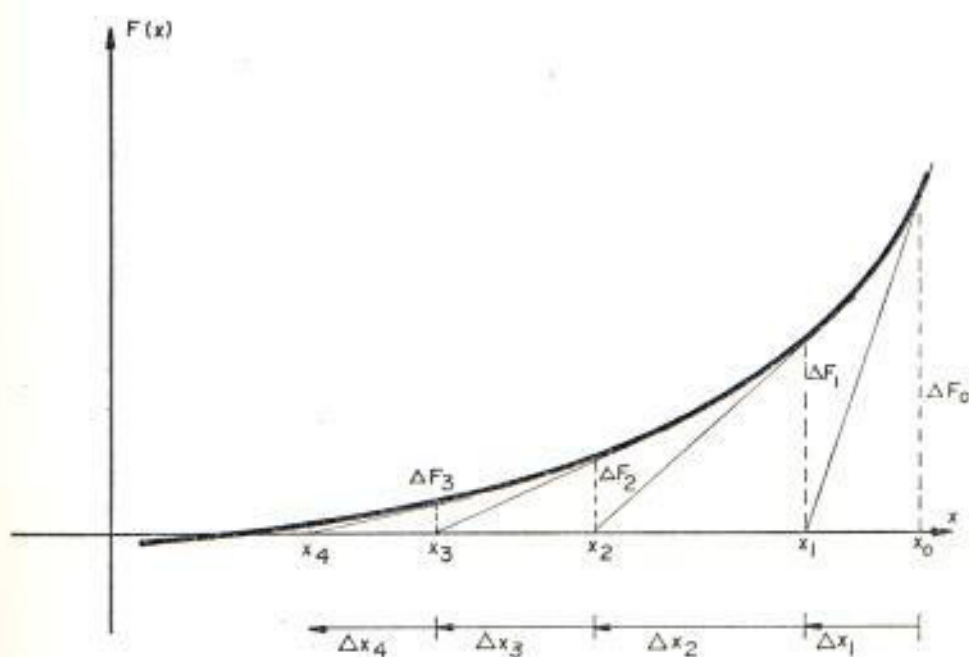


FIGURA N^o 2.4. REPRESENTACION DE ESQUEMA ITERATIVO DE SOLUCION DE LA ECUACION NO LINEAL $F(x)$

Donde:

$$F'(x_0) = -\frac{\Delta F_0}{\Delta x_1}$$

$$F'(x_1) = -\frac{\Delta F_1}{\Delta x_2}$$

$$\cdot \quad \cdot$$

$$F'(x_{h-1}) = -\frac{\Delta F_{h-1}}{\Delta x_h}$$

Luego:

$$\Delta x_h = - \frac{\Delta F_{h-1}}{F'(x_{h-1})} \quad , \quad x_h = x_{h-1} + \Delta x_h$$

Luego la raíz será:

$$x = x_0 + \Delta x_1 + \Delta x_2 + \Delta x_3 + \dots$$

A continuación se da un ejemplo numérico.

Determinaremos la raíz de $F(x) = x^3 - 27$ mediante el método de Newton - Raphson, de antemano nosotros sabemos que es 3.

Asumimos un valor inicial $x_0 = 5$

$$\Delta F_0 = 125 - 27 = 98$$

$$F'(x_0) = 3(5)^2 = 75$$

$$x_1 = x_0 - \frac{98}{75} = 5 - \frac{98}{75} = 3.7$$

$$x_2 = x_1 - \frac{23}{41} = 3.7 - 0.56 = 3.14$$

$$x_3 = x_2 - \frac{4}{30} = 3.14 - 0.08 = 3.06$$

Otra variación del algoritmo básico de Newton - Raphson requiere solo una evaluación de la derivada, $F'(x_0)$ es usado para cada paso en lugar de $F'(x_n)$, la situación es mostrada graficamente en la figura N° 2.5., la tangente inicial se convierte en una secante de igual pendiente - en las subsiguientes iteraciones.

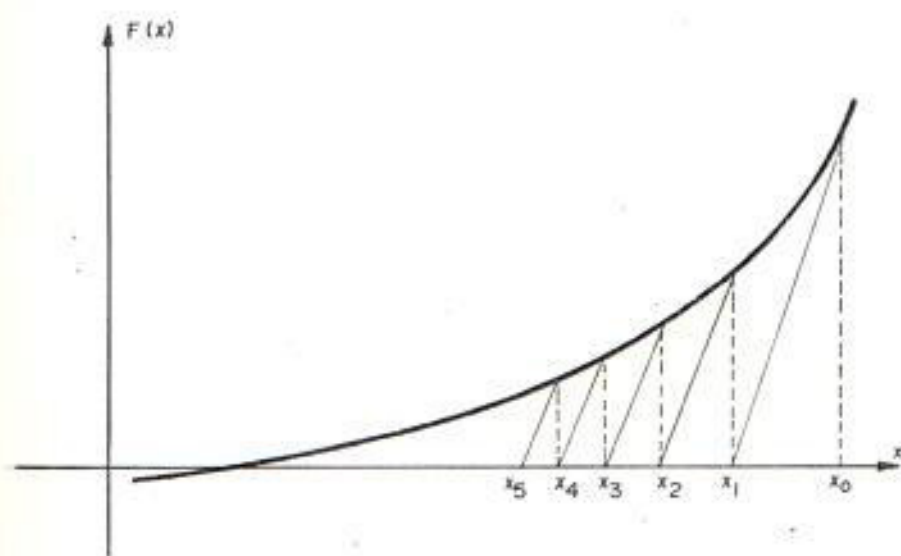
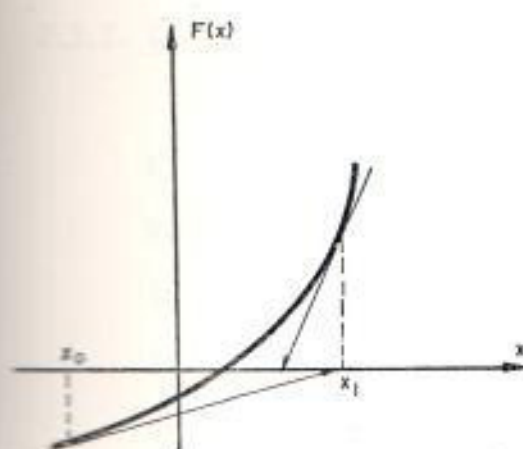
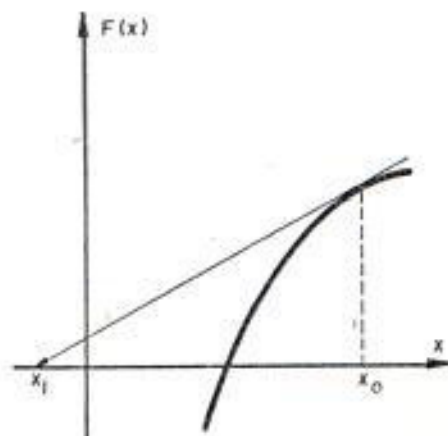


FIGURA N° 2.5. REPRESENTACION DEL ESQUEMA ITERATIVO DE SOLUCION DE $F(x)$ MEDIANTE UNA APROXIMACION CON PENDIENTE FIJA.

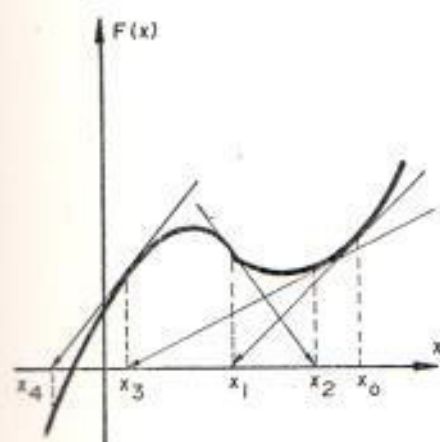
En la siguiente página podremos observar la figura N° 2.6., la misma que está compuesta por cuatro partes. En esta figura se muestran graficamente algunas de las posibilidades de convergencia con el algoritmo de Newton - Raphson.



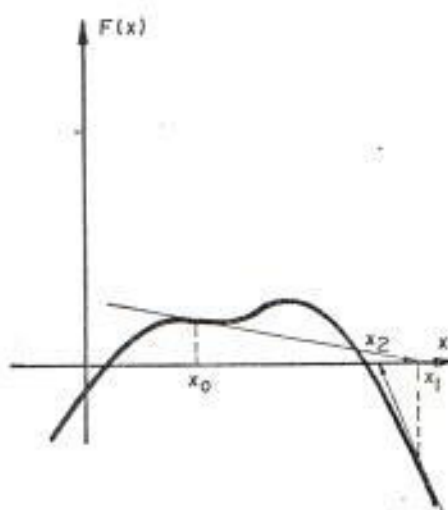
(a) Convergencia oscilatoria



(b) No tiene solución convergente



(c) Convergencia oscilatoria



(d) Convergencia a una raíz no deseada.

FIGURA Nº 2.6. POSIBLES CARACTERISTICAS DE CONVERGENCIA DEL METODO DE NEWTON.

2.3.2. Caso de n dimensiones

El concepto de una dimensión del método de Newton-Raphson puede ser extendido a n dimensiones, todo lo que necesitamos es un análogo n dimensional de la primera derivada (pendiente de la función para el caso unidimensional), para esto supongamos que tenemos n ecuaciones con n incógnitas (2.7).

$$\begin{aligned}
 F_1(x_1, x_2, \dots, x_n) &= 0 \\
 F_2(x_1, x_2, \dots, x_n) &= 0 \\
 \cdot &\cdot \\
 \cdot &\cdot \\
 \cdot &\cdot \\
 F_n(x_1, x_2, \dots, x_n) &= 0 \qquad (2.7)
 \end{aligned}$$

En una aproximación inicial $x_1^0, x_2^0, \dots, x_n^0$ serían la solución de (2.7), con un error $\Delta x_1^1, \Delta x_2^1, \dots, \Delta x_n^1$, luego:

$$\begin{aligned}
 F_1(x_1^0 + \Delta x_1^1, x_2^0 + \Delta x_2^1, \dots, x_n^0 + \Delta x_n^1) &= 0 \\
 F_2(x_1^0 + \Delta x_1^1, x_2^0 + \Delta x_2^1, \dots, x_n^0 + \Delta x_n^1) &= 0 \\
 \cdot &\cdot \\
 \cdot &\cdot \\
 \cdot &\cdot \\
 F_n(x_1^0 + \Delta x_1^1, x_2^0 + \Delta x_2^1, \dots, x_n^0 + \Delta x_n^1) &= 0 \qquad (2.8)
 \end{aligned}$$

Aplicando la expansión de la serie de Taylor a (2.8) y despreciando las derivadas de orden superior a la primera derivada nos queda:

$$F_1(x_1^0, x_2^0, \dots, x_n^0) +$$

$$\frac{\partial F_1}{\partial x_1} \bigg|_{x_1^0} \Delta x_1^1 + \frac{\partial F_1}{\partial x_2} \bigg|_{x_2^0} \Delta x_2^1 + \dots + \frac{\partial F_1}{\partial x_n} \bigg|_{x_n^0} \Delta x_n^1 = 0$$

$$F_2(x_1^0, x_2^0, \dots, x_n^0) +$$

$$\frac{\partial F_2}{\partial x_1} \bigg|_{x_1^0} \Delta x_1^1 + \frac{\partial F_2}{\partial x_2} \bigg|_{x_2^0} \Delta x_2^1 + \dots + \frac{\partial F_2}{\partial x_n} \bigg|_{x_n^0} \Delta x_n^1 = 0$$

·
·
·

$$F_n(x_1^0, x_2^0, \dots, x_n^0) + \tag{2.9}$$

$$\frac{\partial F_n}{\partial x_1} \bigg|_{x_1^0} \Delta x_1^1 + \frac{\partial F_n}{\partial x_2} \bigg|_{x_2^0} \Delta x_2^1 + \dots + \frac{\partial F_n}{\partial x_n} \bigg|_{x_n^0} \Delta x_n^1 = 0$$

$$\frac{\partial F_1}{\partial x_1} \bigg|_{x_1^0} \Delta x_1^1 + \frac{\partial F_1}{\partial x_2} \bigg|_{x_2^0} \Delta x_2^1 + \dots + \frac{\partial F_1}{\partial x_n} \bigg|_{x_n^0} \Delta x_n^1 =$$

$$= -F_1(x_1^0, x_2^0, \dots, x_n^0)$$

$$\frac{\partial F_2}{\partial x_1} \bigg|_{x_1^0} \Delta x_1^1 + \frac{\partial F_2}{\partial x_2} \bigg|_{x_2^0} \Delta x_2^1 + \dots + \frac{\partial F_2}{\partial x_n} \bigg|_{x_n^0} \Delta x_n^1 =$$

$$= -F_2(x_1^0, x_2^0, \dots, x_n^0)$$

$$\frac{\partial F_n}{\partial x_1} \bigg|_{x_1^0} \Delta x_1^1 + \frac{\partial F_n}{\partial x_2} \bigg|_{x_2^0} \Delta x_2^1 + \dots + \frac{\partial F_n}{\partial x_n} \bigg|_{x_n^0} \Delta x_n^1 =$$

$$(2.10)$$

$$= -F_n(x_1^0, x_2^0, \dots, x_n^0)$$

La función F_i evaluada en los valores iniciales - $x_1^0, x_2^0, \dots, x_n^0$, nos da el residuo o error entre la función evaluada o calculada y el valor esperado de la función para la raíz (cero) ya que $x_1^0, x_2^0, \dots, x_n^0$, no son la solución (raíz) si no una aproximación, luego:

$$\Delta F_1^0 = -F_1(x_1^0, x_2^0, \dots, x_n^0)$$

$$\Delta F_2^0 = -F_2(x_1^0, x_2^0, \dots, x_n^0)$$

$$\cdot \quad \cdot \quad \cdot$$

$$\cdot \quad \cdot \quad \cdot$$

$$\cdot \quad \cdot \quad \cdot$$

$$\Delta F_n^0 = -F_n(x_1^0, x_2^0, \dots, x_n^0) \quad (2.11)$$

Reemplazando (2.11) en (2.10) tendremos:

$$\begin{array}{l}
 \frac{\partial F_1}{\partial x_1} \left| \begin{array}{c} \Delta x_1^1 \\ x_1^0 \end{array} \right. + \frac{\partial F_1}{\partial x_2} \left| \begin{array}{c} \Delta x_2^1 \\ x_2^0 \end{array} \right. + \dots + \frac{\partial F_1}{\partial x_n} \left| \begin{array}{c} \Delta x_n^1 \\ x_n^0 \end{array} \right. = \Delta F_1^0 \\
 \frac{\partial F_2}{\partial x_1} \left| \begin{array}{c} \Delta x_1^1 \\ x_1^0 \end{array} \right. + \frac{\partial F_2}{\partial x_2} \left| \begin{array}{c} \Delta x_2^1 \\ x_2^0 \end{array} \right. + \dots + \frac{\partial F_2}{\partial x_n} \left| \begin{array}{c} \Delta x_n^1 \\ x_n^0 \end{array} \right. = \Delta F_2^0 \\
 \cdot \\
 \cdot \\
 \cdot \\
 \frac{\partial F_n}{\partial x_1} \left| \begin{array}{c} \Delta x_1^1 \\ x_1^0 \end{array} \right. + \frac{\partial F_n}{\partial x_2} \left| \begin{array}{c} \Delta x_2^1 \\ x_2^0 \end{array} \right. + \dots + \frac{\partial F_n}{\partial x_n} \left| \begin{array}{c} \Delta x_n^1 \\ x_n^0 \end{array} \right. = \Delta F_n^0
 \end{array}$$

(2.12)

Colocando (2.12) en forma matricial:

$$\begin{bmatrix} \frac{\partial F_1}{\partial x_1} \left| \begin{array}{c} \Delta x_1^1 \\ x_1^0 \end{array} \right. & \frac{\partial F_1}{\partial x_2} \left| \begin{array}{c} \Delta x_2^1 \\ x_2^0 \end{array} \right. & \dots & \frac{\partial F_1}{\partial x_n} \left| \begin{array}{c} \Delta x_n^1 \\ x_n^0 \end{array} \right. \\
 \frac{\partial F_2}{\partial x_1} \left| \begin{array}{c} \Delta x_1^1 \\ x_1^0 \end{array} \right. & \frac{\partial F_2}{\partial x_2} \left| \begin{array}{c} \Delta x_2^1 \\ x_2^0 \end{array} \right. & \dots & \frac{\partial F_2}{\partial x_n} \left| \begin{array}{c} \Delta x_n^1 \\ x_n^0 \end{array} \right. \\
 \cdot & \cdot & \cdot & \cdot \\
 \cdot & \cdot & \cdot & \cdot \\
 \cdot & \cdot & \cdot & \cdot \\
 \frac{\partial F_n}{\partial x_1} \left| \begin{array}{c} \Delta x_1^1 \\ x_1^0 \end{array} \right. & \frac{\partial F_n}{\partial x_2} \left| \begin{array}{c} \Delta x_2^1 \\ x_2^0 \end{array} \right. & \dots & \frac{\partial F_n}{\partial x_n} \left| \begin{array}{c} \Delta x_n^1 \\ x_n^0 \end{array} \right. \end{bmatrix} \begin{bmatrix} \Delta x_1^1 \\ \Delta x_2^1 \\ \cdot \\ \cdot \\ \cdot \\ \Delta x_n^1 \end{bmatrix} = \begin{bmatrix} \Delta F_1^0 \\ \Delta F_2^0 \\ \cdot \\ \cdot \\ \cdot \\ \Delta F_n^0 \end{bmatrix}$$

(2.13)

En una notación matricial más reducida tendremos:

$$|J^{(0)}| \quad |\Delta x^{(1)}| = |\Delta F^{(0)}| \quad (2.14)$$

Donde la matriz de las derivadas parciales de primer orden es la matriz jacobiano y es el análogo representativo en n dimensiones de la pendiente de la función en una dimensión, en las n dimensiones la matriz jacobiano $|J|$ representa la pendiente de los hiperplanos tangente⁽²⁾ que aproximan las funciones a la raíz deseada.

El siguiente paso en el método de Newton - Raphson es determinar el vector $|\Delta x|$ en (2.13), esto a través de (2.14) quedaría:

$$|\Delta x^{(1)}| = |J^{(0)}|^{-1} \quad |\Delta F^{(0)}| \quad (2.15)$$

Donde $|J^{(0)}|^{-1}$ es la inversa de la matriz jacobiano evaluado en el valor inicial, luego determinada la primera corrección de las incógnitas $|\Delta x^{(1)}|$, determinamos una nueva aproximación a la solución mediante (2.16).

$$|x^{(1)}| = |x^{(0)}| + |\Delta x^{(1)}| \quad (2.16)$$

Con esta nueva aproximación para $|x|$ repetimos nuevamente el proceso en forma iterativa hasta que el error $|\Delta F|$ esté dentro de una tolerancia dada.

El método de Newton - Raphson convergerá cuadráticamente si las funciones tienen primeras derivadas continuas en la vecindad de la solución.

Colocando (2.14) y (2.16), para una iteración h cualquiera tendremos:

$$|J^{(h-1)}| \quad |\Delta x^{(h)}| = |\Delta F^{(h-1)}| \quad (2.17)$$

$$|x^{(h)}| = |x^{(h-1)}| + |\Delta x^{(h)}| \quad (2.18)$$

El algoritmo es iterado hasta que $\Delta F^{(h)}$ o $\Delta x^{(h)}$ estén dentro de una tolerancia especificada.

2.4. EL METODO DE NEWTON - RAPHSON PARA EL FLUJO DE POTENCIA

El método de Newton - Raphson puede ser aplicado en diferentes maneras para la resolución del problema del flujo de Potencia, así las variantes más conocidas son la forma rectangular y la forma polar, en nuestro caso las ecuaciones de potencia y el Jacobiano serán calculados con el voltaje en coordenadas rectangulares, pero las in

cógnitas que son obtenidas a través de la derivación del método de Newton - Raphson son las correcciones de la magnitud y ángulo del voltaje, desde este punto de vista la solución es hallada en forma polar.

2.4.1. Derivación y análisis de las ecuaciones de Newton-Raphson para el flujo de potencia

En la sección 2.1., pudimos observar que para resolver el problema del flujo de potencia, debemos resolver un conjunto de ecuaciones no lineales como la mostrada en la ecuación (2.6).

$$P_k + jQ_k = \bar{V}_k \sum_{m=1}^n \bar{Y}_{km}^* \bar{V}_m^*$$

Donde:

$$\bar{V}_k = V_k e^{j\delta_k} = e_k + j f_k \quad (2.19)$$

$$\bar{V}_m = V_m e^{j\delta_m} = e_m + j f_m$$

$$\bar{Y}_{km} = Y_{km} e^{j\theta_{km}} = G_{km} + jB_{km}$$

Matemáticamente hablando las ecuaciones complejas de flujo de carga son no analíticas y por lo tanto no pueden ser diferenciadas en forma compleja, por

lo cual a fin de aplicar el método de Newton, esta es separada en sus partes reales e imaginarias para así obtener dos sistemas de ecuaciones, como dijimos anteriormente podemos usar coordenadas polares o rectangulares, así:

$$P_k = P(v, \delta) \quad \text{ó} \quad P(e, f) \quad (2.20) \text{ y } (2.22)$$

$$Q_k = Q(v, \delta) \quad \text{ó} \quad Q(e, f) \quad (2.21) \text{ y } (2.23)$$

Usando coordenadas polares o sea (2.20) y (2.21) de la aplicación del método de Newton obtenemos la relación lineal de cambios en la función P o Q para pequeñas variaciones en las variables v y δ , entonces:

$$\Delta P_k = \sum_{m=1}^n \frac{\partial P_k}{\partial \delta_m} \Delta \delta_m + \sum_{m=1}^n \frac{\partial P_k}{\partial v_m} \Delta v_m \quad (2.24)$$

$$\Delta Q_k = \sum_{m=1}^n \frac{\partial Q_k}{\partial \delta_m} \Delta \delta_m + \sum_{m=1}^n \frac{\partial Q_k}{\partial v_m} \Delta v_m \quad (2.25)$$

Si escogemos el número 1 para la barra oscilante (2.24) y (2.25) pueden ser puestos en forma matricial así:

ΔP_2	$\frac{\partial P_2}{\partial \delta_2}$	$\frac{\partial P_2}{\partial \delta_3}$	\dots	$\frac{\partial P_2}{\partial \delta_n}$	$\frac{\partial P_2}{\partial V_2}$	$\frac{\partial P_2}{\partial V_3}$	\dots	$\frac{\partial P_2}{\partial V_n}$	$\Delta \delta_2$
ΔP_3	$\frac{\partial P_3}{\partial \delta_2}$	$\frac{\partial P_3}{\partial \delta_3}$	\dots	$\frac{\partial P_3}{\partial \delta_n}$	$\frac{\partial P_3}{\partial V_2}$	$\frac{\partial P_3}{\partial V_3}$	\dots	$\frac{\partial P_3}{\partial V_n}$	$\Delta \delta_3$
.
ΔP_n	$\frac{\partial P_n}{\partial \delta_2}$	$\frac{\partial P_n}{\partial \delta_3}$	\dots	$\frac{\partial P_n}{\partial \delta_n}$	$\frac{\partial P_n}{\partial V_2}$	$\frac{\partial P_n}{\partial V_3}$	\dots	$\frac{\partial P_n}{\partial V_n}$	$\Delta \delta_n$
ΔQ_2	$\frac{\partial Q_2}{\partial \delta_2}$	$\frac{\partial Q_2}{\partial \delta_3}$	\dots	$\frac{\partial Q_2}{\partial \delta_n}$	$\frac{\partial Q_2}{\partial V_2}$	$\frac{\partial Q_2}{\partial V_3}$	\dots	$\frac{\partial Q_2}{\partial V_n}$	ΔV_2
ΔQ_3	$\frac{\partial Q_3}{\partial \delta_2}$	$\frac{\partial Q_3}{\partial \delta_3}$	\dots	$\frac{\partial Q_3}{\partial \delta_n}$	$\frac{\partial Q_3}{\partial V_2}$	$\frac{\partial Q_3}{\partial V_3}$	\dots	$\frac{\partial Q_3}{\partial V_n}$	ΔV_3
.
ΔQ_n	$\frac{\partial Q_n}{\partial \delta_2}$	$\frac{\partial Q_n}{\partial \delta_3}$	\dots	$\frac{\partial Q_n}{\partial \delta_n}$	$\frac{\partial Q_n}{\partial V_2}$	$\frac{\partial Q_n}{\partial V_3}$	\dots	$\frac{\partial Q_n}{\partial V_n}$	ΔV_n

(2.26)

Para las barras de voltaje controlado desaparecen los desajustes ΔQ y ΔV (ya que V es especificado y la única incógnita es δ) y las respectivas filas y columnas en la matriz de (2.26).

Poniendo (2.26) en una notación más reducida tendremos:

$$\begin{bmatrix} \Delta P_k \\ \hline \Delta Q_k \end{bmatrix} = \begin{bmatrix} J1_{km} & J2_{km} \\ \hline J3_{km} & J4_{km} \end{bmatrix} \begin{bmatrix} \Delta \delta_m \\ \hline \Delta V_m \end{bmatrix} \quad (2.27)$$

Donde:

$$[J] = \begin{bmatrix} J1_{km} & J2_{km} \\ J3_{km} & J4_{km} \end{bmatrix} \quad \text{es la matriz jacobiano y}$$

$$J1_{km} = \frac{\partial P_k}{\partial \delta_m}$$

$$J2_{km} = \frac{\partial P_k}{\partial V_m}$$

$$J3_{km} = \frac{\partial Q_k}{\partial \delta_m}$$

$$J4_{km} = \frac{\partial Q_k}{\partial V_m}$$

ΔV_m y $\Delta \delta_m$ son las correcciones de la magnitud y ángulo del voltaje (incógnitas).

$P_k = P_k^{esp} - P_k$ Desajuste entre la potencia activa neta especificada y calculada en la barra k .

(2.28)

$Q_k = Q_k^{esp} - Q_k$ Desajuste entre la potencia reactiva neta especificada y calculada en la barra k .

(2.29)

Donde:

$$P_k^{esp} = P_G - P_L$$

$$Q_k^{esp} = Q_G - Q_L$$

Para evaluar los desajustes de potencia activa y reactiva necesitamos calcular P_k y Q_k de (2.6)

$$P_k + jQ_k = \bar{V}_k \sum_{m=1}^n \bar{Y}_{km}^* \bar{V}_m^*$$

$$\bar{I}_k^* = a_k - j b_k = \sum_{m=1}^n \bar{Y}_{km}^* \bar{V}_m^* \quad (2.30)$$

$$\begin{aligned}
 &= \sum_{m=1}^n (G_{km} - jB_{km})(e_m - jf_m) \\
 &= \sum_{m=1}^n (e_m G_{km} - f_m B_{km}) \\
 &\quad - j \sum_{m=1}^n (f_m G_{km} + e_m B_{km})
 \end{aligned}$$

reemplazando (2.30) y (2.19) en (2.6), tendremos:

$$P_k + jQ_k = (e_k + jf_k)(a_k - jb_k)$$

$$P_k + jQ_k = (e_k a_k + f_k b_k) + j(f_k a_k - e_k b_k)$$

Separando en parte real e imaginaria obtenemos:

$$P_k = e_k a_k + f_k b_k$$

$$Q_k = f_k a_k - e_k b_k$$

$$P_k = e_k \sum_{m=1}^n (e_m G_{km} - f_m B_{km}) + f_k \sum_{m=1}^n (f_m G_{km} + e_m B_{km}) \quad (2.31)$$

$$Q_k = f_k \sum_{m=1}^n (e_m G_{km} - f_m B_{km}) - e_k \sum_{m=1}^n (f_m G_{km} + e_m B_{km}) \quad (2.32)$$

Entonces con (2.31) y (2.32), (2.28) y (2.29) podemos evaluar los desajustes de potencia del sistema de poten

cia exceptuando la barra oscilante y las barras de voltaje controlado para (2.29).

