



UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA
La Universidad Católica de Loja

ÁREA TECNICA

**TÍTULO DE INGENIERO EN ELECTRONICA Y
TELECOMUNICACIONES**

**Control de voltaje en redes de distribución con variación de carga y la
presencia de un Generador Distribuido (DG).**

TRABAJO DE TITULACIÓN

AUTOR: Maza Alejandro, Wilson Fernando.

DIRECTOR: Castro Mendieta, José Raúl, Dr.

LOJA – ECUADOR

2018



Esta versión digital, ha sido acreditada bajo la licencia Creative Commons 4.0, CC BY-NY-SA: Reconocimiento-No comercial-Compartir igual; la cual permite copiar, distribuir y comunicar públicamente la obra, mientras se reconozca la autoría original, no se utilice con fines comerciales y se permiten obras derivadas, siempre que mantenga la misma licencia al ser divulgada. <http://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>

2018

APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN

Doctor.

José Raúl Castro Mendieta.

DOCENTE DE LA TITULACIÓN

De mi consideración:

El presente trabajo de titulación: Control de voltaje en redes de distribución con variación de carga y la presencia de un generador distribuido, realizado por Maza Alejandro Wilson Fernando, ha sido orientado y revisado durante su ejecución, por cuanto se aprueba la presentación del mismo.

Loja, septiembre de 2018

f)

DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS

“Yo **Maza Alejandro Wilson Fernando** declaro ser autor del presente trabajo de titulación: *Control de voltaje en redes de distribución con variación de carga y la presencia de un generador distribuido (DG)*, de la Titulación de Electrónica y Telecomunicaciones, siendo José Raúl Castro Mendieta director del presente trabajo; y eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones legales. Además, certifico que las ideas, conceptos, procedimientos y resultados vertidos en el presente trabajo investigativo, son de mi exclusiva responsabilidad.

Adicionalmente declaro conocer y aceptar la disposición del Art. 88 del Estatuto Orgánico de la Universidad Técnica Particular de Loja que en su parte pertinente textualmente dice: “Forman parte del patrimonio de la Universidad la propiedad intelectual de investigaciones, trabajos científicos o técnicos y tesis de grado o trabajos de titulación que se realicen con el apoyo financiero, académico o institucional (operativo) de la Universidad”

f.

Autor: Maza Alejandro, Wilson Fernando.

Cédula: 1104585383

DEDICATORIA

Dedico este trabajo a mis padres por todo el apoyo brindado en estos años de estudio, ya que sin ellos no estaría logrando este tan ansiado objetivo. A mi madre Delfina por haber sido el pilar más importante para obtener este logro. En especial a la memoria de mi Padre Juan José que desde el cielo me brindo la fuerza y sabiduría para poder culminar mi carrera universitaria.

A mis hermanos por el apoyo brindado durante todo el transcurso de mis estudios ya que han sido un gran apoyo durante estos años.

AGRADECIMIENTO

Primeramente, agradezco a Dios por darme la fuerza y sabiduría para poder tomar el camino correcto durante estos años de estudio.

A mi familia por todo el apoyo brindado que sin ellos no hubiera alcanzado este logro

Agradezco a mi tutor Dr. Raúl Castro quien con su experiencia y sabiduría supo guiarme por el camino correcto durante mi investigación.

INDICE DE CONTENIDOS

CARÁTULA.....	i
APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN	ii
DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS.....	iii
DEDICATORIA.....	iv
AGRADECIMIENTO	v
RESUMEN.....	1
ABSTRACT.....	2
INTRODUCCIÓN.....	3
CAPITULO I.....	6
REVISION DE LA LITERATURA	6
CAPITULO II.....	10
ANALISIS Y ESTUDIO DE LA RED DE DISTRIBUCION ELECTRICA	10
2.1 Red de distribución eléctrica.....	11
2.2 OLTC.....	11
2.3 IEEE 13 Node Test Feeder.....	14
2.4 Generación Distribuida.....	17
CAPITULO III.....	18
OPENDSS ESTUDIO Y SIMULACION	18
3.1 Componentes de OpenDSS	19
3.1.1 Control de OpenDSS desde Matlab.....	19
3.1.2 Iniciar la interfaz COM.....	21
3.1.3 Obtener información de OpenDSS desde Matlab.....	22
3.1.4 Añadiendo y Editando elementos desde Matlab.....	23
3.2 Simulación de la red IEEE nodo 13 (OpenDSS).....	24
CAPITULO IV	27
ANALISIS Y SIMULACION DE LA RED IEEE 13 NODE EN MATLAB APLICANDO VARIACIÓN DE LA CARGA Y PRESENCIA DE UN DG.....	27
4.1 Simulación de la red en Matlab.....	28
4.2 Comprobación de valores obtenidos sin presencia del DG y de la carga.....	29
4.3 Simulación de disturbios en la red.....	32
4.4 Introducción del DG y la carga en Simulink.....	33

4.5 Variaciones de voltaje en las barras de la red.....	34
4.6 Perfil del voltaje en el bus piloto.....	36
4.7 Análisis de Perdidas de la red por introducción del DG y la carga.....	37
CONCLUSIONES.....	39
RECOMENDACIONES	41
REFERENCIAS.....	42
ANEXOS.....	45

RESUMEN

Los sistemas de distribución son el medio que permite que la energía eléctrica se entregue a los centros de consumo desde la red de transmisión. Estos sistemas están expuestos a fallas de diferentes tipos que dependerá de los dispositivos de protección para asegurar el suministro de energía durante su operación. Actualmente en muchos sistemas de distribución están apareciendo nuevos actores como la generación distribuida, que aumenta las perturbaciones en el sistema.

Debido al crecimiento acelerado de las redes eléctricas y la exigencia de la calidad del servicio hacia los consumidores, el sistema de protección de los sistemas eléctricos de distribución ha adquirido mayor importancia. Lo que se busca en un sistema de distribución que sea fiable, es decir que el servicio sea constante y de ínfima capacidad a la percepción del usuario, donde se pueda disminuir las perturbaciones al momento de introducir una nueva carga al sistema y un generador distribuido, mediante transformadores OLTC.

Palabras clave: Generación distribuida, redes de distribución, OpenDSS, OLTC.

ABSTRACT

The distribution system is the medium that allows the electrical energy to be delivered to the centers of consumption from the transmission network. These systems are exposed to failures of different kinds that will depend on the protection devices to ensure power supply during operation. Currently in many distribution systems new plaintiffs are emerging as distributed generation, which increases the perturbation in the system.

Due accelerated growth of electrical networks and the demand for the quality of service to consumers, the systems of protection of the electrical system of distribution has gained greater importance. What is sought in a systems of distribution that is reliable, that the service is constant and of negligible capacity to the perception of the consumer, where it can reduce the disturbances when introducing a new load to the system and a distributed generator, using OLTC transformers.

Keywords: Distributed generation, distribution networks, OpenDSS, OLTC.

INTRODUCCIÓN.

Debido al constante crecimiento de la población y alta demanda de consumo de la energía eléctrica, se busca optimizar las redes de distribución para que el consumidor disponga de una buena calidad de servicio y las empresas distribuidoras de energía eléctrica minimicen sus pérdidas. Las energías renovables se están convirtiendo en un importante actor en las redes de distribución como una solución a la creciente demanda. Estos generadores, conocidos como Generación Distribuida pueden inyectar energía para satisfacer la demanda y están conectados cerca de los consumidores finales (Castro, Saad, Lefebvre, Asber, & Lenoir, 2016b).

Las redes de distribución eléctrica entregan potencia activa y reactiva a los clientes. En sistemas eléctricos de potencia, la demanda de energía activa y reactiva es variable y desbalanceado (Castro et al., 2016b). Esto provoca variaciones de voltaje en todos los puntos de la red.

Esta variación de voltaje, las empresas eléctricas controlada utilizando bancos de capacitores, los cuales inyectan potencia reactiva a la red para mantener el voltaje dentro de los límites establecidos. De forma similar pero inversa, se puede controlar el voltaje usando bancos de inductores, los cuales consumen potencia reactiva. Finalmente, las empresas eléctricas usan transformadores OLTC (cambiador de tomas bajo carga) aumentando o disminuyendo el número de devanados del transformador por medio de un mecanismo cambiador de tomas, para controlar el voltaje (Redfern, n.d.). Es estas redes se las denomina redes pasivas.

Las variaciones de carga y la Generación Distribuida (DG) causan variaciones de voltaje en las barras de la red. El DG podría cambiar la dirección del flujo de potencia. Muchos investigadores han trabajado en solucionar los disturbios sobre el voltaje, la reducción de pérdidas usando DGs y otros autores buscan el óptimo dimensionamiento del DG (Jose & Mendieta, 2016)(Richardot et al., n.d.).

Para realizar el modelamiento de la red se usará el software para simular redes de distribución eléctrica llamado OpenDSS (R. C. Dugan, 2009). Simulink de Matlab

permite trabajar en firma conjunta con OpenDSS (Tseng, 2014). Trabajando en forma conjunta (OpenDSS – Matlab) permite simular la red de distribución y realizar todo el análisis variaciones de y comparar los resultados.

Para demostrar y analizar la variación y el control del Voltaje realizado por el OLTC se simulará la red desbalanceada de distribución IEEE 13-node test feeder (Caldon, Coppa, Sgarbossa, & Turri, 2013) usando OpenDSS y Matlab. Para crear la primera variación de voltaje se le añadirá a la red de distribución un DG en una de las barras, observaremos la variación de voltaje y el control del voltaje por parte del OLTC. Una segunda variación de voltaje obtendremos al añadir una carga eléctrica. La variación del voltaje en las barras y como se estabiliza el voltaje al momento de aplicar el OLTC se podrá observar en la simulación (Castro et al., 2016b).

Gracias al avance tecnológico y computacional, existen otros métodos utilizados por las empresas eléctricas para mantener el voltaje estable ante variaciones de la carga para disminuir el trabajo de los OLTCs como inyectar coordinadamente potencia reactiva y activa generada por los DGs o otras fuentes. A esto se le denomina redes activas

Otros investigadores usan la potencia activa y reactiva de los DGs (redes activas) no solamente para mantener el voltaje entre los límites establecidos, sino para minimizar las pérdidas y disminuir el número de actuaciones del OLTC (Castro, Saad, Lefebvre, Asber, & Lenoir, 2016a).

Objetivo principal.

Simular y analizar redes de Distribución eléctrica con variaciones de carga y con la presencia de un DG.

Objetivos específicos.

- 1) Estudiar y simular una red de distribución eléctrica (OpenDSS).
- 2) Analizar el impacto de disturbios (variación de carga y presencia de un DG) en redes de distribución eléctrica.
- 3) Comprobar la variación de OLTC para mantener el voltaje dentro de los límites.

CAPITULO I

REVISION DE LA LITERATURA

Algunos autores presentan trabajos de investigación en la óptima ubicación y el adecuado tamaño de los DGs en redes de distribución eléctrica (Fernercom, 2007) que apoyan activamente la red.

El Control de Voltaje Coordinado (CVC) maneja óptimamente la potencia reactiva en las diferentes partes de la red para mantener el voltaje dentro de los límites permitidos. Para coordinar adecuadamente la red debe permitir realizar un control en las fuentes de potencia reactiva. La potencia reactiva de los DGs en redes de distribución pueden ayudar al CVC para mantener el voltaje regulado (Richardot et al., n.d.).

Otras investigaciones (Castro et al., 2016b) representan el voltaje de variación en cada área de control por las variaciones en algunas barras seleccionados llamados " barra piloto ". Entonces, el objetivo es mantener los voltajes en las barras pilotos dentro de un rango predeterminado.

Mantener los voltajes dentro de los límites de operación pre asignados y minimizar las pérdidas usando la potencia reactiva de los DG conectados a la red son los principales objetivos. El trabajo de (Cavvaro & Carli, 2017) propone dos tipos de soluciones, las cuales son puramente locales. El DG actualiza la cantidad de potencia reactiva que se inyectará basándose únicamente en mediciones locales de la magnitud de los voltajes.

En (Castro et al., 2016a) el control de voltaje coordinado (CVC) en la red de distribución ajusta el voltaje en la barra piloto. El CVC utiliza el problema de múltiples objetivos para minimizar la variación de voltaje en la barra piloto y minimizar las pérdidas en potencia activa y reactiva. En este trabajo, los autores proponen varios métodos de optimización para resolver el problema del control de voltaje en la red.