Ahora deduciremos los elementos del jacobiano, para lo cual pondremos la ecuación (2.6) en forma exponencial.

$$P_k + jQ_k = V_k e^{j\delta_k} \sum_{m=1}^n Y_{km} e^{-j\theta_{km}} V_m e^{-j\delta_m} \quad (2.33)$$

Primero obtendremos los elementos fuera de la diagonal del Jacobiano, es decir para $k \neq m$, para lo cual derivamos parcialmente (2.33) con respecto a δ_m y V_m , así:

$$\frac{\partial}{\partial \delta_m} (P_k + jQ_k) = -jV_k e^{j\delta_k} Y_{km} e^{-j\theta_{km}} e^{-j\delta_m}$$

$$\frac{\partial P_k}{\partial \delta_m} + j \frac{\partial Q_k}{\partial \delta_m} = -j V_k \bar{Y}_{km}^* \bar{V}_m^*$$

Pero:

$$\bar{V}_m^* = a_m - j b_m = \bar{Y}_{km}^* \bar{V}_m^*, \text{ reemplazando}$$

$$\frac{\partial P_k}{\partial \delta_m} + j \frac{\partial Q_k}{\partial \delta_m} = -j(e_k + j f_k) (a_m - j b_m) \quad (2.34)$$

$$= (f_k a_m - e_k b_m) - j (e_k a_m + f_k b_m)$$

$$\begin{aligned} \frac{\partial (P_k + jQ_k)}{\partial V_m} &= V_k e^{j\delta_k} Y_{km} e^{-j\theta_{km}} e^{-j\delta_m} \\ &= \frac{1}{V_m} V_k e^{j\delta_k} Y_{km} e^{-j\theta_{km}} V_m e^{-j\delta_m} \end{aligned}$$

$$\begin{aligned} \frac{\partial P_k}{\partial V_m} + j \frac{\partial Q_k}{\partial V_m} &= \frac{1}{V_m} \bar{V}_k \bar{Y}_{km}^* \bar{V}_m^* \\ &= \frac{1}{V_m} (e_k + j f_k) (a_m - j b_m) \end{aligned}$$

$$\frac{\partial P_k}{\partial V_m} + j \frac{\partial Q_k}{\partial V_m} = \frac{1}{V_m} (e_k a_m + f_k b_m) + j \frac{1}{V_m} (f_k a_m - e_k b_m) \quad (2.35)$$

De (2.34) y (2.35), tenemos todos los elementos fuera de la diagonal de la matriz jacobiano.

$$J_{1_{km}} = \frac{\partial P_k}{\partial \delta_m} = (f_k a_m - e_k b_m) \quad (2.36)$$

$$J_{2_{km}} = \frac{\partial P_k}{\partial V_m} = \frac{1}{V_m} (e_k a_m + f_k b_m)$$

$$J_{3_{km}} = \frac{\partial Q_k}{\partial \delta_m} = -(e_k a_m + f_k b_m)$$

$$J_{4_{km}} = \frac{\partial Q_k}{\partial V_m} = \frac{1}{V_m} (f_k a_m - e_k b_m)$$

Ahora evaluaremos los elementos de la diagonal de la matriz jacobiano o sea cuando $k = m$

$$\begin{aligned} \frac{\partial}{\partial \delta_k} (P_k + jQ_k) &= j V_k e^{j\delta_k} \sum_{m=1}^n Y_{km} e^{-j\theta_{km}} V_m e^{-j\delta_m} \\ &\quad - j V_k e^{j\delta_k} Y_{kk} e^{-j\theta_{kk}} V_k e^{-j\delta_k} \end{aligned}$$

$$\begin{aligned} \frac{\partial P_k}{\partial \delta_k} + j \frac{\partial Q_k}{\partial \delta_k} &= j (P_k + jQ_k) - j V_k^2 (G_{kk} - j B_{kk}) \\ &= (-Q_k + j P_k) + (-V_k^2 B_{kk} - j V_k^2 G_{kk}) \end{aligned}$$

$$\frac{\partial P_k}{\partial \delta_k} + j \frac{\partial Q_k}{\partial \delta_k} = (-Q_k - V_k^2 B_{kk}) + j(P_k - V_k^2 G_{kk}) \quad (2.37)$$

$$\begin{aligned} \frac{\partial}{\partial V_k} (P_k + jQ_k) &= e^{j\delta_k} \sum_{m=1}^n Y_{km} e^{-j\theta_{km}} V_m e^{-j\delta_m} \\ &\quad + V_k e^{j\delta_k} Y_{kk} e^{-j\theta_{kk}} e^{-j\delta_k} \end{aligned}$$

multiplicando y dividiendo para V_k tenemos:

$$\begin{aligned}
\frac{\partial P_k}{\partial V_k} + j \frac{\partial Q_k}{\partial V_k} &= \frac{1}{V_k} V_k e^{j\delta_k} \sum_{m=1}^n Y_{km} e^{-j\theta_{km}} V_m e^{-j\delta_m} \\
&+ \frac{1}{V_k} V_k e^{j\delta_k} Y_{kk} e^{-j\theta_{kk}} V_k e^{-j\delta_k} \\
&= \frac{1}{V_k} (P_k + j Q_k) + \frac{1}{V_k} V_k^2 (Y_{kk}^*) \\
&= \frac{1}{V_k} (P_k + j Q_k) + \frac{1}{V_k} V_k^2 (G_{kk} - jB_{kk})
\end{aligned}$$

$$\frac{\partial P_k}{\partial V_k} + j \frac{\partial Q_k}{\partial V_k} = \frac{1}{V_k} (P_k + V_k^2 G_{kk}) + j \frac{1}{V_k} (Q_k - V_k^2 B_{kk}) \quad (2.38)$$

De (2.37) y (2.38), tenemos todos los elementos diagonales de la matriz jacobiano y que escribimos a continuación junto con los elementos fuera de la diagonal.

$$J_{1_{km}} = \frac{\partial P_k}{\partial \delta_m} = (f_k a_m - e_k b_m)$$

$$J_{2_{km}} = \frac{\partial P_k}{\partial V_m} = \frac{1}{V_m} (e_k a_m + f_k b_m)$$

$$J_{3_{km}} = \frac{\partial Q_k}{\partial \delta_m} = -(e_k a_m + f_k b_m)$$

$$J_{4_{km}} = \frac{\partial Q_k}{\partial V_m} = \frac{1}{V_m} (f_k a_m - e_k b_m)$$

$$J_{1_{kk}} = \frac{\partial P_k}{\partial \delta_k} = (-Q_k - V_k^2 B_{kk}) \quad (2.39)$$

$$J_{2_{kk}} = \frac{\partial P_k}{\partial V_k} = \frac{1}{V_k} (P_k + V_k^2 G_{kk})$$

$$J_{3_{kk}} = \frac{\partial Q_k}{\partial \delta_k} = (P_k - V_k^2 G_{kk})$$

$$J_{4_{kk}} = \frac{\partial Q_k}{\partial V_k} = \frac{1}{V_k} (Q_k - V_k^2 B_{kk})$$

Observando las ecuaciones (2.36) y (2.39), de la matriz jacobiano notamos que si en los subjacobianos J2 y J4 pasamos la magnitud del voltaje del segundo miembro de la ecuación al lado izquierdo esto produce que:

$$\frac{\partial P_k}{\partial \delta_m} = V_m \frac{\partial Q_k}{\partial V_m}$$

$$\frac{\partial Q_k}{\partial \delta_m} = -V_m \frac{\partial P_k}{\partial V_m}$$

Lo cual es más conveniente ya que con unicamente

evaluar los elementos fuera de la diagonal de los subjacobianos J_1 y J_3 automáticamente quedan determinados los otros dos subjacobianos, modificados para expresar esto, usaremos una nueva notación para los subjacobianos y que es la notación estandar usada en la referencia⁽⁵⁾.

$$H_{km} = \frac{\partial P_k}{\partial \delta_m} = f_k a_m - e_k b_m$$

$$N_{km} = V_m \frac{\partial P_k}{\partial V_m} = e_k a_m + f_k b_m$$

$$J_{km} = \frac{\partial Q_k}{\partial \delta_m} = - (e_k a_m + f_k b_m)$$

$$L_{km} = V_m \frac{\partial Q_k}{\partial V_m} = f_k a_m - e_k b_m \quad (2.40)$$

Nótese de (2.40) que:

$$H_{km} = L_{km}$$

$$N_{km} = -J_{km}$$

$$H_{kk} = \frac{\partial P_k}{\partial \delta_k} = (-Q_k - V_k^2 B_{kk})$$

$$N_{kk} = V_k \frac{\partial P_k}{\partial V_k} = P_k + V_k^2 G_{kk}$$

$$J_{kk} = \frac{\partial Q_k}{\partial \delta_k} = B_k - V_k^2 G_{kk}$$

$$L_{kk} = V_k \frac{\partial Q_k}{\partial V_k} = Q_k - V_k^2 B_{kk} \quad (2.41)$$

Ahora bien si modificamos los subjacobianos como en (2.40) y (2.41), las ecuaciones (2.24) y (2.25), deben ser modificadas para que ΔP_k y ΔQ_k sigan siendo los mismos luego, (2.24) y (2.25) para no ser alteradas deben quedar así:

$$\Delta P_k = \sum_{m=1}^n \frac{\partial P_k}{\partial \delta_m} \Delta \delta_m + \sum_{m=1}^n V_m \frac{\partial P_k}{\partial V_m} \frac{\Delta V_m}{V_m} \quad (2.42)$$

$$\Delta Q_k = \sum_{m=1}^n \frac{\partial Q_k}{\partial \delta_m} \Delta \delta_m + \sum_{m=1}^n V_m \frac{\partial Q_k}{\partial V_m} \frac{\Delta V_m}{V_m} \quad (2.43)$$

reemplazando la notación estandar para los subjacobianos tendremos:

$$\Delta P_k = \sum_{m=1}^n H_{km} \Delta \delta_m + \sum_{m=1}^n N_{km} \frac{\Delta V_m}{V_m} \quad (2.44)$$

$$\Delta Q_k = \sum_{m=1}^n J_{km} \Delta \delta_m + \sum_{m=1}^n L_{km} \frac{\Delta V_m}{V_m} \quad (2.45)$$

Poniendo (2.44) y (2.45) en una notación matricial reducida tenemos:

$$\begin{bmatrix} \Delta P_k \\ \Delta Q_k \end{bmatrix} = \begin{bmatrix} H_{km} & N_{km} \\ J_{km} & L_{km} \end{bmatrix} \begin{bmatrix} \Delta \delta_m \\ \frac{\Delta V_m}{V_m} \end{bmatrix} \quad (2.46)$$

Donde (2.46) representa un sistema de ecuaciones lineales por resolver para δ y $\Delta V/V$ y que se obtuvo de la aplicación del método de Newton a las ecuaciones no lineales (2.6) del problema del flujo de potencia, donde el vector de desajuste de potencia se puede evaluar mediante (2.28) y (2.29) y los elementos del jacobiano los evaluamos mediante (2.40) y (2.41), claro está que debemos tener un juego de voltajes asumidos o de la última iteración para evaluarlos y los datos de la matriz admitancia de barra.

A continuación formularemos las ecuaciones de Newton para el sistema de potencia de ejemplo de la figura N° 2.1.

En este sistema de 5 barras, escogiendo la barra oscilante como 1, tendremos que en cada barra tenemos 2 variables conocidas y 2 por determinar.

Con lo que tendremos 6 incógnitas a determinar, por lo tanto requerimos plantear la ecuación (2.31) en las ba

BARRA	TIPO BARRA	VARIABLE CONOCIDA	A DETERMINAR
2	DE GENERACION	P_2, V_2	δ_2
3	DE CARGA	P_3, Q_3	V_3, δ_3
4	DE GENERACION	P_4, V_4	δ_4
5	DE CARGA	P_5, Q_5	V_5, δ_5

rras 2, 3, 4 y 5 y la ecuación (2.32) en las barras 3 y 5, con lo cual tenemos 6 ecuaciones no lineales y 6 incógnitas por determinar a continuación planteamos las ecuaciones de Newton (2.44) y (2.45) en forma matricial:

$$\begin{bmatrix} \Delta P_2 \\ \Delta P_3 \\ \Delta P_4 \\ \Delta P_5 \\ \hline \Delta Q_3 \\ \Delta Q_5 \end{bmatrix} = \begin{bmatrix} H_{22} & H_{23} & H_{24} & & & \\ H_{32} & H_{33} & H_{34} & & & \\ H_{42} & H_{43} & H_{44} & H_{45} & & \\ & & H_{54} & H_{55} & & \\ \hline J_{32} & J_{33} & J_{34} & & & \\ & & J_{54} & J_{55} & & \end{bmatrix} \begin{bmatrix} N_{23} \\ N_{33} \\ N_{43} & N_{45} \\ N_{55} \\ \hline L_{33} \\ L_{55} \end{bmatrix} \begin{bmatrix} \Delta \delta_2 \\ \Delta \delta_3 \\ \Delta \delta_4 \\ \Delta \delta_5 \\ \hline \frac{\Delta V_3}{V_3} \\ \frac{\Delta V_5}{V_5} \end{bmatrix}$$

En esta ecuación matricial lineal tenemos 6 ecuaciones y 6 incógnitas por determinar y que son las correcciones de la magnitud y ángulo del voltaje $\Delta\delta_2, \Delta\delta_3, \Delta\delta_4, \Delta\delta_5$, $\Delta V_3/V_3$ y $\Delta V_5/V_5$, los residuos $\Delta P_2, \Delta P_3, \Delta P_4, \Delta P_5, \Delta Q_3$ y ΔQ_5 , pueden ser evaluados de (2.31) y (2.32) y la matriz jacobiano puede ser evaluada de (2.40) y (2.41), por lo tanto el problema puede ser resuelto en un proceso iterativo como veremos en la siguiente sección.

Observe como la estructura simétrica dispersa de la matriz jacobiano depende del patrón de conexión de la red, aunque el arreglo de arriba es de ayuda conceptual, no es muy conveniente para programación en la computadora, el arreglo que se muestra a continuación es mejor para la programación.

		2	3	4	5			
ΔP_2	2	H_{22}	H_{23}	N_{23}	H_{24}		$\Delta\delta_2$	
ΔP_3	3	H_{32}	H_{33}	N_{33}	H_{34}		$\Delta\delta_3$	
ΔQ_3	3	J_{32}	J_{33}	L_{33}	J_{34}		$\Delta V_3/V_3$	
ΔP_4	4	H_{42}	N_{43}	N_{43}	H_{44}	H_{45}	N_{45}	$\Delta\delta_4$
ΔP_5	5				H_{54}	H_{55}	N_{55}	$\Delta\delta_5$
ΔQ_5	5				J_{54}	J_{55}	L_{55}	$\Delta V_5/V_5$

Como antes, la estructura dispersa es simétrica y hay una correspondencia uno a uno entre los elementos de la matriz admitancia y las submatrices (1 x 1, 1x2, 2 x 1, 2 x 2) del Jacobiano.

Con esta disposición de sistema de ecuaciones lineales, la matriz jacobiano que se forma contendrá filas dobles - y/o filas simples, las filas simples corresponderán a los nodos de generación y las filas dobles a los nodos de carga, esto también producirá cuatro tipos de submatrices dentro del jacobiano dependiendo de que si se tiene una combinación de nodos: carga - carga, carga - generación, generación - generación y generación - carga, cada uno de estos casos los encontramos en la matriz jacobiano representada previamente, es de notar que esta matriz es simétrica en estructura y asimétrica con respecto a los valores de sus elementos.

2.4.2. ESTRATEGIA DE SOLUCION POR MEDIO DEL METODO DE NEWTON-RAPHSON

La estrategia de solución por medio del método de Newton-Raphson consiste en:

1. Asumir un juego de voltajes iniciales en todas las barras.

2. Reemplazando los voltajes asumidos o de la última iteración y los parámetros de la admitancia de barra en (2.31) y (2.32), obtenemos las potencias netas inyectadas en cada barra.
3. Restando estas potencias calculadas en 2) de las potencias especificadas en las barras, determinamos ΔP_k y ΔQ_k .
4. Con los voltajes asumidos o de la iteración más reciente y los parámetros de la matriz admitancia de barra a través de (2.40) y (2.41) evaluamos la matriz jacobiana, entonces en (2.46) determinamos las correcciones de la magnitud y ángulo del voltaje.

Es decir tenemos que evaluar:

$$\begin{bmatrix} \Delta \delta_m \\ \frac{\Delta V_m}{V_m} \end{bmatrix} = \begin{bmatrix} H_{km} & N_{km} \\ J_{km} & L_{km} \end{bmatrix}^{-1} \begin{bmatrix} \Delta P_k \\ \Delta Q_k \end{bmatrix} \quad (2.47)$$

5. Una vez determinadas las correcciones de la magnitud y ángulo del voltaje actualizamos la magnitud y ángulo del voltaje, y regresamos a 2) hasta que los desajustes de potencia en barra este por debajo de un

error permitido.

A continuación se muestra el algoritmo de solución del método de Newton - Raphson. (Figura N° 2.7.).

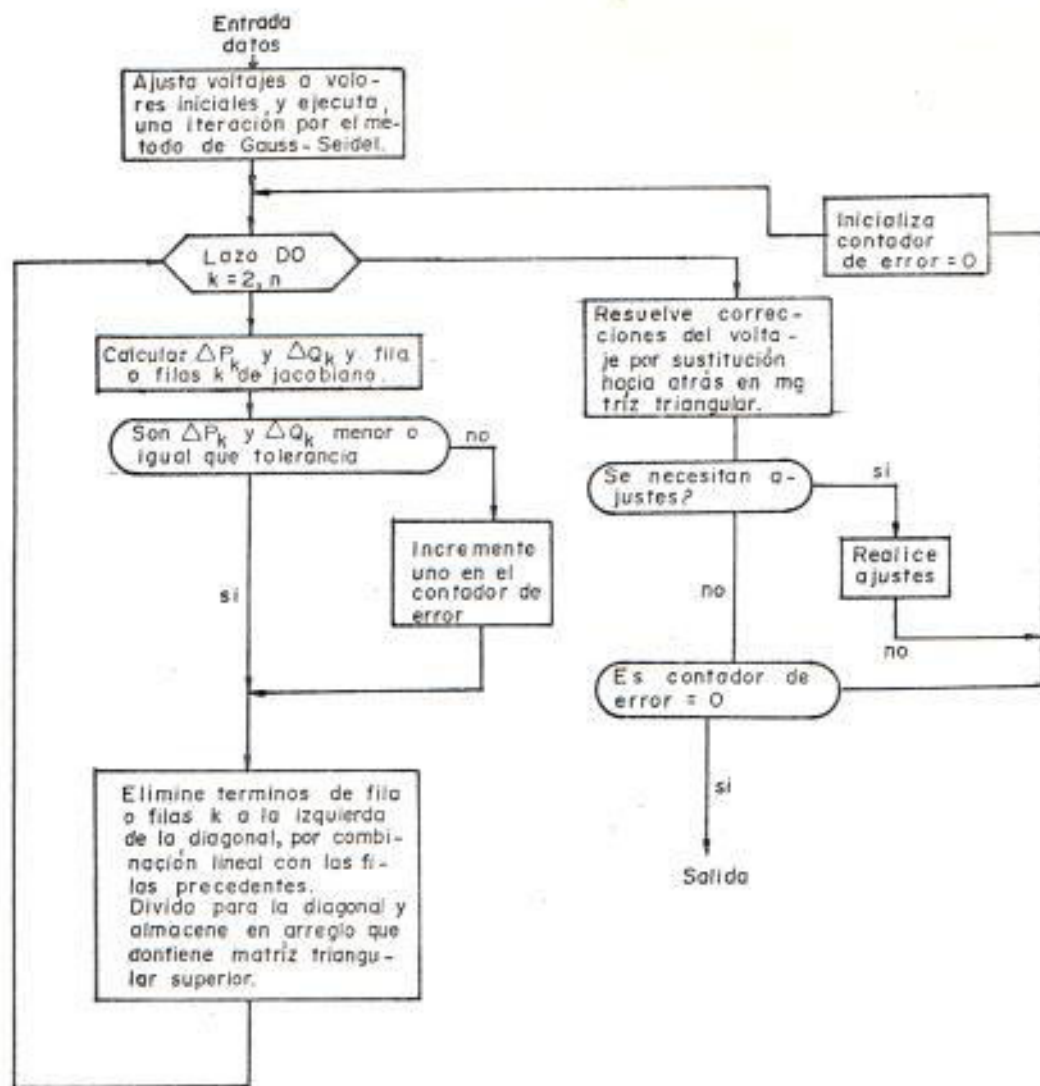


FIGURA N° 2.7. DIAGRAMA DE FLUJO SIMPLIFICADO DEL ALGORITMO DE NEWTON PARA EL FLUJO DE POTENCIA

2.4.3. Ventajas y Desventajas del Método de Newton - Raphson sobre los métodos iterativos (Gauss - Seidel)

El método de Newton - Raphson tiene poderosas propiedades de convergencia sobre otros métodos, así tiene una característica de convergencia cuadrática cuando se usa un punto de partida adecuado para la solución y sin ajustes en la solución, converge en una mucho menor cantidad de iteraciones con respecto al método de Gauss- Seidel independientemente del tamaño del sistema de potencia a resolver, el número de iteraciones típicas de este método es de 3 a 5 iteraciones para flujos de potencia sin ajustes, estas características se deben a que el método de Newton - Raphson a diferencia del método de Gauss - Seidel no se limita a resolver a ciegas el sistema de ecuaciones no lineales (2.6), sino que usa un segundo sistema de ecuaciones lineales (2.46) que determina los factores de corrección para el primer sistema de ecuaciones⁽⁵⁾, así de esta manera (2.6) se acerca más rápido a los valores de P y Q especificados ya que con (2.46) vamos acercándonos en pasos definidos al voltaje solución, es decir con la ecuación (2.46) retroalimentamos el error en el voltaje, por supuesto debido a la no linealidad de las ecuaciones originales (2.6), la solución de (2.46) no necesariamente conduce a las potencias especificadas en (2.6), a menos que las correcciones sean

relativamente pequeñas.

El tiempo total de solución del método de Newton es menor que el de Gauss - Seidel, debido a la mucho menor cantidad de iteraciones del método, aunque el tiempo por iteración del método de Newton es mayor que el de una iteración de Gauss - Seidel, por la cantidad de cálculos que son requeridos para la evaluación del jacobiano y resolver (2.46).

El método de Newton no tiene las deficiencias que presentan los métodos iterativos como son su inhabilidad para resolver problemas con reactancias de transferencia negativas y que no pueden ser enlazadas con reactancias positivas de una línea, dichas reactancias surgen del circuito equivalente de transformadores con 3 devanados, tampoco tiene problema con ramales de alta y baja impedancia que concurren a un mismo nodo, a todas estas situaciones se les denomina problemas de mal condicionamiento del sistema⁽⁸⁾.

Sin embargo de estas ventajas, el método de Newton Raphson no fue competitivo para flujos de potencia de sistemas grandes, debido a las siguientes desventajas:

Requería de gran cantidad de memoria

Excesivo tiempo de cálculo por iteración.

Si no se escogía un punto adecuado de partida para los voltajes iniciales podía el método diverger.

En cambio los métodos iterativos si bien es cierto que eran de convergencia lenta y sujetos a situaciones de mal condicionamiento, sus requerimientos de memoria eran mínimos y directamente proporcional al tamaño del sistema a resolver, mientras que en el método de Newton los requerimientos de memoria y tiempo de cálculo por iteración se volvían críticos con el incremento en tamaño del sistema a resolver, además que en los métodos iterativos el valor inicial de partida del voltaje solo retrasa la convergencia, pero no decide el éxito o fracaso de la convergencia.

Como se puede ver el método de Newton tiene ventajas muy importantes sobre los métodos iterativos, por lo que se hizo fue investigar el método para ir eliminando sus desventajas como eran el excesivo consumo de memoria y tiempo de cálculo por iteración, de estas investigaciones se concluyó que la dificultad no estaba en el método mismo sino en el procedimiento para resolver el sistema de ecuaciones lineales (2.46), reconocido este hecho, condujo al descubrimiento de las técnicas de programación y de manejo de las matrices dispersas

(matriz admitancia y jacobiano) del sistema^(7,9,10) que aprovechan las características que poseen los sistemas de potencia.

Estas técnicas numéricas y de programación son:

1. Métodos de eliminación ordenada optimamente para resolución de (2.46).
2. Esquemas de triangulación por fila en matrices.
3. Esquemas de memoria compacta.
4. Esquemas de fila de trabajo compacta.

Además para asegurar un punto de partida favorable al método de Newton se usa un ciclo inicial de voltajes - obtenidos mediante Gauss - Seidel.

Con estas técnicas y una muy hábil programación se cubrieron las desventajas de memoria y tiempo de cálculo del método de Newton para resolución de flujo de grandes - sistemas de potencia con lo cual definitivamente el método de Newton superó a los métodos iterativos (Gauss - Seidel) terminando por imponerse.

En el siguiente Capítulo detallaremos estas técnicas que hicieron competitivo el método de Newton sobre los otros métodos.

C A P I T U L O I I I

TECNICAS NUMERICAS Y DE PROGRAMACION

Un punto importante que se debe tener en cuenta cuando se va a implementar un programa de computadora es que este tendrá su mayor eficiencia cuando aprovecha las características físicas que posee el sistema que se va a resolver, por lo tanto el algoritmo del programa debe ser implementado con técnicas numéricas y de programación orientadas en este sentido.

Entonces un programa para resolución del flujo de carga de un sistema de potencia debe explotar sus características que son:

- Simetría de la red
- Dispersidad de la red

3.1. ALMACENAMIENTO APROVECHANDO SIMETRIA Y DISPERSIDAD

Las características mencionadas se han ido aprovechando en

ese orden cuando almacenamos las matrices representativas del sistema, estas matrices en el caso que usamos el método de Newton - Raphson para la resolución del flujo de carga son:

- La matriz admitancia de barra ($|Y|$)
- La matriz Jacobiano ($|J|$)

Generalmente estas matrices representativas del sistema llevan sus características, es decir que también son simétricas y dispersas.

3.1.1. Almacenamiento aprovechando simetría

Debido a que con el método de Newton - Raphson para la resolución del flujo de carga debemos construir la matriz admitancia de barra y jacobiano una manera sencilla de ahorrar memoria es aprovechando la simetría de la matriz, osea almacenando unicamente los elementos de la diagonal y del triángulo superior fuera de la diagonal.

La matriz $[Y]$ es simétrica porque si por ejemplo se tiene el elemento Y_{km} que corresponde al negativo de la admitancia del ramal conectado entre las barras k y m , el elemento Y_{mk} corresponderá al mismo valor ya que

todos los elementos de la red son bilaterales.

Entonces para el ejemplo de la figura N° 2.1., se ha almacenado la matriz (Y) en dos formas, en (3.1) como un arreglo de dos dimensiones (matriz cuadrada) que no permite aprovechar simetría y en (3.2) como un arreglo de una dimensión (vector columna) en el que se almacenan los elementos simétricos solo una vez.

$$|Y| = \begin{bmatrix} Y_{11} & Y_{12} & 0 & 0 & Y_{15} \\ Y_{21} & Y_{22} & Y_{23} & Y_{24} & 0 \\ 0 & Y_{32} & Y_{33} & Y_{34} & 0 \\ 0 & Y_{42} & Y_{43} & Y_{44} & Y_{45} \\ Y_{51} & 0 & 0 & Y_{54} & Y_{55} \end{bmatrix}$$

(3.1)

NODOS k-m	Y(i)
1-1	Y(1)
1-2	Y(2)
2-2	Y(3)
1-3	Y(4)
2-3	Y(5)
3-3	Y(6)
1-4	Y(7)
2-4	Y(8)
3-4	Y(9)
4-4	Y(10)
1-5	Y(11)
2-5	Y(12)
3-5	Y(13)
4-5	Y(14)
5-5	Y(15)

(3.2)

Como vemos si almacenamos la matriz $|Y|$ de barra como un arreglo de dos dimensiones ocupamos 25 posiciones de memoria para este ejemplo, en cambio si la almacenamos en forma de vector solo ocupamos 15 posiciones de memoria, pero el direccionamiento ahora no es tan directo sino que tenemos que usar la siguiente ecuación (3.3) para identificar la posición i del elemento Y_{km} en el vector.

$$i = m(m-1)/2 + k \quad (3.3)$$

Note que $k \leq m$ y que los valores de $|Y|$ han sido almacenados en orden de columnas.

De igual manera el jacobiano se almacena como un vector.

3.1.2. Almacenamiento aprovechando Dispersidad

Hasta este momento ya se ha aprovechado la simetría de la red para el ahorro de memoria, entonces el siguiente paso en el ahorro de memoria y tiempo de cálculo será aprovechar la dispersidad que posee todo sistema de potencia, para lo cual veamos que significa que una red es dispersa y como nos aprovecharemos de esto.

Se dice que una red es dispersa, cuando las matrices representativas de dicha red contienen una gran cantidad de ceros fuera de su diagonal, esto se debe a que en una red o sistema de potencia cada ramal entre un par de no dos o barras del sistema corresponde a un elemento fue ra de la diagonal de estas matrices, correspondiendo la posición de fila y columna en la matriz a la nume ración de las barras extremas del ramal, luego como no necesariamente cada barra del sistema tiene un ramal de conexión con cada una de las otras barras del sistema, quiere decir que van a existir barras que no se encuentran interconectadas directamente con un ramal (línea o transformador) y dichas posiciones en las matrices $|Y|$ y $|J|$ del sistema son representadas por ceros, en conclusión quiere decir que las matrices $|Y|$ y $|J|$ del sistema son dispersas, o sea que contienen un buen porcentaje de ceros fuera de su diagonal.

Entonces aprovechamos la dispersidad de una matriz, almacenando únicamente los elementos diferentes de cero de dicha matriz, a esto denomina Esquema de memoria COMPACTA.

En conclusión solo almacenamos y realizamos operaciones con los elementos diferentes de cero, pero hay que no

tar que si bien optimizamos memoria usando almacenamiento compacto la localización de los elementos de la matriz en la memoria se vuelve mucho más compleja.

Para la matriz $|Y|$ de la red de 5 barras de la figura N° 2.1., mediante el uso de un esquema de memoria compacta aprovechando dispersidad y simetría solo ocupamos 11 posiciones de memoria en lugar de 15, observe la diferencia en (3.3) y (3.4).

Aprovechando solo simetría (3.3)

Nodos	k-m	Y(i)
1-1		Y(1)
1-2		Y(2)
2-2		Y(3)
1-3		0
2-3		Y(5)
3-3		Y(6)
1-4		0
2-4		Y(8)
3-4		Y(9)
4-4		Y(10)
1-5		Y(11)
2-5		0
3-5		0
4-5		Y(14)
5-5		Y(15)

Aprovechando simetría y dispersidad (memoria com (3.4) pacta).

<u>Nodos k-m</u>	<u>Y(i)</u>
1-1	Y(1)
1-2	Y(2)
2-2	Y(3)
2-3	Y(4)
3-3	Y(5)
2-4	Y(6)
3-4	Y(7)
4-4	Y(8)
1-5	Y(9)
4-5	Y(10)
5-5	Y(11)

3.2. TECNICAS NUMERICAS

La necesidad de resolver grandes sistemas de ecuaciones lineales simultáneas dispersas de la forma $|J| |X| = |C|$; ocurre frecuentemente en varias áreas de ingeniería especialmente en sistemas de potencia eléctrico.

En particular en la resolución de flujos de carga surgen grandes sistemas de ecuaciones lineales a resolver, por ejemplo en el método de Newton - Raphson para flujo de carga debemos resolver un juego de ecuaciones lineales (3.5).

$$\begin{bmatrix} \Delta P_k \\ \Delta Q_k \end{bmatrix} = \begin{bmatrix} H_{km} & N_{km} \\ J_{km} & L_{km} \end{bmatrix} \begin{bmatrix} \Delta \delta_m \\ \Delta V_m/V_m \end{bmatrix} \quad (3.5)$$

Que en forma general y simplificada puede ser expresada - como (3.6)

$$|C| = |J| |X| \quad (3.6)$$

Donde:

- |C| es un vector columna conocido (desajustes de potencia en barra)
- |X| es el vector incógnita a ser determinado (correcciones - de V y δ)
- |J| es la matriz coeficiente del sistema de ecuaciones (jaco biano)

Como vemos de (3.6) para determinar las correcciones del voltaje |X| podríamos invertir la matriz |J|, pero esto no es lo más conveniente si observamos que a medida que transcurre el tiempo, los sistemas de potencia crecen en tamaño y complejidad, por lo cual cada vez ha sido necesario ir mejorando más y más las técnicas numéricas de solución, además la inversión no sólo que es lenta sino que para los sistemas de hoy puede exceder fácilmente las capacidades de memoria disponibles

y tampoco permite explotar la característica de dispersidad de la matriz $[J]$ porque la inversa de $[J]$ es llena.

Por lo cual en esta sección se irán explicando las técnicas numéricas en el orden que se han ido desarrollando e incorporando mejoras.

Entonces frente a la necesidad de resolver $[X]$ en (3.6), sin invertir $[J]$ se recurrió a otra herramienta matemática muy útil en su época, que es el método de eliminación de Gauss, y que se presenta a continuación.

3.2.1. Eliminación de Gauss

Sea el siguiente sistema de ecuaciones lineales (3.7) que se quiere resolver por eliminación.

$$C_1 = J_{1,1} X_1 + J_{1,2} X_2 + \dots + J_{1,n-1} X_{n-1} + J_{1,n} X_n$$

$$C_2 = J_{2,1} X_1 + J_{2,2} X_2 + \dots + J_{2,n-1} X_{n-1} + J_{2,n} X_n$$

⋮

$$C_{n-1} = J_{n-1,1} X_1 + J_{n-1,2} X_2 + \dots + J_{n-1,n-1} X_{n-1} + J_{n-1,n} X_n$$

$$C_n = J_{n,1} X_1 + J_{n,2} X_2 + \dots + J_{n,n-1} X_{n-1} + J_{n,n} X_n$$

(3.7)

En forma matricial (3.8)

$$\begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_{n-1} \\ C_n \end{bmatrix} = \begin{bmatrix} J_{1,1} & J_{1,2} & \cdots & J_{1,n-1} & J_{1,n} \\ J_{2,1} & J_{2,2} & \cdots & J_{2,n-1} & J_{2,n} \\ \vdots & \vdots & & \vdots & \vdots \\ J_{n-1,1} & J_{n-1,2} & \cdots & J_{n-1,n-1} & J_{n-1,n} \\ J_{n,1} & J_{n,2} & \cdots & J_{n,n-1} & J_{n,n} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{n-1} \\ X_n \end{bmatrix}$$

$$|C| = |J| |X| \quad (3.8)$$

Entonces el procedimiento para la resolución de $|X|$ consiste en:

- a. Una serie de operaciones y combinaciones lineales en tre filas de la matriz coeficientes $|J|$ aumentada - por el vector independiente $|C|$ de tal forma que $|J|$ se transforma en una matriz triángular superior con diagonal unitaria que la llamaremos $|U|$, el vector $|C|$ modificado por las combinaciones entre filas lo llama^{re}mos $|F|$.
- b. Con la matriz triángular $|U|$ y el vector modificado - $|F|$ efectuamos un proceso de sucesivos despejes de

las variables $|X|$ comenzando por X_n y finalizando con X_1 en ese orden, a este proceso se denomina de sustitución hacia atrás.

Los pasos son los siguientes:

Paso 1

Se multiplica la primera fila de $|J|$ aumentada por la columna $|C|$ por el recíproco del pivot de la fila, a continuación se eliminan cada uno de los elementos bajo el pivot de la primera columna, es decir eliminamos $J_{2,1}^{(0)}$ de la segunda fila por combinación lineal de ésta con la primera fila derivada, para ello a la segunda fila se le añade la nueva primera fila multiplicada por el negativo del elemento que se quiere eliminar, esta misma operación se repite entre las subsiguientes filas y la primera fila derivada, quedando (3.9), las operaciones que se han realizado en cada fila están representadas en (3.10) y (3.11).

Donde:

F_i es la fila i que está siendo modificada por la eliminación y que queda modificada por las operaciones

raciones expresadas a la derecha de F_j .

El superíndice indica el orden del sistema derivado o sea las veces que la cantidad ha sido alterada.

$$\begin{bmatrix} 1 & J_{1,2}^{(1)} & \dots & J_{1,n-1}^{(1)} & J_{1,n}^{(1)} & C_1^{(1)} \\ 0 & J_{2,2}^{(1)} & \dots & J_{2,n-1}^{(1)} & J_{2,n}^{(1)} & C_2^{(1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & J_{n-1,2}^{(1)} & \dots & J_{n-1,n-1}^{(1)} & J_{n-1,n}^{(1)} & C_{n-1}^{(1)} \\ 0 & J_{n,1}^{(1)} & \dots & J_{n,n-1}^{(1)} & J_{n,n}^{(1)} & C_n^{(1)} \end{bmatrix} \quad (3.9)$$

$$J_{1,j}^{(1)} = (1/J_{11}^{(0)}) J_{1,j}^{(0)}$$

$$F_1 : 1 / J_{11}^{(0)} \times F_1$$

$$F_2 : F_2 + (-J_{21}^{(0)}) \times F_1$$

⋮
⋮
⋮
⋮

$$F_{n-1} : F_{n-1} + (-J_{n-1,1}^{(0)}) \times F_1$$

$$F_n : F_n + (-J_{n,1}^{(0)}) \times F_1$$

(3.10)

$$C_1^{(1)} = (1/J_{11}^{(0)}) C_1^{(0)} \quad J = 2 \dots n$$

$$J_{i,j}^{(1)} = J_{i,j}^{(0)} + (-J_{i,1}^{(0)}) J_{1,j}^{(1)} \quad C_i^{(1)} = C_i^{(0)} + (-J_{i,1}^{(0)}) C_1^{(1)}$$

$$i, j = 2, \dots, n \quad (3.11)$$

Paso 2:

Se multiplica la segunda fila modificada por el reciproco de su Pivot, es decir se normaliza la fila quedando su pivot unitario.

A continuación se eliminan cada uno de los elementos bajo el pivot de la segunda columna (3.12) para lo cual se añade a la fila cuyo elemento se va a eliminar la segunda fila normalizada multiplicada por el negativo del elemento a eliminar, las operaciones son mostradas en (3.13) y (3.14).

$$\begin{bmatrix} 1 & J_{1,2}^{(1)} & \dots & J_{1,n-1}^{(1)} & J_{1,n}^{(1)} & C_1^{(1)} \\ 0 & 1 & \dots & J_{2,n-1}^{(2)} & J_{2,n}^{(2)} & C_2^{(2)} \\ \cdot & & & & & \\ \cdot & & & & & \\ \cdot & & & & & \\ \cdot & & & & & \\ 0 & 0 & \dots & J_{n-1,n-1}^{(2)} & J_{n-1,n}^{(2)} & C_{n-1}^{(2)} \\ 0 & 0 & \dots & J_{n,n-1}^{(2)} & J_{n,n}^{(2)} & C_n^{(2)} \end{bmatrix}$$

(3.12)

$$\begin{aligned}
 F_2 &: (1/J_{2,2}^{(1)}) F_2 \\
 &\cdot \\
 &\cdot \\
 &\cdot \\
 &\cdot \\
 F_{n-1} &: F_{n-1} + (-J_{n-1,2}^{(1)}) F_2 \\
 F_n &: F_n + (-J_{n,2}^{(1)}) F_2
 \end{aligned} \tag{3.13}$$

$$\begin{aligned}
 J_{2,j}^{(2)} &= (1/J_{22}^{(1)}) J_{2,j}^{(1)} & c_2^{(2)} &= (1/J_{22}^{(1)}) c_2^{(1)} & j &= 3 \dots n \\
 J_{i,j}^{(2)} &= J_{i,j}^{(1)} + (-J_{i,2}^{(1)}) J_{2,j}^{(2)} & c_i^{(2)} &= c_i^{(1)} + (-J_{i,2}^{(1)}) c_2^{(2)} & i, j &= 3 \dots n
 \end{aligned} \tag{3.14}$$

Continuando con este procedimiento para cada una de las filas siguientes se transforma $[J]$ a una matriz triangular superior con diagonal unitaria (3.15), las ecuaciones generales del método se muestran en (3.16).

$$\left[\begin{array}{cccccc}
 1 & J_{1,2}^{(1)} & \dots & J_{1,n-1}^{(1)} & J_{1,n}^{(1)} & c_1^{(1)} \\
 0 & 1 & \dots & J_{2,n-1}^{(2)} & J_{2,n}^{(2)} & c_2^{(2)} \\
 \cdot & & & & & \cdot \\
 \cdot & & & & & \cdot \\
 \cdot & & & & & \cdot \\
 0 & 0 & \dots & 1 & J_{n-1,n}^{(n-1)} & c_{n-1}^{(n-1)} \\
 0 & 0 & \dots & 0 & 1 & c_n^{(n)}
 \end{array} \right] \tag{3.15}$$

$$\begin{aligned}
 J_{k,j}^{(k)} &= (1/J_{kk}^{(k-1)}) J_{k,j}^{(k-1)} & C_k^{(k)} &= (1/J_{kk}^{(k-1)}) C_k^{(k-1)} \\
 J_{i,j}^{(k)} &= J_{i,j}^{(k-1)} + (-J_{i,k}^{(k-1)}) J_{k,j}^{(k)} & C_i^{(k)} &= C_i^{(k-1)} + (-J_{i,k}^{(k-1)}) C_k^{(k)} \\
 \therefore k < i \quad i, j &= 2, \dots, n & & (3.16)
 \end{aligned}$$

Como vemos aquí la matriz $[J]$ ha sido triangulada a una matriz equivalente $[U]$ y $|C|$ se ha modificado para dar un vector $|F|$ (3.17) y (3.18) y el proceso de triangulación ha ido progresando hacia adelante en orden de columnas (Figura N° 3.1), a esto se denomina un proceso de sustitución hacia adelante.

El proceso de triangulación también puede progresar hacia adelante en orden de filas (figura N° 3.2), lo que nos conducirá a la misma matriz $[U]$ y vector $|F|$ (3.17), (3.18).

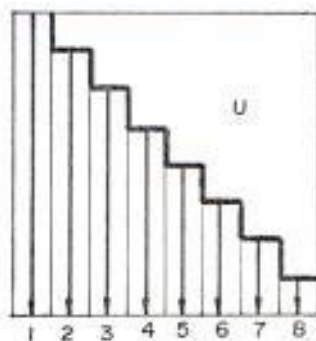


FIGURA N° 3.1. PROCESO DE TRIANGULACION POR COLUMNA.

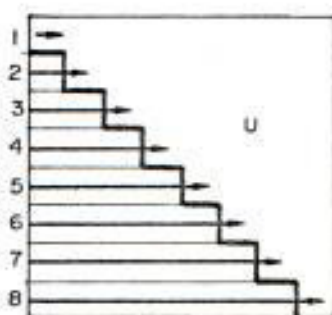


FIGURA N° 3.2. PROCESO DE TRIANGULACION POR FILA

$$|U| = \begin{bmatrix} 1 & J_{1,2}^{(1)} & \dots & J_{1,n-1}^{(1)} & J_{1,n}^{(1)} \\ 0 & 1 & \dots & J_{2,n-1}^{(2)} & J_{2,n}^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & J_{n-1,n}^{(n-1)} \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix} \quad (3.17)$$

$$[F] = \begin{bmatrix} c_1^{(1)} \\ c_2^{(2)} \\ \vdots \\ c_{n-1}^{(n-1)} \\ c_n^{(n)} \end{bmatrix} \quad (3.18)$$

Desde el punto de vista de programación es mucho más eficiente eliminar en orden de filas (figura N° 3.2), el

procedimiento es el siguiente:

Paso 1 :

Multiplicamos los elementos de la primera fila por el recíproco de su pivot (J_{11}) como se indica en (3.19):

$$J_{1,j}^{(1)} = (1/J_{1,1}^{(0)}) J_{1,j}^{(0)} \quad C_1^{(1)} = (1/J_{1,1}^{(0)}) C_1^{(0)} \quad J = 2 \dots n \quad (3.19)$$

Paso 2:

El segundo paso como está indicado en (3.20) y (3.21) es eliminar J_{21} de la segunda fila, a través de combinación lineal de esta con la nueva primera fila y luego normalizar la segunda fila, esto es dividir toda la fila para su nuevo pivot derivado.

$$\begin{bmatrix} 1 & J_{1,2}^{(1)} & J_{1,3}^{(1)} & \dots & J_{1,n}^{(1)} & C_1^{(1)} \\ 0 & 1 & J_{2,3}^{(2)} & \dots & J_{2,n}^{(2)} & C_2^{(2)} \\ J_{3,1}^{(0)} & J_{3,2}^{(0)} & J_{3,3}^{(0)} & \dots & J_{3,n}^{(0)} & C_3^{(0)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ J_{n,1}^{(0)} & J_{n,2}^{(0)} & J_{n,3}^{(0)} & \dots & J_{n,n}^{(0)} & C_n^{(0)} \end{bmatrix} \quad (3.20)$$

$$\begin{aligned}
 J_{2,j}^{(1)} &= J_{2,j}^{(0)} + (-J_{2,1}^{(0)})J_{1,j}^{(1)} & C_2^{(1)} &= C_2^{(0)} + (-J_{2,1}^{(0)})C_1^{(1)} & J=2\dots n \\
 J_{2,j}^{(2)} &= (1/J_{2,2}^{(1)})J_{2,j}^{(1)} & C_2^{(2)} &= (1/J_{2,2}^{(1)})C_2^{(1)} & J=3\dots n
 \end{aligned}$$

(3.21)

Paso 3:

El tercer paso como indicado en (3.22) y (3.23) consiste en eliminar los elementos a la izquierda de la diagonal de la fila 3 y luego dividir los elementos derivados que quedan de la fila para el elemento diagonal de la fila derivada (pivot).

$$\left[\begin{array}{cccccccc}
 1 & J_{1,2}^{(1)} & J_{1,3}^{(1)} & J_{1,4}^{(1)} & \dots & J_{1,n}^{(1)} & C_1^{(1)} \\
 0 & 1 & J_{2,3}^{(2)} & J_{2,4}^{(2)} & \dots & J_{2,n}^{(2)} & C_2^{(2)} \\
 0 & 0 & 1 & J_{3,4}^{(3)} & \dots & J_{3,n}^{(3)} & C_3^{(3)} \\
 J_{4,1}^{(0)} & J_{4,2}^{(0)} & J_{4,3}^{(0)} & J_{4,4}^{(0)} & \dots & J_{4,n}^{(0)} & C_4^{(0)} \\
 \vdots & & & & & & \vdots \\
 \vdots & & & & & & \vdots \\
 \vdots & & & & & & \vdots \\
 \vdots & & & & & & \vdots \\
 J_{n,1}^{(0)} & J_{n,2}^{(0)} & J_{n,3}^{(0)} & J_{n,4}^{(0)} & \dots & J_{n,n}^{(0)} & C_n^{(0)}
 \end{array} \right]$$

(3.22)

$$\begin{aligned}
 J_{3,j}^{(1)} &= J_{3,j}^{(0)} + (-J_{31}^{(0)})J_{1,j}^{(1)} & C_3^{(1)} &= C_3^{(0)} + (-J_{31}^{(0)})C_1^{(1)} & J=2, \dots, n \\
 J_{3,j}^{(2)} &= J_{3,j}^{(1)} + (-J_{32}^{(1)})J_{2,j}^{(2)} & C_3^{(2)} &= C_3^{(1)} + (-J_{32}^{(1)})C_2^{(2)} & J=3, \dots, n \\
 J_{3,j}^{(3)} &= (1/J_{3,3}^{(2)})J_{3,j}^{(2)} & C_3^{(3)} &= (1/J_{33}^{(2)})C_3^{(2)} & J=4, \dots, n
 \end{aligned}
 \tag{3.23}$$

Si continuamos así hasta la n-esima fila tenemos:

$$\begin{bmatrix}
 1 & J_{1,2}^{(1)} & J_{1,3}^{(1)} & \dots & J_{1,n}^{(1)} & C_1^{(1)} \\
 0 & 1 & J_{2,3}^{(2)} & \dots & J_{2,n}^{(2)} & C_2^{(2)} \\
 0 & 0 & 1 & \dots & J_{3,n}^{(3)} & C_3^{(3)} \\
 \vdots & & & & & \vdots \\
 0 & 0 & 0 & \dots & 1 & C_n^{(n)}
 \end{bmatrix}
 \tag{3.24}$$

Puede ser notado en (3.22) que hasta el paso 3, el trabajo de triangulación sobre las filas 1,2 y 3 ha sido concluído y que desde la fila 4 en adelante hasta la fila n, estas no han sufrido aun modificación alguna, esto es muy conveniente porque podemos ir evaluando cada fila del jacobiano $[J]$ y a la vez ir triangulando la respectiva fila, esto será explica

do más adelante en técnicas de programación.

Una vez obtenida la matriz triangular superior $|U|$ y el vector modificado $|F|$ (3.25) resolvemos $|X|$ mediante sucesivos despejes de la variable, desde X_n hasta X_1 en ese orden (3.26), a este proceso denominamos de sustitución hacia atrás.

$$|F| = |U| |X|$$

$$\begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ \vdots \\ F_{n-1} \\ F_n \end{bmatrix} = \begin{bmatrix} 1 & U_{1,2} & U_{1,3} \cdots U_{1,n-1} & U_{1,n} \\ & 1 & U_{2,3} \cdots U_{2,n-1} & U_{2,n} \\ & & 1 & \cdots U_{3,n-1} & U_{3,n} \\ & & & \vdots & \vdots \\ & & & \vdots & \vdots \\ & & & & 1 & U_{n-1,n} \\ & & & & & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ \vdots \\ X_{n-1} \\ X_n \end{bmatrix}$$

$$F_i = C_i^{(i)} \quad \therefore i = 1 \dots n$$

$$U_{i,j} = J_{i,j}^{(i)}$$

$$i, j = 1 \dots n \quad \therefore i < j$$

(3.25)

$$X_n = F_n$$

$$X_{n-1} = F_{n-1} - U_{n-1,n} X_n$$

.

.

.

.

$$X_3 = F_3 - \dots - U_{3,n-1} X_{n-1} - U_{3,n} X_n$$

$$X_2 = F_2 - U_{2,3} X_3 - \dots - U_{2,n-1} X_{n-1} - U_{2,n} X_n$$

$$X_1 = F_1 - U_{1,2} X_2 - U_{1,3} X_3 - \dots - U_{1,n-1} X_{n-1} - U_{1,n} X_n$$

(3.26)

EJEMPLOS:

Sea el siguiente sistema que se quiere resolver, lo resolveremos por las dos formas de eliminación de Gauss.

$$2 = 3X_1 - X_2 - X_3$$

$$-3/2 = -X_1 + 4X_2 - 2X_3$$

$$-1/2 = -X_1 - 2X_2 + 4X_3$$

En forma matricial tenemos:

$$|C| = |J| |X|$$

$$\begin{bmatrix} 2 \\ -3/2 \\ -1/2 \end{bmatrix} = \begin{bmatrix} 3 & -1 & -1 \\ -1 & 4 & -2 \\ -1 & -2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

a) Por sustitución hacia adelante en orden de columnas

$$\begin{bmatrix} 3 & -1 & -1 & 2 \\ -1 & 4 & -2 & -3/2 \\ -1 & -2 & 4 & -1/2 \end{bmatrix} \begin{matrix} F_1:(1/3)F_1 \\ \\ \end{matrix} \begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ -1 & 4 & -2 & -3/2 \\ -1 & -2 & 4 & -1/2 \end{bmatrix} \begin{matrix} \\ F_2:F_2+(1)F_1 \\ \end{matrix}$$

$$\begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ 0 & 11/3 & -7/3 & -5/6 \\ -1 & -2 & 4 & -1/2 \end{bmatrix} \begin{matrix} \\ \\ F_3:F_3+(1)F_1 \end{matrix} \begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ 0 & 11/3 & -7/3 & -5/6 \\ 0 & -7/3 & 11/3 & 1/6 \end{bmatrix} \begin{matrix} \\ F_2:(3/11)F_2 \\ \end{matrix}$$

$$\begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ 0 & 1 & -7/11 & -5/22 \\ 0 & -7/3 & 11/3 & 1/6 \end{bmatrix} \begin{matrix} \\ \\ F_3:F_3+(7/3)F_2 \end{matrix} \begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ 0 & 1 & -7/11 & -5/22 \\ 0 & 0 & 24/11 & -4/11 \end{bmatrix} \begin{matrix} \\ \\ F_3:11/24)F_3 \end{matrix}$$

$$\begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ 0 & 1 & -7/11 & -5/22 \\ 0 & 0 & 1 & -1/6 \end{bmatrix}$$

Resolución por sustitución hacia atrás

$$[F] = [U] [X]$$

$$\begin{bmatrix} 2/3 \\ -5/22 \\ -1/6 \end{bmatrix} = \begin{bmatrix} 1 & -1/3 & -1/3 \\ 0 & 1 & -7/11 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} \quad \begin{array}{l} 2/3 = X_1 - 1/3X_2 - 1/3X_3 \\ -5/22 = X_2 - 7/11X_3 \\ -1/6 = X_3 \end{array}$$

$$X_3 = -1/6$$

$$X_3 = -1/6$$

$$X_2 = -5/22 + 7/11X_3$$

$$X_2 = -1/3$$

$$X_1 = 2/3 + 1/3X_2 + 1/3X_3$$

$$X_1 = 1/2$$

b. Por sustitución hacia adelante en orden de filas

$$\begin{bmatrix} 3 & -1 & -1 & 2 \\ -1 & 4 & -2 & -3/2 \\ -1 & -2 & 4 & -1/2 \end{bmatrix} \begin{array}{l} F_1: (1/3)F_1 \\ \\ \end{array} \begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ -1 & 4 & -2 & -3/2 \\ -1 & -2 & 4 & -1/2 \end{bmatrix} \begin{array}{l} \\ F_2: F_2 + (1)F_1 \\ \end{array}$$

$$\begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ 0 & 11/3 & -7/3 & -5/6 \\ -1 & -2 & 4 & -1/2 \end{bmatrix} \begin{array}{l} \\ F_2: (3/11)F_2 \\ \end{array}$$

$$\begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ 0 & 1 & -7/11 & -5/22 \\ -1 & -2 & 4 & -1/2 \end{bmatrix} \begin{array}{l} \\ \\ F_3: F_3 + (1)F_1 \end{array}$$

$$\begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ 0 & 1 & -7/11 & -5/22 \\ 0 & -7/3 & 11/3 & 1/6 \end{bmatrix} \quad F_3: F_3 + (7/3)F_2$$

$$\begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ 0 & 1 & -7/11 & -5/22 \\ 0 & 0 & 24/11 & -4/11 \end{bmatrix} \quad F_3: F_3(11/24)$$

$$\begin{bmatrix} 1 & -1/3 & -1/3 & 2/3 \\ 0 & 1 & -7/11 & -5/22 \\ 0 & 0 & 1 & -1/6 \end{bmatrix}$$

U
F

$$\begin{bmatrix} 2/3 \\ -5/22 \\ -1/6 \end{bmatrix} = \begin{bmatrix} 1 & -1/3 & -1/3 \\ 0 & 1 & -7/11 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} \quad \begin{array}{l} 2/3 = X_1 - 1/3X_2 - 1/3X_3 \\ -5/22 = X_2 - 7/11X_3 \\ -1/6 = X_3 \end{array}$$

$$X_3 = -1/6$$

$$X_3 = -1/6$$

$$X_2 = -5/22 + 7/11X_3$$

$$X_2 = -1/3$$

$$X_1 = 2/3 + 1/3X_2 + 1/3X_3$$

$$X_1 = 1/2$$

3.2.2. Optimización de Dispersidad y secuencia de Procesamiento

Hasta este momento hemos aprovechado simetría y dispersidad para almacenar las matrices $|Y|$ y $|J|$, pero tam-

bién debemos almacenar la matriz triangular $|U|$ que resulta del proceso de eliminación, se puede comprobar - que al triangular una matriz dispersa cualquiera, la tendencia es hacia llenar la matriz diagonalizada $|U|$, o sea, de destruir los ceros y disminuir la dispersidad.

Por lo tanto el siguiente paso en la optimización de memoria es tratar de optimizar la dispersidad de la matriz que se está triangulando tanto como sea posible, de tal forma que la matriz resultante $|U|$ también sea dispersa, de esta manera será útil usar un esquema de almacenamiento compacto.

Entonces las investigaciones fueron orientadas en este sentido y se descubrió que la optimización - de la dispersidad durante el proceso de eliminación - de Gauss dependía de la secuencia u orden en que las filas de la matriz son procesadas es decir de la secuencia con que se realizan las operaciones de - triangulación o eliminación⁽⁷⁾, de esta manera surgen varios esquemas de ordenamiento de las ecuaciones y por ende de la matriz que se va a procesar.

Si el método de eliminación de Gauss se realiza siguiendo una secuencia ordenada de procesamiento (esquema de ordenamiento), entonces el método toma el

nombre de "ELIMINACION ORDENADA".

3.2.3. Variación de Dispersidad con secuencia de procesamiento

Demostraremos con un ejemplo como varía la dispersidad durante el proceso de eliminación⁽⁷⁾.

Considere la red del ejemplo de la figura N° 3.3.

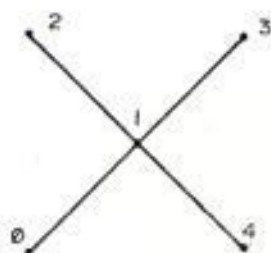


FIGURA N° 3.3. RED DE CUATRO NODOS

Esta consiste de 4 nodos independientes numerados desde el 1 hasta el 4 y un nodo de referencia numerado como 0, la ecuación en forma matricial (3.27) muestra el patrón de la matriz admitancia $[Y]$ de acuerdo a la numeración dada en la figura N° 3.3.

$$[Y] = \begin{bmatrix} X & X & X & X \\ X & X & & \\ X & & X & \\ X & & & X \\ 1 & 2 & 3 & 4 \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} \quad (3.27)$$

Durante el proceso de eliminación pueden suceder 3 cosas con los elementos de la fila que se está procesando, como lo veremos con el ejemplo:

- Que algunos elementos no sufran modificación
- Que algunos elementos sufran modificación
- Que se creen nuevos elementos. (se destruyen ceros)

La ecuación (3.28) muestra el patrón después de procesar la primera fila y la (3.31) después del primer paso de eliminación.

$$Y' = \begin{bmatrix} 1 & X' & X' & X' \\ X & X & & \\ X & & X & \\ X & & & X \end{bmatrix} \quad (3.28)$$

$$Y'' = \begin{bmatrix} 1 & X' & X' & X' \\ 0 & X' & \textcircled{X} & \textcircled{X} \\ X & & X & \\ X & & & X \end{bmatrix} \quad (3.29)$$

$$Y''' = \begin{bmatrix} 1 & X' & X' & X' \\ 0 & X' & \textcircled{X} & \textcircled{X} \\ 0 & \textcircled{X} & X' & \textcircled{X} \\ X & & & X \end{bmatrix} \quad (3.30)$$

$$Y^{IV} = \begin{bmatrix} 1 & X' & X' & X' \\ 0 & X' & \textcircled{X} & \textcircled{X} \\ 0 & \textcircled{X} & X' & \textcircled{X} \\ 0 & \textcircled{X} & \textcircled{X} & X' \end{bmatrix} \quad (3.31)$$

En la ecuación (3.29) se muestra la eliminación del primer elemento de la segunda fila, como vemos se crean dos nuevos elementos el Y_{23} y el Y_{24} , de igual manera al eliminar el primer elemento de la tercera fila, se crea un nuevo ramal o elemento el Y_{35} , y al finalizar el proceso de eliminación de los elementos bajo la diagonal de la primera columna, observamos que en la segunda columna se tienen ahora dos nuevos elementos bajo su pivot.

Hasta este punto vemos como matemáticamente se introducen tres nuevos elementos en la matriz, esto también puede ser interpretado físicamente desde el punto de vista de teoría de redes que dice que:

Si eliminamos un nodo de una red que contiene dos o más nodos adyacentes unidos a él, a través de ramales, entonces la red equivalente contendrá nuevos ramales entre cada uno de los nodos adyacentes que no se encuentran interconectados por un ramal, y los que ya se encuentran interconectados, el ramal original sufrirá -

una modificación de su admitancia, producida por el paralelismo de un nuevo ramal (ver figuras N^os. 3.4 y 3.5.).

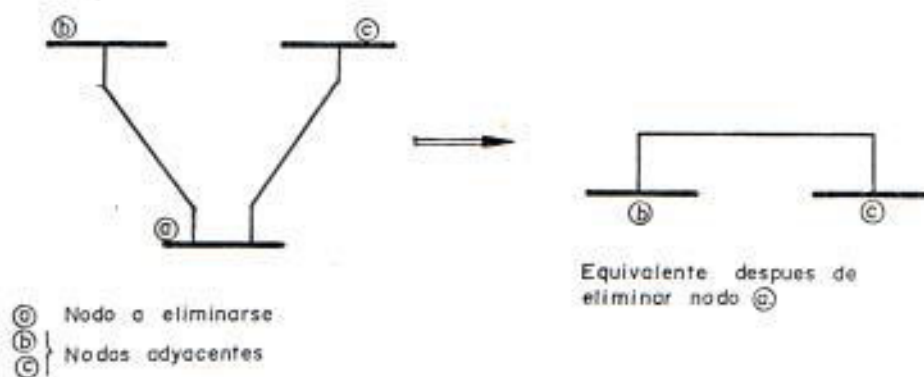


FIGURA N^o 3.4. RED DE TRES NODOS Y DOS RAMALES

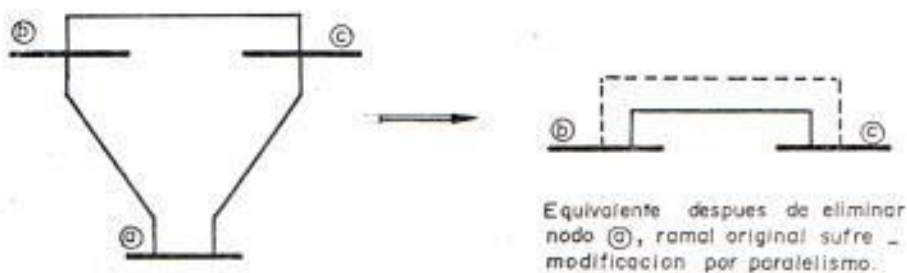


FIGURA N^o 3.5. RED DE TRES NODOS Y TRES RAMALES

En cambio si se elimina un nodo que contiene unicamente un ramal es decir sólo un nodo adyacente, simplemente se elimina el ramal.

Entonces de acuerdo con esto al hacer cero todos los elementos bajo el pivot de la primera columna, es como

que si hubieramos eliminado el nodo 1 y la red equivalente será una red con solo 3 nodos independientes (ver figura N° 3.6) y que contiene 3 nuevos ramales entre los nodos independientes, estos son los tres nuevos elementos que contiene (3.31) en su triángulo superior.

Continuando con el proceso de eliminación en la ecuación (3.31), obtenemos la matriz triangular $|U|$ (3.32).

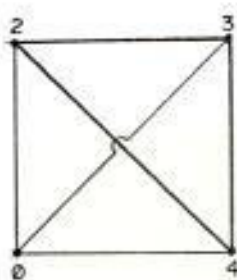


FIGURA N° 3.6. RED EQUIVALENTE DE RED DE CUATRO NODOS INICIALES

$$|U| = \begin{bmatrix} 1 & X' & X' & X' \\ & 1 & X & X \\ & & 1 & X \\ & & & 1 \end{bmatrix} \quad (3.32)$$

Como vemos en este ejemplo, si triangulamos la matriz $|Y|$ de acuerdo con la numeración de barras dada se pierde la dispersidad de la matriz y además debemos realizar más operaciones no previstas con los nue

vos elementos creados; ahora observemos el efecto de procesar la cuarta fila primero o lo que es lo mismo - al nodo que era 1 numerarlo como 4 y al nodo que era 4 como 1 (Ver figura N° 3.7).