El aumento de voltaje también puede mitigarse utilizando métodos de control de voltaje activo, manejando adecuadamente la potencia activa y reactiva de las fuentes de generación. El control de voltaje activo puede, en muchos casos, disminuir significativamente los costos totales de la red de distribución (Castro et al., 2016b).

Según (Kulmala, Repo, & Jarventausta, 2014) los métodos de control de voltaje activo deben utilizar información de toda la red de distribución y usar técnicas de optimización o técnicas de análisis para coordinar adecuadamente el flujo de potencia en la red y mantener el voltaje dentro de los límites pre establecidos. Los métodos de control de voltaje pueden determinar sus acciones de control basadas en reglas de control o usar algún tipo de algoritmo de optimización. La mayoría de los algoritmos de control de voltaje propuestos en publicaciones están centralizados, pero también se han propuesto métodos distribuidos que utilizan sistemas multiobjetivos.

Para poder ubicar adecuadamente los DGs en una red de distribución, para poder encontrar el tamaño correcto del DG es necesario simular la red de distribución y usar técnicas de optimización. Los autores (Meghasai, Monger, Vega, & Krishnaswami, 2015) usan el software OpenDSS. Para modelar la red de distribución y para usar técnicas de optimización y técnicas de análisis usan Matlab. OpenDSS es el software que permite simular una red de distribución desbalanceada a la que se le puede añadir nuevas cargas y nuevos elementos (DGs).

El software OpenDSS es el encargado de calcular el flujo de potencia de la red de distribución. OpenDSS puede trabajar simultáneamente con Matlab creando una herramienta que permite realizar diferentes análisis. OpenDSS es un software gratuito que se especializa en redes eléctricas (Calderaro, Galdi, Lamberti, Piccolo, & Graditi, 2014).

Cuando la red tiene grandes cambios en la demanda (grandes variaciones de cargas) produce una variación de voltaje en las barras de la red. La red de distribución usa los OLTC para corregir el voltaje. La norma ANSI propone que los límites de variación de voltaje deben mantenerse dentro de $\pm 3\%$ de su valor nominal (Meghasai et al., 2015).

CVC ayuda a corregir el voltaje de en las barras coordinando óptimamente los recursos de potencia reactiva en la red. De esta manera, CVC disminuye el trabajo mecánico del OLTC.

Disponer de métodos computacionales para simular y analizar los impactos de disturbios eléctricos en redes de distribución permitirá buscar mejores técnicas de control y comprobar su funcionamiento.

CAPITULO II

ANALISIS Y ESTUDIO DE LA RED DE DISTRIBUCION ELECTRICA

2.1 Red de distribución eléctrica.

La red de distribución eléctrica o sistema de Distribución de Energía Eléctrica es el conjunto de elementos que conforman un circuito con una determinada tensión nominal. Los sistemas eléctricos de distribución pueden clasificarse por su nivel de tensión (tabla 2.1) de acuerdo a criterios y de normas (Red & Energía, n.d.).

En Ecuador el nivel de voltaje estándar para sistemas eléctricos de Baja tensión es de 120 V y una frecuencia de 60Hz. De acuerdo al Consejo Nacional de Electricidad (CONELEC) se tiene la siguiente clasificación de niveles de voltaje en nuestro país (Rodrigo, 2013).

Tabla 2.1: Rangos especificados por el consejo nacional de electricidad.

Distribución.	Baja tensión	120 V – 600 V
Distribución.	Media tensión	600 V – 40 kV
Transmisión.	Muy alta tensión	69 kV – 138 kV – 230 kV

Fuente: CONELEC

Elaboración: CONELEC

2.2 OLTC.

Los Voltajes en una red de distribución varían en el tiempo. La variación del voltaje depende de la variación de la carga eléctrica. El sistema de distribución debe mantener su voltaje nominal para garantizar la calidad de energía a sus clientes.

Un cambiador de tomas bajo carga (OLTC) tiene como función principal modificar la relación de transformación de los transformadores de potencia (Dohnal, 2013), como respuesta a un cambio de carga en el sistema, permitiendo regular el voltaje de salida del transformador a los niveles requeridos sin la interrupción del servicio (Sergio, 2012).

La Figura 2.1 muestra un transformador de cambio de toma de carga (transformador OLTC). Los transformadores OLTC combinan un transformador reductor con un

regulador de voltaje. Normalmente se requieren dos transformadores OLTC por subestación a fin de tener capacidades de transferencia de carga para fines de mantenimiento del regulador (Lane, Chen, Croda, Welch, & Editor, n.d.).



Figura 2.1. Transformador de cambio de tomas de carga

Fuente: (Lane et al., n.d.).

Elaboración: (Lane et al., n.d.).

2.2.1 Estructura general de un OLTC

Un OLTC está estructurado por los siguientes componentes: un Ruptor (diverter switch), un Selector (selector switch) y, puede contener un Preselector.

La estructura básica de un OLTC está compuesta por los siguientes elementos (Lane et al., n.d.):

- 1) Caja de comando: contiene todas las partes mecánicas y eléctricas necesarias para el accionar el transformador OLTC.
- 2) Mecanismo de transmisión: es el encargado de realizar cambios de una toma a otra.
- 3) Cabeza del OLTC: es un receptáculo metálico situado en la parte superior al que le sirve de soporte.
- 4) Selector (selector switch): realiza la selección de dos tomas a conmutar a través de dos contactos móviles. Preselector: conecta el devanado principal con el devanado de regulación del transformador.
- 5) Ruptor (diverter switch): es el elemento que realiza la transferencia de intensidad de carga desde la toma en servicio a la próxima toma seleccionada sin interrumpir el servicio.

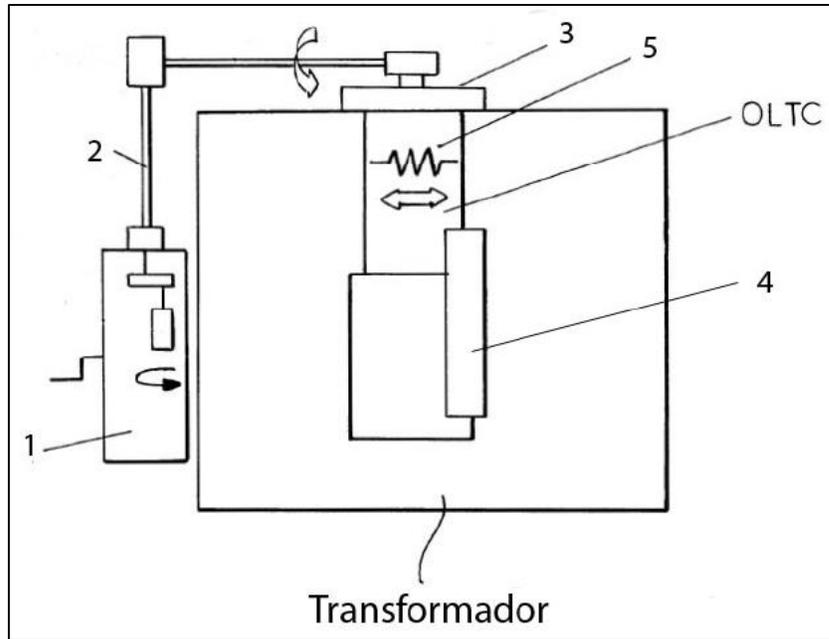


Figura 2.2. Estructura general de un OLTC

Fuente:(Sergio, 2012).

Elaboración: (Sergio, 2012).

2.3 IEEE 13 Node Test Feeder.

La red IEEE 13 Node Test Feeder es una red de distribución eléctrica desbalanceada que se utiliza para probar características comunes de análisis de distribución. Las características principales de esta red son: pequeño, relativamente cargado, tiene un solo regulador de tensión en la subestación, líneas aéreas y subterráneas, dos condensadores de derivación, transformador de línea y consta de un total de 9 cargas desequilibradas entre trifásicas y monofásicas (Planning, 1991).

2.3.1 Modelos de Carga: Las cargas pueden ser conectadas en una barra (carga puntual) o distribuidos uniformemente a lo largo de una sección lineal (carga). Las cargas pueden ser trifásicas (balanceadas o no balanceadas) o monofásico. Las cargas trifásicas se conectan en estrella (Wye) o triangulo

(delta), mientras que las cargas monofásicas pueden conectarse línea a neutro o línea a línea. Todas estas cargas pueden ser modeladas como kW constante y kVAr (PQ), impedancia constante (Z) o Corriente constante (I) (IEEE Distribution System Subcommittee, 2001).

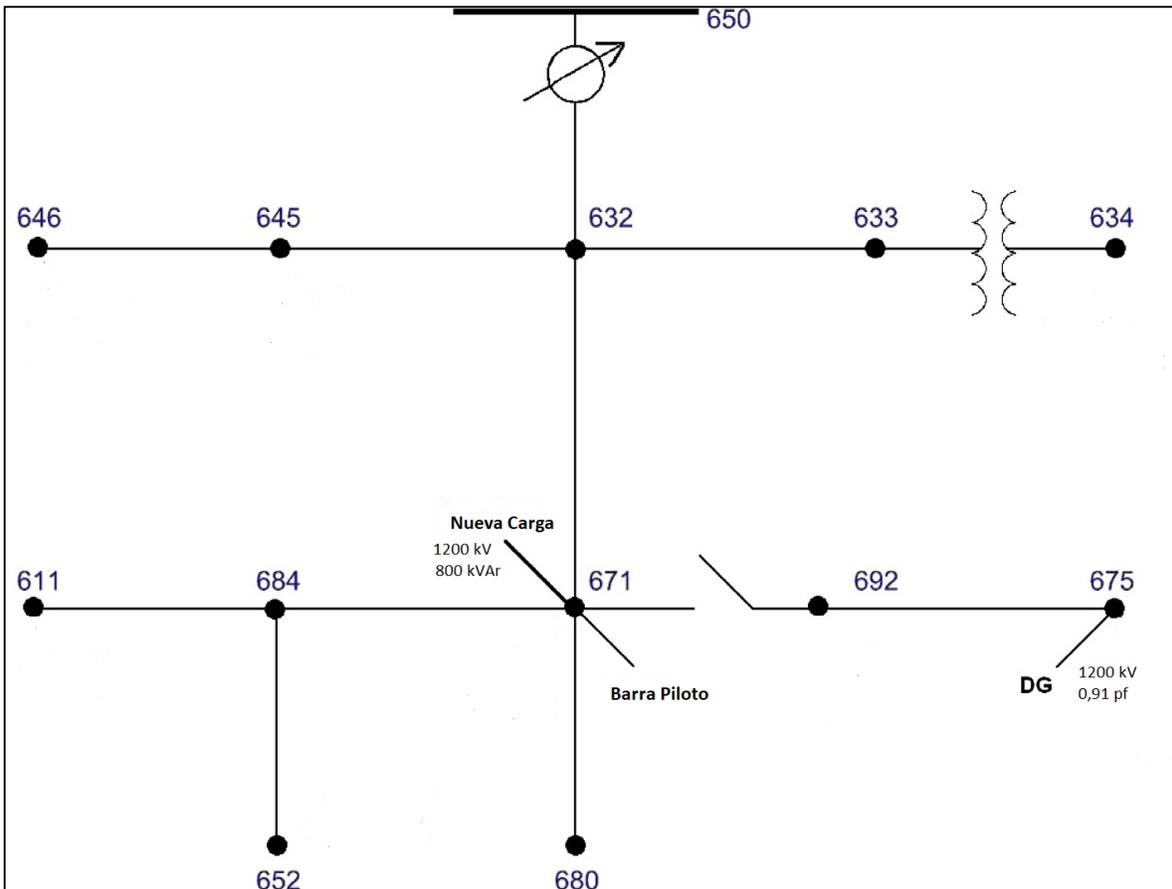


Figura 2.3. Red de distribución IEEE 13 Node Test Feeder.

Fuente: (IEEE Distribution System Subcommittee, 2001).

Elaboración: (IEEE Distribution System Subcommittee, 2001).

En la Figura 2.3 podemos observar la red de distribución IEEE 13 node en el cual trabajaremos en este proyecto, también podemos observar las barras en donde vamos a trabajar introduciendo el DG con una potencia de 1200 kW con un factor de potencia de 0,91 (barra 675) y la variación de la carga trifásica en triángulo con

una Potencia activa 1200 kW y una potencia reactiva de 800 kVAr (barra 671). La barra 671 es la barra más sensible a las variaciones de la carga. Esta barra es llamada barra piloto.