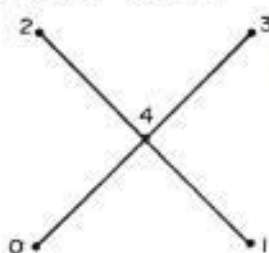


FIGURA N° 3.7. RED DE CUATRO NODOS CON RENUMERACION OPTIMA DE NODOS

La ecuación (3.33) muestra el patrón de $|Y|$ con la nueva numeración de barras.

$$|Y| = \begin{array}{cccc} 1 & 2 & 3 & 4 \\ \left[\begin{array}{cccc} X & & & X \\ & X & & X \\ & & X & X \\ X & X & X & X \end{array} \right] & \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \end{array} \end{array} \quad (3.33)$$

La ecuación (3.34), muestra el patrón después de procesar la primera fila y la ecuación (3.35) muestra la matriz triangulada final $|U|$.

$$Y' = \begin{array}{cccc} \left[\begin{array}{ccc} 1 & & X \\ & X & X \\ & & X & X \\ & X & X & X \end{array} \right] & \end{array} \quad (3.34)$$

$$|U| = \begin{bmatrix} \bar{1} & & & X \\ & 1 & & X \\ & & 1 & X \\ & & & 1 \end{bmatrix} \quad (3.35)$$

Estos dos ejemplos demuestran como la secuencia de operaciones o reordenamiento de filas y columnas - influyen sobre el número de operaciones y términos diferentes de cero (dispersidad).

Las matrices $|U|$ de las ecuaciones (3.32) y (3.35), son equivalentes y pueden ser usadas indistintamente en la mayoría de aplicaciones.

Como vemos poniendo $|Y|$ en la forma (3.33) hemos minimizado la cantidad de ceros que se destruyen durante el proceso de eliminación, esto puede ser visto gráficamente.

En la figura N° 3.7., los nodos que se eliminan primero, son los nodos que contienen un solo ramal y por ende un solo nodo adyacente, así su eliminación no trae consigo la creación de nuevos ramales, mientras que en el primer caso primero se eliminaba un nodo que contenía tres nodos adyacentes, la red equivalente después de eliminar el nodo 1 con la nueva nu

meración de barras será (figura N° 3.8):

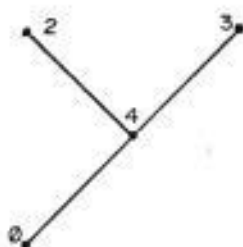


FIGURA N° 3.8. RED EQUIVALENTE DE RED DE CUATRO NODOS
CON RENUMERACION OPTIMA

Como vemos no hay ningún nuevo ramal añadido, por lo tanto en la ecuación (3.34) no se ha destruído ningún cero.

Puesto que la secuencia en la cual las filas son procesadas, influye sobre el número de operaciones y la dispersidad, entonces para cada red debe existir una secuencia de procesamiento la cual requiere el número mínimo de operaciones y una secuencia posiblemente pero no necesariamente la misma que requiere el mínimo de memoria.

En conclusión para triangular la matriz jacobiano, debemos tener una secuencia de operaciones o lo que es lo mismo asignar una numeración de barras o nodos óptima a esto se denomina esquema de ordenamiento de nodos, de esta forma minimizamos la creación de

nuevos términos diferentes de cero ahorrando memoria si usamos almacenamiento compacto.

3.2.4. Esquema de ordenamiento de Nodos

Para obtener una secuencia de operaciones óptimas existen varios esquemas de ordenamiento, que se pueden clasificar en:

- Estáticos; y,
- Dinámicos.

Estudiaremos sólo tres esquemas que son, los que normalmente se usan.

ESQUEMA 1:

Se denomina también esquema de ordenamiento estático, y es el más sencillo y fácil de programar, se denomina es tático porque se asigna la numeración de barras de acuerdo a una cuenta inicial de la cantidad de ramales conectados a cada barra sin tomar en cuenta lo que ocurre dinámicamente durante el proceso de eliminación como la creación y eliminación de ramales que alteran la cuenta inicial de ramales por nodo, el esquema

consiste en asignar la numeración más baja a los nodos que contienen la menor cantidad de ramales, y la numeración más alta a los nodos con mayor cantidad de ramales finalizando con los nodos que contienen la mayor cantidad de ramales. En la cuenta de ramales por nodo, los ramales al nodo de referencia no son incluidos, los pasos son los siguientes:

1. Los nodos con solo un ramal son numerados primero en cualquier secuencia así al eliminar estos nodos, el efecto es disminuir la red en un ramal por cada nodo de estos eliminado.
2. Los nodos con dos ramales son numerados en cualquier secuencia, de tal forma que al ser eliminados se eliminan dos ramales ya existentes y se crea un nuevo ramal entre los dos nodos adyacentes al nodo eliminado, y si ya existe un ramal entre los nodos adyacentes se modifica.

Al final de estas operaciones la red reducida - tendrá una menor cantidad de ramales que antes, el número de ramales por nodo entre los que quedan aún concordará con su cuenta original excepto en

los casos en que ocurre modificación de un ramal ya existente (paralelismo).

- Después los nodos con tres ramales en la red original (algunos de los cuales podrían ahora tener solo dos ramales) son eliminados, esto elimina tres ramales ya existentes y crea o modifica tres ramales entre los nodos adyacentes, en este momento la red no tendrá más y posiblemente menos ramales que antes.

Así se continua hasta finalizar con todos los nodos en orden ascendente a la cantidad de ramales de la cuenta inicial.

EJEMPLO:

La siguiente red mostrada en la figura N° 3.9., será numerada de acuerdo a cada uno de los esquemas de ordenamiento.

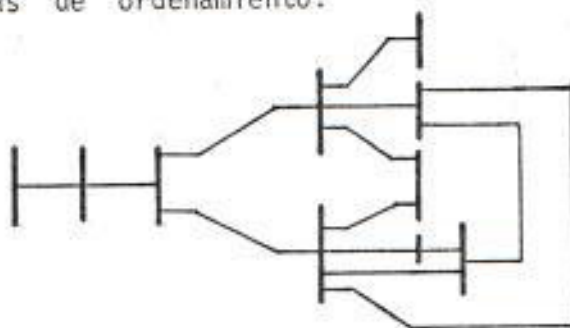


FIGURA N° 3.9. RED DE 10 NODOS (EJEMPLO)

$$|Y| = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \end{matrix} & \begin{bmatrix} X & & & & X & & & & & \\ & X & & & & & & & X & \\ & & X & & & & & & X & X \\ & & & X & & & & X & & X \\ X & & & & X & X & & & & \\ & & & & X & X & & & X & X \\ & & & & & & X & X & X & X \\ & & & X & & & X & X & & X \\ & X & X & & & X & X & & X & \\ & & X & X & & X & X & X & & X \end{bmatrix} \end{matrix}$$

$$|U| = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{matrix} \\ \begin{matrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{matrix} & \begin{bmatrix} & & & & X & & & & & \\ & 1 & & & & & & & X & \\ & & 1 & & & & & & X & X \\ & & & 1 & & & & X & & X \\ & & & & 1 & X & & & & \\ & & & & & 1 & & & X & X \\ & & & & & & 1 & X & X & X \\ & & & & & & & 1 & \otimes & X \\ & & & & & & & & 1 & \otimes \\ & & & & & & & & & 1 \end{bmatrix} \end{matrix} \quad (3.36)$$

De acuerdo con el esquema 1 la red quedaría numerada como en la figura N° 3.10. y el patrón de las matri

ces $|Y|$ y la matriz triangulada $|U|$ como en (3.36).

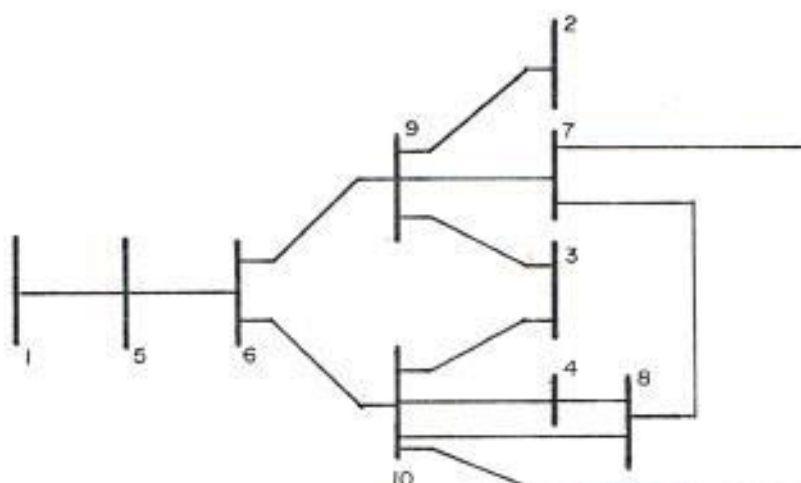


FIGURA N°3.10. RED DE 10 NODOS NUMERADA DE ACUERDO A ESQUEMA 1

ESQUEMA 2 :

Se denomina también esquema de ordenamiento dinámico, similar al esquema anterior se numeran las barras o nodos de acuerdo a la cantidad de ramales conectados, comenzando con la numeración más baja para los nodos con menor cantidad de ramales, pero a diferencia del esquema estático la cuenta inicial de ramales por nodo se va modificando a medida que se progresa en el proceso de eliminación porque por ejemplo al eliminar los nodos con un ramal, los nodos adyacentes a estos se ven disminuidos en un ramal, lo mismo sucede cuando ocurre paralelismos.

De acuerdo con este esquema por ejemplo en los nodos -

numerados como 3, 4 y 5 en el ejemplo anterior y que contienen 2 ramales inicialmente cada uno, después de la eliminación del nodo 1 el nodo 5 tendrá solo un ramal, luego en el esquema dinámico este nodo debe ser procesado a continuación del 1 en lugar del 2 que aún continúa con dos ramales.

Entonces en el ejemplo anterior con el nuevo esquema el nodo 5 pasa a ser 3 y viceversa.

La figura N° 3.11., muestra la red del ejemplo con la numeración de nodos de acuerdo al esquema 2 y (3.37) muestra el patrón de las matrices $|Y|$ y $|U|$.

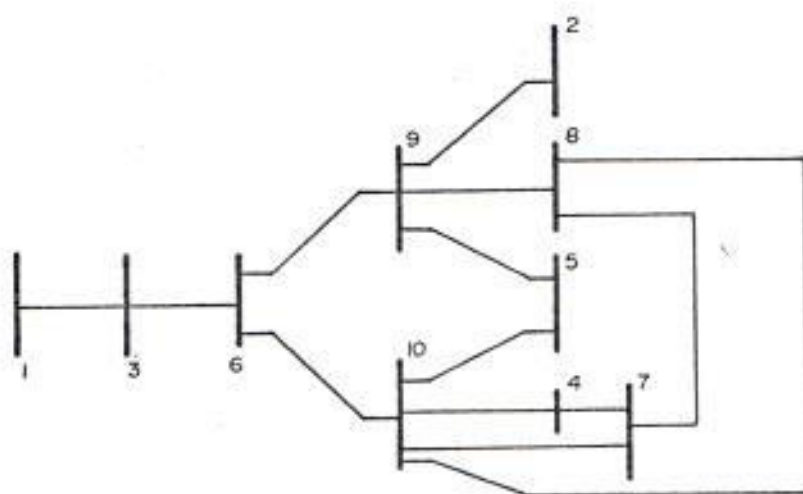


FIGURA N° 3.11. RED DE 10 NODOS NUMERADA DE ACUERDO A ESQUEMA 2

$$[Y] = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \end{matrix} & \begin{bmatrix} X & & X & & & & & & & \\ & X & & & & & & & & X \\ X & & X & & & X & & & & \\ & & & X & & & X & & & X \\ & & & & X & & & & & X \\ & & X & & & X & & & X & X \\ & & & X & & & X & X & & X \\ & & & & & & X & X & X & X \\ X & & & & X & X & & X & X & \\ & & & X & X & X & X & X & & X \end{bmatrix} \end{matrix}$$

$$[U] = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \end{matrix} & \begin{bmatrix} 1 & & X & & & & & & & \\ & 1 & & & & & & & & X \\ & & 1 & & & X & & & & \\ & & & 1 & & & X & & & X \\ & & & & 1 & & & & X & X \\ & & & & & 1 & & & X & X \\ & & & & & & 1 & X & & X \\ & & & & & & & 1 & X & X \\ & & & & & & & & 1 & \otimes \\ & & & & & & & & & 1 \end{bmatrix} \end{matrix}$$

(3.37)

Como vemos de (3.37) y (3.36) el esquema 2 es más óptimo que el esquema 1 ya que $|U|$ en (3.37) es más dispersa que $|U|$ en (3.36).

Ahora veamos que ocurre con la dispersidad de $|U|$, - cuando se tiene una numeración de nodos aleatoria como la mostrada en la figura N° 3.12.

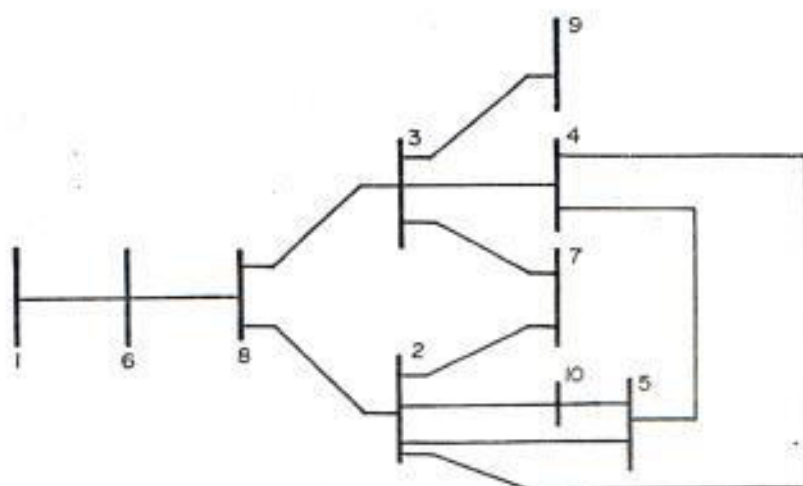


FIGURA N° 3.12. RED DE 10 NODOS CON NUMERACION ALEATORIA

	1	2	3	4	5	6	7	8	9	10
1	X					X				
2		X		X	X		X	X		X
3			X	X			X	X	X	
4		X	X	X	X					
5		X		X	X					X
6	X					X		X		
7		X	X				X			
8		X	X			X		X		
9			X						X	
10		X			X					X

$$|U| = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 1 & & & & & X & & & & \\ & 1 & & X & X & & X & X & & X \\ & & 1 & X & & & X & X & X & \\ & & & 1 & X' & & \otimes & \otimes & \otimes & \otimes \\ & & & & 1 & & \otimes & \otimes & \otimes & X'' \\ & & & & & 1 & & X & & \\ & & & & & & 1 & \otimes & \otimes & \otimes \\ & & & & & & & 1 & \otimes & \otimes \\ & & & & & & & & 1 & \otimes \\ & & & & & & & & & 1 \end{bmatrix} \quad (3.38)$$

Como vemos de (3.38) es más conveniente usar un esquema de ordenamiento de nodos para minimizar la cantidad de términos introducidos por el proceso de eliminación (optimizar dispersidad), también es más eficiente un esquema dinámico que uno estático.

ESQUEMA 3:

El esquema 3 es también un esquema de ordenamiento dinámico, los nodos son numerados de tal forma que a cada paso del proceso de eliminación se introduzca el menor número de elementos diferentes de ce

ro al efectuar su eliminación.

Es decir los nodos son numerados tal que en cada paso - de eliminación el próximo nodo a ser eliminado es uno que introduce el menor número de ramales equivalentes nuevos.

La selección de un esquema es un compromiso entre la velocidad de ejecución y el número de veces que el resultado va a ser usado, para el método de Newton - Raphson el esquema 2 se ha observado que es el mejor⁽⁸⁾.

3.3. TECNICAS DE PROGRAMACION

Las técnicas de programación son usualmente omitidas en la presentación de métodos de solución, en este caso, sin embargo, donde la programación remarca la diferencia entre el éxito o fracaso del método, especial atención debe ser dada a las técnicas de programación, ellas y el algoritmo numérico son inseparables, en estas técnicas los métodos de manejo de matrices dispersas deben ser usados, es decir un programa implementado con estas técnicas procesa y almacena únicamente los elementos diferentes de cero de las matrices, los elementos de las matrices diferentes de cero son almacenadas en tablas solidamente empaquetadas, esto

es un esquema de memoria compacta, y su direccionamiento es facilitado por tablas de índices auxiliares, a los nodos del sistema son asignadas numeraciones, siguiendo algunos de los esquemas de ordenamientos vistos en la sección anterior, este ordenado minimizará el llenado durante la triangulación de matrices en la eliminación gaussiana.

Las técnicas serán descritas en términos de un ejemplo de 7 nodos, mostrado en la figura N° 3.13., el ordenado no es óptimo intencionalmente para mostrar los efectos de llenado.

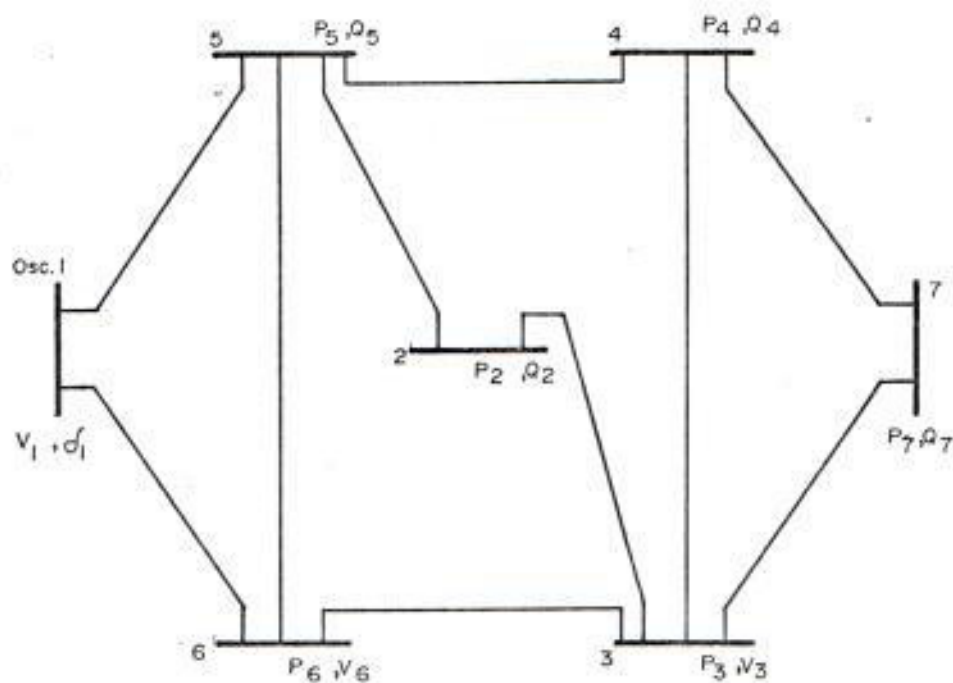


FIGURA N° 3.13. RED DE 7 NODOS

3.3.1. Tablas de almacenamiento para un ejemplo dado

A continuación se dan las tablas que pueden ser desarrolladas para almacenar la información para el flujo de potencia, en base al ejemplo de 7 nodos de la figura N° 3.13., cuyas condiciones dadas en los nodos se encuentran mostradas en la misma figura, así tenemos que los nodos 3 y 6 son de generación, los nodos 2, 4, 5 y 7 son de carga y el nodo 1 es el oscilante.

Las siguientes tablas son necesitadas:

TABLA I

LOC	POTENCIAS	NETAS	
		P	Q/V
1		-	-
2		P_2	Q_2
3		P_3	V_3
4		P_4	Q_4
5		P_5	Q_5
6		P_6	V_6
7		P_7	Q_7

TABLA II

ADMITANCIAS DIAGONALES

LOC	G_{kk}	B_{kk}	LOC. EN III
1	G_{11}	B_{11}	1
2	G_{22}	B_{22}	3
3	G_{33}	B_{33}	5
4	G_{44}	B_{44}	9
5	G_{55}	B_{55}	12
6	G_{66}	B_{66}	16
7	G_{77}	B_{77}	19
8	-	-	21

TABLA III

ADMITANCIA FUERA DE LA DIAGONAL

LOC	M	G_{km}	B_{km}
1	5	G_{15}	B_{15}
2	6	G_{16}	B_{16}
3	3	G_{23}	B_{23}
4	5	G_{25}	B_{25}
5	2	G_{32}	B_{32}
6	4	G_{34}	B_{34}
7	6	G_{36}	B_{36}
8	7	G_{37}	B_{37}
9	3	G_{43}	B_{43}
10	5	G_{45}	B_{45}
11	7	G_{47}	B_{47}
12	1	G_{51}	B_{51}
13	2	G_{52}	B_{52}
14	4	G_{54}	B_{54}
15	6	G_{56}	B_{56}
16	1	G_{61}	B_{61}
17	3	G_{63}	B_{63}
18	5	G_{65}	B_{65}
19	3	G_{73}	B_{73}
20	4	G_{74}	B_{74}
21	-	-	-

TABLA IV
VOLTAJES

LOC	e_k	f_k			
1	e_1	f_1	=	V_1	+ j0
2	e_2	f_2	=	V_1	+ j0
3	e_3	f_3	=	V_3	+ j0
4	e_4	f_4	=	V_1	+ j0
5	e_5	f_5	=	V_1	+ j0
6	e_6	f_6	=	V_6	+ j0
7	e_7	f_7	=	V_1	+ j0

Donde:

TABLA I En esta tabla se almacenan las potencias ac tivas y reactivas netas especificadas en cada nodo, en los nodos de generación - el voltaje especificado es almacenado en lugar de Q, esto es redundante para el algoritmo básico, pero es necesario cu an do hay límites de reactivo impuestos en los nodos de generación.

TABLA II La tabla II contiene los elementos de la diag o nal de la matriz admitancia, la tercera colu m

na contiene la posición donde comienza la información de los elementos fuera de la diagonal del nodo respectivo en la tabla III, por ejemplo la fila 3 comienza en la localización 5 de la tabla III y la fila 4 comienza en la localización 9.

TABLA III La tabla III contiene los elementos diferentes de cero fuera de la diagonal de la matriz admitancia, aunque la matriz admitancia es simétrica en valor, esta es almacenada en forma doble para ahorrar tiempo en su localización, la columna M da el número de la columna de cada elemento.

TABLA IV En esta tabla se almacenan los voltajes de nodo, este vector de voltajes puede ser inicializado como sigue.

En los nodos de carga la magnitud de voltaje es ajustada igual a la del nodo oscilante, y en los nodos de generación la magnitud del voltaje es ajustada al valor especificado en los datos y el ángulo es ajustado igual al del nodo oscilante.

3.3.2. Cálculo de elementos del Jacobiano

Aquí se muestra la secuencia para el cálculo del jacobiano, se presenta únicamente la formación del jacobiano correspondiente al nodo 2, lo cual es suficiente para indicar como el resto de los elementos se rían calculados, ya que el nodo 2 es un nodo de carga tendremos una fila doble en el jacobiano luego para el ejemplo de la figura N° 3.13., tendremos:

$$k = 2$$

$$m = 3$$

$$a_3 - jb_3 = (G_{23} - jB_{23}) (e_3 - j f_3)$$

$$\underline{\underline{H_{23}}} = a_3 f_2 - b_3 e_2$$

$$\underline{\underline{J_{23}}} = -(a_3 e_2 + b_3 f_2)$$

$$m = 5$$

$$a_5 - j b_5 = (G_{25} - j B_{25}) (e_5 - j f_5)$$

$$\underline{\underline{H_{25}}} = a_5 f_2 - b_5 e_2$$

$$\underline{\underline{N_{25}}} = a_5 e_2 + b_5 f_2$$

$$\underline{J_{25}} = -N_{25}$$

$$\underline{L_{25}} = H_{25}$$

$$P_2 + jQ_2 = (e_2 + j f_2) \{ (e_3 - j b_3) + (e_5 - j b_5) \}$$

$$V_2^2 = (e_2^2 + f_2^2)$$

$$\underline{H_{22}} = -Q_2 - B_{22} V_2^2$$

$$\underline{N_{22}} = P_2 + G_{22} V_2^2$$

$$\underline{J_{22}} = P_2 - G_{22} V_2^2$$

$$\underline{L_{22}} = Q_2 - B_{22} V_2^2$$

$$\underline{\Delta P_2} = P_2^{esp.} - P_2$$

$$\underline{\Delta Q_2} = Q_2^{esp.} - Q_2$$

Los símbolos subrayados son los elementos que constituyen la doble fila del jacobiano correspondiente al nodo 2. En la próxima iteración, los valores de todos los elementos habrán cambiado debido a que los voltajes: $e + j f$ han sido cambiados con respecto a la primera iteración.

3.3.3. Esquema de Triangularización

El proceso de eliminación de Gauss para la triangu-

larización del jacobiano se lo realiza en forma de fila ya que este esquema nos permite que las filas del jacobiano pueden ser formadas y operadas por eliminación al mismo tiempo, de tal forma que los requerimientos de memoria se ahorran al máximo, y nunca excederán la capacidad de una matriz triangular superior. Para cumplir con esto se define lo que se llama una fila de trabajo, este procedimiento consiste en:

1. Construída la matriz admitancia de barra y almacenada en tablas compactas, se construye la fila que corresponde al jacobiano, esta fila del jacobiano se almacena en un vector fila de trabajo, cuya capacidad corresponde unicamente para almacenar una fila del jacobiano aumentado por una posición más para los residuos ΔP ó ΔQ .
2. Una vez construída la fila se realizan las operaciones de eliminación de Gauss sobre ella (se triangula).
3. Una vez triangulada la fila del jacobiano y aumentado por la columna del residuo del vector independiente ΔP ó ΔQ , se transfiere toda la fila ope

rada a otra posición de memoria o tabla que irá a contener todo el jacobiano triangulado en forma compacta ; luego se vuelve a construir la siguiente fila del jacobiano en la fila de trabajo y se triangula ésta, haciendo operaciones entre la fila de trabajo y las filas del jacobiano ya triangulados que se encuentran en la tabla que contiene a la matriz jacobiano-triangulado, una vez triangulada esta nueva fila, se la transfiere a la tabla del jacobiano triangulado. La figura N° 3.14., ilustra graficamente este proceso

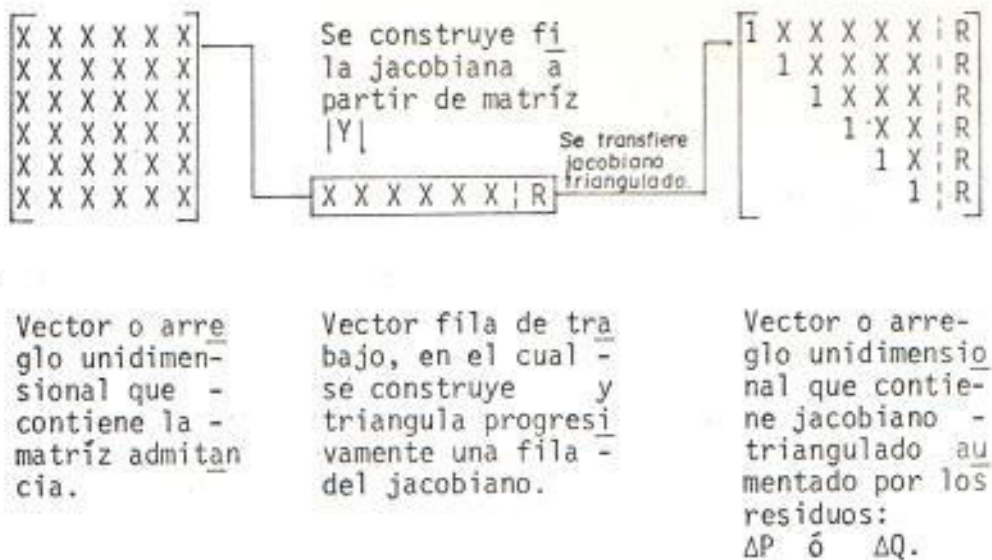


FIGURA N° 3.14. PROCESO DE FORMACION Y TRIANGULACION DE JACOBIANO.

A continuación se muestra como triangulamos las filas 2,3 y 4 del jacobiano del ejemplo de la figura N° 3.13.

Fila 2 como fue formado en 3.3.2. 2

	2	3	4	5	6	7	
H	N	H		H	N		ΔP
J	L	J		J	L		ΔQ

Fila 2 después de la triangulacion y normalizacion. 2

Nueva fila 3 3

1	N' 1	H' J'		H' J'	N' L'		$\Delta P'$ $\Delta Q'$
H	N	H	H N		H	H N	ΔP

Fila 3 es ahora combinada con la fila 2 para eliminar H_{32} , N_{32} , y es normalizada. 2

3

1	N' 1	H' J'		H' J'	N' L'		$\Delta P'$ $\Delta Q'$
		1	$H'N'$	H'	N'	H'	$H'N'$

2

3

Nueva fila 4 4

1	N' 1	H' J'		H' J'	N' L'		$\Delta P'$ $\Delta Q'$
		1	$H'N'$	H'	N'	H'	$H'N'$
		H	H N	H	N	H	H N
		J	J L	J	L	J	J L

2

3

4

Después de triangularizar y normalizar la fila 4.

1	N' 1	H' J'		H' J'	N' L'		$\Delta P'$ $\Delta Q'$
		1	$H'N'$	H'	N'	H'	$H'N'$
			1	N' 1	H' J'	N' L'	H' $J'L'$

3.3.4. Esquema de almacenamiento para jacobiano triangulado

El esquema de almacenamiento del jacobiano triangularizado para el ejemplo de 7 nodos es mostrado a continuación.

	2	3	4	5	6	7
2	HN JL	H J		HN JL		
3	HN	H	HN		H	HN
4		H J	HN JL	HN JL		HN JL
5	HN JL		HN JL	HN JL	H J	
6		H		HN	H	
7		H J	HN JL			HN JL

$$= \begin{bmatrix} \Delta\delta_2 \\ \Delta V_2/V_2 \\ \Delta\delta_3 \\ \Delta\delta_4 \\ \Delta V_4/V_4 \\ \Delta\delta_5 \\ \Delta V_5/V_5 \\ \Delta\delta_6 \\ \Delta\delta_7 \\ \Delta V_7/V_7 \end{bmatrix} = \begin{bmatrix} \Delta P_2 \\ \Delta Q_2 \\ \Delta P_3 \\ \Delta P_4 \\ \Delta Q_4 \\ \Delta P_5 \\ \Delta Q_5 \\ \Delta P_6 \\ \Delta P_7 \\ \Delta Q_7 \end{bmatrix}$$

MATRIZ JACOBIANO ORIGINAL

1	N'	H'		H' N'			
	1	J'		J' L'			
		1	H' N'	H N	H'	H' N'	
			1 N'	H' N'	H	H' N'	
			1	J' L'	J	J' L'	
				1 N'	H'	H N	
				1	J'	J L	
					1	H N	
						1 N'	
							1

MATRIZ JACOBIANO TRIANGULARIZADO

Donde:

Las áreas sombreadas son los términos llenados o sea creados durante el proceso de eliminación.

Las primas indican valores cambiados durante la eliminación.

TABLA VI

ESQUEMA DE ALMACENAMIENTO DE MATRIZ JACOBIANO TRIANGULADO

TABLA V

ESQUEMA DE ALMACENAMIENTO DE LOS RESIDUOS ΔP y ΔQ .

ΔP_2	ΔQ_2	1
ΔP_3	x	-10
ΔP_4	ΔQ_4	21
ΔP_5	ΔQ_5	35
ΔP_6	x	-44
ΔP_7	ΔQ_7	47

	1	N_{22}		21	N_{44}
	2	-3		22	5
F	3	H_{23}		23	H_{45}
i	4	J_{23}	F	24	J_{45}
1	5	5	i	25	N_{45}
a	6	H_{25}	1	26	L_{45}
	7	J_{25}	a	27	-6
2	8	N_{25}		28	H_{46}
	9	L_{25}	4	29	J_{46}
	10	4		30	7
	11	H_{34}		31	H_{47}
	12	N_{34}		32	J_{47}
F	13	5		33	N_{47}
i	14	H_{35}		34	L_{47}
1	15	N_{35}		35	N_{55}
a	16	-6	F	36	-6
	17	H_{36}	i	37	H_{56}
	18	7	1	38	J_{56}
3	19	H_{37}	a	39	7
	20	N_{37}		40	H_{57}
			5	41	J_{57}
				42	N_{57}
				43	L_{57}
			F	44	7
			i	45	H_{67}
			1	46	N_{67}
			a	47	N_{77}
			F	48	-
			i		
			1		
			a		

- llega a ser $\Delta\delta$.
- llega a ser $\Delta V/V$ después de la sustitución hacia atrás.
- para los nodos de generación no tiene importancia a menos que viole el reactivo y se convierta en nodo de carga.

La descripción muestra como la matriz jacobiano y la columna de residuos ΔP o ΔQ son formadas, triangularizadas y almacenadas en tablas compactas V y VI.

El direccionamiento en este esquema de memoria es tal que cualquier fila del jacobiano puede ser identificada y las coordenadas (columnas) de sus elementos diferentes de cero directamente interpretados, en la siguiente sección se explica esto en más detalle.

3.3.5. Esquema de Direccionamiento

Las primeras dos columnas de cualquier fila k en la tabla V contiene los residuos $\Delta P_k, \Delta Q_k$, resultantes del procesamiento de la fila k . Si la fila k pertenece a un nodo de generación el ingreso ΔQ_k es dejado en blanco, la tercera columna de la tabla V contiene la dirección de la localización de partida de la fila k simple o doble del jacobiano triangularizado almacenado en la tabla VI. Si la dirección tiene un prefijo negativo, la fila es simple y corresponde a un nodo de generación, de otra forma la fila es do

ble y corresponde a un nodo de carga.

Los ingresos en la tabla VI son una mezcla de cantidades reales representativas, de elementos del jacobiano triangularizado H , J , N , L y enteros identificadores que identifican las columnas de los elementos.

El primer ingreso en alguna fila k que pertenece a un nodo de generación es el identificador de la primera columna diferente de cero a la derecha de la diagonal, si el identificador es negativo el próximo ingreso que continua es H_{km} , donde m es la columna señalada por el identificador, seguido por el identificador para la próxima columna diferente de cero. Si el identificador es positivo los siguientes dos ingresos son H_{km} y N_{km} seguido por el próximo identificador.

El primer ingreso para una fila doble (nodo de carga) es N_{kk} seguido por un identificador de la primera columna diferente de cero a la derecha de la diagonal. Si el identificador es negativo, los próximos dos ingresos son H_{km} y J_{km} , seguido por el identificador de la próxima columna diferente de

cero.

El fin de una fila es señalada cuando un contador alcanza la dirección de partida de la próxima fila (el ingreso de la fila $k + 1$ en la columna 3 de la tabla V). A medida que el proceso de triangularización procede, es necesario recuperar e identificar elementos de las filas previamente procesadas.

Este esquema facilita esta operación.

Cuando $k = n$ es decir se tiene el jacobiano completamente triangularizado, el proceso de sustitución hacia atrás puede ser ejecutado, fila por fila desde $k = n$, hasta $k = 2$ recuperando filas y residuos.

3.3.6. Esquema de eliminación Gaussiana (EJEMPLO)

Es necesario usar alguna clase de esquema de fila de trabajo para el proceso de eliminación Gaussiana sobre la matriz jacobiano del método de Newton.

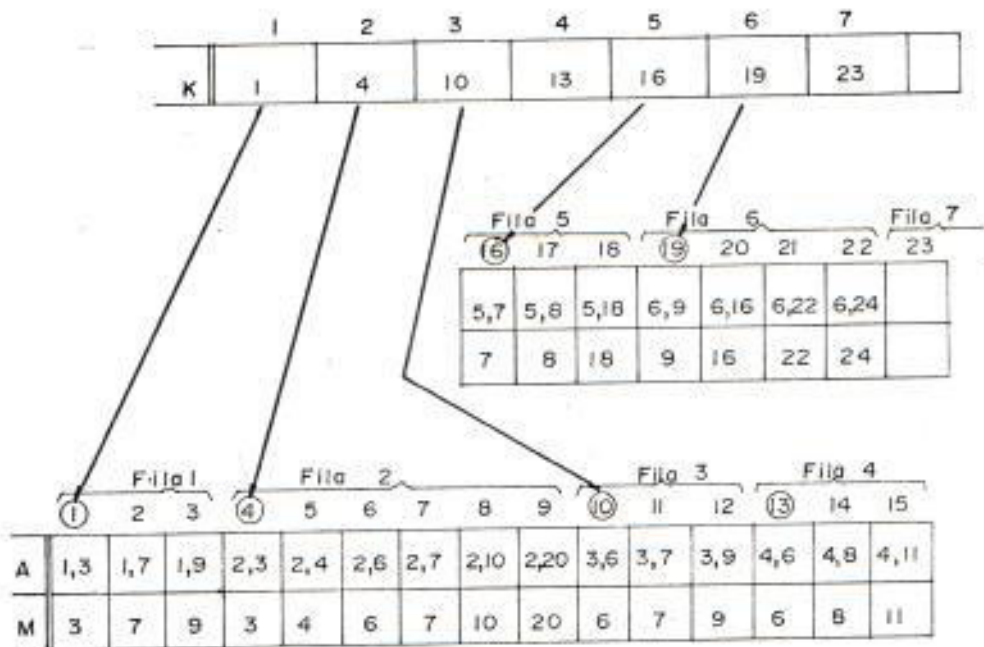
A continuación se muestra uno de estos esquemas, a fin de aclarar la idea esencial del esquema, esto es mostrado primero para una matriz real asimétrica -

sin introducir las complicaciones adicionales de la estructura especial de la matriz jacobiano del flujo de potencia.

Asumimos una matriz en la cual la eliminación es procesada fila por fila, las primeras 6 filas ya han sido trianguladas y normalizadas (sus términos diagonales son unitarios), la fila 7 será procesada a continuación.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
1	1	X					X	X																	
2		1	X	X			X	X			X										X				
3			1				X	X		X															
4				1			X	X			X														
5					1		X	X											X						
6						1		X								X						X		X	
7		X	X				X				X						X								

La matriz procesada parcialmente es almacenada en un arreglo compacto como se muestra a continuación:



La tabla k indica la posición de partida de cada fila en las tablas A y M.

La tabla A contiene los valores en números reales de los elementos de la matriz A.

La tabla M los números de columna de los elementos diferentes de cero de la matriz A.

El esquema de almacenamiento para la matriz jacobiano triangularizado del flujo de potencia combina las tablas A y M en un solo arreglo unidimensional.

Las siguientes ilustraciones dan una idea como la fila 7 es procesada.

La fila 7 es procesada por medio de tablas de trabajo de una sola dimensión llamadas KOL, NEXT, ROW.

	1	2	3	4	5	6	7	8	9
KOL	2	4	7	11	17				
NEXT	2	3	4	5	0		"0" indica el fin de la fila.		
ROW	X	X	X	X	X				

KOL contiene los índices de columnas de los elementos diferentes de cero de la fila 7.

NEXT indica la dirección del próximo término en secuencia de columna.

ROW contiene los elementos de la matriz a ser procesados.

El primer paso se eliminar los términos diferentes de cero en la columna 2, multiplicando los elementos de la fila 2 previamente procesadas por este elemento

y restándolos de los correspondientes elementos de la fila 7, aquí estamos interesados con el almacenamiento de estos elementos, a fin de ayudar al seguimiento del proceso, la porción de la fila 2 de las tablas A y M, es repetido.

	4	5	6	7	8	9
M	3	4	6	7	10	20
A	X	X	X	X	X	X

	1	2	3	4	5	6	7	8	9
KOL	2	4	7	11	17	③	← KOL(6) = M(4)		
NEXT	⑥	3	4	5	0	②	← Va a la próxima localización (2)		
ROW	X	X	X	X	X	⊗	← ROW(6) = -ROW(1)*A(4)		

Va a la próxima localización (6).

La fila 2 contiene un término diferente de cero en la columna 3, pero la fila 7 no, por lo tanto este término no es ingresado en la localización 6 de KOL, NEXT, ROW, los ingresos cambiados o añadidos son encerrados en un círculo.

	1	2	3	4	5	6	7	8	9
KOL	2	4	7	11	17	3			
NEXT	6	3	4	5	0	2			
ROW	X	⊗	X	X	X	X	ROW(2) = ROW(2) - ROW(1)*A(5)		

La fila 2 contiene un término diferente de cero en la

columna 4 y lo mismo la fila 7, por lo tanto no habrá un nuevo ingreso.

	1	2	3	4	5	6	7	8	9
KOL	2	4	7	11	17	3	⑥		
NEXT	6	⑦	4	5	0	2	③		
ROW	X	X	X	X	X	X	⊗		

$\text{KOL}(7) = M(6)$
 Va a la próxima localización (3)
 $\text{ROW}(7) = -\text{ROW}(1) * A(6)$
 Va a la próxima localización (7).

La fila 2 contiene un término en la columna 6, pero la fila 7 no, por lo tanto este término es ingresado en la localización 7 de KOL, NEXT, ROW.

	1	2	3	4	5	6	7	8	9
KOL	2	4	7	11	17	3	6		
NEXT	6	7	4	5	0	2	3		
ROW	X	X	⊗	X	X	X	X		

$\text{ROW}(3) = \text{ROW}(3) - \text{ROW}(1) * A(7)$

La fila 2 contiene un término en la columna 7 y la fila 7 también.

	1	2	3	4	5	6	7	8	9
KOL	2	4	7	11	17	3	6	10	$KOL(8)=M(8)$
NEXT	6	7	8	5	0	2	3	4	Va a la próxima localización (4)
ROW	X	X	X	X	X	X	X	X	$ROW(8)=-ROW(1)*A(8)$ Va a la próxima localización (8)

La fila 2 contiene un término en la columna 10, pero -
la fila 7 no.

	1	2	3	4	5	6	7	8	9
KOL	2	4	7	11	17	3	6	10	20 $KOL(9)=M(9)$
NEXT	6	7	8	5	9	2	3	4	0 Ultimo término de la fila.
ROW	X	X	X	X	X	X	X	X	X $ROW(9)=-ROW(1)A(9)$ Va a la próxima localización (9).

La fila 2 contiene un término en la columna 20, la fila 7 no, esto completa la eliminación de la columna 2 de la fila 7.

El próximo paso es eliminar la columna 3 de la fila 7.

	1	2	3	4	5	6	7	8	9	10
KOL	2	4	7	11	17	3	6	10	20	
NEXT	6	7	8	5	9	2	3	4	0	
ROW	X	X	X	X	X	X	X	X	X	$ROW(7)=ROW(7)-ROW(6)*A(10)$

La fila 3 contiene un término en la columna 6 y la fila 7 también.

	1	2	3	4	5	6	7	8	9	10
KOL	2	4	7	11	17	3	6	10	20	
NEXT	6	7	8	5	9	2	3	4	0	
ROW	X	X	(X)	X	X	X	X	X	X	ROW(3)=ROW(3)-ROW(6) *A(11)

	1	2	3	4	5	6	7	8	9	10
KOL	2	4	7	11	17	3	6	10	20	(9)-KOL(10)=M(12)
NEXT	6	7	(10)	5	9	2	3	4	0	(8)-Va a la próxima localización(8).
ROW	X	X	X	X	X	X	X	X	X	(X)-ROW(10)=-ROW(6)*A(12)

Esto completa la eliminación de la columna 3 de la fila 7.

El próximo paso es eliminar la columna 4 de la fila 7. La tabla es mostrada al final de este paso.

	1	2	3	4	5	6	7	8	9	10	11	12	13
KOL	2	4	7	11	17	3	6	10	20	9	8		
NEXT	6	7	(11)	5	9	2	3	4	0	8	(10)		
ROW	X	X	X	X	X	X	X	X	X	X	X		

Puesto que la fila 7 no tiene un elemento en la columna 5, la próxima columna a ser eliminada es la columna 6. Las tablas son mostradas en el final de este paso.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
KOL	2	4	7	11	17	3	6	10	20	9	8	16	22	24
NEXT	6	7	11	12	9	2	11	4	13	8	10	5	14	0
ROW	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Comenzando en la localización 11, los ingresos en KOL - son transferidos a la tabla M y los ingresos en la fila son multiplicados por el recíproco de ROW(3) y transferido a la tabla A en el orden indicado por el arreglo - NEXT.

Así:

$$M(23) = KOL(11)$$

$$M(24) = KOL(10)$$

$$M(25) = KOL(8)$$

$$A(23) = ROW(11)/ROW(3)$$

$$A(24) = ROW(10)/ROW(3)$$

$$A(25) = ROW(8)/ROW(3), \text{etc.}$$

	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
A	6,9	6,16	6,22	6,24	7,8	7,9	7,10	7,11	7,16	7,17	7,20	7,22	7,24		
M	9	16	22	24	8	9	10	11	16	17	20	22	24		

Comienzo de la fila 8

	1	2	3	4	5	6	7	8
K	1	4	10	13	16	19	23	32

3.3.7. Esquema de eliminación Gaussiana para jacobiano del flujo de potencia

Hay cuatro tipos de combinaciones de elementos en el jacobiano para el flujo de potencia, dependiendo de los dos tipos de nodos, de carga o de generación.

Aquí mostraremos la situación para la eliminación de la columna 5 de la fila 11 de un problema hipotético para cada una de las 4 posibles combinaciones mencionadas. La matriz jacobiano ha sido triangularizada hasta la fila 10.

La fila 5 es almacenada en un arreglo compacto que no es mostrado aquí a fin de ayudar a visualizar el proceso; esta representada como una fila de una matriz. Ver figura N° 3.15.

La fila de trabajo análoga a la tabla ROW en el ejemplo precedente ahora consiste de 4 tablas ROW_H, ROW_J, ROW_N y ROW_L, como en el ejemplo precedente las operaciones son guiadas por las tablas KOL y NEXT.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	
1																						
2																						
3																						
4																						
5					X		X		X		X	X		X		X			X			
6					↓																	
7					↓																	
8					↓																	
9					↓																	
10					↓																	
11					X	X	X			X	X		X	X				X				X

FIGURA Nº 3.15. MATRIZ EJEMPLO PARA ELIMINACION DE UN ELEMENTO DE JACOBIANO

Las tablas o arreglos son mostrados como ellos aparecerían al final del paso de eliminación indicado.

Nodo	5	11
Tipo	Generación	Generación

A continuación se muestra la fila correspondiente al nodo 5 ya triangularizada y la fila 11 que se va a procesar, esto es una representación para comprensión, en realidad así no está almacenado sino en forma compacta.

	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
FILA 5	1		HN		HN		H	H		H		HN			H		
FILA 11	H	H	HN			H	H		H	H				H			HN

Aquí se presenta la fila 11 ya trinagularizada tal como está empaquetado en la memoria.

	1	2	3	4	5	6	7	8	9	10	11	12	13
NEXT	2	3	10	5	11	7	12	13	0	4	6	8	9
KOL	5	6	7	10	11	13	14	18	21	9	12	16	19
ROWH	X	X	X	X	X	X	X	X	X	X	X	X	X
ROWJ													
ROWN			X						X	X		X	
ROWL													

2.	NODO	5	11
	TIPO	CARGA	GENERACION

	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
FILA 5	1N		HN		HN		H	H		H		HN			H		
	1		JL		JL		J	J		J		JL			J		
FILA 11	HN	H	HN			H	H		H	H				H			HN

	1	2	3	4	5	6	7	8	9	10	11	12	13
NEXT	2	3	10	5	11	7	12	13	0	4	6	8	9
KOL	5	6	7	10	11	13	14	18	21	9	12	16	19
ROWH	X	X	X	X	X	X	X	X	X	X	X	X	X
ROWJ													
ROWN	X		X						X	X		X	
ROWL													

3)	NODO	5	11
	TIPO	CARGA	CARGA

	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
FILA 5	HN		HN		HN	H		H		HN			H				
	JL		JL		JL	J		J		JL			J				
FILA 11	HN	H	HN		H	HN		H	H				H				HN
	JL	J	JL		J	JL		J	J				J				JL

	1	2	3	4	5	6	7	8	9	10	11	12	13
NEXT	2	3	10	5	11	7	12	13	0	4	6	8	9
KOL	5	6	7	10	11	13	14	18	21	9	12	16	19
ROWH	X	X	X	X	X	X	X	X	X	X	X	X	X
ROWJ	X	X	X	X	X	X	X	X	X	X	X	X	X
ROWN	X		X		X				X	X		X	
ROWL	X		X		X				X	X		X	

4.	NODO	5	11
	TIPO	GENERACION	CARGA

5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

FILA 5	1		HN		HN		HN	H		H		HN			H		
FILA 11	H	H	HN			H	HN		H	H					H		HN
	J	J	JL			J	JL		J	J					J		JL

1 2 3 4 5 6 7 8 9 10 11 12 13

NEXT	2	3	10	5	11	7	12	13	0	4	6	8	9
KOL	5	6	7	10	11	13	14	18	21	9	12	16	19
ROWH	X	X	X	X	X	X	X	X	X	X	X	X	X
ROWJ	X	X	X	X	X	X	X	X	X	X	X	X	X
ROWN			X		X	X			X	X		X	
ROWL			X		X	X			X	X		X	

CAPITULO IV

APLICACION DEL METODO DE NEWTON - RAPHSON EN LA SOLUCION DEL FLUJO DE POTENCIA

En este Capítulo, se aplica el método de Newton - Raphson, en la solución del flujo de potencia de un ejemplo, utilizando un programa completo que usa este método con todas sus técnicas, dicho programa ha sido ambientado en el sistema 4341 de la Escuela Superior Politécnica del Litoral (ESPOL), y ya que uno de los objetivos de este trabajo es permitir el avance que sobre el método de Newton - Raphson se puede hacer, se desarrollará aquí adicionalmente la estructura del programa, las funciones de sus principales subrutinas, forma de usarlo, etc., que permitirán cumplir con este objetivo.

Este programa tiene una serie de facilidades extendidas sobre el método de Newton - Raphson como son los transformadores con cambio automático de Tap para el control del voltaje en barra local o remota, chequeo de límite de reactivo en los nodos de generación, control de intercam-

bio de potencia entre áreas y contingencias.

Las contingencias son las salidas de servicio de líneas y/o transformadores, de generadores y/o carga, el programa maneja estas situaciones con subrutinas especiales (se verán en las siguientes secciones), que permiten añadir, cambiar, sacar de servicio, reconectar o remover líneas y/o transformadores, todo esto lo hacen estas subrutinas modificando las tablas almacenadas que contienen los datos del sistema a resolver, el proceso es puramente computacional, una vez modificadas las tablas que contienen por ejemplo a la matriz admitancia para el caso de líneas y/o transformadores o los datos de barra para el caso de generadores y/o cargas, el jacobiano es nuevamente reevaluado y se inicia un proceso iterativo con la solución anterior o los datos iniciales.

Para el control de intercambio de potencia entre áreas la red del sistema es dividida en varias subredes mutuamente exclusivas o llamadas áreas. La potencia neta especificada de intercambio entre áreas es controlada por un generador dentro de cada área (nodo oscilante del área).

El nodo oscilante del sistema es también el nodo oscilante de su propia área.

La potencia real en el nodo oscilante de cada área es una variable a ser determinada por la solución del flujo de potencia.

4.1. ESTRUCTURA DEL PROGRAMA A USAR

El programa a usar consta de 69 subrutinas, una función del programa (no de biblioteca) y un programa principal llamado "MAIN" que es el que maneja internamente todo el programa, el control interno del programa es manejado por el usuario mediante códigos de control que se suministran al programa conjuntamente con los datos, a continuación se da la estructura del programa, donde los números de la izquierda indican los respectivos códigos de control, que cuando son ingresados el programa llamará a las subrutinas a la derecha de dicho código, para ejecutar la función deseada que se indica en la próxima sección, en la tabla VIII, se muestra la acción a tomar con el respectivo código de control, por ejemplo con el Código de control 7 se pueden cambiar los datos de líneas y/o transformadores, para una mejor comprensión de esta estructura, tome en cuenta que:

1. Los nombres de subrutinas entre paréntesis son subrutinas que a su vez invocan o llaman a otra u

otras subrutinas que a la vez pueden llamar a más subrutinas.

2. Los nombres de subrutinas que no se encuentran entre paréntesis son subrutinas que no invocan a ninguna otra subrutina.
3. Las barras // indican que el programa puede o no ejecutar la subrutina o subrutinas entre barras, dependiendo de una opción dada en el código de control.
4. Los puntos: indican que la función o subrutinas a continuación son llamadas por la subrutina que los precede.

El programa principal antes de la ejecución de los códigos de control llama a las subrutinas INIT, PGHEAD, CLOCK, COMPRT y ERROR, a su vez la subrutina PGHEAD llama a las subrutinas DATE y CLOCK, esta subrutina siempre llama a estas otras dos subrutinas aunque en los códigos de control no se lo muestra para abreviar la estructura.


```

1 (PGHEAD) (PGHEAD)
2 COMNTS
3 (PGHEAD) / (RECALL) / (DACL) / (PGHEAD)
      MSIN      MSIN
      MSOUT     MSOUT
      DABIN
4 (LCHNGE)
      MSIN
      MSOUT
      (PGHEAD)
      BASLIN
      SORT
5 (BCHNGE)
      (PGHEAD)
      I2 SRCH
      I2 INST
      I4 INST
      I2 DELT
      I4 DELT
6 (ALCON)
      (PGHEAD)
7 (LCHNGE)
      (ADDIT) (CONVERT) (OUTAGE) (RECLOS) (REMOVE)
      I4 SRCH I4SRCH I4SRCH I4 SRCH I4SRCH
      I4 INST I4INST I4 DELT I4 INST I4 DELT
      I2 INST I2INST I2 DELT I2 INST I2 DELT
      DEHIST I2DELT INHIST DEHIST DEHIST
      INHIST I4 DELT INHIST
8 (BCHNGE)
      I2 SRCH
      I2 INST
      I4 INST
      I2 DELT
      I4 DELT
9 RDAREA / (LOPUT) / (BOPUT) / (MOPUT) / (NUDECK)
      INCONV MSIN (PGHEAD) MSIN MSIN
              MSOUT (PGHEAD) MSOUT MSOUT
              (PGHEAD) (PGHEAD) (PGHEAD)
10 (PSYCHO)
      MSIN
      MSOUT
      (CHOP) (PGHEAD)
11 (EDIT) (OPTIMO) / MSOUT / CLOCK (ROLOUT) (INSURE) (MATGEN)
      MSIN MSIN MSIN (PGHEAD) EXNODE
      MSOUT MSOUT MSOUT (PGHEAD) EXBRAN
                                      (LITCHCO) .. EXTIES
                                      ELIM
                                      BAKSUB
      (ADJUST) CLOCK (FLOW) (MOPUT) (PREOUT) / (TIEOUT) / (OUTPUT)
      MSIN MSIN MSIN (PGHEAD) (PGHEAD) RDAREA .. INCONV
      MSOUT MSOUT MSOUT MSIN (PGHEAD) MSIN
                                      (PGHEAD)
      / (SINIT) (SISOLV) CLOCK (MOPUT) (SISUMM) (SIOPUT) / CLOCK
      MSOUT (PGHEAD) MSIN (PGHEAD) (SIOPUT) / CLOCK
      MSIN SINODE MSOUT MSIN (PGHEAD) .. INCONV
      SIBRAN (PGHEAD) MSOUT MSIN
      MSIN MSIN
12 (SAVE) / (DASAVE) /
      DATE DATE
      TIME TIME
      MSOUT DPRIME
      MSIN DABOUT
      DFLUSH
      MSOUT
      MSIN
13 (LISTIT)
      (PGHEAD)
14 (REDEF)
      MSOUT
15 AREAIN
22 (LODCHG)
      (PGHEAD)
30 EXIT Fin del Programa...
  
```

ESTRUCTURA DEL PROGRAMA

Como usted puede ver, si usted desea reducir el programa (puede ser por razones de capacidad del computador o por alguna aplicación), por ejemplo supongamos que nunca va a ejecutar el código de control 12, entonces las subrutinas SAVE y DASAVE pueden ser eliminadas, siempre y cuando no sean invocadas en alguna otra parte de la estructura (llamadas por otro código de control). Igualmente TIME, DPRIME, DABOUT, y DFLUSH, que no son llamadas en ninguna otra parte del programa pueden ser eliminadas, no así DATE, MSIN y MSOUT que son subrutinas que son llamadas por otras subrutinas del programa, con esta acción se han reducido 6 subrutinas al programa que imposibilitan la ejecución del control 12, pero el resto del programa permanece con todas sus capacidades.

4.2. FUNCION DE PRINCIPALES SUBRUTINAS

Para una mayor comprensión del programa a usarse a continuación se dan la función específica de las subrutinas que constituyen el programa.

ADDIT Rutina para hacer adición de líneas y/o transformadores en las tablas del caso base, llamada para contingencia de líneas y transformadores.

ADJUST	Rutina que realiza ajuste a los voltajes entre iteraciones a las barras con control de voltaje.
ALCON	Rutina que es llamada cuando se requieren leer nuevas constantes del programa.
AREAIN	Rutina para leer información referente a intercambio entre áreas.
BAKSUB	Rutina para ejecutar proceso de sustitución hacia atrás en el jacobiano triangulado y <u>au</u> mentado.
BASLIN	Rutina para ensamblar datos de línea y <u>trans</u> formadores del caso base.
BCHNGE	Rutina para leer datos de barra del caso - base y/o cambios en los datos de barra.
BOPUT	Rutina que imprime información de datos de barra
CHOP	Rutina que es llamada para imprimir tabla de <u>his</u> toria de cambios (hechos para contingencias).
CLOCK	Rutina que retorna ceros para el tiempo.
COMNTS	Rutina que es llamada para leer e imprimir <u>co</u> mentarios desde los datos.

COMPRT	Rutina para imprimir comentarios actualizados.
CONVERT	Rutina llamada para contingencias que involucran modificación de datos de línea y/o transformadores.
DABIN	Rutina para leer un arreglo A de longitud N desde el Archivo de acceso directo histórico.
DABOUT	Rutina para escribir un arreglo A de longitud N, hacia el archivo de acceso directo histórico.
DACALL	Rutina para cargar a memoria un caso de flujo desde el archivo de acceso directo histórico.
DASAVE	Rutina para grabar un caso de flujo hacia el archivo de acceso directo histórico.
DATE	Rutina que retorna caracteres de mes, día y año.
DEHIST	Rutina llamada para borrar parte de información contenida en la tabla de historia de cambios (solo es llamado por rutinas que ejecutan contingencia de líneas y/o transformadores).

- DFLUSH Rutina que almacena información en el archivo de acceso directo histórico, haciéndola avanzar a través de los registros del archivo, a la vez que actualiza el apuntador de las pistas vacías del archivo si es necesario.
- DPRIME Lee número del próximo registro en el archivo de acceso directo histórico.
- EDIT Rutina que chequea validez de los datos de entrada antes de cada solución.
- ELIM Rutina que ejecuta proceso de eliminación - Gaussiana sobre cada fila del jacobiano.
- ERROR Rutina para imprimir en el reporte de salida los mensajes de error generados en el programa.
- EXBRAN Extrae información del próximo ramal a ser procesado y que se encuentra almacenado en el archivo de acceso directo de trabajo.
- EXNODE Rutina que extrae información del próximo nodo a ser procesado y almacenado en el archivo de acceso directo de trabajo.
- EXTIES Rutina que extrae la información desde el

archivo de acceso directo de trabajo, sobre las líneas de interconexión entre áreas.

- FLOW Rutina para calcular los flujos de potencia activa y reactiva en las líneas y transformadores del sistema después de la solución.
- I2DELT Rutina para borrar los elementos L(I) y L(J) de una tabla o arreglo desde L(1) hasta L(N) condensándola en forma apropiada. (las variables son de longitud 2).
- I4DELT Rutina para borrar los elementos L(I) y L(J) de longitud 4 de una tabla L(1).....L(N), condensando la tabla en forma apropiada.
- I2INST Rutina que causa que los contenidos de L(I) y L(J) sean insertados en la tabla L(1)....L(N), tal que L(I) y L(J) respectivamente preceden al L(I) y L(J) original (las variables son de longitud 2).
- I4INST Rutina que causa que los contenidos de L(I) y L(J) sean insertados en la tabla de L(1)...L(N) tal que L(I) y L(J), respectivamente preceden a L(I) y L(J) original (variables de longitud 4).
- I2SRCH Rutina que realiza la búsqueda en una tabla -

Lav. No. 207 - 067

L(L(1)L(N)) del primer valor mayor o igual que en el contenido de la variable LX - (para variables de longitud 2).