2.3.1 El alimentador de prueba de nodo IEEE 13

En resumen la red IEEE 13 node test presenta las siguientes características (Planning, 1991)(Saha & Johnson, 2016).

1. Nivel de voltaje del alimentador de 4,16 kV (Red de distribución Tabla 2.1).
2. Un regulador de tensión (OLTC) que consta de tres unidades monofásicas conectadas en wye colocada en la barra 650 (figura 2.3).
3. Líneas aéreas y subterráneas con ajuste de fase.
4. Bancos de condensadores de derivación (barra 675 y 611).
5. Transformador en línea entre la barra 633 y 634.
6. Cargas eléctricas distribuidas y no balanceadas (tabla 2.2).

La Tabla 2.2 muestra los valores de carga por defecto para la red IEEE 13. Valores que se están utilizando en la simulación (IEEE Distribution System Subcommittee, 2001).

Tabla 2.2. Datos de carga puntual para IEEE 13.

Node	Load	Ph-1	Ph-1	Ph-2	Ph-2	Ph-3	Ph-3
	Model	kW	kVAr	kW	kVAr	kW	kVAr
634	Y-PQ	160	110	120	90	120	90
645	Y-PQ	0	0	170	125	0	0
646	D-Z	0	0	230	132	0	0
652	Y-Z	128	86	0	0	0	0
671	D-PQ	385	220	385	220	385	220

675	Y-PQ	485	190	68	60	290	212
692	D-I	0	0	0	0	170	151
611	Y-I	0	0	0	0	170	80
TOTAL		1158	606	973	627	1135	753

Fuente: (IEEE Distribution System Subcommittee, 2001).

Elaboración: (IEEE Distribution System Subcommittee, 2001)

2.4 Generación Distribuida.

La generación distribuida (DG) se la define como cualquier tecnología que genera electricidad a pequeña escala, y que se encuentra cerca al consumidor (Fernercom, 2007).

El DG utilizado en el presente proyecto es un generador constante, el cual se lo ingresara en la barra 675 con las siguientes especificaciones: DG de 1200 kW, \pm 0,91 pf en el tiempo de simulación ($t = 100$ s). Se escogió la barra 675 debido a que varios autores (Castro et al., 2016b) que realizaron un estudio de esta red llegaron a la conclusión de que esta barra es la más óptima para colocar un DG para reducir las pérdidas de la red, disminuyendo así la corriente que atraviesa por la red (Jose & Mendieta, 2016).

Así mismo, los valores de la carga que se introducirá en la barra 671 como carga variable, en el tiempo de simulación ($t = 350$ s), provocarán una variación de voltaje en las barras de la red. La carga trifásica introducida es de (1200 kW y 800 kVAr). La simulación se ejecuta en un tiempo total de 500 s, mostrando así el comportamiento de los voltajes al momento de introducir el DG y la carga a la red y la regulación del voltaje realizada por le OLTC.

CAPITULO III
OPENDSS ESTUDIO Y SIMULACION

OpenDSS es una herramienta de simulación de circuitos eléctricos, que consta de una interfaz gráfica de usuario limitada. Soporta casi todos los análisis de dominio de frecuencia (estado estacionario sinusoidal) que se realizan comúnmente en los sistemas de distribución de energía eléctrica. Se puede manejar desde Excel VBA, MATLAB entre otros (R. C. Dugan, 2009).

3.1 Componentes de OpenDSS

El programa OpenDSS es utilizado para crear circuitos y sistemas eléctricos, el cual utiliza un sin número de elementos los cuales se los puede configurar con gran facilidad (CLARO BÁEZ, 2016). Los elementos que más se utilizan son:

Elementos de suministro de potencia.

- Líneas. Todo tipo de líneas y cables para distribución.
- Transformadores. Multifase y multibobinados.
- Condensadores. En serie y en derivación.

Elementos de conversión de potencia.

- Generadores.
- Cargas disipadoras de energía.
- PVSysyem. Sistemas de energía fotovoltaica incluyendo panel e inversor.
- Almacenaje.

Medidores.

- EnergyMeter. Permite medir potencias y pérdidas.
- Monitor. Permite medir cantidades en un determinado punto del circuito.
- Sensores.

3.1.1 Control de OpenDSS desde Matlab.

Para hacer que funcione OpenDSS sin necesidad de ejecutar su interfaz gráfica (OpenDSS GUI), se debe utilizar una aplicación independiente que permite comunicarse con el COM server (Reales & Impresas, 2012). Además, OpenDSS puede conectarse con otros programas tales como Matlab, Visual Basic, Python o Excel, lo que permite un amplio análisis de información (CLARO BÁEZ, 2016).

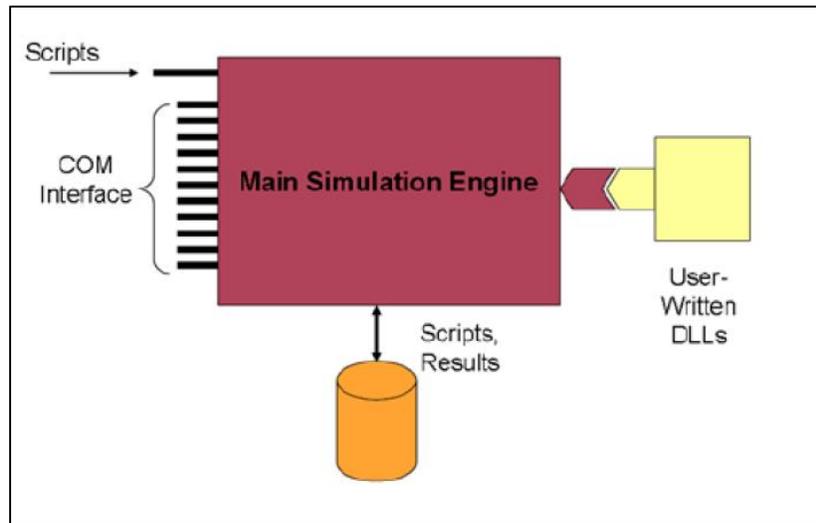


Figura 3.1. Estructura de OpenDSS

Fuente: (R. Dugan, 2009).

Elaboración: (R. Dugan, 2009).

Para iniciar el COM Server se realiza a través de una función creada en Matlab. Cuando OpenDSS ha empezado a inicializarse, se empieza a compilar el circuito a través de su script (.DSS) (R. C. Dugan, 2009).

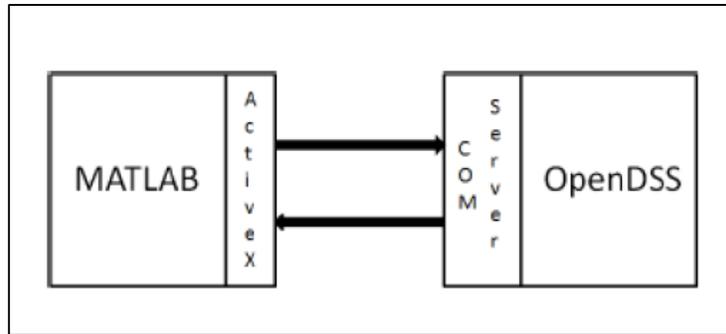


Figura 3.1. Comunicación Matlab y OpenDSS.

Fuente: (CLARO BÁEZ, 2016).

Elaboración: (CLARO BÁEZ, 2016).

3.1.2 Iniciar la interfaz COM

Para iniciar la interfaz COM lo primero es realizar una llamada a la función `DSSStartup`, esto se lo puede lograr de muchas formas, pero la función que se utiliza más habitualmente se presenta a continuación (CLARO BÁEZ, 2016):

$$[DSSCircObj, DSSText, gridpvPath] = DSSStartup$$

La función `DSSStartup` inicia OpenDSS en segundo plano y nos devuelve tres salidas que serán explicados a continuación:

- **DSSCircObj**, es el cual contiene el circuito activo (`DSSCircObj.ActiveCircuit`), el cual aún no se ha compilado y la interfaz de texto para OpenDSS (`DSSCircObj.Text`). `DSSCircObj` va a estar vacío hasta que un circuito sea compilado.
- **DSSText**, es la interfaz de texto contenida dentro de `DSSCircObj`. Se define de esta manera para facilitar el uso dentro de la ventana de comandos de Matlab. `DSSCircObj.Text.Command`, `Text.command` y `DSSText.Command` son los que nos permiten devolver el mismo resultado en forma de interfaz de texto.
- **gridpvPath**, es el que contiene la ruta de localización en el ordenador para la herramienta.

3.1.2.1 *Compilación del circuito.*

Para abrir un circuito creado en OpenDSS desde Matlab, se usa la interfaz de texto con el código "compile". Con esta línea se especifica la ubicación exacta del archivo que vamos a llamar desde Matlab, se debe verificar que la ubicación del archivo sea la correcta de lo contrario el programa nos dará un error y no se podrá compilar, se debe seleccionar el archivo específico que se correrá con el programa OPENDSS (Quintanilla Tovar & Zamora Cárdenas, 2017):

DSSText.Command

```
= 'Compile "(C:\Users\DELL\Documents\gp sistemas estocasticos  
\frecuencia en opendss\IEEE3T1\IEEE13\Matlab\IEEE13_OLTC.dss)";
```

Se debe seleccionar la ruta específica del archivo .DSS que se va utilizar mediante el comando "compile", esto se debe hacer para evitar errores.

3.1.3 **Obtener información de OpenDSS desde Matlab.**

Cuando la interfaz COM ha sido iniciada y el circuito se haya compilado, se procede a utilizar la ventana de comandos para la comunicación entre los dos programas.

La función DSSCircObj.methods se la utiliza para poder ver los métodos que están disponibles, los cuales nos permite poder interactuar con la interfaz. Mediante la instrucción DSSCircObj.get nos muestra la interfaz principal.

El comando DSSCircObj.get nos permite devolver algunos punteros de objetos de la interfaz del servidor COM en OpenDSS. (CLARO BÁEZ, 2016). Con esta línea

podemos cambiar las propiedades de los elementos del circuito en OpenDSS (Quintanilla Tovar & Zamora Cárdenas, 2017)

```
DSSCircuit = DSSCircObj.ActiveCircuit ;
```

Mediante la línea *DSSCircuit.methods*, se podrán ver los métodos adecuados para el circuito resuelto. Nuevamente, usando *DSSCircuit.get*, se observa todos los diferentes campos e interfaces que se encuentran en el circuito (CLARO BÁEZ, 2016):

```
DSSCircuit.methods
```

```
DSSCircuit.get
```

3.1.4 Añadiendo y Editando elementos desde Matlab

Lo que más se utiliza en la interfaz de texto es añadir y editar elementos del circuito. Para ello se utiliza los comandos de OpenDSS "new" y "edit", mediante los cuales se puede añadir diferentes elementos, movidos o cambiados a través de Matlab como se puede observar en el siguiente ejemplo (CLARO BÁEZ, 2016):

```
DSSCircuit.Generators.get ;
```

```
DSSText.command = ' new generator.PV bus1 = n292757 phases = 3 kv  
= 34.5 kw500 pf = 1 enabled = true' ;
```

```
DSSText.command = ' edit generator.PV bus1 = n1325391 kv = 13.2' ;
```

```
DSSCircuit.ActiveElement.BusNames
```

3.2 Simulación de la red IEEE nodo 13 (OpenDSS).

La red de distribución en la cual nos basaremos es IEEE 13 Node Test Feeder, la cual nos permitirá evaluar datos similares a una red real mediante simulaciones, las cuales nos darán una idea al comportamiento de una red de distribución real.

La Figura 3.3 nos muestra la red IEEE 13 Node Test Feeder simulado en OpenDSS, la cual nos muestra la red con la que estaremos trabajando, en el anexo se puede observar el código en OpenDSS para la red.