- I4SRCH Rutina de búsqueda del valor mayor o igual que LX de una tabla L(1).....L(N), donde la variable debe ser de longitud 4.
- INCONV Esta es una función del programa que es invocada por la subrutina RDAREA y su objetivo es generar los números correspondientes a las áreas a ser suprimidas.
- INHIST Rutina que graba los datos de líneas y transformadores en contingencia en la tabla de historias de cambios (solo llamada por rutinas que ejecutan contingencia de líneas y/o transformadores).
- INIT Rutina que es llamada por el programa principal MAIN apenas inicia su ejecución, para inicializar un archivo de acceso directo de trabajo, que contendrá información (del caso de flujo a ser resuelto) que no se puede mantener en la memoria principal.
- INSURE Rutina que ejecuta una iteración de Gauss-Seidel para obtener los valores de partida -

- del voltaje para el método de Newton - Raphson y de este modo asegurar la posibilidad de convergencia.
- ITCHCO Rutina que construye las ecuaciones para el control de intercambio entre áreas.
- LCHNGE Rutina para ingresar los datos de líneas y transformadores del caso base y/o para cambios - en ellos (contingencia).
- LISTIT Rutina que lista los títulos de los casos - contenidos en un archivo histórico de acceso - secuencial o directo.
- LODCHG Rutina que permite ingresar los factores para modificar el nivel de carga de cada área especificada así como también los VAR estáticos.
- LOPUT Rutina que imprime los datos de líneas y transformadores en el reporte de salida.
- MATGEN Esta es la rutina de solución principal del problema del flujo de potencia, aquí la matriz jacobiano del sistema es formada y - triangularizada por eliminación gaussiana ordenada, las correcciones de la magnitud y ángulo - del voltaje son calculadas por sustitución hacia atrás, para lo cual esta subrutina llama a

las subrutinas ELIM y BAKSUB, respectivamente, el triángulo superior del jacobiano reducido es almacenado en un arreglo en la memoria como se mostró para un ejemplo en la tabla VI.

- MOPUT Rutina que imprime información miscelanea en el reporte de salida, como por ejemplo el resumen de intercambio de potencia entre áreas, número de iteraciones para la solución, máximo número de iteraciones, tolerancias para la convergencia, etc.
- MSIN Rutina que lee información desde el archivo - de acceso directo de trabajo.
- MSOUT Rutina que escribe información hacia el archivo de acceso directo de trabajo.
- NUDECK Crea un archivo en formato de entrada de la solución de un caso de flujo.
- OPTIMO Esta es otra de las rutinas más importantes ya que ella establece la secuencia de ordenado óptimo para minimizar el llenado del jacobiano durante el proceso de eliminación (optimiza dispersidad del jacobiano triangulado).
- OUTAGE Rutina para sacar de servicio líneas o transformadores.

OUTPUT	Rutina que imprime reporte de salida principal.
PGHEAD	Rutina que es llamada para la impresión de títulos, y encabezados como fecha, tiempo, número - de página, etc.
PREOUT	Rutina que prepara e imprime salida preliminar antes del reporte de salida principal.
PSYCHO	Rutina que recupera para impresión la historia de cambios por contingencias en el sistema.
RDAREA	Rutina para leer datos que indican las áreas que se desean suprimir, retornando los números de las áreas a ser suprimidas.
RECALL	Rutina para cargar a memoria un caso de flujo desde un archivo secuencial histórico.
RECLOS	Rutina para reconectar una línea o transformador que previamente fue sacado de servicio mediante la subrutina OUTAGE.
REDEF	Rutina para redefinir la tabla de historia de cambios de línea como de caso base.
REMOVE	Rutina llamada para eliminar datos de líneas y/o transformadores de las tablas del caso base.

- ROLOUT Rutina que almacena los datos de barras y la matriz admitancia en tablas compactas que son transferidas al archivo de acceso directo de trabajo, a estos datos almacenados en esta forma son denominados el "roll de datos".
- SAVE Rutina que es llamada para grabar un caso de flujo en un archivo secuencial histórico.
- SIBRAN Extrae información del próximo ramal a ser procesado para un flujo simplificado (flujo D.C) y que se encuentra almacenado en el Roll de datos simplificados en el archivo de acceso directo de trabajo.
- SIINIT Es una rutina análoga a ROLOUT o sea realiza la misma función pero generando un roll de datos para un flujo simplificado.
- SINODE Recupera información de datos de nodos desde el roll de datos simplificados.
- SIOPUT Rutina que imprime reporte de salida principal - del flujo simplificado.
- SISOLV Rutina usada para la opción de un flujo DC., ella forma y triangula la matriz jacobiano simplificada mediante eliminación gaussiana y mediante -

sustitución hacia atrás resuelve el problema - del flujo D.C.

- SISUMM Rutina que prepara e imprime salida preliminar antes del reporte de salida principal del flujo simplificado.
- SORT Rutina que sirve para secuenciar las líneas y transformadores del caso base.
- TIEOUT Rutina que imprime el flujo de potencia - en las líneas de interconexión entre áreas, y un resumen de transformadores y barras reguladas o controladas.
- TIME Rutina que retorna caracteres para hora, minutos y segundos.

4.3. MANUAL DEL USUARIO

Con el objeto de posteriores usos del programa utilizado en esta tesis para la solución del flujo de potencia de un ejemplo de aplicación, en esta sección se da el manual para el uso de dicho programa, la capacidad de este y otros tópicos relacionados con el manejo del mismo.

El método de solución de este programa, es el mismo

que se ha desarrollado en los capítulos precedentes, es decir el método de Newton - Raphson, en el cual la matriz jacobiano es formada, aumentada por la columna de desajustes de potencia activa y reactiva, triangulada y resuelta por sustitución hacia atrás, para determinar las correcciones a la magnitud y ángulo del voltaje, en cada iteración varios criterios son examinados para el chequeo de la convergencia, ellos son:

1. El desajuste de potencia absoluto activa y reactiva en cada barra es comparado con una tolerancia especificada.
2. El desajuste absoluto total del sistema (esto es el sumatorio de los desajustes de P y Q, respectivamente en cada una de las barras del sistema) es comparado con una tolerancia dada.
3. Cuando todas las correcciones de la magnitud y ángulo del voltaje en cada barra llegan a ser tan pequeñas que una nueva iteración no sería significativa en el resultado.

El orden óptimo una vez establecido, es grabado como parte de los datos y es modificado para cambios en con

figuración de la red (adición y remoción de líneas y barras), esta modificación es realizada si el número de cambios en la configuración de la red excede una constante del programa.

Varios modos de ordenado de las barras están disponibles en el programa, ellos son listados en la tabla VII que se muestra a continuación, y que corresponden a los modos de ordenado explicados en el Capítulo III.

TABLA VII

<u>MODO</u>	<u>DESCRIPCION</u>
0	Esquema N° 3 de Tinney - Walker
1	Esquema N° 2 de Tinney - Walker
2	Esquema N° 1 de Tinney - Walker

Si ningún modo es especificado en los datos de entrada, el programa asume el modo 0 de ordenado.

Otra opción que da este programa, es que cuando las condiciones de MVAR del sistema de potencia a resolver no son consideradas, una solución de flujo de potencia simplificada puede ser usada, bajo esta consideración todos los voltajes son asumidos que tienen una magnitud de 1.0.P.U., y difieren en ángulo únicamente.

Es decir que unicamente se requiere determinar los ángulos de los voltajes de barra, con lo cual los sistemas de ecuaciones a resolver se simplifican.

Otra de las facilidades que da este programa es que la solución de una corrida de flujo de carga puede ser grabada como se encuentra en memoria y posteriormente recuperada en memoria para futuras corridas de flujo - partiendo de la solución anterior, esto es útil en el caso de contingencias, con lo cual le evita al programa nuevamente leer datos originales y los subsiguientes pasos para esa solución, lo cual le involucra más tiempo principalmente en el acceso de datos, la solución de la corrida del flujo puede ser grabada en dos tipos de archivo que son:

- De acceso secuencial; y,
- De acceso directo.

A este archivo se denominará Archivo de Acceso Secuencial Histórico en el un caso o Archivo de Acceso Directo Histórico en el otro caso, identificando de esa forma el archivo de que se trate y es histórico porque contiene una solución pasada.

El uso de esta facilidad del programa y otras es po

sible mediante los códigos de control específicos que permiten hacerlo y que se dan a continuación. En la tabla VIII se da un resumen de todos los códigos de control del programa y su función.

TABLA VIII

RESUMEN DE CODIGOS DE CONTROL Y SU FUNCION

CODIGO	F U N C I O N
1	Lee, imprime y almacena la siguiente línea como título del caso.
2	Lee e imprime la siguiente línea como comentario.
3	Lee caso desde el archivo de acceso secuencial o directo histórico.
4	Lee las siguientes líneas como datos de líneas del caso base.
5	Lee las siguientes líneas como datos de barras del caso base.
6	Lee la siguiente línea como cambios de las constantes del programa.
7	Lee las siguientes líneas como cambios a los datos de líneas.
8	Lee las siguientes líneas como cambios a los datos de barras.
9	Imprime un archivo en formato de entrada del caso en memoria.

- 10 Imprime tabla con la historia de los cambios de las líneas.
 - 11 Solución del problema de flujo de potencia (caso) e impresión de resultados.
 - 12 Graba el caso en memoria en un archivo de acceso secuencial o directo histórico.
 - 13 Imprime título de los casos grabados en el archivo histórico.
 - 14 Limpia la tabla de historia de cambios de líneas.
 - 15 Lee las siguientes líneas como datos de áreas.
 - 22 Lee las siguientes líneas como ajustes de los niveles de carga de las áreas.
 - 30 Fin de corrida.
 - ? Este signo indica que la corrida será suprimida apenas encuentra un error en los datos, es decir es un código de control de error, y si se usa debe ser la primera línea en los datos.
-

La información que sigue es aplicable a la versión del programa de flujo de potencia (PFLOW) que fue am bientado en el sistema 4341 de la ESPOL, bajo CMS, el programa se encuentra en el lenguaje Fortran 66 y com pilado bajo este nivel de lenguaje.

CAPACIDAD DEL PROGRAMA

1500 barras, todas con capacitores/reactores fijos.
2500 líneas, todas con dos capacidades y carga de línea
500 transformadores, todos con cambio de tap bajo carga.
25 transformadores desfasadores
500 barras con regulación de MVAR
40 áreas con control automático de intercambio.

OPERACION Y CONTROL DEL PROGRAMA

En adelante se dan los procedimientos para preparar los datos de líneas/transformadores y barra necesarios para el programa de flujo de potencia (PFLOW). Para obtener resultados correctos del programa de flujo es necesario preparar un archivo completo y exacto de los datos del caso base, también se describirán los códigos de control y las constantes que son usadas por el programa de flujo.

REQUERIMIENTOS DEL ARCHIVO DE DATOS

El programa a través del terminal recibe el nombre del archivo que contiene los datos del caso base que va a ser corrido.

Este archivo tiene que contener:

1. Líneas con los códigos de control
2. Datos de líneas y transformadores
3. Datos de barras
4. Datos de áreas.

Los datos de líneas/transformadores y barras son requeridos y ellos siempre deben ser suministrados. Los datos de área pueden ser omitidos; si lo son, los nombres de las áreas saldrán como espacios en blanco.

Además, el usuario debe incluir líneas con código de control con uno o dos dígitos. Los factores constantes, tales como el número máximo de iteraciones y la tolerancia para la convergencia, pueden ser incluidos como opción del usuario, sino se los incluye, el programa usa sus propios valores.

El archivo de datos debe ser creado y grabado pre-

viamente como un archivo de CMS, donde la identificación de un archivo de CMS consta de: Nombre y tipo de archivo, el tipo del archivo debe ser necesariamente PFLOW, y el nombre es fijado por el usuario - cumpliendo con los requerimientos de CMS del sistema 4341.

Para subsiguientes casos que requieren el archivo de datos del caso que ha sido usado previamente en el cual se pueden añadir cambios que reflejan la nueva característica de operación del sistema, los siguientes dos métodos pueden ser usados para obtener los nuevos datos del nuevo caso:

Los datos del caso precedente pueden ser retenidos en memoria y actualizados leyendo los cambios de datos necesarios, unicamente con los códigos de control apropiados. Este método es normalmente usado cuando varios consecutivos casos subsiguientes requieren los mismos datos del caso inicial, es decir durante una corrida en el computador se ejecutan en cadena el caso base o inicial y los subsiguientes casos.

Los datos del caso inicial pueden ser leídos a memoria desde un archivo de acceso secuencial o directo histórico y éste combinado con los cambios de datos re

queridos.

El método a ser usado así como también otras instrucciones de procesamiento son especificadas por el usuario mediante los códigos de control y operación.

HOJAS DE FORMATO DE DATOS

Se recomienda la preparación de hojas que contengan los formatos para registrar los datos de líneas / transformadores barras y datos de área así de esta manera se garantizará que la edición del archivo de datos sea digitado rápidamente y sin errores.

Estas hojas son mostradas en las figuras N° 4.1, 4.2, y 4.3, respectivamente.

PUNTOS DECIMALES

Algunos campos contienen datos que requieren normalmente un punto decimal, lo que se indican en la hoja de formato de datos, el programa asume un punto decimal en esta posición.

Otros campos no requieren un punto decimal, por lo que la hoja de formato de datos no lo muestra. El programa

asume en estos casos que el punto decimal está al final del campo.

El usuario sin embargo puede poner el punto decimal, - cuando la posición asumida por el programa no se ajusta a sus necesidades.

Es necesario señalar que todos los espacios en blanco en los campos numéricos son asumidos como ceros por el programa a menos que explícitamente sea indicado lo contrario. Por este motivo se debe tener mucho cuidado para ajustar en forma justificada a la derecha todas las entradas en los campos numéricos - que no incluyan punto decimal.

Errores de este tipo producirán números por lo menos diez veces mayores que el deseado.

CODIGOS DE CONTROL PARA LA OPERACION DEL PROGRAMA

Todo el control interno del programa se lo consigue con códigos de control con uno o dos dígitos que van en una línea de control del archivo de datos, en las columnas 2 y 3, si se trata de un código de control de dos dígitos o en la columna 3 si se trata de un dígito.

En la tabla VIII se incluye la lista de estos códigos de control y sus funciones.

Adicionalmente al campo del código de control las líneas de control pueden contener otros campos que son dados en la descripción específica de cada línea de control.

La operación de estos códigos de control, es descrita a continuación.

LECTURA DE LINEA DE TITULO CODIGO DE CONTROL 1

El código de control 1 hace que el programa lea y almacene lo contenido entre las columnas 1 a 80 como el título del caso.

La información del título es retenida como parte de los datos.

Una línea de título no es requerida por el programa sin embargo si es usada, su contenido es incluido en la primera línea de cada página de salida. Si los mismos datos son usados en subsiguientes corridas, el mismo título será impreso a menos que sea cambiado por un subsiguiente código de control 1 en el momento -

de reprocesar los datos grabados.

1. Entre un 1 en la columna 3 de la línea de control
2. Entre la información del título en las columnas 1 a 80 de la siguiente línea. La línea de código de control 1 debe ser seguida por una línea de título.

LECTURA DE LINEA DE COMENTARIOS CODIGO DE CONTROL 2

El código de control 2 hace que el programa lea e imprima el contenido de las columnas 1 a 80 de la siguiente línea del archivo de datos. Los comentarios son almacenados por el programa hasta un máximo de 20. El programa -numera las líneas de comentarios en forma ascendente del 1 al 20, con este número se puede modificar el comentario respectivo con código de control 2. Las líneas de comentarios a partir de 21 son impresas en la salida, pero no almacenadas.

1. Entre un 2 en la columna 3 de la línea de control
2. Entre las informaciones de comentario en las columnas 1 a 80 de la siguiente línea del archivo de datos.
3. Si uno de los 20 comentarios almacenados, debe ser reemplazado, entre el número del comentario en las columnas 9 y 10, ajustado a la derecha, en la línea que contiene el código de control 2.

LECTURA DE ARCHIVO HISTORICO. CODIGO DE CONTROL 3

El código de control 3 hace que el programa lea (cargue a memoria) un caso desde el archivo de acceso secuencial o directo histórico. Esta línea de control debe también especificar el número de registro del caso y la identificación del archivo histórico.

Como dijimos el caso puede estar grabado en un archivo de acceso secuencial o un archivo de acceso directo. Hay una ventaja en velocidad para leer y escribir en archivos de acceso directo, pero ocupan más memoria estos archivos que los secuenciales, así que se recomienda que los casos que no van a ser usados con mucha frecuencia y que requieren ser almacenados, sean transferidos a archivos secuenciales, mediante una secuencia apropiada de los códigos de control 3 y 12.

PARA ARCHIVO SECUENCIAL

1. Entre un 3 en la columna 3 de la línea de control.
2. Entre el número de registro del caso en las columnas 9 y 10, ajustado a la derecha. El número de registro del caso no puede ser mayor que 99.
3. Entre el número de la unidad lógica con que está direccionado el archivo de acceso secuencial histórico y

en el cual se encuentra el caso a ser leído en las columnas 14 y 15 de la línea de control, ajustado a la derecha. Si estas columnas son dejadas en blanco, el programa asume que el número de la unidad lógica es 3.

4. Entre la identificación del archivo histórico en las columnas 16 a 23 (la identificación constará como máximo de 8 caracteres alfanuméricos). Esta identificación aparecerá en el reporte de salida y ayuda a la identificación de los resultados.
5. Para suprimir el rebobinado del archivo secuencial, ingrese el dígito 1 en la columna 31 de la línea de control. El uso de esta opción se recomienda cuando el próximo caso a ser leído (cargado) tiene un número de registro mayor que el caso previamente cargado. Si después que el rebobinado ha sido suprimido, el número del registro del próximo caso a ser cargado es menor o igual al precedente ya procesado el archivo será rebobinado y la búsqueda y carga del siguiente caso procederá en forma normal.

PARA ARCHIVO DIRECTO

1. Entre un 3 en la columna 3 de la línea de control.
2. Entre el número del registro del caso a ser cargado, en las columnas 9 y 10, ajustado a la derecha. El número de registros de casos no puede ser de más de 2 dígitos.

3. Entre el número de la unidad lógica con que está direccionado el archivo de acceso directo y en el cual se encuentra el caso a ser cargado, en las columnas 14 y 15 de la línea de control, ajustado a la derecha. Si estas columnas son dejadas en blanco el programa asume un número de unidad lógica igual a 16.
4. Entre la identificación del archivo histórico en las columnas 16 a 23 (la identificación tendrá 8 caracteres alfanuméricos como máximo).
5. Entre el dígito 1 en la columna 30 de la línea de control, esto indicará al programa que se está procesando un archivo de acceso directo.

NUMERO DE UNIDAD LOGICA

Si ningún número de unidad lógica (llamado también en CMS número de dispositivo) es especificado, el programa asume 3 para archivo secuencial y 16 para archivo directo, sin embargo el usuario puede definir otros números de unidad lógica, pero siempre y cuando estén entre alguno de los números válidos en este programa para los archivos secuenciales o directos. Los números de unidad lógica permitidos se encuentran listados en la tabla IX que se muestra a continuación.

TABLA IX

NUMERO DE UNIDADES LOGICA		PERMITIDOS
NUMERO DE UNIDAD LOGICA	ARCHIVO	DE ACCESO
	3	Secuencial
	4	Secuencial
9	a 15	Secuencial
16	a 20	Directo

LECTURA DE DATOS DE LINEAS Y TRANSFORMADORES

DEL CASO BASE

CODIGO DE CONTROL 4

El código de control 4 indica al programa que a continuación vienen las líneas que contienen los datos de líneas y transformadores del caso base. El programa procede a leer la información de las siguientes líneas hasta que encuentra una línea que contiene los dígitos 9999 en las columnas 1 - 4, que definen el fin de los datos de líneas.

Con el código de control 4, el programa asume que un caso base está siendo procesado, por lo que incluye un listado de los datos de líneas y transformadores en el reporte de salida. Este listado puede ser eliminado -

entrando el dígito 1 en la columna 10 de la línea con el código de control 4.

1. Entre un 4 en la columna 3 de la línea de control.
2. Entre un 1 en la columna 10 de esta misma línea de control, si no desea un listado de los datos de entrada en el reporte de salida.
3. La primera línea con datos de línea y transformador debe ir a continuación de la línea de control 4. La última línea con datos de línea y transformador debe ser seguida por una línea que contenga 9999 en las columnas 1-4.
4. El grupo de datos de líneas y transformadores debe ser seguido por una línea con código de control que por lo general es el control 5 para datos de barras.

El programa aceptará las descripciones para un máximo de 2500 líneas, 500 de las cuales pueden ser transformadores con cambio de taps bajo carga. Cada línea puede tener carga de línea en MVA y dos capacidades, una normal y otra de emergencia. Hasta 25 transformadores desfasadores pueden ser representados; estos mantienen un ángulo fijo con relación a una barra terminal.

IDENTIFICACION DEL CIRCUITO (COLUMNAS 1 - 16)

El contenido de estos campos identifica una línea o transformador por el número de sus barras terminales. Una línea o transformador debe ser identificado una sola vez ya que el programa asume que entradas duplicadas corresponden a circuitos paralelos. El número de circuito de entradas paralelas es asignado por el programa, de acuerdo al orden de su entrada. El programa asume que la primera entrada para una línea entre dos barras es el circuito 1, la segunda entrada entre las mismas dos barras el circuito 2, y así en adelante. Aunque no es un requisito, se recomienda que los circuitos paralelos sean entrados en orden sucesivo.

Si un circuito de un grupo de circuitos paralelos es especificado como transformador, los otros circuitos paralelos también deben ser especificados como transformadores. Los transformadores en paralelo deben tener el tap en el mismo lado.

DE BARRA

COLUMNAS 1 - 4

Entre el número de una de las barras terminales del circuito. Si se está identificando un LTC, el

número de la barra donde está conectado el cambiador de tap físico, debe ser entrado en este espacio.

Para identificar el fin de datos de línea se debe entrar un 9999 en éste sub-campo. A continuación debe ir una línea con un código de control.

CODIGO DE CAMBIO

COLUMNA 6

Este sub-campo no es tomado en cuenta en la lectura de los datos del caso base. Es usado cuando se hacen cambios a los datos de líneas y/o transformadores del caso base es decir en contingencia, para una mayor descripción refiérase a la operación del código de control 7.

A BARRA

COLUMNAS 9 - 12

Entre el número de la otra barra terminal del circuito. Si se trata de un LTC, aquí se debe entrar el número de la barra que está opuesta al cambiador de tap físico.

NUMERO DE CIRCUITO

COLUMNA 14

La columna 14 es usada unicamente en el caso de -

procesar cambios a los datos del caso base. Es igno
rado este campo para los datos del caso base.

DUÑO DE LA LINEA

COLUMNA 16

Este campo define a que barra terminal pertenece la línea. Todas las pérdidas de la línea son asignadas al propietario de la línea. Si el dueño de la línea es aquel que está del lado de envío, es decir el del campo DE BARRA, entre la letra F en éste campo. Si el dueño de la línea es aquel que está del lado de recepción, campo A BARRA, entre la letra T en este campo. Si se deja el campo en blanco, el programa asume que la propiedad pertenece a DE BARRA.

RESISTENCIA EN PORCENTAJE

COLUMNAS 18 - 23

Este campo contiene la resistencia de cada línea o transformador expresada como porcentaje de cualquier base MVA conveniente. Entre en las columnas 18 - 23 la resistencia en porcentaje del circuito. Si la resistencia del circuito es insignificante deje este campo en blanco.

REACTANCIA EN PORCENTAJE

COLUMNAS 24 - 29

Este campo contiene la reactancia de cada línea o -

transformador expresada como porcentaje de la misma - base MVA usada para describir la resistencia en porcentaje.

Entre en las columnas 24 - 29 la reactancia en porcentaje del circuito. Circuitos sin resistencia ni reactancia no podrán ser procesados por el programa y resultarán en un mensaje de error fatal.

El método de solución de éste programa de flujo, no es por lo general, afectado por los rangos extremos de impedancia en una barra, como lo son otros métodos más convencionales. Magnitudes normales de impedancia no deben causar dificultades de convergencia.

CARGA DE LA LINEA

COLUMNAS 30 - 35

Este campo contiene la carga total de la línea en MVAR. El programa divide el total para dos y crea una sección "PI" asignando una mitad a cada barra-terminal.

Entre el total de carga de línea en MVAR del circuito entre las columnas 30 - 35. Este campo puede quedar en blanco para los circuitos que no tienen carga de línea.

TAP DEL TRANSFORMADOR

COLUMNAS 36 - 40

Los transformadores se distinguen de las líneas de transmisión por la entrada en el campo tap del transformador. Se requiere una entrada para cada transformador a ser estudiado. Si se trata de un transformador con tap fijo, entre aquí la posición física del tap.

La posición del tap es la relación actual del transformador en por unidad, no su desviación de 1.0.

CAMBIO AUTOMATICO DE TAP

COLUMNAS 41 - 50

Cualquier transformador puede ser definido con cambio automático de tap, entrando relaciones mínima y máxima en el campo apropiado (columnas 41 - 45 y 46 - 50). Los datos entrados son las relaciones mínima y máxima que pueden ser permitidas para mantener el voltaje en una barra. El voltaje a ser mantenido es especificado en la entrada de datos de barra. El programa determina la relación, dentro de los límites definidos (y que pueden ser modificados por el usuario), requerida para mantener el voltaje en una barra determinada. Si no se especifica ninguna barra en el campo de barra controlada (columnas 56 - 60), el

voltaje del lado del tap del transformador es man
nido.

TAP MINIMO COLUMNAS 41 - 45

Entre la relación mínima permisible para mantener voli
taje en la barra especificada.

TAP MAXIMO COLUMNAS 46 - 50

Entre la relación máxima permisible para mantener voli
taje en la barra especificada.

ANGULO DE DESFASAJE COLUMNAS 51 - 55

Si se va a representar un transformador desfasador ,
entre el ángulo de desfasaje en éste campo. El án
gulo es especificado en grados. Este ángulo es man
tenido con respecto a la barra terminal especificada
en la entrada DE BARRA.

BARRA CONTROLADA COLUMNAS 56 - 60

Si se va a representar un LTC, entre en éste campo
el número de barra donde se va a controlar el voli
taje. Si se deja este campo en blanco, el voltaje de

la barra terminal especificada en la entrada DE BARRA (lado del tap), será mantenido.

Si la barra controlada no es una de las terminales - del transformador, el signo de la entrada en este campo determina la dirección con la que el tap es movido para aumentar el voltaje en la barra controlada. Una barra controlada en la dirección del lado sin tap del LTC es indicada por una entrada negativa en este campo. Esto se indica con un signo menos (-) a la izquierda del número de la barra controlada.

CAPACIDADES EN MVA

COLUMNAS 61 - 68

El programa calcula la carga en MVA para que cada línea tenga entrada de límites en los campos de capacidades en MVA. Los números de las barras terminales, sus nombres, capacidades y flujos en MVA son impresos en dos tablas al comienzo del reporte. La primera tabla enumera todas las líneas de una capacidad que tengan flujos dentro de la capacidad.

La segunda tabla enumera todas las líneas de una capacidad que tengan flujos fuera de la capacidad. La primera tabla puede ser suprimida si se lo desea (véase la línea de control 11).

Entre las capacidades normal y de emergencia de cualquier línea donde se lo desee, para determinar la carga en MVA de cada línea. Las capacidades son entradas en MVA entre las columnas 61 - 64 y 65 - 68. El flujo es calculado en la barra del que no es dueño de la línea.

MVA BASE

COLUMNAS 69 - 72

El programa asume que la resistencia y reactancia del circuito están en base 100 MVA. Sin embargo, si el circuito está en una base diferente, el programa convertirá la resistencia y reactancia de esa base a 100 MVA, si es que se entra la base vieja en este campo. El programa automáticamente vuelve a esperar entradas con base 100 MVA, luego de procesar el circuito anterior. Por lo tanto, cada circuito que tenga una base diferente a 100 MVA, debe llevar una entrada en este campo.

Entre los MVA base de este circuito si es diferente de 100 MVA.

IDENTIFICACION DE LINEA

COLUMNAS 73 - 80

Este campo no es considerado por el programa, pero -

puede ser usado para identificar la línea por su descripción alfanumérica.

LECTURA DE DATOS DE BARRA DEL CASO BASE CODIGO DE CONTROL 5

El código de control 5 define el comienzo de las líneas de datos de barra en el archivo de datos de entrada. Cuando el programa encuentra la línea con el código de control 5, inicia la lectura de las líneas de datos de barra hasta que encuentra una línea con los dígitos 99 99 en las columnas 1 - 4, pues ésta determina el fin de los datos de barra.

Cuando el programa encuentra el código de control 5, anula todos los datos de barra y área almacenados pero los datos de línea no son afectados. Un listado de los datos de barra es impreso en el reporte de salida pero puede ser eliminado entrando un 1 en la columna 10 de la línea de código de control 5.

1. Entre un 5 en la columna 3 de la línea de control.
2. Si no se desea un listado de los datos de entrada, entre un 1 en la columna 10 de esta línea de control.
3. La línea de control 5 debe ser seguida por una línea de datos de barra. La última línea de datos de barra debe ser seguida por una línea con los

- dígitos 9999 en las columnas 1 - 4.
4. El código de control 5 puede ser usado para redefinir completamente los datos de barra, sin afectar a los datos de líneas y transformadores en efecto, o a la tabla que contiene la historia de cambios en las líneas.
 5. Los datos de barra deben seguir a los datos de líneas y transformadores si es que alguno de estos ha entrado.

El programa de flujo acepta hasta 1500 descripciones de barras. Hasta 500 barras pueden ser reguladas por generación reactiva. Cualquiera de las 1500 barras pueden tener capacitores y/o reactores.

IDENTIFICACION DE BARRA

COLUMNAS 1 - 21

El contenido de estos campos identifica una barra - por su nombre y su número. También se especifica - si la barra es regulada o no, o si es la barra oscilante del sistema. Pueden omitirse números al asignar números de identificación a las barras, pero el número más grande que puede asignarse a una barra es el 1500. Debe anotarse que la omisión de grandes - bloques de números puede perjudicar la ejecución óptima del programa.

NUMERO DE BARRA

COLUMNAS 1 - 4

Entre el número de barra exactamente como fue usado en los datos de línea y transformador. Todas las entradas en éste campo deben estar ajustadas a la derecha ya que el programa asume los espacios en blanco como ceros. La última entrada en este campo debe ser 9999.

La línea con 9999 señala el fin de los datos de barra y debe ser seguida inmediatamente por una línea con código de control.

CÓDIGO DE CAMBIO

COLUMNA 6

Para la corrida del caso base este código de cambio no es usado y por lo tanto este campo es ignorado. En contingencias el código de cambio denota el tipo de cambio a ser hecho sobre los datos del caso base. Más detalle de este código se dá en el código de control 8.

REGULACION

COLUMNA 8

Este campo designa a la barra como barra oscilante, barra con voltaje controlado por variación de genera-

ción reactiva (local o remota), o barra con carga y generación fijas.

Entre un cero o un espacio en blanco, un 1 o un 2 de acuerdo a lo que sigue:

0 o espacio en blanco = la barra no es regulada y tiene carga y generación fijas.