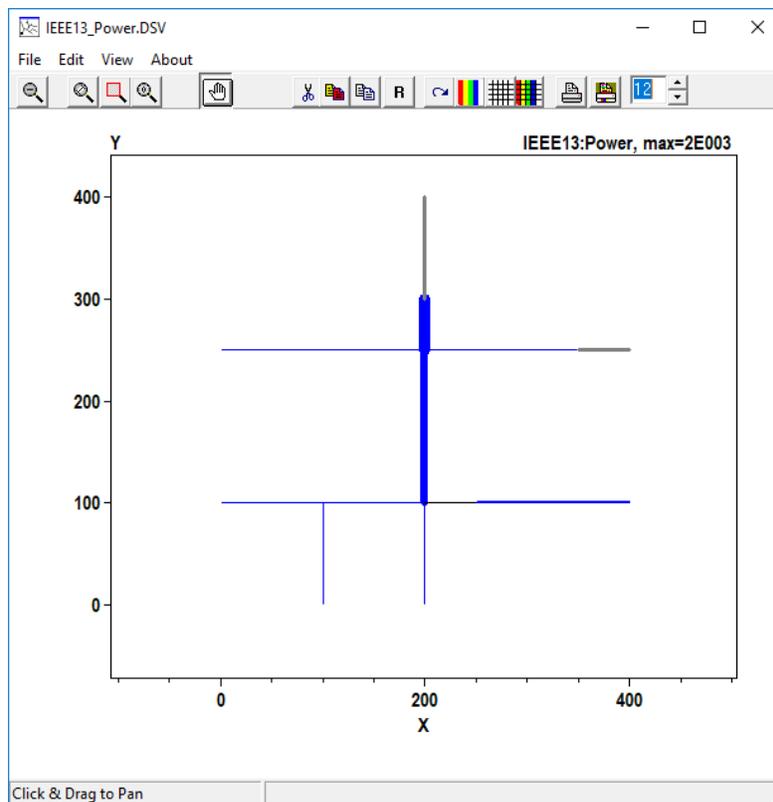


Figura 3.3. Red IEEE nodo 13 en OpenDSS.

Fuente: Wilson Maza.

Elaboración: Wilson Maza.

En la siguiente Figura podemos observar los valores de los voltajes en la barra 675 en la cual será introducido el DG.

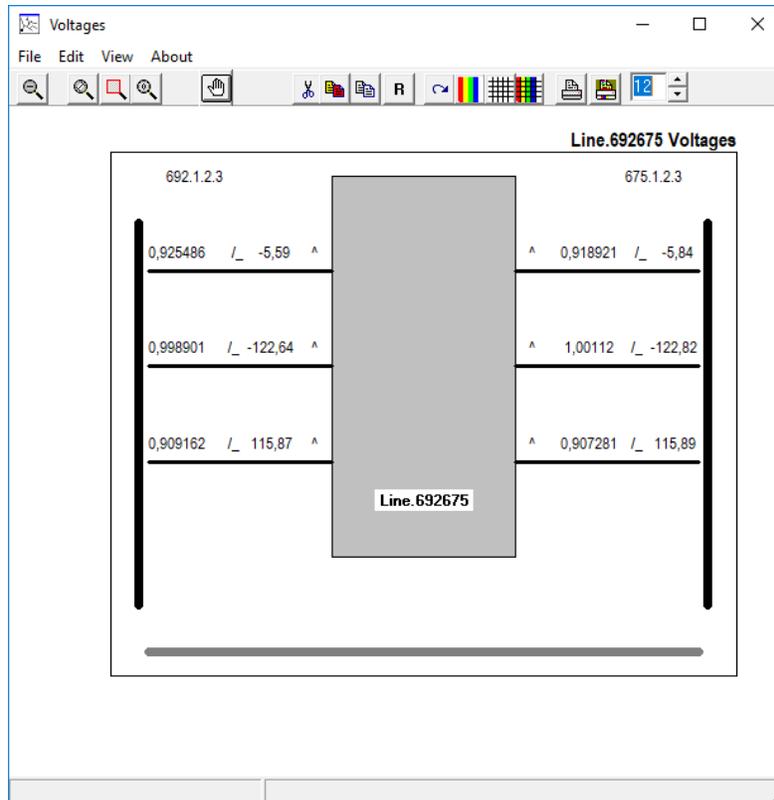


Figura 3.4. Voltajes en las barras 675 y 692.

Fuente: Wilson Maza.

Elaboración: Wilson Maza.

Como podemos observar en la Figura 3.4 nos muestra los voltajes que de las barras 675 y 692 en cada una de las fases de la red, los cual nos ayudara a verificar como se están comportando los voltajes en cada una de las barras.

En la Figura 3.5 observamos los valores de voltaje de la barra 671 en donde será ingresada la carga y los valores de voltaje de la barra 692.

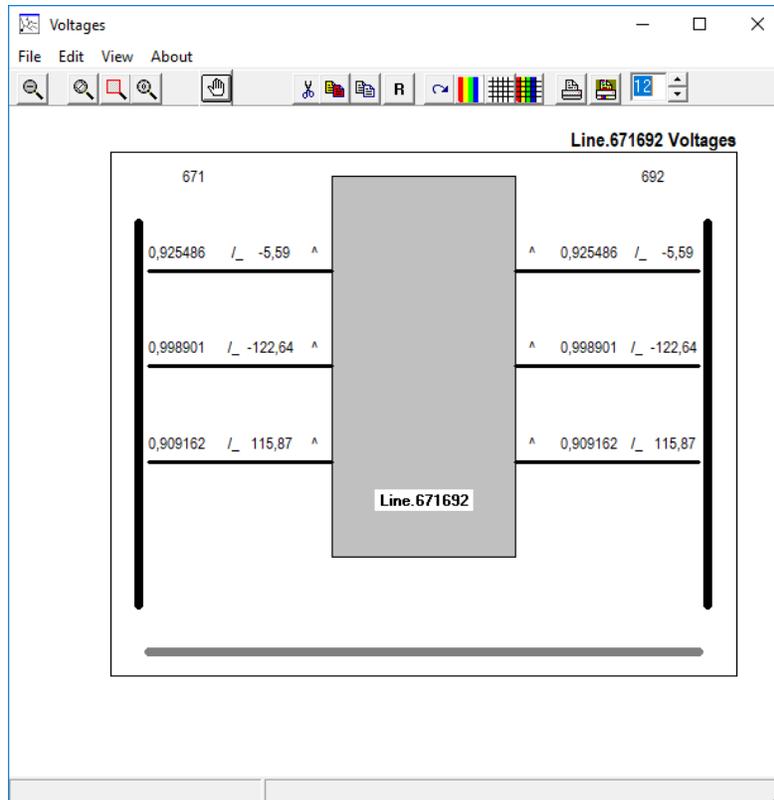


Figura 3.5. Voltajes en las barras 671 y 692.

Fuente: Wilson Maza.

Elaboración: Wilson Maza.

Como podemos observar en la Figura 3.5 se puede visualizar los voltajes que están atravesando esta barra con la presencia de una carga lo que nos permitirá observar los comportamientos de estos voltajes al momento de introducir la carga correspondiente.

CAPITULO IV

**ANALISIS Y SIMULACION DE LA RED IEEE 13 NODE EN MATLAB
APLICANDO VARIACIÓN DE LA CARGA Y PRESENCIA DE UN DG.**

4.1 Simulación de la red en Matlab.

En la Figura 4.1 se observa el bloque creado en simulink de la red IEEE 13 node test feeder, en donde podemos observar las salidas y las funciones que ocupamos para ese bloque de simulación, los cuales se describen a continuación.

- Tap: muestra el número de cambios (taps) que realiza el OLTC para poder regular el voltaje de la red.
- V1: muestra el voltaje que atraviesa cada una de las barras de la red.
- Q5: muestra la variación de las perdidas por potencia activa y reactiva de la red.
- opendssOLTC: función por la cual se llama a la red IEEE nodo 13 simulada en OpenDSS.
- OLTC: función en la cual se simula el transformador OLTC

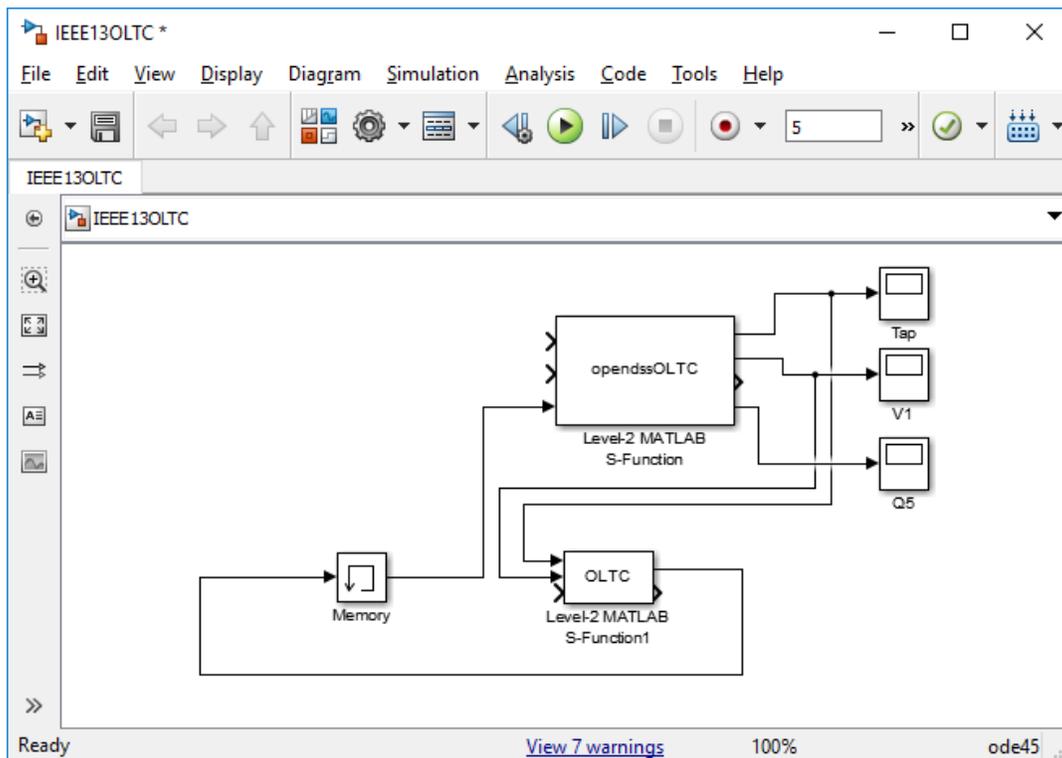


Figura 4.1. Simulación de la red IEEE 13-node en simuink.

Fuente: Wilson Maza.

Elaboración: Wilson Maza.

Para la simulación de la red IEEE 13-node test feeder en Matlab creamos la función OpenDSSOLTC.m. Esta función llama a la red IEEE 13-node que simulamos en el programa OpenDSS, aquí podemos realizar los cambios previstos en la red. Para poder simular el transformador OLTC procedemos a crear la función OLTC.m (el programa en Matlab se lo puede observar en el Anexo) el cual contiene todas las características del transformador OLTC utilizado en el proyecto con sus respectivas entradas y salidas. Mediante la herramienta simulink de Matlab realizaremos un diagrama de bloques en el cual contendrá las dos funciones antes mencionadas las cuales nos permitirán observar los voltajes de cada una de las barras de la red IEEE 13-node sin presencia de cargas y DGs.

Como se puede ver en la Figura 4.1 nos muestra el diagrama de bloques de la red IEEE 13-node simulado en la interfaz gráfica simulink. Como se puede observar en la interfaz de simulink la función opendssOLTC nos muestra las diferentes salidas, las cuales se está analizando como son: Taps, el voltaje V1 y las pérdidas Q5, cabe recalcar que se puede hacer un análisis de diferentes parámetros según sean los requerimientos que se soliciten.

4.2 Comprobación de valores obtenidos sin presencia del DG y de la carga.

Las siguientes Figuras que se muestran a continuación se analizarán como se están comportando los voltajes en las barras, los cuales no presentan variaciones debido a que no se cuenta la presencia del DG y de la carga. Este procedimiento nos permite validar los resultados del programa con los valores de información de voltajes con la información de la red IEEE13 estándar.

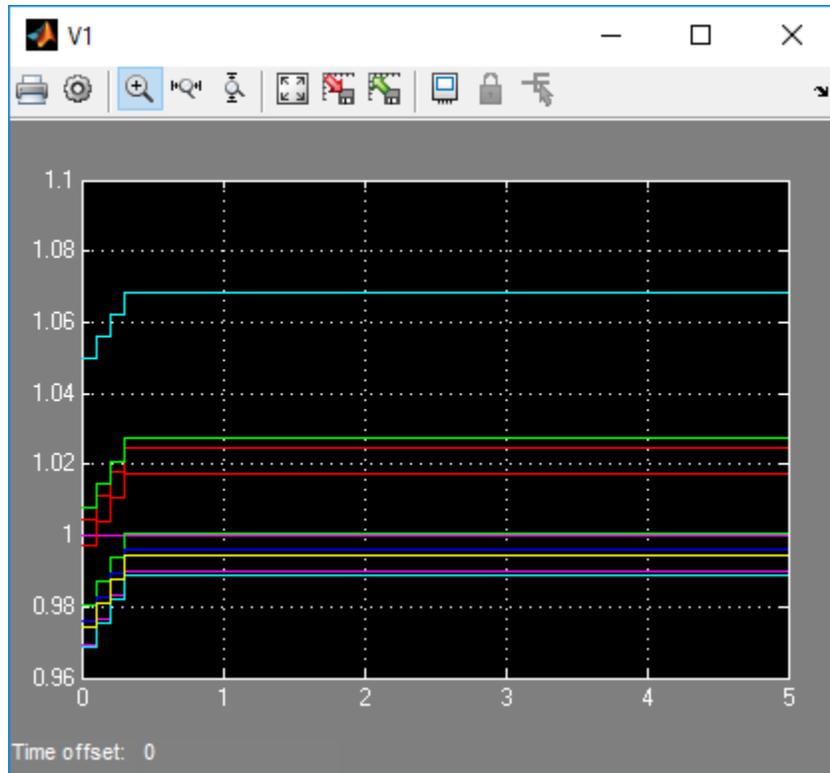


Figura 4.2. Figura de los voltajes sin presencia de DG y sin carga para validar la información.

Fuente: Wilson Maza.

Elaboración: Wilson Maza

La Figura 4.2 nos muestra los voltajes de la red en cada una de las barras. Como no existe la presencia de un DG y una nueva carga, los voltajes se mantienen constantes. Al inicio se pueden observar unas pequeñas variaciones que son provocadas debido a que el programa se está inicializando.

Las Figuras 4.3 y 4.4 nos muestran los comportamientos de los TAPs y de las pérdidas de potencia activa y reactiva (vector Q5) de la red simulados en Matlab sin la presencia de una carga ni del DG. Como podemos observar las Figuras nos muestran valores constantes, debido a que no existe ningún tipo de cambios que interfiera en la red. Estos datos han sido validos con la información que se obtiene en la red estándar IEEE 13.

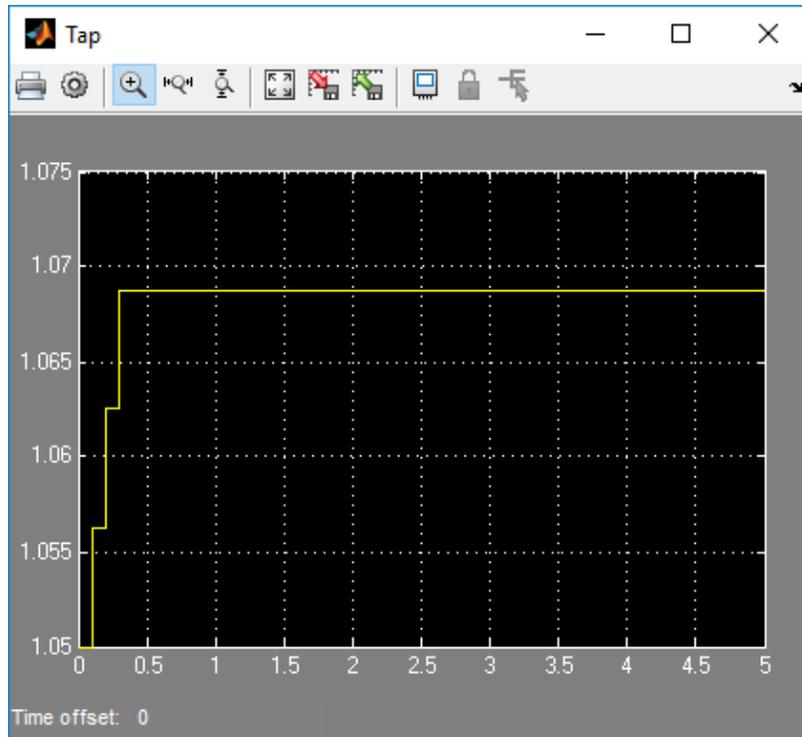


Figura 4.3. Grafica de los saltos del TAP.

Fuente: Wilson Maza.

Elaboración: Wilson Maza

La Figura 4.3 nos muestra los saltos de los TAP del OLTC para regular el voltaje. Los saltos al inicio del programa se deben a la inicialización del programa. Como se puede observar en la figura no existe ningún cambio, esto se debe a que no existe la presencia de un DG o de una carga. Debido a esto los Taps se mantienen en el mismo lugar por lo que no necesitaría realizar los saltos para estabilizar el voltaje en la red.

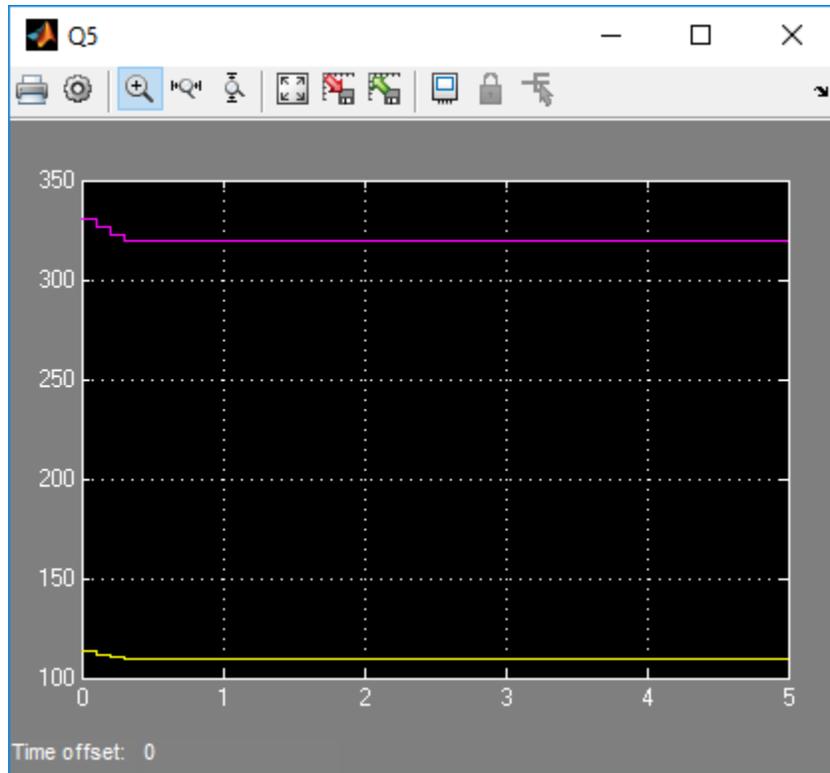


Figura 4.4. Grafica de las pérdidas Q.

Fuente: Wilson Maza.

Elaboración: Wilson Maza.

La Figura 4.4 nos muestra las pérdidas de potencia activa y reactiva de la red. Como podemos ver las pérdidas son constantes debido a que no hay presencia de un DG ni de una carga en la red.

4.3 Simulación de disturbios en la red.

La simulación durara 500 s. Para la introducción del DG y la nueva carga a la red IEEE 13 utilizaremos las funciones `OpenDSSOLTC.m` (el código en Matlab se puede observar en el anexo), en el cual se especificará las barras en donde se introducirá el DG y la nueva carga con los siguientes valores: DG de 1200 kW, $\pm 0,91$ pf ($t = 100$ s) en la barra 675, en el tiempo ($t = 350$ s) se agrega una nueva carga para simular una perturbación (de 1200 kW y 800 kVAr) en la barra 671.

4.4 Introducción del DG y la carga en Simulink.

La siguiente Figura nos muestra el diagrama de bloques en la interfaz de simulink de la red IEEE 13, en donde se puede observar los parámetros del DG y la nueva carga aplicándose el tiempo de simulación y el número de iteraciones. Cada segundo de simulación se realiza 30 iteraciones.

En la salida obtendremos las gráficas de los cambios en el OLTC (TAPs), del voltaje que atraviesa cada una de las barras (V1) y las pérdidas de potencia activa y reactiva de la red (Q5). Observando cómo actúa el OLTC para controlar el voltaje y mantener dentro de los límites establecidos previamente.

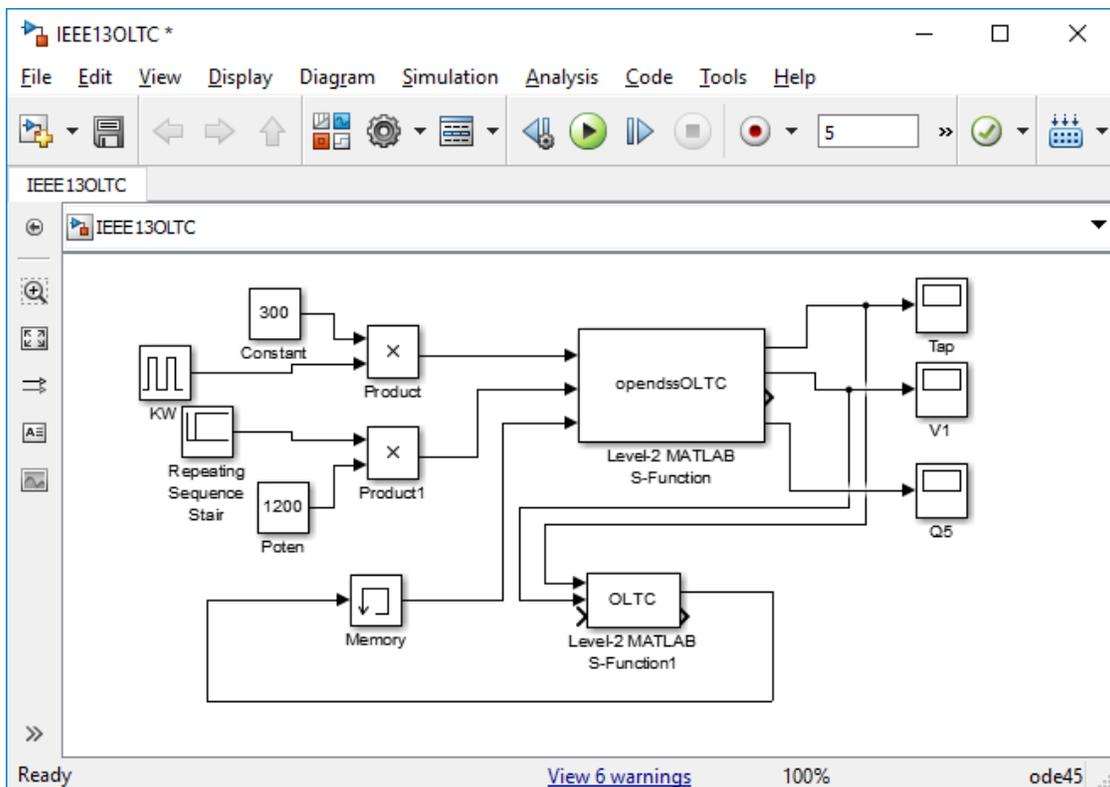


Figura 4.5. Simulación de la red IEEE 13 con DG y carga.

Fuente: Wilson Maza.

Elaboración: Wilson Maza.

Como se puede observar en la Figura 4.5 se muestra el nuevo bloque creado en Simulink con el nombre IEEE13OLTC, que ya contienen el DG y la carga con los valores que previamente se especificó. En la función opendssOLTC se realiza la programación especificando en que barras irán conectados el DG y la carga.

4.5 Variaciones de voltaje en las barras de la red.

La Figura 4.6 nos muestra la variación del voltaje en cada una de las barras de red, los cuales nos permiten hacer un análisis de cómo está actuando el OLTC para poder mantener los voltajes dentro de los valores establecidos. Como se puede observar algunos de los voltajes dependen mucho de la ubicación de las barras. Las barras más cercanas de donde se conecta el DG y la carga se ven más afectadas porque por estas barras atraviesa una corriente diferente.