1 = la barra es regulada y se está controlando a ella misma o a otra barra remota. El campo BARRA - CONTROLADA (Columnas 51 - 55) señala a la barra controlada. El voltaje en las columnas 23 - 26 de la línea del voltaje controlado, es mantenido mientras el rango de variación de la generación reactiva sea suficiente (el rango está dado en los sub-campos de generación, columnas 41 - 50). Si la generación reactiva alcanza los mínimos o máximos especificados, el voltaje de la barra variará.

2 = la barra es la barra oscilante. Si no se especifica una barra oscilante, el programa terminará con un error fatal. Si más de una barra oscilante es designada, el programa asumirá a la última barra encontrada con un 2 en la columna 8 como la barra oscilante.

NOMBRE DE LA BARRA COLUMNAS 10 - 21

Entre la descripción alfanumérica de la barra.

VOLTAJE DE LA BARRA COLUMNAS 23 - 26

Este campo contiene el estimado en por unidad del voltaje de la barra. El voltaje en la barra oscilante permanecerá siempre fijo con el valor designado en este campo.

Entre en las columnas 23 - 26 el estimado en por undad del voltaje en la barra. Para la barra oscilante del sistema, barras local o remotamente controladas por generación de M var, y barras controladas por transformadores LTC, la entrada debe ser el voltaje deseado.

En general, por el método de solución usado, el valor inicial de las barras no controladas no tiene efecto en la convergencia de la solución

ANGULO DE LA BARRA COLUMNAS 27 - 30

Este campo contiene un estimado para el ángulo en grados del voltaje estimado para la barra con relación a la barra oscilante. El ángulo designado en la barra os

cilante permanecerá siempre constante.

Entre en las columnas 27 - 30 el estimado del ángulo en grados del voltaje relativos a la barra oscilante. Si no se dispone de un estimado, este campo puede ser dejado en blanco.

En general, por el método de solución usado, el estimado inicial del ángulo de cualquier barra, a excepción de la barra oscilante, no tiene ningún efecto en los resultados finales.

GENERACION

COLUMNAS 31 - 55

Estos campos contienen la generación deseada en megawatts para cada barra, los megavars de generación en las barras con generación fija de Mvar, y los límites de Mvar y número de barra controlada para las barras reguladas.

MEGAWATTS

COLUMNAS 31 - 35

Entre la generación deseada de Megawatts, ajustada a la derecha. Si se deja en blanco este campo, el programa asume cero este valor.

MEGAVARS

COLUMNAS 36 - 40

Si una generación reactiva constante es deseada, entre la generación en Mvar en este campo, ajustada a la derecha. Si se desea generación reactiva variable, este campo debe quedar en blanco.

LIMITE MINIMO DE MVAR

COLUMNAS 41 - 45

Entre el límite mínimo en Mvar (límite mínimo para la generación). Si esta cantidad es negativa, incluya un signo menos (-) a la izquierda de la cantidad entrada.

LIMITE MAXIMO DE MVAR

COLUMNAS 46 - 50

Entre la generación máxima en Mvar (límite superior de la generación). La entrada es el límite de la generación reactiva en la barra.

BARRA CONTROLADA

COLUMNAS 51 - 55

Si esta es una barra con límites de generación reactiva y código de regulación igual a uno(1), entre en este campo el número de la barra donde se desea controlar el voltaje. El voltaje a ser mantenido se lo

toma del campo VOLTAJE DE BARRA de la barra designada. Si el campo BARRA CONTROLADA es dejado en blanco, el voltaje de la barra cuya línea de información está - siendo leída, será el mantenido.

CARGA COLUMNAS 56 - 65

Estos campos contienen la carga deseada en Megawatts y Megavars para todas las barras de carga.

MW COLUMNAS 56 - 60

Entre la carga deseada en Mw para todas las barras de carga. Si no existe carga, este campo debe que dar en blanco.

MVAR COLUMNAS 61 - 65

Entre la carga deseada en Mvar para todas las ba rras de carga. Si no existe carga, este campo debe quedar en blanco.

CAPACITORES/REACTORES, MVAR COLUMNAS 66 - 70

Este campo contiene los Mvar totales de capacitores estáticos o reactores. Estos totales son converti-

dos por el programa a valores en por unidad de susceptancia en base 100 Mva.

Si la barra tiene capacitores estáticos, entre su capacidad en Mvar como un número positivo (no se requiere el signo más). Si la barra tiene reactores, entre su capacidad en Mvar como un número negativo, con un signo menos precediendo el número. Si la barra no tiene capacitores ni reactores deje el campo en blanco.

AREA

COLUMNAS 71 - 72

Este campo contiene un número que identifica al sistema, compañía o división. El programa agrupa a todas las barras con igual número de área en el listado de salida e imprime la carga y generación total para cada área. El número más grande para un área es 40. Si se deja en blanco este campo, la barra es adjudicada al área número uno. Una designación exacta de este campo es indispensable para el control del intercambio de áreas.

LECTURA DE CAMBIO DE CONSTANTES

CODIGO DE CONTROL 6

El código de control 6 hace que el programa lea en la línea siguiente, constantes especiales que el programa usa

rã en lugar de las que tiene predefinidas.

1. Entre un 6 en la columna 3 de la línea de control. Todas las otras columnas deben permanecer en blanco.
2. Entre en la siguiente línea los valores que deben reemplazar a las constantes predefinidas. Aquellos campos que queden en blanco significarán que las constantes predefinidas no sufrirán cambio.
3. Los cambios de constantes deben ser hechos antes de que se lea la línea con el código de control 11. En la tabla X se muestra el formato de la línea del cambio de constantes.

*
Nota: En la tabla X vemos que el valor de estas constantes es determinado cuando se procesa el caso base de acuerdo a:

$$\text{TOLERANCIA} = 0.01 + \frac{\text{N}^{\circ} \text{ de líneas} + \text{N}^{\circ} \text{ de barras}}{4000.0}$$

Pueden ser cambiadas luego de que son leídos los datos de barra.

TABLA X

DESCRIPCION DE CONSTANTES

COLUMNAS CONSTANTE DEL PROGRAMA

D E S C R I P C I O N

3 - 9	.001000	Criterio de convergencia para el desajuste(mismatch) en barra de potencia en p.u.
11 -17	.00100	Criterio de convergencia para el desajuste en potencia Q en p.u.
19 - 2	0.0050	Tolerancia de voltaje en por unidad para transformador LTC.
38 -42	0.950	Criterio de voltaje en p.u. para la tabla de voltajes excepcionalmente bajos.
43 -47	1.050	Criterio de voltaje en p.u. para la tabla de voltajes excepcionalmente altos
49 -55	.010000	*Criterio de convergencia para el desajuste absoluto de potencia P en p.u.
57 -63	.010000	*Criterio de convergencia para el desajuste absoluto de potencia Q en p.u.

LECTURA DE CAMBIOS EN DATOS DE
LINEAS Y TRANSFORMADORES

CODIGO DE CONTROL 7

El código de control 7 indica al programa que a continuación vienen las líneas que contienen los datos de cambios de líneas y transformadores de un caso precedente y debe ser seguido inmediatamente por la primera línea que contiene los cambios de líneas y transformadores.

Cuando el programa encuentra la línea que contiene el código de control 7, el programa lee y procesa los cambios de datos de líneas y transformadores hasta que una línea que contiene los dígitos 9999 en las columnas 1-4 es hallada, esta línea indica el fin de los datos de cambio.

Un listado de los cambios hechos es impreso en el reporte de salida. Este listado puede ser eliminado ingresando un 1 en la columna 10 de la línea que contiene al código de control 7.

1. Entre un 7 en la columna 3 de la línea de control.
2. Entre un 1 en la columna 10 de la línea de control, si no se desea un listado de los cambios realizados.
3. La línea de control debe ser seguida por la primera

línea que contiene cambios de datos de línea o transformadores.

4. La última línea de cambios de datos de líneas y transformadores debe ser seguida por una línea con 9999 en las columnas 1-4.

Los cambios a los datos de líneas y transformadores son identificados mediante un código de cambio en la columna 6 en una línea con el mismo formato como las de datos de líneas y transformadores y que contiene los cambios. Los códigos y la acción tomada por el programa para cada uno de ellos son dados más abajo.

El número total de cambios de cada tipo que son grabados en la tabla de historia de cambios de líneas para ser impresos cuando sea deseada es dada también.

No hay límite para el número de cambios que pueden ser hechos; sino únicamente al número de cambios que pueden ser grabados.

CODIGO DE CAMBIO	ACCION DEL PROGRAMA	MAXIMO NUMERO GRABADO
0	El circuito designado es añadido a los datos de líneas y transformadores.	90

1	El circuito designado es eliminado desde los datos.	95
2	El circuito designado es sacado de servicio, pero es retenido para posterior reconexión del circuito.	50
3	El circuito designado, previamente sacado de servicio, es restaurado a los datos (reconexión).	
4	Todos los nuevos valores especificados para el circuito designado son ingresados en los datos.	100

ADICION (CODIGO 0)

El código de cambio 0 es usado para añadir líneas o transformadores a un caso precedente.

1. Entre un 0 en la columna 6 (campo del código de cambio)
2. Complete los campos restantes de acuerdo al procedimiento descrito bajo el código de control 4.
3. Si nuevas barras son establecidas, estas deben ser añadidas a los datos de barra del caso.

Si un transformador LTC es añadido, el voltaje a ser mantenido en la barra controlada por el, es seleccionado de el

voltaje ya especificado para dicha barra si algún otro dispositivo estaba regulando dicha barra, de otra forma el voltaje es ajustado a 1.0.P.U., si esto no es deseado, entonces debe hacerse una conversión de dicha barra para especificar el voltaje que se desea mantener.

Si un transformador LTC es añadido en paralelo a otro LTC ya existente que tiene un determinado rango de tap, el tap del LTC añadido será ajustado igual al tap del LTC existente.

REMOCIÓN (CODIGO 1)

El código de cambio 1 es usado para eliminar en forma permanente líneas y transformadores de un caso precedente.

Si los circuitos eliminados con el código de cambio 1 necesitan ser nuevamente restaurados al sistema, entonces ellos deben ser añadidos con el código de cambio 0. Cuando un circuito paralelo es eliminado, el programa renumera los circuitos restantes. Por ejemplo, los circuitos que eran el N° 4 y el N° 5 llegan a ser después N° 3 y N° 4, respectivamente. Esto debe ser considerado por el usuario para futuras eliminaciones que se hagan en los circuitos paralelos que quedan, aun

cuando las siguientes eliminaciones sean subsiguientes.

1. Entre un 1 en la columna 6
2. Complete los campos de IDENTIFICACION DEL CIRCUITO para cada circuito a ser eliminado, de acuerdo al procedimiento descrito bajo el código de control 4, ningún ingreso adicional es requerido.

SALIDA DE SERVICIO (CODIGO 2)

Mediante el código de cambio 2, los circuitos pueden ser eliminados de un caso, pero ellos son almacenados en una tabla que les permite ser reconectados posteriormente mediante el código de cambio 3.

1. Entre un 2 en la columna 6
2. Complete los campos de IDENTIFICACION DEL CIRCUITO para cada circuito a ser eliminado de acuerdo al procedimiento descrito bajo el código de control 4, ningún ingreso adicional es requerido.

RECONEXION (CODIGO 3)

El código de cambio 3 es usado para restaurar o reconectar circuitos que han sido eliminados mediante el código de

cambio 2.

1. Entre un 3 en la columna 6
2. Complete los campos de IDENTIFICACION DEL CIRCUITO para circuitos a ser reconectados, de acuerdo al procedimiento descrito bajo el código de control 4.

El número del circuito reingresado debe ser el mismo como fue especificado cuando el circuito fue sacado de servicio, ningún ingreso adicional es requerido.

3. Cuando el circuito es reconectado, este será reconectado con un número de circuito inmediatamente mayor al número de circuito más alto ya existente.

CONVERSION (CODIGO 4)

El código de cambio 4 es usado para cambiar cualquier dato para una línea o transformador de un caso precedente.

Solo los campos no blancos son cambiados a excepción del caso en que se cambie una de las constantes de línea (R, X o BC), en este caso se requiere que todos los tres valores sean ingresados; de otra forma los campos que quedan en blanco serán interpretados como cero y almacenados como su nuevo valor.

Los cambios a transformadores puede ser hecho ingresando solo el valor que tiene que ser cambiado. Un circuito puede ser cambiado de un transformador a una línea, ingresando un cero (0) en el campo del tap.

Cambios en el voltaje a ser mantenido por un LTC debe ser hecho mediante un apropiado cambio de barra.

Un transformador desfasador puede ser convertido a una línea de transmisión ingresando un cero (0) en el campo ANGULO DE DESFASAJE.

Si líneas paralelas son convertidas a transformadores LTC o desfasadores, estas deben ser convertidas en orden creciente del número de circuito.

Un circuito paralelo debe ser eliminado o sacado de servicio previa a los conversión del resto de circuitos o transformadores LTC.

LECTURA DE CAMBIOS EN DATOS DE BARRA CODIGO DE CONTROL 8

El código de control 8 define el comienzo de las líneas que contienen cambios a los datos de barra del caso precedente.

Cuando el programa encuentra la línea con el código de control 8, este lee y procesa las líneas con los cambios a los datos de barra hasta que una línea con 9999 digitado entre las columnas 1 y 4 es hallada, esta línea indica el fin de líneas con cambios a los datos de barra.

Un listado con los cambios hechos es impreso en el reporte de salida, este listado puede ser eliminado ingresando un 1 en la columna 10 de la línea de control.

1. Entre un 8 en la columna 3 de la línea de control.
2. Si no se desea un listado de los cambios realizados, ingrese el dígito 1 en la columna 10 de la línea de control.
3. La línea de control debe ser seguida por la primera línea que contiene cambios a los datos de barras.
4. La última línea de cambio a los datos de barra deben ser seguida por una línea que contenga a los dígitos 9999 en las columnas 1 a 4.
5. Si hay cambios de datos de barras y de líneas, los cambios de datos de barras deben seguir a los cambios de datos de líneas.

Los cambios de datos de barra son identificados mediante un código de cambio en la columna 6 en una línea del

mismo formato de las líneas de datos de barra y que contiene los cambios.

Los códigos para cambios y la acción tomada por el programa es resumida a continuación:

CODIGO DE CAMBIO	ACCION DEL PROGRAMA
0	La barra designada es añadida a los datos de barra.
1	La barra designada es eliminada de los datos de barra.
2	Cambia los datos de generación para la barra designada por los nuevos valores ingresados.
3	Cambia los datos de carga para la barra designada por los nuevos valores ingresados.
4	Cambia todos los valores de la barra designada por los nuevos ingresos (no blancos).

ADICION(CODIGO 0)

El código de cambio cero es usado para añadir nuevas barras sobre los de un caso precedente. Intentar añadir una barra ya existente es un error fatal.

1. Entre un 0 en la columna 6
2. Detalle cada nueva barra haciendo los ingresos apropiados en todos los campos de datos de barra, de acuerdo al procedimiento descrito bajo el código de control 5.

REMOCION (CODIGO 1)

El código de cambio 1 es usado para eliminar barras de un caso precedente. Intentar eliminar una barra que no se encuentra en los datos es un error fatal. Una vez que una barra es eliminada mediante el código 1 la única forma en que puede ser restaurada es mediante el código de cambio 0 ya que ningún dato de la barra eliminado ha sido grabado.

1. Entre 1 en la columna 6
2. Identifique la barra a ser eliminada, completando el campo NUMERO DE BARRA, ningún ingreso adicional es requerido. Todos los datos pertinentes a la barra eliminada son eliminados, esto no significa que todas las líneas y transformadores conectados a la barra - son eliminados, estos deben ser eliminados mediante cambios a los datos de líneas y transformadores.

CAMBIO DE GENERACION (CODIGO 2)

El código de cambio 2 es usado para cambiar valores de la generación de barras ya existentes. Los campos para los Mw y MVAR de generación son sustituidos por los nuevos valores ingresados.

Si la barra es regulada, ningún cambio es hecho en los límites ni puede una barra ser convertida a barra regulada con este código.

1. Entre un 2 en la columna 6
2. Complete el campo NUMERO DE BARRA para la barra a ser cambiada de acuerdo al procedimiento descrito bajo el código de control 5.
3. Entre los datos de cambio apropiados en los subcampos de MW y MVAR de GENERACION. Si uno de estos dos subcampos es dejado en blanco este es asumido como cero. Ningún ingreso adicional es requerido.

CAMBIO DE CARGA (CODIGO 3)

El código de cambio 3 es usado para cambiar los valores de carga en una barra existente. Los campos de MW y MVAR de CARGA son sustituidos por los nuevos valores ingresados.

1. Entre un 3 en la columna 6
2. Complete el campo NUMERO DE BARRA para la barra a ser cambiada de acuerdo al procedimiento bajo el código de control 5.
3. Entre la nueva carga deseada en los subcampos MW y - MVAR de CARGA, si uno de estos dos subcampos es dejado en blanco este es asumido como cero. Ningún ingreso adicional es requerido.

CONVERSION - TODOS LOS VALORES LISTADOS SON CAMBIADOS (CODIGO 4)

El código de cambio 4 es usado cuando uno o varios cambios - deben ser hechos sobre una barra existente. El programa reemplaza los valores correspondientes de la barra designada con los nuevos valores ingresados en la línea de cambio de datos de barra. Si algún campo es dejado en blanco, dicho campo es ignorado y ningún cambio es realizado sobre el dato correspondiente a este campo.

2. Entre un 4 en la columna 6
2. Entre el número de la barra a ser cambiada en el campo NUMERO DE BARRA.

3. Entre los nuevos valores en la forma apropiada de acuerdo a su respectivo campo, como se especifica bajo el código de control 5.

Si el campo que indica el tipo o código de la barra es ingresado como cero (0) y la barra es una barra regulada por reactivo (MVAR) en el caso precedente al cual se están haciendo los cambios entonces dicha barra será eliminada de la lista de barras reguladas, es decir se convierte a barra de carga.

Si un valor de 1 es ingresado en el campo de código de barra, la barra será ubicada dentro de la lista de barras reguladas si no se encontraba. El campo del código de barra no necesita ser ingresado para cambios en los otros parámetros de una barra regulada.

Cualquier barra puede ser cambiada a barra oscilante reemplazando un 2 en el campo del código de barra, entonces la barra oscilante en efecto es reemplazada por la nueva barra oscilante y su generación y cargas previas permanecen inalteradas (esta barra no es ubicada en la lista de barras reguladas).

El código de control 9 causa que el programa imprima las tablas de barras, líneas y transformadores, y áreas corrientemente en el archivo directo de trabajo. Opcionalmente el usuario puede obtener un archivo en formato de entrada del programa de los datos del caso en efecto con los valores como se hallan almacenados en el momento en que se lo pide, cuando se ejecuta esta opción en el reporte de salida se imprimirá también un listado de este archivo.

Cualquiera de los tres tipos de datos puede ser suprimido de la salida por medio de opciones del usuario.

1. Entre un 9 en la columna 3 de la línea de control.
2. Si se desea suprimir la tabla de datos de líneas y transformadores, ingrese el dígito 1 en la columna 30 de la línea de control.
3. Si la tabla de datos de barra desea ser suprimida, ingrese el dígito 1 en la columna 31 de la línea de control.
4. Si la tabla con datos de área desea ser suprimida, ingrese el dígito 1 en la columna 32 de la línea de control.
5. Si un archivo en formato de entrada de los datos del

caso en efecto es deseado ingrese el dígito 1 en la columna 33 de la línea de control. Si se desea obtener el archivo en formato de entrada pero suprimir las tablas de barras líneas y transformadores y áreas entonces ingrese el dígito 2 en la columna 33, en am los casos el listado de este archivo es impreso en el reporte de salida.

6. Para suprimir en las tablas y/o en el archivo en formato de entrada una o más áreas ingrese el dígito 1 en la columna 40 de la línea de control. Esto causa que se lea la siguiente línea y que debe contener los números de las áreas a ser suprimidas.

Los números de las áreas son ingresados en campos de dos columnas (hay un total de 40 campos para las 40 áreas). Si la clave T0 es ingresada en uno de los campos las áreas correspondientes a los números entre los campos inmediatamente precedente y subsiguiente inclusive son suprimidas. La opción T0 puede ser usada cualquier número de veces en la línea que está siendo leída.

IMPRESION DE LA TABLA DE HISTORIA CODIGO DE CONTROL 10
DE CAMBIOS DE LINEA

El código de control 10 causa que el programa imprima todos los

cambios de líneas y transformadores que se han realizado y que se encuentran almacenados en la tabla de historia de cambios de línea, recuerde que en esta tabla cada cambio tiene un límite para retener cada tipo de cambio.

1. Entre un 10 en las columnas 2 y 3 de la línea de control. Todas las otras columnas deben permanecer en blanco.

SOLUCION AL PROBLEMA DE FLUJO DE CARGA CODIGO DE CONTROL 11

El código de control 11 hace que el programa inicie la solución del problema de flujo de carga. También inicia la producción del reporte normal de salida luego de la solución.

El circuito descrito por el caso base es examinado rigurosamente para determinar un reordenamiento óptimo para el proceso de la solución.

1. Entre un 11 en las columnas 2 - 3 de la línea de control.
2. Entre el nuevo límite de iteraciones en las columnas 7 - 10, ajustado a la derecha; si es que el límite pre-establecido no es el deseado. El programa asume un límite de 20 iteraciones.

3. Entre un 1 en la columna 15 de la línea de control si la magnitud y ángulo del voltaje de un caso precedente desea ser usado como valores iniciales, de otro modo el programa inicializa los voltajes tal como se indica en la tabla IV en la sección 3.3.1.
4. Si un caso resuelto previamente desea ser impreso únicamente, entre la clave NOSOLVE en las columnas 16 a 22.
5. Entre las opciones deseadas como se describen en las tablas XI y XII para un flujo normal y uno simplificado, respectivamente.
6. La línea con el código de control 11 no debe ser leída por el programa antes que todos los datos requeridos hallan sido leídos y procesados.
7. Si la opción de supresión de áreas ha sido seleccionada, la línea conteniendo los números de áreas a ser suprimidas debe inmediatamente seguir a la línea de control con el código 11.

TABLA XI: OPCIONES CON EL CODIGO DE CONTROL 11

COLUMNAS	O P C I O N
30	Si esta columna es 1, una solución de flujo simplificada es iniciada, de otra forma la solución de flujo normal es iniciada.
31	Si esta columna es 1, se produce un listado de todas las líneas de interconexión y sus flujos. De otra manera no se produce este listado.

- 32 Si esta columna 1, un resumen de transformadores se produce en el reporte de salida. De otro modo no se produce el resumen.
- 33 Si esta columna es 1, un resumen de barras reguladas se produce, de otra manera este resumen no se produce.
- 34 Si esta columna es 1, el resumen de las líneas que no están sobrecargadas es suprimido del reporte de salida.
- 35 Esta columna contiene el modo de optimización
0 esquema 3 de Tinney - Walker
1 esquema 2 de Tinney - Walker
2 esquema 1 de Tinney - Walker
- Si no se especifica nada en esta columna, el programa asume el modo cero.
- 36 Si esta columna es 1, la tabla de intercambio de áreas es suprimida del reporte. De otra manera es producida.
- 37 Si esta columna es 1, los taps de los transformadores con LTC son congelados antes de la solución.
- 38 - 39 Estas columnas contienen el número de copias del reporte de salida que se desea. Entre el número de copias ajustado a la derecha. Si se deja el campo en blanco, se producirá una copia.
- 40 Si esta columna es 1, la línea a continuación de

la del código 11 es leída asumiendo que contiene los números de las áreas cuyos reportes de salida no son deseados. Los números de las áreas deben ser entrados en campos de dos columnas, en cualquier orden, y en cualquier campo (un total de 40 campos). Si la palabra T0 es entrada en uno de los campos, los números de áreas en los campos anterior y posterior son asumidos como límites incluyentes para supresión. La opción T0 puede ser usada cualquier número de veces en combinación con supresiones individuales de áreas.

- 41 Si esta columna es 1, el caso es solucionado con el control automático de intercambio de áreas.
- 42 Si esta columna es 1, se produce un sumario de desajuste en barras.

TABLA XII
OPCIONES SOLO PARA EL FLUJO SIMPLIFICADO

COLUMNAS	VALOR	O P C I O N
32	1	Imprime análogo DC del voltaje del sistema completo en orden creciente del número de barra.
	2	Imprime análogo DC del voltaje en orden creciente de sus valores.
	3	Imprime ambos de los resúmenes descritos arriba.
33	1	Imprime análogo DC del voltaje por área (si no son suprimidas) en orden creciente de número de barra.
	2	Imprime análogo DC del voltaje por área (a menos que sean suprimidas) en orden creciente de su valor.
	3	Imprime ambos resúmenes descritos arriba.

GRABACION DE ARCHIVO HISTORICO CODIGO DE CONTROL 12

El código de control 12 hace que el programa grabe un caso pre-establecido en la memoria principal en un archivo histórico, de esta forma el caso grabado en el archivo histórico puede ser recuperado posteriormente (con el código de control 3) y combinado con cambios que se requieren hacer y obtener un nuevo caso que puede ser corrido.

El archivo histórico puede ser de dos tipos, de acceso secuencial o de acceso directo.

PARA ARCHIVO SECUENCIAL

1. Entre un 12 en las columnas 2 y 3 de la línea de control.
2. Entre el número de registro del caso a ser grabado en las columnas 9 y 10, ajustado a la derecha. El número de registro del caso no puede ser mayor que 99. Si este campo es dejado en blanco el caso es grabado como el último registro en el archivo secuencial.
3. Entre el número de la unidad lógica con que estará direccionado el archivo en las columnas 14 y 15, ajustado a la derecha. Si este campo es dejado en blanco

el programa asumirá el número de la unidad lógica como 3, para mayor detalle sobre el número de unidad lógica che que acerca de este en el código de control 3 y en la ta bla IX.

4. Entre la identificación del archivo histórico en las co lumnas 16 a 23 (la identificación constará como máximo de 8 caracteres alfanuméricos).
5. Para suprimir el rebobinado del archivo secuencial ingrese el dígito 1 en la columna 31 de la línea de control. El uso de esta opción se recomienda cuando el próximo caso a ser grabado tiene un número de registro mayor que el caso que esta siendo grabado. Si después que el rebobinado ha sido suprimido, el número de registro del próximo caso a ser grabado es menor o igual al precedente, el archivo será rebobinado y procederá en forma normal.

Cuando un caso es grabado en un archivo secuencial - con un número de registro específico, cualquier caso en el archivo con un número mayor o igual que el espe cificado será destruido.

PARA ARCHIVO DIRECTO

1. Entre un 12 en las columnas 2 y 3 de la línea de con trol.

2. Entre el número de registro del caso a ser grabado, en las columnas 9 y 10, ajustado a la derecha. El máximo número de registros permitidos es dos. Los números pueden ser saltados al grabar los casos pero todos los números deben estar en un rango de solo dos dígitos. Si las columnas 9 y 10 son dejadas en blanco, el programa grabará el caso con el número de registro más bajo disponible. Si un número de registro es especificado en las columnas 9 y 10, y ya existe un caso grabado con el mismo número de registro, el nuevo caso reemplaza al anterior en el archivo directo bajo el número de registro especificado.
3. Entre el número de unidad lógica con que el archivo directo va a ser direccionado, en las columnas 14 y 15, ajustado a la derecha. Si este campo es dejado en blanco, el programa asume un número de unidad lógica de 16, para el uso de otro número de unidad lógica para el direccionamiento del archivo histórico directo vea la tabla IX.
4. Entre la identificación del archivo histórico en las columnas 16 a 23 (máximo 8 caracteres alfanuméricos).
5. Entre el dígito 1 en la columna 30 de la línea de control esto indicará al programa que se va a grabar el caso en mención en un archivo histórico de acceso directo.

Si se desea borrar un caso ya grabado con un determinado número de registro de un archivo histórico de acceso directo use el código de control 12 ingresando un 1 en la columna 30 de la línea de control, ingresando el número de registro del caso a ser borrado en las columnas 9 y 10, ajustado a la derecha, el número de la unidad lógica en las columnas 14 y 15 e ingrese la clave DELETE en las columnas 16 a 21.

Ninguno de los otros casos grabados con diferentes número de registro en el archivo histórico de acceso directo es afectado con esta acción.

Debe tomarse en cuenta que antes de que un archivo histórico de acceso directo sea usado por primera vez este debe ser inicializado por el programa de flujo de potencia.

Para inicializar un archivo histórico de acceso directo ingrese una línea de control con código de control 12, con el dígito 1 en la columna 30 y la clave INITIAL entre las columnas 16 y 22.

PRECAUCION

La inicialización de un archivo histórico de acceso di

recto ya existente, borrará cada uno de los casos grabados previamente en dicho archivo.

IMPRESION DE TITULOS DE CASOS CODIGO DE CONTROL 13
EN ARCHIVO HISTORICO

El código de control 13 es usado para imprimir los títulos de todos los casos en un archivo histórico.

1. Entre un 13 en las columnas 2 y 3 de la línea de control.
2. Entre el número de unidad lógica con el que está direccionado el archivo histórico, en las columnas 14 y 15 - ajustado a la derecha. Si el campo es dejado en blanco, el programa asume un número de unidad lógica 3 para un archivo histórico de acceso secuencial y 16 para un archivo histórico de acceso directo.
3. Entre la identificación del archivo histórico en las columnas 16 a 23.
4. Si el archivo histórico es de acceso directo entre el dígito 1 en la columna 30 de la línea de control.

ANULA HISTORIA DE CAMBIOS DE LINEA CODIGO DE CONTROL 14

El código de control 14 causa que el programa anule el registro de los cambios hechos a los datos de líneas y trans

formadores.

1. Entre un 14 en las columnas 2 y 3 de la línea de control, todas las otras columnas deben quedar en blanco.

LECTURA DE DATOS DE AREA

CODIGO DE CONTROL 15

El código de control 15 hace que el programa lea y almacene el contenido de las siguientes líneas como datos de área. Una línea con 99 en las columnas 3 - 4 señala el fin de los datos de área.

1. Entre un 15 en las columnas 2 - 3 de la línea de control. Todas las otras columnas deben quedar en blanco.
2. La línea de control 15 debe ser seguida por la primera línea de datos de área. La última línea de datos de área debe ser seguida por una línea con 99 en las columnas 3-4.
3. Los datos de área deben ser leídos antes que el programa encuentre la línea con el código de control 11.

El programa de flujo acepta hasta 40 designaciones de áreas. El nombre especificado para cada área aparecerá encabezando cada página del reporte. Los datos para el -

control automático del intercambio entre áreas es entrado por intermedio de este código de control. La selección de si un caso es o no resuelto con control automático - de intercambio de área se lo hace en la línea de código de control 11.