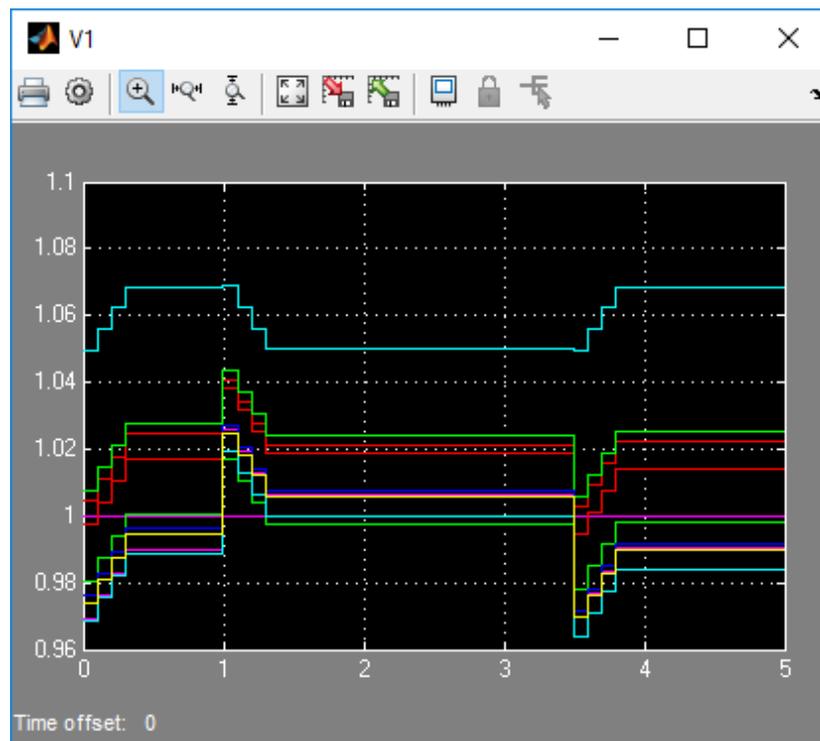


Figura 4.6. Voltajes en cada una de las barras de la red IEEE 13.

Fuente: Wilson Maza.

Elaboración: Wilson Maza.

Al momento que el DG ingresa a la red a los 100 s, se puede observar como existe un aumento del voltaje por lo que el OLTC procede a realizar los saltos debidos para poder mantener los voltajes dentro de los límites establecidos, para luego mantenerlos constantes. En el tiempo 350 s se introduce la nueva carga como se puede observar en la Figura el voltaje disminuye por lo que el OLTC realiza los nuevos saltos requeridos para poder mantener el voltaje sobre los límites que permite la red.

La Figura 4.7 nos muestra como los TAPs están actuando para regular el voltaje en los límites establecidos, se puede observar el número de saltos que realizan los TAPs al momento de ingresar el DG y la carga para realizar óptimamente el control de voltaje.

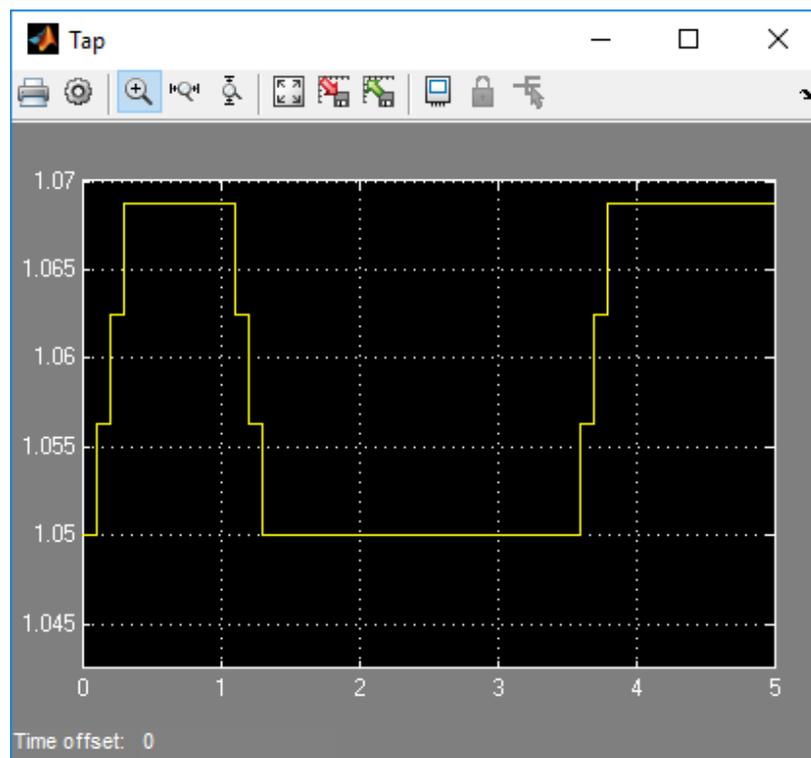


Figura 4.7. Gráfico de los saltos de TAPs.

Fuente: Wilson Maza.

Elaboración: Wilson Maza.

Como se puede observar en la Figura 4.7 en el tiempo 100 s se introduce un DG a la red lo que causa que el voltaje aumente, provocando que los taps realicen los saltos necesarios para mantener el voltaje dentro de los rangos permitidos de la red. En el tiempo 350 s se introduce una carga en la red lo que provoca una disminución en el voltaje debido a esto los taps vuelven a realizar los saltos necesarios que la red necesita para mantenerse dentro del rango establecido.

4.6 Perfil del voltaje en el bus piloto.

En la Figura 4.8 podemos observar los perfiles de los voltajes en las barras en donde se introducen la carga y el DG, en comparación al voltaje en donde no se introduce carga ni DG. observando estos perfiles nos podemos dar cuenta como el OLTC realiza el control de voltaje al momento de introducir dichos valores. al inicio de la simulación podemos ver una perturbación, dicha perturbación se debe a que el programa necesita inicializarse.

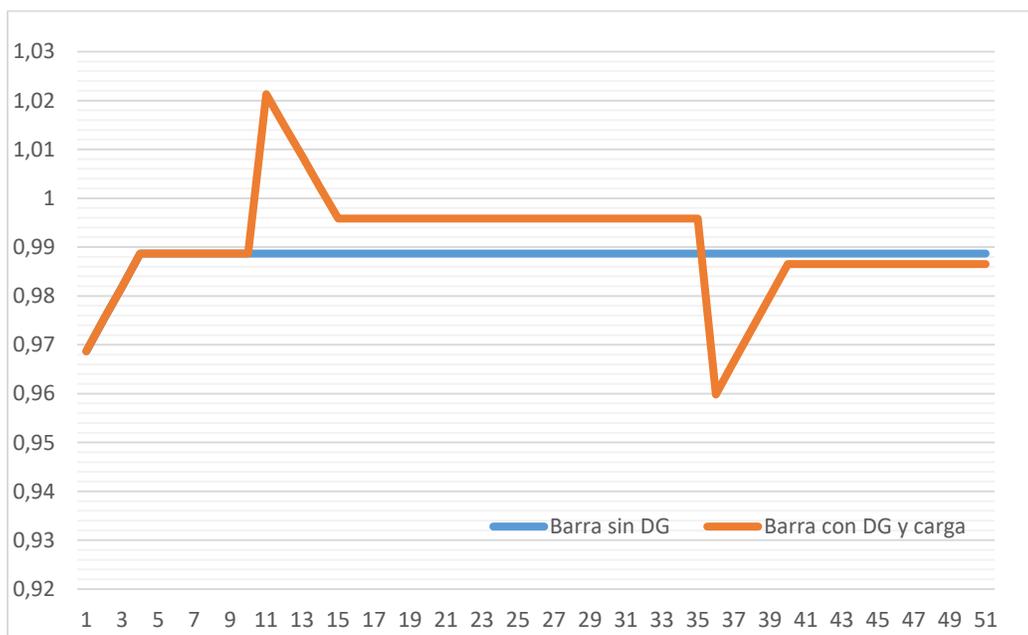


Figura 4.8. Perfil del voltaje con presencia del DG y la carga VS perfil sin presencia de DG y carga.

Fuente: Wilson Maza.

Elaboración: Wilson Maza.

4.7 Análisis de Pérdidas de la red por introducción del DG y la carga.

Las pérdidas que se dan al momento que se introduce el DG y la nueva carga se puede observar en la siguiente Figura, el cual nos muestra las pérdidas de potencia activa y reactiva respectivamente.

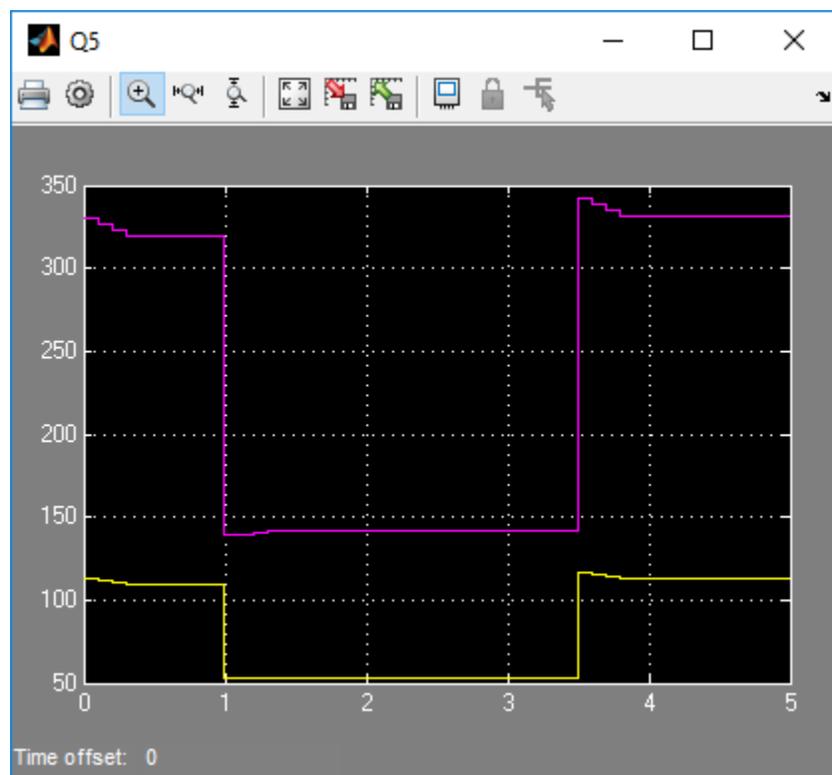


Figura 4.9. Pérdidas Q5 de la red.

Fuente: Wilson Maza.

Elaboración: Wilson Maza.

Como podemos observar en la Figura 4.9 las pérdidas disminuyen al momento que se introduce el DG en la red, esto se debe a que el DG es óptimo que no varía en el tiempo. Analizando esta grafica nos podemos dar cuenta que los DGs

óptimamente colocados en la red disminuyen considerablemente las pérdidas debido a que la corriente que atraviesa la red es menor gracias a la presencia del DG. También podemos observar que las pérdidas aumentan al momento de ingresar la nueva carga, ya que la corriente aumenta y debe compensar el ingreso de la misma aumentando las pérdidas de toda la red.

CONCLUSIONES

- Se ha desarrollado un programa que permite demostrar y observar como el voltaje en las barras de una red de distribución varían ante la presencia de un DG y ante las variaciones de la carga.
- El primer objetivo específico nos permitió estudiar y simular una red estándar de distribución (IEEE 13 node test feeder) utilizando un software libre llamado OPenDSS. Este software es utilizado para simular redes eléctricas y obtener toda la información del flujo de potencia. En este trabajo, OpenDSS trabaja en forma conjunta con Matlab. El programa realizado permite variar los elementos de la red de distribución (Añadir un DG y añadir una carga) para observar los cambios de voltaje en las barras de la red.
- Los resultados obtenidos permiten analizar el impacto en el voltaje cuando se introducen nuevos actores en la red de distribución. Los resultados permiten validar el tamaño del DG y de la carga adicional que puede añadirse a la red.
- El tercer objetivo específico de control del voltaje en las barras de la red de distribución, se lo ha logrado realizar utilizando Matlab para calcular el número de cambios (taps) que tiene que realizar el OLTC para mantener el voltaje dentro de los límites establecidos. De esta manera, se observa el trabajo que realiza el OLTC.
- El control del voltaje por medio del OLTC permite desarrollar sistemas flexibles. Sin embargo, se observa que es necesario trabajar en técnicas de optimización para reducir el trabajo de los OLTC y minimizar las variaciones del voltaje.

- Los resultados obtenidos fueron validados utilizando las tablas de información de la red IEEE analizada. Las mediciones nos permiten observar que los programas realizados nos ayudan a entender el trabajo real que tienen los elementos presentes en las redes de distribución.

RECOMENDACIONES

Es necesario hacer otros trabajos de titulación que nos permita encontrar: a) el lugar adecuado y el tamaño del DG, b) buscar nuevas técnicas que nos permita minimizar el trabajo del OLTC y c) buscar técnicas de optimización para disminución de las pérdidas en redes eléctricas

REFERENCIAS.

- Calderaro, V., Galdi, V., Lamberti, F., Piccolo, A., & Graditi, G. (2014). Voltage support control of unbalanced distribution systems by reactive power regulation. *2014 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, 1–5.
- Caldon, R., Coppa, M., Sgarbossa, R., & Turri, R. (2013). A simplified algorithm for OLTC control in active distribution MV networks. *AEIT Annual Conference, 2013*, (October), 1–6. <https://doi.org/10.1109/AEIT.2013.6666788>
- Castro, J. R., Saad, M., Lefebvre, S., Asber, D., & Lenoir, L. (2016a). Coordinated voltage control in distribution network with the presence of DGs and variable loads using pareto and fuzzy logic. *Energies*, *9*(2), 1–16. <https://doi.org/10.3390/en9020107>
- Castro, J. R., Saad, M., Lefebvre, S., Asber, D., & Lenoir, L. (2016b). Optimal voltage control in distribution network in the presence of DGs. *International Journal of Electrical Power and Energy Systems*, *78*, 239–247. <https://doi.org/10.1016/j.ijepes.2015.11.081>
- Cavraro, G., & Carli, R. (2017). Local and distributed voltage control algorithms in distribution network, *8950*(c), 1–10. <https://doi.org/10.1109/TPWRS.2017.2731682>
- CLARO BÁEZ, J. P. (2016). Modelado y simulación de Smart Grid con OpenDSS y Matlab.
- Dohnal, D. (2013). On-load tap changers for power transformers. *MR Knowledge Base*, *24*.
- Dugan, R. (2009). OpenDSS Introductory Training Level 1 Introduction.
- Dugan, R. C. (2009). Systems OpenDSS Why was DSS Developed ? Why

- OpenDSS ? *Architecture*, (June), 8–11.
- Fernercom. (2007). Guía Básica de la Generación Distribuida. *Fundación de La Energía de La Comunidad de Madrid*, 65.
- IEEE Distribution System Subcommittee, a. (2001). IEEE 13 Node Test Feeder Report, 1–11.
- Jose, C., & Mendieta, C. (2016). INTELLIGENT DISTRIBUTION VOLTAGE CONTROL WITH DISTRIBUTED GENERATION Jose CASTRO MENDIETA MANUSCRIPT – BASED THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE IN PARTIAL FULFILLMENT FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.
- Kulmala, A., Repo, S., & Jarventausta, P. (2014). Coordinated voltage control in distribution networks including several distributed energy resources. *IEEE Transactions on Smart Grid*, 5(4), 2010–2020.
<https://doi.org/10.1109/TSG.2014.2297971>
- Lane, H., Chen, T., Croda, T. G., Welch, S., & Editor, A. (n.d.). *ELECTRIC POWER*.
- Meghasai, Monger, S., Vega, R., & Krishnaswami, H. (2015). Simulation of smart functionalities of photovoltaic inverters by interfacing OpenDSS and Matlab. *2015 IEEE 16th Workshop on Control and Modeling for Power Electronics, COMPEL 2015*, (Lv). <https://doi.org/10.1109/COMPEL.2015.7236443>
- Planning, I. D. (1991). Radial Distribution Test Feeders, 6(3), 975–985.
- Quintanilla Tovar, E. B., & Zamora Cárdenas, E. A. (2017). Desarrollo de una interfaz Matlab-Openss para simulacion de sistemas fotovoltaicos. *Jóvenes En La Ciencia*, 2(1), 1117–1121.
- Reales, E. De, & Impresas, Ó. (2012). Proyecto Fin de Carrera.
- Red, C. I., & Energía, D. E. D. D. E. (n.d.). CAPÍTULO I DE ENERGÍA ELÉCTRICA.

Redfern, C. G. M. A. (n.d.). AUTOMATIC COMPENSATION VOLTAGE CONTROL STRATEGY FOR ON-LOAD TAP CHANGER TRANSFORMERS WITH DISTRIBUTED GENERATIONS, 1–5.

Richardot, O., Viciu, A., Bésanger, Y., Hadjsaïd, N., Member, S., & Kieny, C. (n.d.). Networks Using Distributed Generation, 1–6.

Rodrigo, M. C. A. (2013). Universidad Politécnica Salesiana Sede Quito. *Tesis*, 1–100. <https://doi.org/10.20868/UPM.thesis.39079>

Saha, S., & Johnson, N. (2016). Modeling and Simulation in XENDEE IEEE 13 Node Test Feeder.

Sergio, A. (2012). Capítulo 2. El conmutador bajo carga, 6–14.

Tseng, K. (2014). INTRODUCTION TO MATLAB.

ANEXOS

ANEXO I: Código en OpenDSS de la red IEEE 13 node test feeder

Clear

new circuit.IEEE13

~ basekv=115 pu=1.0001 phases=3 bus1=SourceBus

~ Angle=30

! advance angle 30 deg so result agree with published angle

~ MVAsc3=20000 MVASC1=21000 ! stiffen the source to approximate inf source

!SUB TRANSFORMER DEFINITION

! Although this data was given, it does not appear to be used in the test case results

! The published test case starts at 1.0 per unit at Bus 650. To make this happen, we will change the impedance

! on the transformer to something tiny by dividing by 1000 using the DSS in-line RPN math

New Transformer.Sub Phases=3 Windings=2 XHL=(8 1000 /)

~ wdg=1 bus=SourceBus conn=delta kv=115 kva=5000 %r=(.5 1000 /) XHT=4

~ wdg=2 bus=650 conn=wye kv=4.16 kva=5000 %r=(.5 1000 /) XLT=4

! FEEDER 1-PHASE VOLTAGE REGULATORS

! Define low-impedance 2-wdg transformer

New Transformer.Reg1 phases=1 XHL=0.01 kVAs=[1666 1666]

~ Buses=[650.1 RG60.1] kVs=[2.4 2.4] %LoadLoss=0.01

new regcontrol.Reg1 transformer=Reg1 winding=2 vreg=122 band=2 ptratio=20 ctpri=700 R=3 X=9

New Transformer.Reg2 phases=1 XHL=0.01 kVAs=[1666 1666]

~ Buses=[650.2 RG60.2] kVs=[2.4 2.4] %LoadLoss=0.01

new regcontrol.Reg2 transformer=Reg2 winding=2 vreg=122 band=2 ptratio=20 ctpri=700 R=3 X=9

New Transformer.Reg3 phases=1 XHL=0.01 kVAs=[1666 1666]

~ Buses=[650.3 RG60.3] kVs=[2.4 2.4] %LoadLoss=0.01

new regcontrol.Reg3 transformer=Reg3 winding=2 vreg=122 band=2 ptratio=20 ctpri=700 R=3 X=9

!TRANSFORMER DEFINITION

New Transformer.XFM1 Phases=3 Windings=2 XHL=2

~ wdg=1 bus=633 conn=Wye kv=4.16 kva=500 %r=.55 XHT=1

~ wdg=2 bus=634 conn=Wye kv=0.480 kva=500 %r=.55 XLT=1

!LINE CODES

redirect IEEELineCodes.dss

```

// these are local matrix line codes
// corrected 9-14-2011
New linecode.mtx601 nphases=3 BaseFreq=60
~ rmatrix = (0.3465 | 0.1560 0.3375 | 0.1580 0.1535 0.3414 )
~ xmatrix = (1.0179 | 0.5017 1.0478 | 0.4236 0.3849 1.0348 )
~ units=mi
New linecode.mtx602 nphases=3 BaseFreq=60
~ rmatrix = (0.7526 | 0.1580 0.7475 | 0.1560 0.1535 0.7436 )
~ xmatrix = (1.1814 | 0.4236 1.1983 | 0.5017 0.3849 1.2112 )
~ units=mi
New linecode.mtx603 nphases=2 BaseFreq=60
~ rmatrix = (1.3238 | 0.2066 1.3294 )
~ xmatrix = (1.3569 | 0.4591 1.3471 )
~ units=mi
New linecode.mtx604 nphases=2 BaseFreq=60
~ rmatrix = (1.3238 | 0.2066 1.3294 )
~ xmatrix = (1.3569 | 0.4591 1.3471 )
~ units=mi
New linecode.mtx605 nphases=1 BaseFreq=60
~ rmatrix = (1.3292 )
~ xmatrix = (1.3475 )
~ units=mi
New linecode.mtx606 nphases=3 BaseFreq=60
~ rmatrix = (0.7982 | 0.3192 0.7891 | 0.2849 0.3192 0.7982 )
~ xmatrix = (0.4463 | 0.0328 0.4041 | -0.0143 0.0328 0.4463 )
~ Cmatrix = [257 | 0 257 | 0 0 257]
~ units=mi
New linecode.mtx607 nphases=1 BaseFreq=60
~ rmatrix = (1.3425 )
~ xmatrix = (0.5124 )
~ cmatrix = [236]
~ units=mi

!LOAD DEFINITIONS
New Load.671 Bus1=671.1.2.3 Phases=3 Conn=Delta Model=1 kV=4.16 kW=1155 kvar=660
New Load.634a Bus1=634.1 Phases=1 Conn=Wye Model=1 kV=0.277 kW=160 kvar=110
New Load.634b Bus1=634.2 Phases=1 Conn=Wye Model=1 kV=0.277 kW=120 kvar=90
New Load.634c Bus1=634.3 Phases=1 Conn=Wye Model=1 kV=0.277 kW=120 kvar=90
New Load.645 Bus1=645.2 Phases=1 Conn=Wye Model=1 kV=2.4 kW=170 kvar=125
New Load.646 Bus1=646.2.3 Phases=1 Conn=Delta Model=2 kV=4.16 kW=230 kvar=132
New Load.692 Bus1=692.3.1 Phases=1 Conn=Delta Model=5 kV=4.16 kW=170 kvar=151
New Load.675a Bus1=675.1 Phases=1 Conn=Wye Model=1 kV=2.4 kW=485 kvar=190
New Load.675b Bus1=675.2 Phases=1 Conn=Wye Model=1 kV=2.4 kW=68 kvar=60
New Load.675c Bus1=675.3 Phases=1 Conn=Wye Model=1 kV=2.4 kW=290 kvar=212
New Load.611 Bus1=611.3 Phases=1 Conn=Wye Model=5 kV=2.4 kW=170 kvar=80

```

New Load.652 Bus1=652.1 Phases=1 Conn=Wye Model=2 kV=2.4 kW=128 kvar=86
New Load.670a Bus1=670.1 Phases=1 Conn=Wye Model=1 kV=2.4 kW=17 kvar=10
New Load.670b Bus1=670.2 Phases=1 Conn=Wye Model=1 kV=2.4 kW=66 kvar=38
New Load.670c Bus1=670.3 Phases=1 Conn=Wye Model=1 kV=2.4 kW=117 kvar=68

!CAPACITOR DEFINITIONS

New Capacitor.Cap1 Bus1=675 phases=3 kVAR=600 kV=4.16
New Capacitor.Cap2 Bus1=611.3 phases=1 kVAR=100 kV=2.4

!Bus 670 is the concentrated point load of the distributed load on line 632 to 671 located at 1/3 the distance from node 632

!LINE DEFINITIONS

New Line.650632 Phases=3 Bus1=RG60.1.2.3 Bus2=632.1.2.3 LineCode=mtx601 Length=2000 units=ft
New Line.632670 Phases=3 Bus1=632.1.2.3 Bus2=670.1.2.3 LineCode=mtx601 Length=667 units=ft
New Line.670671 Phases=3 Bus1=670.1.2.3 Bus2=671.1.2.3 LineCode=mtx601 Length=1333 units=ft
New Line.671680 Phases=3 Bus1=671.1.2.3 Bus2=680.1.2.3 LineCode=mtx601 Length=1000 units=ft
New Line.632633 Phases=3 Bus1=632.1.2.3 Bus2=633.1.2.3 LineCode=mtx602 Length=500 units=ft
New Line.632645 Phases=2 Bus1=632.3.2 Bus2=645.3.2 LineCode=mtx603 Length=500 units=ft
New Line.645646 Phases=2 Bus1=645.3.2 Bus2=646.3.2 LineCode=mtx603 Length=300 units=ft
New Line.692675 Phases=3 Bus1=692.1.2.3 Bus2=675.1.2.3 LineCode=mtx606 Length=500 units=ft
New Line.671684 Phases=2 Bus1=671.1.3 Bus2=684.1.3 LineCode=mtx604 Length=300 units=ft
New Line.684611 Phases=1 Bus1=684.3 Bus2=611.3 LineCode=mtx605 Length=300 units=ft
New Line.684652 Phases=1 Bus1=684.1 Bus2=652.1 LineCode=mtx607 Length=800 units=ft