NUMERO DE AREA

COLUMNAS 3 - 4

Este campo contiene el número de área correspondiente al número entrado en el campo AREA de la forma DATOS DE BARRA. Entre el número de área ajustado a la derecha.

La última entrada en este campo debe ser 99, que al ser leído define el fin de datos de área y debe ser seguido por una línea con código de control.

BARRA DE CONTROL

COLUMNAS 5 - 8

Si va a ser usada la opción de control automático de intercambio de área, entre aquí el número de la barra de control de esta área. La generación de Megawattios (Mw) de esta barra podrá tomar cualquier valor para mantener constante el intercambio del área. La barra oscilante del sistema debe controlar el intercambio de su área. Si se especifica otra barra que no sea la oscilan

te, el programa la reemplazará con la barra oscilante.

INTERCAMBIO NETO

COLUMNAS 9 - 14

Entre el intercambio neto deseado para esta área en MW. Exportaciones son entradas como números positivos. Importaciones son indicadas con un signo menos (-) a la izquierda de la entrada.

TOLERANCIA

COLUMNAS 15 - 19

Entre el límite en MW que permitirá que el valor calculado de intercambio de área se desvie para cualquiera de los dos lados, del intercambio deseado.

NOMBRE DEL AREA

COLUMNAS 20 - 55

Entre la designación alfanumérica para el área entre las columnas 20 - 55 (36 caracteres en total). Este nombre será reproducido en cada página de salida que contenga barras con este número de área, y en la tabla de intercambio de área. Si no se suministra esta entrada, el nombre del área aparecerá como espacios en blanco.

LECTURA DE FACTORES PARA CAMBIO CODIGO CONTROL 22
DE CARGAS DE AREA.

El código control 22 indica al programa el comienzo de -
datos para cambiar las cargas de área, esta línea de
control debe inmediatamente ser seguida por la primera
línea que contiene los datos para cambios de las -
cargas de área. Cuando el programa encuentra la línea
de control con el código 22, el programa lee y prog
cesa las siguientes líneas como datos para cambios -
de las cargas de las áreas, hasta que una línea -
con los dígitos 99 en las columnas 3 y 4 es hallada,
esta línea señala el fin de estos datos.

En cualquiera de las áreas sus cargas pueden ser -
ajustadas. Sólo aquellas áreas cuyas cargas van a ser
ajustadas necesitan ser ingresadas mediante este código.
Cada carga dentro del área designada es cambiada.

1. Entre un 22 en las columnas 2 y 3 de la línea de con-
trol.
2. La primera línea con los datos para cambios de las car-
gas de área debe inmediatamente seguir a esta línea de
control.

La última línea con los datos para cambios de las -
cargas de área debe ser seguida inmediatamente por

una línea con los dígitos 99 en las columnas 3 y 4.

El código de control 22 puede ser usado cuando el nivel de carga en una o más áreas es cambiado. La carga sobre cada barra en el área designada es ajustada multiplicando la carga anterior por un factor suministrado por el usuario.

NUMERO DE AREA (COLUMNA : 3-4)

Entre el número del área donde las cargas serán ajustadas

FACTORES DE AJUSTES(COLUMNAS: 5 - 22)

FACTORES DE LOS MW DE CARGA (COLUMNAS 5 - 10)

Entre el factor en por unidad a ser aplicado a los MW de carga de cada barra en el área designada.

FACTOR DE LOS MVAR DE CARGA (COLUMNA: 11 - 16)

Entre el factor en por unidad a ser aplicado a los MVAR de carga de cada barra en el área designada.

Los capacitores / reactores fijos de barras no son afectados.

FACTORES DE LOS MVAR DE CAPACITORES / REACTORES (COLUMNAS: 17-22)

Entre el factor en por unidad a ser aplicado a los MVAR totales de los capacitores / reactores de cada barra en el área designada.

Un ingreso en blanco en cualquiera de estos campos no cambiará la correspondiente cantidad del área.

Un ingreso de un cero en cualquiera de estos campos ajustará a cero la correspondiente cantidad del área.

La última línea de cambio de carga debe ser seguida por una línea con 99 en las columnas 3 y 4.

FIN DE LA CORRIDA DE FLUJO CODIGO DE CONTROL 30

El código de control 30 define el fin de una corrida del programa de flujo de carga. Debe ser la última línea de un archivo de datos de entrada del programa PFLOW.

Cuando el programa encuentra el código de control 30, este cesa el procesamiento de cualquier línea de control posterior y retorna el control a CMS.

1. Entre un 30 en las columnas 2 y 3 de la línea de control, todas las otras columnas deben quedar en blanco.

LINEA DE CONTROL DE ERROR

SIGNO DE INTERROGACION EN
LA COLUMNA 1

Una línea de control con un signo de interrogación (?) en la columna 1 se denomina línea de control de error y debe ser la primera línea de entrada para cada caso de flujo independiente y sucesivos a ser procesados en cadena durante una corrida particular de flujo de carga. La línea de control de error previene de errores que se pueden dar durante la ejecución de uno de los casos de flujo y aún quedan otros casos de flujo independientes por ejecutar en esa corrida. Cuando un error ocurre en un caso, el programa salta las siguientes líneas de control y datos correspondiente a ese caso y procede a buscar una próxima línea de control de error, si los sucesivos casos independientes tienen una línea de control de error al comienzo de cada caso, el programa hallará una línea de control de error al comienzo del próximo caso, entonces este siguiente caso es procesado, si ninguna línea de control de error es hallada en la búsqueda, entonces el programa finaliza la corrida, es decir la línea de control de error al comienzo de cada caso independiente a ejecutarse en cadena, asegura que los casos que

no tienen error serán ejecutados en esa corrida particular.

Este código de control de error puede también ser usado para asegurar que todos los errores son detectados en un caso antes que los restantes datos sean saltados. Por ejemplo si algún error es detectado en los datos de cambio de líneas los subsiguientes datos de cambios de barra y cambios de datos de área, etc., que normalmente serían saltados pueden ser chequeados, si una línea de control de error es incluida inmediatamente precediendo cada línea de control, los datos asociados con aquella línea de control serán completamente chequeados para posibles errores. Adicionalmente si la línea con el código de control 11 es precedida por una línea de control de error, todos los errores asociados con la resolución del flujo son ejecutados, esto no significa que el caso será resuelto ni que un reporte de salida será generado si algún error total es detectado previamente.

1. Entre un signo de interrogación (?) en la columna 1 de la línea de control, el resto de columnas debe quedar en blanco.
2. La línea de control de error debe ser la primera línea para cada caso independiente subsiguiente a ser procesado en una misma corrida (a menos que se quiera dar por finalizada la corrida apenas se detecte un

error).

3. Ciertos errores pueden ser detectados por el programa, en cuyos casos la corrida es inmediatamente finalizada y la línea de control de error es sin importancia.

EJECUCION DEL PROGRAMA DE FLUJO PFLOW

El programa PFLOW que usa el método de Newton - Raphson en la resolución del problema de flujo de potencia - ha sido ambientado en el sistema 4341 de la ESPOL, bajo el modo CMS, aquí se da la información necesaria para ejecutar dicho programa en este sistema, se asume que el usuario tiene los conocimientos básicos para tener una sesión de trabajo bajo este modo de operación en caso contrario debe referirse a los manuales sobre este modo operativo que se disponen en el Centro de Servicios Computacionales de la ESPOL, los pasos son los siguientes:

1. Cree el archivo que contendrá los datos del caso de flujo a resolver, para esto use el comando XEDIT (en forma - abreviada únicamente X) conjuntamente con el nombre y tipo del archivo, el nombre del archivo de datos puede ser cualquier nombre aceptado por el sistema

en CMS, el tipo del archivo debe necesariamente ser PFLOW, por ejemplo supongamos que el usuario decide que el nombre del archivo que contendrá los datos se llame DATOS, entonces para crear el archivo en CMS digite en la línea de comandos de este modo:

X DATOS PFLOW

2. Una vez creado el archivo de datos, usted se encuentra en modo de edición y puede editar el archivo de datos, usando los subcomandos de este modo, es decir puede añadir, insertar o borrar líneas al archivo; digite los códigos de control y los datos del sistema de potencia a resolver de acuerdo a sus necesidades y siguiendo el manual del usuario dado en esta sección.
3. Elija un nombre para su archivo donde se grabarán los resultados de la ejecución del programa PFLOW, este nombre puede ser cualquier nombre aceptado por el sistema en CMS excepto el nombre que eligió para el archivo de datos, supongamos que usted eligió SALIDA.
4. Este paso debe usarlo si usted en su archivo de datos usa el código de control 12 o sea para grabar un

archivo histórico de acceso secuencial o directo, elija un nombre para su archivo histórico y que sea aceptado por el sistema en CMS, supongamos que usted eligió FLUJOS.

En el caso de que usted use el código de control 3 para recuperar un caso de el archivo histórico, usted debe elegir necesariamente el nombre del archivo histórico como debió haber sido grabado con anterioridad dicho archivo histórico.

5. Ejecute el programa de flujo PFLOW mediante el comando EJEPFLOW de la siguiente manera:

En modo CMS en la línea de comando digite:

EJEPFLOW (Nombre del archivo de datos) (Nombre del archivo de resultados)

y luego presione la tecla ENTER, inmediatamente aparecerá en pantalla un logotipo de la Facultad de Ingeniería Eléctrica, debe esperar un momento hasta que automáticamente saldrá en pantalla su archivo de resultados para que usted pueda chequearlo.

Si el nombre del archivo de datos que hemos supuesto es DATOS y el del archivo de resultados es SALIDA ,

entonces usted debe ejecutar el programa digitando - EJEFLOW DATOS SALIDA y presionando la tecla ENTER, automáticamente se creará un archivo que contiene el reporte de salida del flujo de carga del sistema que se manda a resolver este archivo de resultados tendrá como nombre, el nombre que usted eligió, en este caso SALIDA y el tipo de este archivo será PFLOW, luego si usted desea llamar a este archivo deberá - usar su identificación que es SALIDA PFLOW.

6. Si usted requirió el paso (4) o sea que necesita grabar o recuperar un caso de flujo desde un archivo histórico secuencial o directo, adicional - al nombre del archivo de datos y del nombre del archivo de resultados, usted debe proporcionar el nombre del archivo histórico en el comando EJEFLOW así:

EJEFLOW (NOMBRE DEL ARCHIVO DE DATOS) (NOMBRE DEL ARCHIVO DE RESULTADOS) (NOMBRE DEL ARCHIVO HISTORICO)

Para el ejemplo tendríamos:

EJEFLOW DATOS SALIDA FLUJOS

Cuando se usa la opción de un archivo histórico, automáticamente se creará este archivo en el minidisco del usuario con la siguiente identificación para este ejemplo:

HISFILES FLUJOS

o

HISFILED FLUJOS

El nombre de HISFILES identificará al archivo histórico si se trata de archivo histórico de acceso secuencial y el nombre HISFILED en el caso de que se trate de un archivo histórico de acceso directo.

4.4. EJEMPLO DE APLICACION

El problema de flujo de potencia que resolveremos aquí mediante la aplicación del método de Newton - Raphson será el Sistema Guayaquil, incorporando el nuevo punto de alimentación a 69 kv en la S/E Pascuales de INECEL - que incluye la instalación de un transformador de 150 MVA, 138/69 kv con cambio automático de Tap bajo carga (LTC) y nuevas líneas de 69 kv que alimentan las S/E de distribución de EMELEC.

El objetivo que tenemos con este ejemplo es compro-

bar algunas de las ventajas del método de Newton - Raphson y las facilidades que nos da también este programa como son el control de intercambio automático de potencia entre áreas, y el control de voltaje en una barra remota (a distancia) mediante la regulación automática de un transformador LTC, para lo cual en el ejemplo propuesto definimos dos áreas, la una que es el Sistema Nacional Interconectado (SNI) - controlado por el Paute y la otra que es EMELEC - controlado por sus unidades de generación del Salitral, el SNI ha sido simplificado por:

Razones de espacio en los diagramas de flujo de potencia adjuntos en las figuras N^{os}. 4.4. y 4.5., y porque entre el SNI completo y el reducido no hay diferencias de voltaje en las barras, además que se necesita correr una gran cantidad de flujos, con lo cual se ahorra trabajo en el análisis ya que lo que nos interesa es:

- La determinación de la regulación de voltaje en un punto de las líneas de EMELEC, controlada por el transformador LTC de Pascuales; y,
- El intercambio automático de potencia entre áreas, para saber como debe operar la generación del salitral

para no sobrepasar la energía contratada por EMELEC a INECEL.

La S/E Pascuales de INECEL, servirá a las S/E de distribución de EMELEC, mediante 3 líneas de 69 kv que las llamaremos Línea A, Línea B, Línea C, donde:

La Línea A alimentará las S/E Mapasingue, Fisa, Sauce, Cridesa, Cervecería y Germania.

La Línea B alimentará las S/E Ceibos y Alborada y la Línea C, alimentará las S/E Boyacá, Atarazana y Américas, como se puede apreciar en las figuras N° 4.4. y 4.5.

En la tabla XIII se dan los datos de líneas y transformadores para el sistema a resolver y en las tablas XIV y XV, los datos de barras, los datos de generación y carga corresponde a los proporcionados por EMELEC para el año 1.985, tanto para demanda máxima y mínima. Con todo esto se estructuró un archivo de datos para correr el ejemplo y determinar la ubicación del punto de regulación (punto en el cual el transformador LTC tratará de mantener el voltaje constante a 0.99 por unidad para demanda máxima y mínima), tal que las bandas de voltaje (ecuación 4.1) de las S/E de EMELEC, servidas desde el transformador LTC de Pascuales sean las mínimas.

$$\% \text{ BANDA DE VOLTAJE} = (V_{\text{Demanda mínima}} - V_{\text{Demanda máxima}}) \times 100 \quad (4.1)$$

Para este efecto se hicieron varias corridas de flujo con el punto de regulación ubicado en diferentes localizaciones a lo largo de la línea C, se escogió la línea C ya que es la más larga de entre las 3, y la de mayor carga a lo largo de toda la línea, ya que se tiene casi toda su carga al final de la línea, así tenemos que el tap Américas que es la derivación a las S/E mencionadas está a 17 km. de la S/E Pascuales de INECEL.

En la tabla XVI se muestran los voltajes para demanda mínima y máxima y la banda de voltaje para diferentes puntos de localización, cada punto de regulación involucra dos corridas de flujo, comparando las bandas de voltajes que se obtuvieron en las diferentes corridas y que se muestran en la tabla XVI se encontró que el punto de regulación más adecuado está en el tap Américas, que como se puede apreciar es en este punto donde se produce el mayor número de menores bandas de voltaje.

En la tabla XVII se dá un resumen del intercambio automático de potencia entre las dos áreas, con lo cual vemos que en Guayaquil debe generarse 33.70 MW para

que la potencia neta recibida del SNI sea 288.50 Mw.

En la figura N° 4.4., se muestra el Diagrama de Flujo de Potencia para la condición de demanda máxima y en la figura N° 4.5., para demanda mínima, ambos flujos corresponden para el punto de regulación de voltaje a 0.99 por unidad (como se lo puede apreciar en ellos) en el tap Américas, los otros flujos correspondientes a los otros puntos de regulación, no son mostrados aquí, pero su resumen se incluye en la tabla XVI también.

Para el caso mostrado en los flujos de las figuras N° 4.4 y 4.5., hay que anotar que para demanda mínima no se usó la opción de intercambio automático de potencia entre las dos áreas definidas, ya que la demanda total de Guayaquil para mínima carga está por debajo de la potencia máxima que se desea importar del SNI (143.2 Mw de demanda de potencia contra 288 Mw), también es de anotar que por este mismo motivo no fue necesario generar en Guayaquil para mínima carga, ambas soluciones de flujo para máxima y mínima fueron obtenidas en 10 y 5 iteraciones, respectivamente y en ambos casos el Tap del LTC fue ajustado automáticamente por el programa en 1.060 y 1.012, respectivamente, con lo cual se controló automáticamente el voltaje del punto de regulación en 0.99 p.u. que era lo deseado.

CONCLUSIONES Y RECOMENDACIONES

Al finalizar el presente trabajo se dejan expuestas las bases teóricas del método de Newton - Raphson conjuntamente con las técnicas numéricas y de programación como son:

- La eliminación gaussiana ordenada optimamente
- Los esquemas de ordenamiento óptimo de nodos.
- Los esquemas de triangularización por fila de las matrices.
- Los esquemas de almacenamiento compactos
- El esquema de fila de trabajo.

Todas estas técnicas han sido aplicadas para el manejo y operación con matrices dispersas en la resolución del problema del flujo de potencia mediante el método de Newton - Raphson.

Se deja también como contribución un programa de flujo de potencia profesional ambientado en el Sistema 4341, que usa este método de resolución con cada una de las técnicas enumeradas arriba, en resumen el esfuerzo de esta tesis clarifica el método de resolución de Newton - Raphson y la operación con matrices dispersas, de tal forma que la

teoría y la herramienta (el programa) permiten profundizar más sobre estos tópicos y de esta forma lograr adelantos sobre estas técnicas y además permitiendo su uso en otras aplicaciones en las que resulten sistemas de ecuaciones dispersas, como es común en diferentes aplicaciones del análisis de los sistemas eléctricos de potencia y así de esta forma optimizar recursos computacionales como son tiempo y memoria.

En base a la experiencia en corridas de flujo de potencia con otros programas de flujo de carga de la ESPOL (el LFL para sistemas de hasta 50 barras y el LFLO para sistemas hasta 800 barras) que usan el método de Gauss - Seidel como técnica de resolución del problema de flujo de potencia, podemos concluir que con el método de Newton - Raphson:

- Se ha reducido sustancialmente el número de iteraciones para convergencia del problema, así en el ejemplo que se ha presentado en la sección 4.4., que es un sistema de 88 barras - convergió en 10 iteraciones para demanda máxima y en 5 iteraciones para demanda mínima, en la sección 2.4.3., se mencionó que el número típico de iteraciones para una solución sin ajustes está entre 3 a 5 iteraciones, aquí se obtuvo un mayor número de iteraciones debido a los ajustes en la solución del flujo de potencia, estos ajustes son el intercambio de potencia automático, el control de voltaje

en una barra remota y las restricciones de reactivo en las barras de generación, sin embargo esta cantidad de iteraciones es mucho menor que con las obtenidas con el método de Gauss - Seidel, que para el sistema similar entre 50 y 80 barras toma entre 140 y 160 iteraciones y para un sistema de 300 barras toma entre 800 y 1000 iteraciones (todo esto para voltajes iniciales de 1.0 p.u. a 0 grados y con el SNI incluido el Sistema Guayaquil, que se corrieron para el estudio de la planificación y expansión del Sistema Guayaquil).

- El número de iteraciones es aproximadamente independiente del tamaño del sistema a resolver.

De las técnicas numéricas y de programación se concluye que:

- Para la resolución de los sistemas de ecuaciones lineales debe usarse un esquema de ordenamiento de barras para la secuencia de procesamiento de barras, ya que con estos esquemas minimizamos el llenado de la matriz jacobiano durante el proceso de eliminación (optimizamos la dispersidad), el programa usado aquí permite el uso de cualquiera de los 3 esquemas de ordenamiento, la selección de uno de ellos es un compromiso entre el grado de optimización de la dispersidad (y por ende la memoria de almacenamiento) y la complejidad y tiempo computacional.

Del ejemplo corrido se concluye que:

- Podemos controlar el voltaje en una barra distante mediante transformadores LTC, esta facilidad no la tenían los otros programas de flujo de carga existentes en la ESPOL.
- Se puede controlar con precisión la potencia neta importada o exportada entre áreas.
- Se puede representar transformadores cuyo equivalente tiene reactancias negativas.

Se recomienda:

Usar este programa para estudios de flujos de carga de sistemas de potencia de cualquier tamaño hasta 1500 barras y 2500 líneas, es ideal para estudios de contingencia y múltiples corridas, ya que permite almacenar en un archivo histórico, las corridas de casos base que servirán para el análisis de contingencias en futuras corridas.

Implementar el método de flujo de carga desacoplado rápido (que es un método aproximado (no exacto) de resolución del problema de flujo de potencia) usando este programa profesional, para lo cual se deben reemplazar las subrutinas de so

lución del flujo por nuevas subrutinas con el nuevo método , así de esta forma dará como resultado otro programa con las mismas facilidades que el actual.

Investigar sobre modificación de la inversa de una matríz, cuando la matriz original sufre cambios de uno de sus elementos de la diagonal o fuera de la diagonal, esto es aplicable para contingencias en el caso del método desacoplado rápido, donde las submatrices jacobiano son constantes y sus factores solo necesitan ser calculados una vez.

BIBLIOGRAFIA

1. GILBERTO ENRIQUEZ HARPER, "ANALISIS MODERNO DE SISTEMAS ELECTRICOS DE POTENCIA", LIMUSA, 1.981.
2. J. ARRILLAGA AND C.P.ARNOLD, "COMPUTER MODELLING OF ELECTRICAL POWER SYSTEMS", JOHN WILEY & SONS, 1.983.
3. B. STOTT, "REVIEW OF LOAD - FLOW CALCULATION METHODS", PROC.IEEE. 62(7), 916 - 929.
4. WARD J.B., HALE H.W., "DIGITAL COMPUTER SOLUTION OF POWER FLOW PROBLEMS", AIEE TRANS. POWER APP. SYST. VOL 75 JUNE 1956, pp.398 a 404.
5. VAN NESS J., "ITERATION METHODS FOR DIGITAL LOAD FLOW STUDIES", AIEE TRANS. POWER APP. SYST., VOL 78, AUGUST, 1959, pp. 583 - 588.
6. VAN NESS J., AND J.H. GRIFFIN, "ELIMINATION METHODS FOR LOAD - STUDIES ", AIEE TRANS. POWER APP. SYST., VOL 80, JUNE 1961, pp. 299 - 304.
7. N. SATO AND W.F. TINNEY, "TECHNIQUES FOR EXPLOITING THE SPARSITY OF THE NETWORK ADMITTANCE MATRIX" IEEE TRANS. POWER APPARATUS - AND SYSTEMS, VOL. 82 pp. 944 - 949, DEC. 1963.
8. TINNEY W. AND C. E. HART, "POWER FLOW SOLUTION BY NEWTON'S - METHOD", IEEE TRANS. POWER APPARATUS AND SYSTEMS, VOL. PAG.86,

pp. 1449-1460, NOV. 1967.

9. W.F. TINNEY, J.W. WALKER, "DIRECT SOLUTIONS OF SPARSE NETWORK - EQUATIONS BY OPTIMALLY ORDERED TRIANGULAR FACTORIZATION", PROC . IEEE, VOL. 55 pp. 1801-1809, NOV. 1967.
10. E.C. OGBUOBIRI, W.F. TINNEY, AND J.W. WALKER, "SPARSITY DIRECTED DECOMPOSITION FROM GAUSSIAN ELIMINATION OF MATRICES, "IEEE TRANS. POWER APPARATUS AND SYSTEMS, VOL. 89, pp. 141 - 149, JANUARY , 1970.

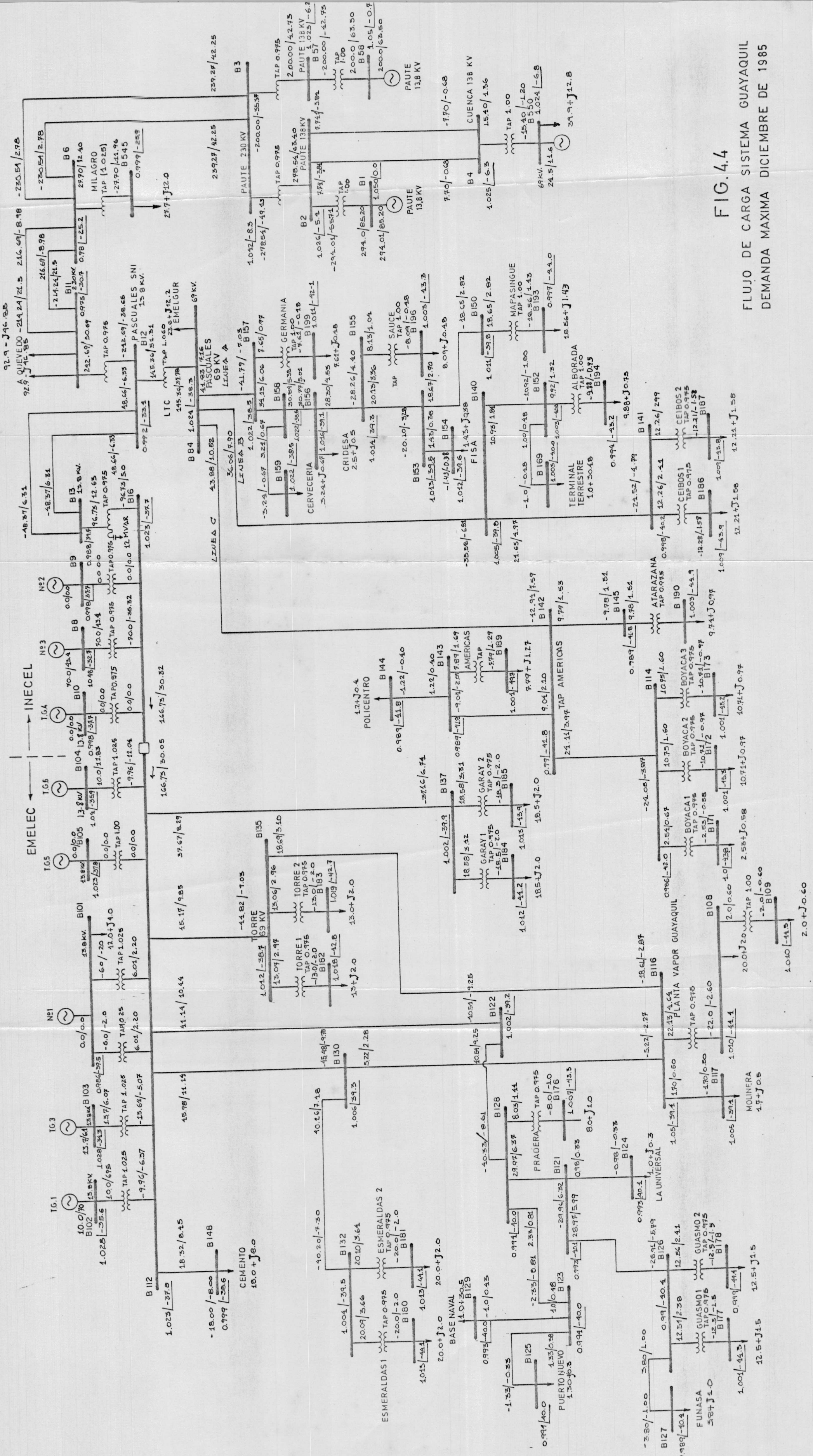


FIG. 4.4

FLUJO DE CARGA SISTEMA GUAYAQUIL
DEMANDA MAXIMA DICIEMBRE DE 1985

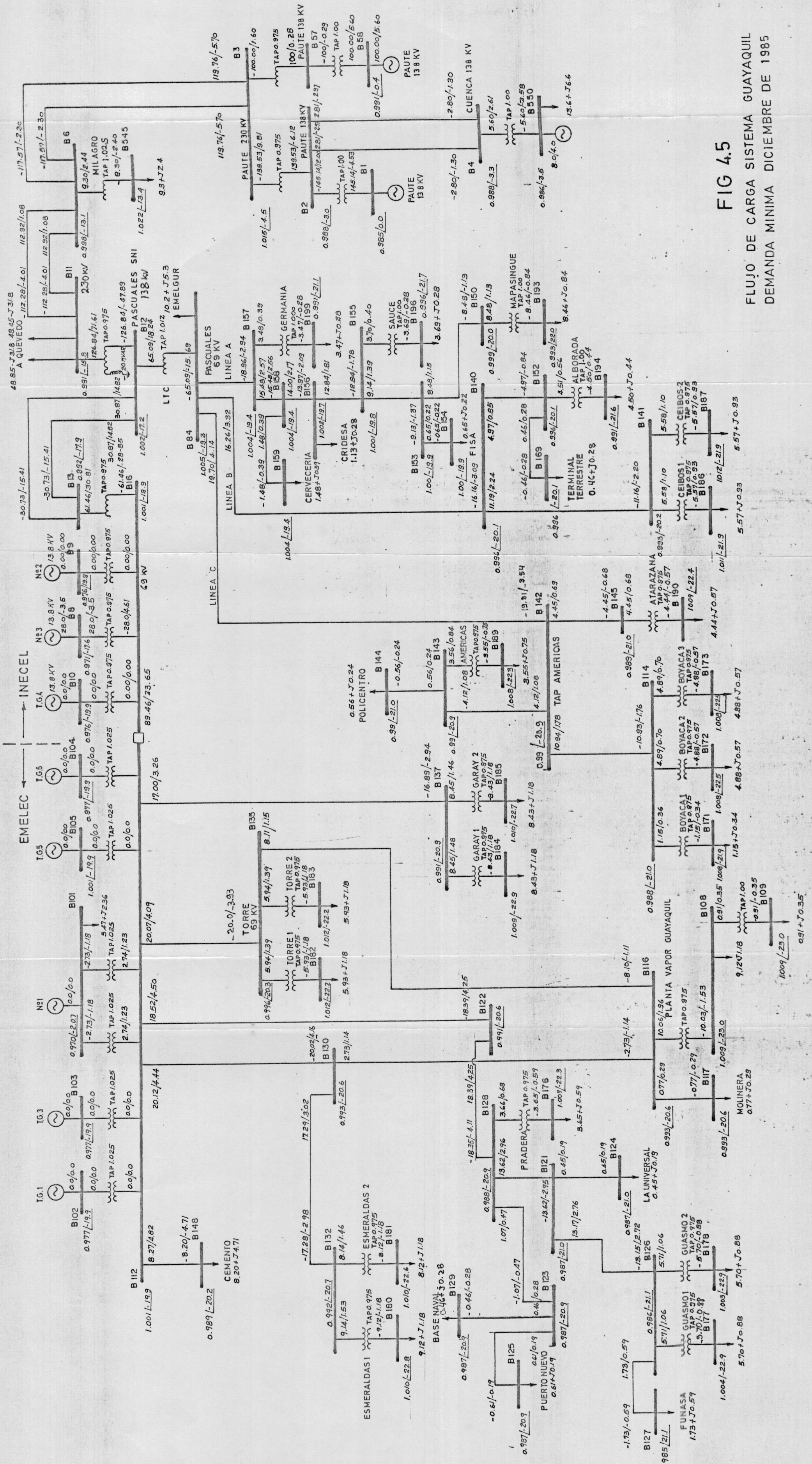


FIG 4.5

FLUJO DE CARGA SISTEMA GUAYAQUIL
 DEMANDA MINIMA DICIEMBRE DE 1985