!SWITCH DEFINITIONS

New Line.671692 Phases=3 Bus1=671 Bus2=692 Switch=y r1=1e-4 r0=1e-4 x1=0.000 x0=0.000 c1=0.000 c0=0.000

Set Voltagebases=[115, 4.16, .48]

!-----

! Alternate Solution Script

! To force the taps to be same as published results, set the transformer taps manually and disable the controls

!-----

!Transformer.Reg1.Taps=[1.0 1.0625]

!Transformer.Reg2.Taps=[1.0 1.0500]

!Transformer.Reg3.Taps=[1.0 1.06875]

Set Controlmode=ON

calcv

!Solve

BusCoords IEEE13Node_BusXY.csv

!-----

```

!-----Show some Results -----
!-----

// Show Voltages LN Nodes
// Show Currents Elem
// Show Powers kVA Elem
// Show Losses
//Show Taps

!-----

Solve
plot circuit Power max=2000 n n C1=$00FF0000

```

ANEXO II: Código de la función OpenDSSOLTC.m en Matlab.

```

function openssT2OLTC(block)

setup(block);
function setup(block)

block.NumInputPorts = 3;
block.NumOutputPorts = 4;

block.SetPreCompInpPortInfoToDynamic;
block.SetPreCompOutPortInfoToDynamic;

block.InputPort(1).Dimensions = 1;
block.InputPort(1).DatatypeID = 0;
block.InputPort(1).Complexity = 'Real';
block.InputPort(1).DirectFeedthrough = true;
block.InputPort(1).SamplingMode = 'Sample';

```

```

block.InputPort(2).Dimensions      = 1;
block.InputPort(2).DatatypeID     = 0;
block.InputPort(2).Complexity     = 'Real';
block.InputPort(2).DirectFeedthrough = true;
block.InputPort(2).SamplingMode   = 'Sample';

block.InputPort(3).Dimensions      = 1;
block.InputPort(3).DatatypeID     = 0;
block.InputPort(3).Complexity     = 'Real';
block.InputPort(3).DirectFeedthrough = true;
block.InputPort(3).SamplingMode   = 'Sample';

block.OutputPort(1).Dimensions     = 1;
block.OutputPort(1).DatatypeID    = 0;
block.OutputPort(1).Complexity    = 'Real';
block.OutputPort(1).SamplingMode  = 'Sample';

block.OutputPort(2).Dimensions     = 13;
block.OutputPort(2).DatatypeID    = 0;
block.OutputPort(2).Complexity    = 'Real';
block.OutputPort(2).SamplingMode  = 'Sample';

block.OutputPort(3).Dimensions     = 82;
block.OutputPort(3).DatatypeID    = 0;
block.OutputPort(3).Complexity    = 'Real';
block.OutputPort(3).SamplingMode  = 'Sample';

block.OutputPort(4).Dimensions     = 2;
block.OutputPort(4).DatatypeID    = 0;
block.OutputPort(4).Complexity    = 'Real';
block.OutputPort(4).SamplingMode  = 'Sample';

block.NumDialogPrms               = 0;

block.SampleTimes = [-1, 0];

```

```

block.SimStateCompliance = 'DefaultSimState';

block.RegBlockMethod('PostPropagationSetup', @DoPostPropSetup);
block.RegBlockMethod('InitializeConditions', @InitializeConditions);
block.RegBlockMethod('Start', @Start);
block.RegBlockMethod('Outputs', @Outputs);
block.RegBlockMethod('Update', @Update);
block.RegBlockMethod('Derivatives', @Derivatives);
block.RegBlockMethod('Terminate', @Terminate);

function DoPostPropSetup(block)
block.NumDworks = 3;

    block.Dwork(1).Name           = 'x1';
    block.Dwork(1).Dimensions     = 1;
    block.Dwork(1).DatatypeID    = 0;
    block.Dwork(1).Complexity    = 'Real';
    block.Dwork(1).UsedAsDiscState = true;

    block.Dwork(2).Name           = 'x2';
    block.Dwork(2).Dimensions     = 1;
    block.Dwork(2).DatatypeID    = 0;
    block.Dwork(2).Complexity    = 'Real';
    block.Dwork(2).UsedAsDiscState = true;

    block.Dwork(3).Name           = 'x3';
    block.Dwork(3).Dimensions     = 1;
    block.Dwork(3).DatatypeID    = 0;
    block.Dwork(3).Complexity    = 'Real';
    block.Dwork(3).UsedAsDiscState = true;

function InitializeConditions(block)

function Start(block)

```

```
block.Dwork(1).Data = 0;
block.Dwork(2).Data = 0;
```

```
function Outputs(block)
```

```
[DSSStartOK, DSSObj, DSSText] = DSSStartup;
```

```
W1 = block.InputPort(1).Data;
```

```
DG = block.InputPort(2).Data;
```

```
Tap = block.InputPort(3).Data;
```

```
if DSSStartOK
```

```
    DSSText.command='Compile (C:\Users\DELL\Documents\gp sistemas
estocasticos\frecuencia en
opendss\IEEE13T1\IEEE13\Matlab\IEEE13_OLTC.dss)';
```

```
DSSCircuit=DSSObj.ActiveCircuit;
```

```
DSSSolution=DSSCircuit.Solution;
```

```
DSSText.Command='New EnergyMeter.Main Line.650632 1';
```

```
VA1 = W1*0.666;
```

```
MyNextVA=VA1;
```

```
MyNextW=W1;
```

```
MyNextTap=Tap;
```

```
MyDG = DG;
```

```
DSSText.Command=['New Load.1 Bus1=671.1.2.3 Phases=3 Conn=Delta Model=1
kV=4.16 kW=' num2str(MyNextW) ' kvar=' num2str(MyNextVA)];
```

```

DSSCircuit.Generators.get;

    DSSText.command = ['new generator.Gen1 Bus1= 675 phases=3 kV=4.16
kw=' num2str(MyDG) ' pf=0.91 enabled=true'];

    DSSCircuit.Generators.get;
DSSCircuit.SetActiveElement('generator.Gen1');
DSSCircuit.ActiveElement.BusNames;

DSSText.Command = ['Transformer.Reg1.Taps=[1.0 ' num2str(MyNextTap)];
DSSText.Command = ['Transformer.Reg2.Taps=[1.0 ' num2str(MyNextTap)];
DSSText.Command = ['Transformer.Reg3.Taps=[1.0 ' num2str(MyNextTap)];

DSSText.Command='Set MaxControlIter=30';

DSSSolution.Solve;

DSSText.Command='Buscoords IEEE13Node_BusXY.csv ! load in bus
coordinates';

V1 = DSSCircuit.AllNodeVmagPUByPhase(1);
Dist1 = DSSCircuit.AllNodeDistancesByPhase(1);
V2 = DSSCircuit.AllNodeVmagPUByPhase(2);
Dist2 = DSSCircuit.AllNodeDistancesByPhase(2);
V3 = DSSCircuit.AllNodeVmagPUByPhase(3);
Dist3 = DSSCircuit.AllNodeDistancesByPhase(3);

Q1 = DSSCircuit.AllElementLosses;

```

```

Q2 = DSSCircuit.Loads.kw;
Q3 = DSSCircuit.Loads.kVar;
Q4 = DSSCircuit.TotalPower;
Q5 = DSSCircuit.Losses;

plot(Dist1, V1, 'k*');
hold on;
plot(Dist2, V2, 'r+');
plot(Dist3, V3, 'bd');
legend('phase A', 'phase B', 'phase C', 'Location', 'SouthEast');

title('Voltage Profile Plot');

ylim([0.9 1.07]);
ylabel('Volts (pu)');
xlabel('Distance from Substation');

hold off;

else
    a='DSS Did Not Start';
    disp(a);

end

block.OutputPort(1).Data = Tap;
block.OutputPort(2).Data = V1;
block.OutputPort(3).Data = Q1;
block.OutputPort(4).Data = Q5/1000;

function Update(block)

block.Dwork(1).Data = block.InputPort(1).Data;
block.Dwork(2).Data = block.InputPort(2).Data;

```

```
function Derivatives(block)
```

```
function Terminate(block)
```

ANEXO III: Código de la función OLTC.m en Matlab.

```
function OLTC(block)
```

```
setup(block);
```

```
function setup(block)
```

```
block.NumInputPorts = 3;
```

```
block.NumOutputPorts = 2;
```

```
block.SetPreCompInpPortInfoToDynamic;
```

```
block.SetPreCompOutPortInfoToDynamic;
```

```
block.InputPort(1).Dimensions = 1;
```

```
block.InputPort(1).DatatypeID = 0;
```

```
block.InputPort(1).Complexity = 'Real';
```

```
block.InputPort(1).DirectFeedthrough = true;
```

```
block.InputPort(1).SamplingMode = 'Sample';
```

```
block.InputPort(2).Dimensions = 13;
```

```
block.InputPort(2).DatatypeID = 0;
```

```
block.InputPort(2).Complexity = 'Real';
```

```
block.InputPort(2).DirectFeedthrough = true;
```

```
block.InputPort(2).SamplingMode = 'Sample';
```

```
block.InputPort(3).Dimensions = 1;
```

```
block.InputPort(3).DatatypeID = 0;
```

```

block.InputPort(3).Complexity = 'Real';
block.InputPort(3).DirectFeedthrough = true;
block.InputPort(3).SamplingMode = 'Sample';

block.OutputPort(1).Dimensions = 1;
block.OutputPort(1).DatatypeID = 0;
block.OutputPort(1).Complexity = 'Real';
block.OutputPort(1).SamplingMode = 'Sample';

block.OutputPort(2).Dimensions = 13;
block.OutputPort(2).DatatypeID = 0;
block.OutputPort(2).Complexity = 'Real';
block.OutputPort(2).SamplingMode = 'Sample';

block.NumDialogPrms = 0;

block.SampleTimes = [-1, 0];

block.SimStateCompliance = 'DefaultSimState';

block.RegBlockMethod('PostPropagationSetup', @DoPostPropSetup);
block.RegBlockMethod('InitializeConditions', @InitializeConditions);
block.RegBlockMethod('Start', @Start);
block.RegBlockMethod('Outputs', @Outputs);
block.RegBlockMethod('Update', @Update);
block.RegBlockMethod('Derivatives', @Derivatives);
block.RegBlockMethod('Terminate', @Terminate);

function DoPostPropSetup(block)
block.NumDworks = 3;

block.Dwork(1).Name = 'x1';
block.Dwork(1).Dimensions = 1;

```

```

block.Dwork(1).DatatypeID      = 0;
block.Dwork(1).Complexity      = 'Real';
block.Dwork(1).UsedAsDiscState = true;

block.Dwork(2).Name            = 'x2';
block.Dwork(2).Dimensions      = 13;
block.Dwork(2).DatatypeID      = 0;
block.Dwork(2).Complexity      = 'Real';
block.Dwork(2).UsedAsDiscState = true;

block.Dwork(3).Name            = 'x3';
block.Dwork(3).Dimensions      = 1;
block.Dwork(3).DatatypeID      = 0;
block.Dwork(3).Complexity      = 'Real';
block.Dwork(3).UsedAsDiscState = true;

function InitializeConditions(block)

function Start(block)

block.Dwork(1).Data = 0;
block.Dwork(2).Data = zeros(1, 13);

function Outputs(block)

    MyNextTap = block.InputPort(1).Data;
    V1 = block.InputPort(2).Data;
    Tap1 = block.InputPort(3).Data;

    Er = (1 - V1(6));

    if Er > 0.009;
        MyNextTap = MyNextTap + 0.00625;
    else

```

```
    if Er < -0.009;
        MyNextTap = MyNextTap - 0.00625;
    else
        MyNextTap = MyNextTap;
    end
end

block.OutputPort(1).Data = MyNextTap;
block.OutputPort(2).Data = V1;

function Update(block)

block.Dwork(1).Data = block.InputPort(1).Data;
block.Dwork(2).Data = block.InputPort(2).Data;

function Derivatives(block)

function Terminate(block)
```